Gru & Vector Gimbal Rocket Design and Launch Report

Ilan Villard*, Krish Mishra*, Taekyung Kim*, Albert Zheng*, Catherine Fang*, Cheng Liu*, Patrick Barry*, James Barritt Chambers*, Sana Churi*, Margaret Hwang*

Georgia Institute of Technology, Atlanta, Georga, 30332, United States

Active control stabilization systems in rocketry are used to supplement or replace fins to improve stability. One form of active stabilization is through movement of the rocket motor itself, in which the position of the motor is used to correct any deviations from a straight flight path. This method, known as motor gimbaling, uses an onboard inertial measurement unit (IMU) to detect and monitor the orientation of the rocket in the air through a closed-loop algorithm. Once the IMU's orientation deviates from its initial position, smaller servo motors are used to rotate the motor in a way that would counteract this change and return the attitude of the rocket back to zero. The GNC Project within the Ramblin' Rocket Club at the Georgia Institute of Technology has designed, built, and launched two mid-powered rockets, named Gru and Vector, with a gimbaled motor system in February 2024. Each rocket consists of different parameters – the one named Gru has a length of 26 inches and uses a G12 motor, while Vector has a length of 22 inches and uses an F10 motor. These variations allow us to compare the effects of different motor burn times, thrust outputs, and rocket conditions on the functionality of the gimbaled motor. The components of the gimbaling system were designed and built by students in house, where 3D printing was primarily used to manufacture components of various systems. This paper details the design process, simulations, manufacturing, and testing of the two gimbaled motor rockets. Furthermore, the paper will discuss the challenges and results encountered in launching the two rockets.

I. Introduction and Background

Traditional rocket stabilization methods rely on fins for stabilization in vertical flight. However, fins are limited in their ability to account for sudden changes in atmospheric conditions such as wind, as well as in any attempt to give the rocket direction. As a result, many alternate stabilization systems have been developed to give rockets more stability and accuracy. One of these methods is motor gimbaling, which adjusts the attitude of the rocket motor to counter deviations from its intended trajectory. The GNC Project within the Ramblin' Rocket Club at the Georgia Institute of Technology designed, built, and launched two mid-powered rockets—Gru and Vector—with gimbaled motor systems in the fall of 2023.

This paper delves into the design choices for the gimbal and rocket system, simulations, manufacturing, and test phases undertaken by the project. The motor gimbal design consisted of two MG92B servo motors that controlled a 3D printed motor mount and were actuated by a Teensy. Two different motors were selected for the two rockets. The rocket named Gru, which had a length of 26 inches, used a G12 motor with a longer burn time, and the rocket named Vector, with a length of 22 inches, used a F10 motor with a shorter burn time. Simulations for the launch were done on OpenRocket software and were used to determine design choices. The avionics software was created by the team and used state estimation to relate the system inputs and outputs through the Teensy. The inputs were determined using a sensor suite consisted of an accelerometer, a barometer, and a gyroscope. Once a deviation from the vertical flight was detected through the sensors, the signal would go through the Teensy, which would acuate the motor gimbal to counteract the change. The two rockets were launched at a launch site in Samson, Alabama on February 3, 2024. The results of this launch are discussed in the Results and Discussions sections.

*Student Member Undergraduate Category
AIAA numbers in order of names: *160356, 160326, 1603498, 1603522, 1405486, 1600100, 1418912, 1603551, 1603466, 1603558*

II. Nomenclature

$\dot{q}$ = time derivative of the quaternion
q = quaternion
ω_x = X component of angular velocity
ω_y = Y component of angular velocity
ω_z = Z component of angular velocity

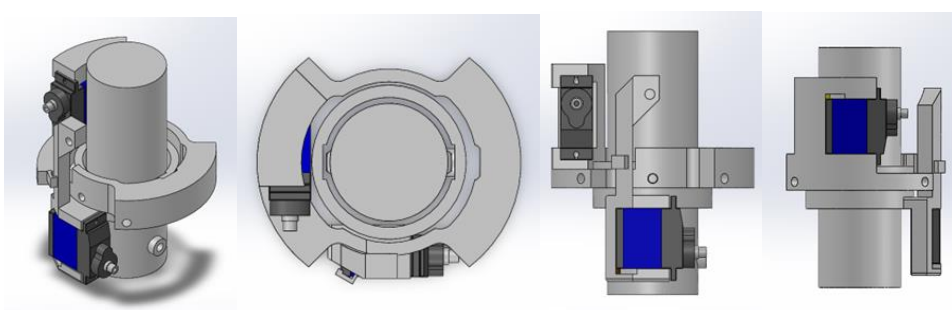
III. Methods and Design

Two rockets, Gru and Vector, were designed to test the gimbal system. The components of Gru and Vector were designed using SolidWorks and manufactured using Prusa MK3 printers. This section details the design process of the main rocket systems.

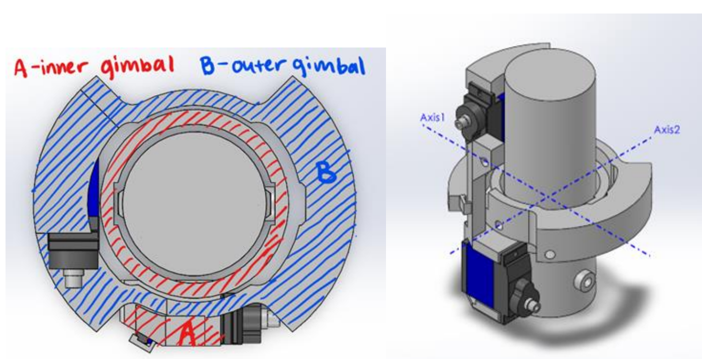
A. Motor Gimbal

To gimbal the motor, the team decided on a 2 degree-of-freedom system to control motion in the yaw and pitch directions. The design of the gimbal was heavily influenced by Joe Barnard and his Signal R2 TVC mount, which relies on controlling the motor with two servos mounted within the airframe [1].

The goal of the gimbal was to be able to rotate the motor a minimum of 5 degrees in both the yaw and pitch directions. Thus, the main physical constraint was keeping the entire system within the 3-inch inner diameter of the tube. To minimize interference with the body tube, two holes were cut into the body tubes that would allow the servo horns to rotate through their full range of motion. For these launches, two 29 mm motors were used, one of length 92 mm and the other of length 156 mm.



(a) Isometric View **(b) Top View** **(c) Side Views**



(d) Identification of outer and inner gimbal **(e) Isometric view with axes of motion marked**

Figure 1. Motor Gimbal Mechanical Design

Yaw was taken to be along Axis 1, controlled by the inner gimbal servo, while pitch was taken to be along Axis 2, controlled by the outer gimbal servo. The inner gimbal servo attached to the motor tube via pushrod connections, as did the outer gimbal servo to the inner gimbal mount. By using the difference in radii of the outer and inner mounts and the height of the inner and outer gimbals on the TVC mount, a maximum degree range of motion could be calculated. The values were as follows.

Table 1. Physical Range of Motion of Gimbal

TVC Mount Component	**Physical Maximum Range of Motion (deg)**
Inner Gimbal	5.72
Outer Gimbal	9.2066

Limits to motion for the inner gimbal resulted from the radial distance between the outer and inner mounts, while limits to the outer gimbal were a result of interference between the inner gimbal servo block and the outer gimbal's rotational motion. Nonetheless, the minimum of 5 degrees of motion was achieved.

For a lightweight and cost-effective design, PLA and PETG filaments were used to 3D print each component of the TVC mount. PETG was favored due to its higher heat deflection temperature, as the motor tube component of the mount would be in direct contact with the motor casing. However, due to material availability and only slightly lower heat deflection, some copies of the mount to be used in the final launches were made of PLA. To streamline the motor integration process, the team opted for heat-set threaded inserts on the outer gimbal, which would allow for the entire motor assembly to be mounted in the rocket via screws going through the body tube. Mounting holes are visible on the outer surface of the outer gimbal in Fig. [1].

B. Avionics

a. Hardware

The thrust-vectored rockets required an avionics system capable of taking in the necessary sensor information for state estimation, actuating the thrust-vectoring system, and deploying parachutes at the appropriate time.

The sensor suite for Gru and Vector included a MPU-6050 IMU, which contained an accelerometer (translational acceleration) and gyroscope (angular velocity), and a BMP 280 barometer (altitude). A moving average filter was implemented for the translational accelerations, taking the mean of the 20 previously recorded values as the system's true acceleration in any given direction. While this approach may be seen as introducing lag error to the system, where the reported acceleration is always behind the true acceleration, the high frequency of IMU readings largely assuaged this concern. Rocket attitude was performed using quaternions instead of Euler angles due to the issue of gimbal lock, where the system ceases to work if a Euler angle reaches 90 degrees [2-3]. Quaternion attitude kinematics updated our attitude using Eq. (1).

$$\dot{q} = \frac{1}{2}\begin{bmatrix} 0 & \omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} q \tag{1}$$

where each ω term represents a gyroscope reading at a given time step. The system's initial attitude was assumed to be pointing upward, a calibration based on leveling the rocket as the avionics system was turned on.

A Teensy microcontroller was used to read the sensors via SPI communication protocol, chosen for its speed in taking in sensor readings. A state estimate for attitude was formed via quaternion dynamics, and this state estimate was used to compute the error from perfectly vertical flight, as the control system's goal was active stabilization. The Teensy actuated two MG92B servo motors using pulse-width modulation (PWM) for setting the servo to hold a specific position. The Teensy also recorded state estimate information and control inputs to an SD card at every time step. All components were soldered into and connected via a protoboard.

A third-party, independent system was utilized for parachute deployment due to safety concerns: each thrust-vectored rocket was equipped with a TeleMega v5.0, a system with its own altimeter and GPS that detects apogee and fires parachutes [4]. All the TeleMega's operations were separate from the Teensy and utilized a separate power source; the TeleMega was also armed with a 9V battery for sparking the igniter attached to the recovery system.

b. Software and Control System

The flight software was implemented and compiled in C++ using the Arduino development environment. A procedural programming strategy was used in which PID controllers and rocket state were represented in structs, and the corresponding functions that manipulated them were organized in their respective headers. A state machine was used to control the activation of flight events. In the Armed state, the rocket calibrates the drifting sensor bias and detects launch through an acceleration spike and manually implemented timer. In the FastAscent state, the controller is activated and this is the only state where the control loop runs. In the SlowAscent state, the motor has burned out the system awaits detection of apogee. In the FreeFall state, it has detected apogee and is awaiting chute deployment by the external TeleMega. In the Landed state, it has detected landing.

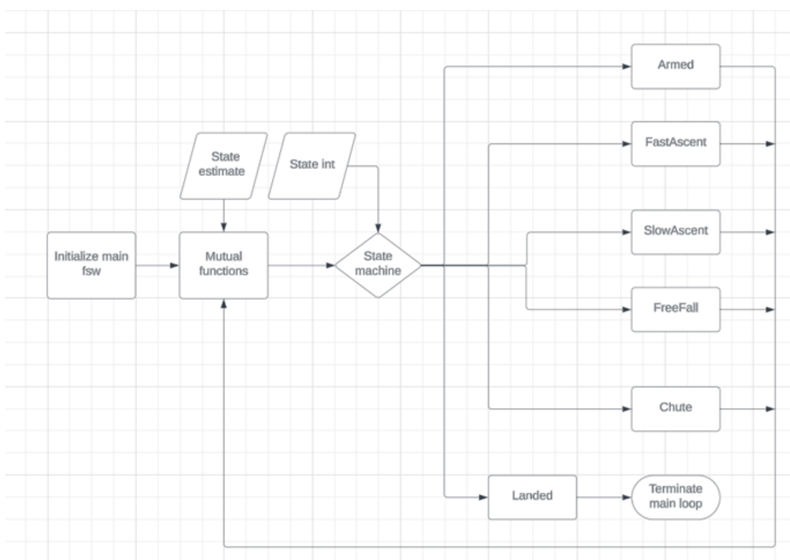


Figure 2: Flight Software Architecture

The control goal of these gimbaled rockets was active attitude stabilization, or maintaining zero pitch and yaw, which was achieved through implementation of a PID controller that took in pitch and yaw error and produced gimbal angle outputs. The gimbal angle was assumed to be equivalent to thrust angle. Since the mount was not a mechanically ideal system, assuming servo angle to equal gimbal angle can become highly inaccurate. Thus, an experimentally derived linear relation was implemented between the servo angle and gimbal angle. The rate of state estimation readings ran as fast as the control loop to ensure the most up to date reading was available at each time step. To ensure the safety and robustness of the controller, servo angle limits of 10 degrees were implemented in software, to match the mechanical actuation limits of the servo mount. The control gains were initially tuned using a custom 3DOF Simulink model of the rocket. However, due to potential deviations of the model from reality, the gains were further refined in a HWIL (hardware-in-the-loop) setting, in which mount responses were observed visually under different gain settings.

Control Loop:

1. Get state estimate (integrated rate gyro readings)
2. Calculate current time step duration
3. Calculate pitch servo angle output
4. Calculate yaw servo angle output
5. Translate servo angle to microseconds
6. Send actuation signal in microseconds

C. Avionics Bay

The main purpose of the avionics bay is to provide an integrated and easily accessible layout of all avionics sensors, hardware, and wiring while structurally constraining the components within the rocket. An overarching goal of the avionics bay was ease of maintenance of avionics components; this was executed by structuring the bay into three key distinct components: bulkheads, electronics tray, and electronics mounts.

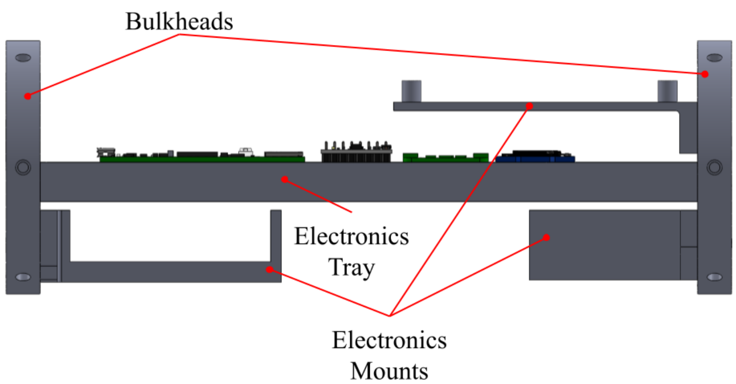


Figure 3: Avionics Bay Layout

Specifically, modularity was achieved via removable bulkheads on both ends of the core avionics vessel. This segmented design allowed us to fully integrate avionics hardware before the rocket body. Once positioned vertically within the rocket, the avionics bay is bolted radially through brass inserts - these heat-sunk inserts provide increased connection strength, ensuring the safety of avionics components.

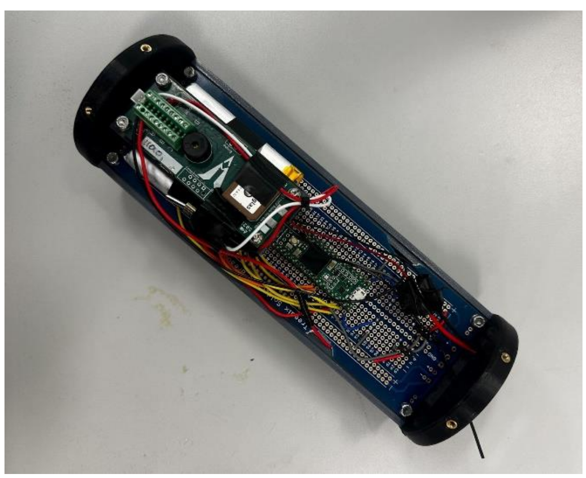


Figure 4: Fully Integrated Avionics Bay

D. Simulations

To obtain simulated data for the gimbaled-motor rockets, various simulations were done using OpenRocket, an opensource rocket simulator. In these simulations, various data such as, estimated apogee, maximum velocity, and the total flight time, were recorded, with inputs of rocket components and component weights. These simulations also allowed predicted the altitude, vertical velocity, and vertical acceleration for each rocket after launch. This data helped the team make design choices, such as weight distribution. The following figures show the results of the simulations.

Table 2. Simulated values of flight for both Gru and Vector. *The center of gravity and pressure are measured from the nose cone's tip.*

Rocket	Motor	Apogee (ft)	Maximum Velocity (ft/s)	Maximum Acceleration (ft/s^2)	Time to Apogee (s)	Flight Time (s)	Burnout Time (s)	Recovery Device Deployed Time (s)	Center of Gravity (in)	Center of Pressure (in)
Gru	G12ST-P	161	38.3	53.4	12.9	17.8	12.6	12.9	19.94	4.5
Vector	F10-2	163	37.3	55.1	7.19	12.3	7.13	7.23	18.37	4.5

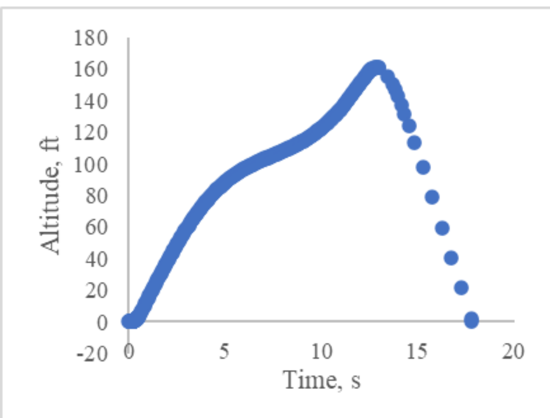


Figure 5. Altitude of Gru as a function of time.

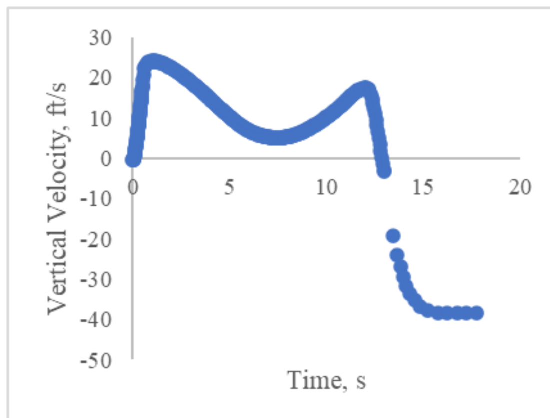


Figure 6. Vertical Velocity of Gru as a function of time.

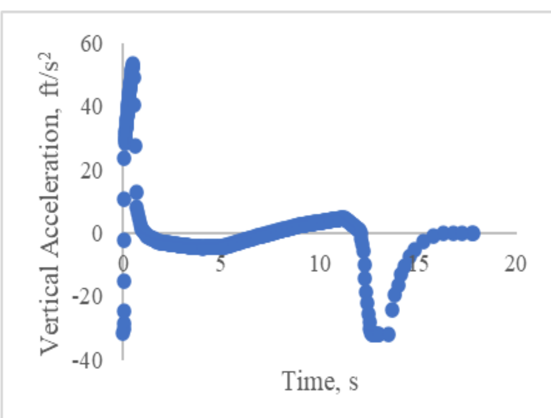


Figure 7. Vertical Acceleration of Gru over time

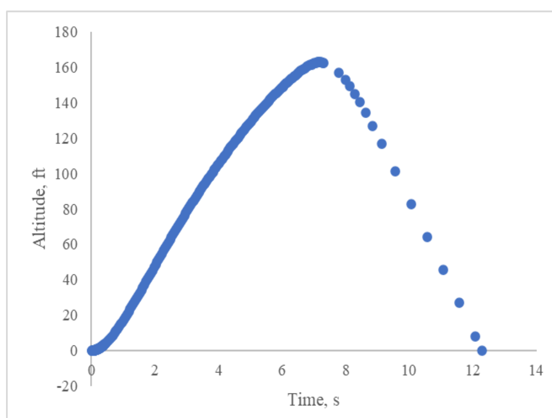
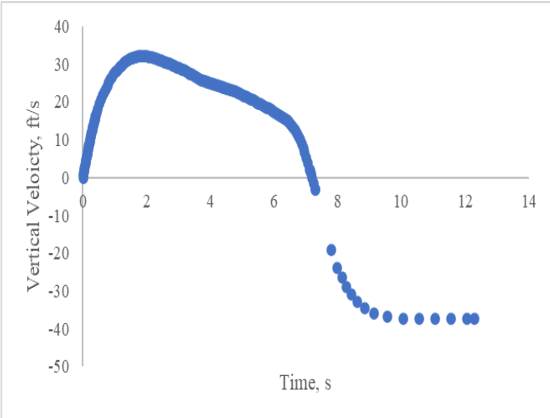


Figure 8. Altitude of Vector as a function of time.



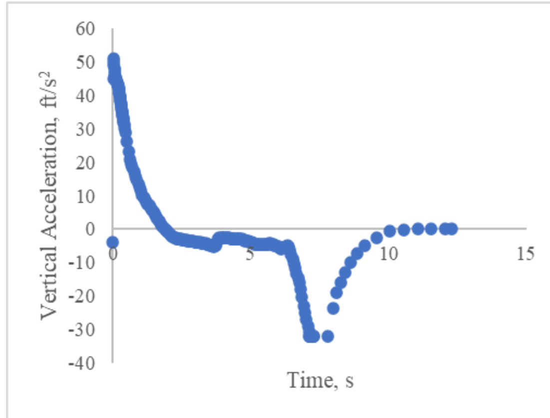


Figure 9. Vertical Velocity of Vector.

Figure 10. Vertical Acceleration of Vector.

The longer rocket, Gru, used a G12 motor with a longer motor burn time and the shorter rocket, Vector, used an F10 motor with a shorter burn time. Both motors were chosen for their flat thrust curves and longer burn times. The correlation between the simulations and flight test will be discussed in the Results and Discussions section.

E. Recovery

Following the detachment of the nose cone from the body tube, an 18" Printed Nylon Parachute was released. The selection of an 18" Printed Nylon Parachute was strategic, chosen to accommodate a 6lbs rocket, theorized to be the maximum weight threshold, while also fitting within the confines of the body tube and remaining cost-effective in terms of parachute material. The parachute was safeguarded against ignition from the black powder charge by utilizing a fire-resistant blanket specifically designed to protect parachutes, procured from Apogee Rockets. This blanket was strategically employed to encase the parachute within the body tube. Upon activation of the black powder charge and subsequent deployment of the parachute from the body tube, the fire-resistant blanket unfurled, allowing the parachute to fully deploy.

The recovery bulkhead was designed to hold black powder charges to release the parachute. The bulkhead would be screwed into the body tube and was easily removed and replaced to set up the recovery system for each launch. Wells were integrated into the bulkhead to hold the black powder, with holes at the bottom of these wells to feed the wires through the bottom of the bulkhead ensuring it would come into contact with the black powder. The full design was 3D printed multiple times to be used in the reoccurring launches. The wire was connected to the onboard TeleMega, which was programmed to fire off the charges at a specific time or altitude. There were two charge wells incorporated into the design to provide redundancy if one charge did not go off. A deployment test was done in order to test the amount of black powder needed to effectively pop off the nose cone and to validate the TeleMega system. During this test, the team filled 1.5g of black powder per well to test first. The 1.5g mass was only successful on 1 of two tests, and therefore a mass of around 2g per well was tested filling up the wells. The 2g mass of black powder tests demonstrated a successful nose cone separation. [4]

F. Testing

Gru and Vector were launched at the SEARS (Southeast Alabama Rocketry Society) launch site in Samson, Alabama. There were unfavorable wind conditions of wind gusts up to 20mph at the site, which were expected to affect performance and stability. Both rockets were expected to launch 3 times each, but due to time constraints and structural integrity issues, only 3 total launches were conducted. Gru launched 2 times, and Vector launched once. The most successful launch was Gru's 2^{nd}, which exhibited successful thrust vectoring for 2-3 seconds before losing momentum and falling to the ground. The rocket reached an apogee of just around 12 feet. The following flight data was collected by the TeleMega:

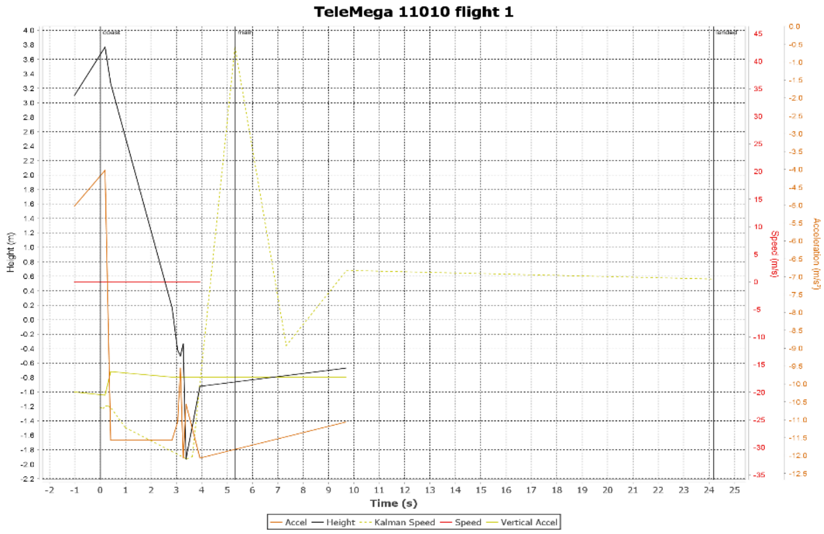


Figure 11. TeleMega Flight Data from the Second Launch of Gru

IV. Results and Discussion

The original goal of the Gru and Vector launches were to observe the capabilities of a motor gimbal to perform active stabilization. During the launch, several factors impacted the ability of the gimbal to perform as designed.

The first issue was the low thrust to weight ratio of both rockets. This value was much lower than expected, either due to errors in the simulations or the wind, causing the rocket to have great difficulty getting off the launch pad. After getting off the launch pad, the rocket was unable to fly very far up either. Based on the data from the TeleMega, the rocket Gru reached an altitude of only around 12 feet, which was significantly lower than the projected apogee of 161 feet. As a result, it was very difficult to observe long periods of thrust vectoring during the short period of time that the rocket was in the air. Had the rocket been able to reach a higher apogee, longer term thrust vectoring would have been observed, rather than just the two to three seconds observed in the second Gru launch.

Next, the strong winds made it much more difficult to thrust vector and had not been accounted for in simulations. Especially for such a light rocket, the 20+mph winds would have had a significant impact on the rocket's flight. This unprecedented condition significantly decreased the rocket's ability to fly as it was designed to. In addition to decreasing the apogee and shortening the flight time, the wind added sideways force upon the rocket. Consequently, it was more difficult for the motor gimbal to correct the attitude of the rocket, as it had to fight against an additional force to produce motion. Therefore, it was difficult to observe the true capabilities of the gimbal system with this external force.

Finally, during the initial correction, it appeared that the gimbal had overcorrected slightly, causing the rocket to tilt too far in the opposite direction. This could have been due to miscalibration of the servos or the wind conditions, which could have affected the rocket's motion as it was correcting its trajectory. As a result, the gimbal was unable to readjust itself in time, and with very little upward momentum, it fell to the ground.

Despite the difficult launch conditions, the GNC team considered the 2nd Gru launch a success as multiple angles of video evidence of the flight proved that attitude controls with the gimbal mount worked as intended.

(a) **(b)** **(c)**

Figure 12. Rocket Flight Over 0.5 Second Window Showing Correction in Vertical Orientation

Figure 12.a displays a deviation in the vertical position of Gru due to lack of stability. Over Figures 12.b and 12.c, the rocket is corrected to its original upwards position with the gimbal system. This correction was done several times over the course of the flight before the over correction that caused the rocket to go down.

Finally, due to the failure to fly for a substantial period of time, no observations could be made on the effects of the different motors and burn times on the ability to gimbal.

V. Conclusions

The Gru and Vector rockets designed by the Ramblin' Rocket Club at Georgia Tech were an attempt to design and test a gimbaled motor system. These were two medium-powered rockets, using a G12 motor and F10 motor respectively. The motor gimbal moved mechanically using MG92B servo motors connected to a 3D printed motor mount and actuated by a Teensy. Various sensors determined deviations from the rocket's vertical position so the servo motors could adjust accordingly using a state estimation model. The flight was simulated on OpenRocket software prior to launch to predict launch results.

Many unforeseen circumstances were encountered during the rocket launch that caused discrepancies between simulated and observed results of the launch. One of the main issues was a lower-than-expected thrust to weight ratio. This caused the rockets to be unable to reach a high enough vertical acceleration to reach a substantial apogee or have enough vertical momentum to maintain upwards flight, making it difficult to observe the active stabilization. Consequently, only one of the launches yielded successful thrust vectoring.

To produce more successful thrust vectoring in the future, possible considerations could include eliminating elements to reduce weight, selecting a different motor, and accounting for a higher margin of error when analyzing simulations.

References

[1] BPS.space, "Thrust Vectoring Mount - Build Signal R2", YouTube, Accessed Sept. 14, 2023. https://www.youtube.com/watch?v=mA7TwcemOh0&list=PLyV774-3p835n5XZFz4JMvF-Griy_uOdV&index=4&ab_channel=BPS.space

[2] Kisabo, A.B., Adebimpe, A.F., Okwo, O.C., Samuel, S.O., "State-Space Modelling of a Rocket for Optimal Control System Design", Journal of Aircraft and Spacecraft Technology, 2019, pp. 128-127. https://thescipub.com/pdf/jastsp.2019.128.137.pdf

[3] Santos, P.D., Oliveira, P., “ADCS Design for a Sounding Rocket with Thrust Vectoring”, Aerotecnica Missili & Spazio, 2023.
https://www.researchgate.net/publication/372228268_ADCS_Design_for_a_Sounding_Rocket_with_Thrust_Vectoring

[4] BPS.space, “Parachute Ejection System - Build Signal R2”, YouTube, Accessed Sept. 28, 2023.
https://www.youtube.com/watch?v=GaGgLGDbEq4&list=PLyV774-3p835n5XZFz4JMvF-Griy_uOdV&index=5&ab_channel=BPS.space