33RD CONGRESS
OF THE INTERNATIONAL COUNCIL
OF THE AERONAUTICAL SCIENCES
STOCKHOLM, SWEDEN, 4-9 SEPTEMBER, 2022

ICAS
2022
SWEDEN

CRITICAL REVIEW OF SOUNDING ROCKET FLIGHT DATA AND COMPUTATIONAL FLUID DYNAMICS EFFORTS

Bjorn Kierulf[1]

[1]Northeastern University, Undergraduate

Abstract

Data from four instrumented rocket flights is analyzed and compared to results from Computational Fluid Dynamics (CFD) calculations. Sensor data from multiple boards is aligned and incorporated into a Kalman Filter (KF). All rockets had pitot tube systems in their nosecones to measure velocity. The pitot tube's velocity is compared to the derivative of altitude and the velocity returned by a Kalman Filter, and found to disagree with both at certain points. An Extended Kalman Filter (EKF) was explored to compensate for the rocket's tilt angle, but limited gyroscope data meant this did not resolve the discrepancies. Many possible sources of the deviations are examined, but none wholly explain the observed behavior. Drag coefficients (CD) based on the direct measurements (pitot tube and accelerometer), the EKF filter values, the pitot tube alone, and the derivative of altitude are compared. The CD's using the direct measurements and the EKF values agree as expected, and for two flights the pitot tube alone agrees closely as well. The altitude derived drag coefficients are generally higher than the others, because of underestimation of the velocity.

Computational Fluid Dynamics (CFD) calculations are performed for two flights using a $\frac{1}{3}$ symmetry domain, mesh refinement regions, the $k-\omega$ SST turbulence model, and a pseudo-transient solver. Both cases are demonstrated to be mesh-independent. For one flight, the CFD drag coefficient was substantially off (17%) from the calculated CD, however this flight has the largest discrepancies between data sources. The second flight's CFD drag coefficient was very close (.7%) to the calculated CD, and this flight's data was much more internally consistent. In light of the presented data, trends, and discrepancies, the CFD cases appear to be within the margin of error of the experimental data.

Keywords: CFD, Kalman Filter, Pitot tube, Sounding rocketry, Calibration procedures

1. Introduction

The author is a member of Project Redshift within the Aerospace club at Northeastern (AeroNU). Project Redshift is a student-led team whose long-term goal is to launch a liquid fueled rocket to fulfill the Dollar-per-Foot (DPF) challenge hosted by the Friends of Amateur Rocketry (FAR). The project has launched several incremental test vehicles with integrated measurements systems, mainly a pitot tube aimed at measuring velocity during flight. This paper presents a detailed review of four of flights of instrumented rockets containing multiple sensors and boards. It presents an overview of the data analysis methods employed, pitot tube readings and two different filtering algorithms, and their results for each flight.

1.1 Sounding Rockets and CFD

Existing work in this area includes comparison of CFD predictions with experimental data from a wind tunnel for the case of a supersonic sounding rocket. In his thesis, Hammargren looks at the variation of C_D, $C_{N\alpha}$, and X_{cp} over a range of Mach numbers, for multiple rockets. In general the CFD results predict the value of coefficients within 10% when compared with experimental data, with some disparities. The CFD results disagree with experimental data on certain trends with respect to Mach number, such as the drag divergence Mach number, and variation of coefficients in the transonic region [13]. Northeastern University does not have a wind tunnel that is large or fast enough to emulate flight conditions, so Project Redshift employs sounding rockets as an alternative means of gathering experimental data.

A NASA contractor report from 1971 provides background on meteorological sounding systems, where a rocket would deliver a payload of a falling sphere to the high atmosphere. Data collected by the falling sphere and by radar tracking could be used to calculate atmospheric density and temperature profiles as a function of altitude. This report gives an overview of the different sources of error for a pitot-static system developed by Sandia National Laboratories. Time lag, angle of attack, and the gauge itself had uncertainties on the order of 2% under 70 kilometers. Above 70 kilometers, gauge uncertainty was 8% and low-density and hypersonic flow corrections were estimated at 10-15% uncertainty. The report presents the design of a low cost pitot probe for measuring density between 50 and 100 kilometers. One key difference between that design and Project Redshift's pitot tube systems is the size of the orifice. Bollermann suggests a 0.7 inch diameter orifice pitot tube with a 3.5 inch diameter hemisphere surrounding - this is to minimize angle-of-attack and rarefied gas effects. The report also descrivbes a gamma-ray-backscatter technique for measuring the atmospheric density. This sensor is present in the Nike-Apache rocket, and is insensitive to shock layers present on the rocket forebody [7].

One example of a similar collegiate rocketry team also employing pitot tube measurement systems is Laval University's GAUL project, and their High V rocket. This rocket includes pitot tube, payload, CO2 deployment, and composite fin subsystems, which are documented in the 2018 technical project report [8]. It is not clear from available resources whether this rocket has successfully collected data from it's pitot tube system.

All of the computational fluid dynamics calculations in this paper use the Finite Volume Method, a discrete version of the integral form of the Navier-Stokes equations. Basics of the finite volume method are described by Jasak in his Ph.D. Thesis, including topics such as causes of error, discretisation, Reynolds averaging in turbulence models, and more [14]. A more detailed explanation of the numerical methods used in this paper can be found in the Ansys Fluent Theory Guide [4].

1.2 Filtering Algorithms

This paper examines flight data from instrumented rockets and uses filtering algorithms as a method to evaluate the performance of pitot tubes. The fundamental principle behind a Kalman Filter is the weighted fusion of a system model with real measurements. This lessens the effect of sensor noise, while states are ultimately grounded to measured values. The simplest version of the filter uses a linear system model, and operates in two steps - prediction and update. The prediction step propagates the current state forward in time using the system dynamics model. The update step corrects the predicted state values by giving some weight to measurements. The relative weight given to the prediction vs the measurements is dependent on the relative uncertainty of each, tracked using sensor and process covariance matrices [12].

All of these operations can be expressed as matrix algebra, as shown in the equations below. In this notation, x is the state vector and z is the measurement vector. F is the system dynamics matrix, and B is a matrix that maps the effect of the control input u on the state. P is the error covariance of the state. R and Q are covariance matrices for the measurement and the process, respectively. The measurement residual is $\tilde{y}$, and H maps the states to the measurement vector. K is the Kalman gain, which weights and maps incorporation of the measurements into the state. In this notation, k is the current time step, '-' indicates a predicted estimate and '+' indicates an updated estimate. A superscript of T represents the transpose, and I represents the identity matrix.

$$x_k^- = F x_{k-1}^+ + B u_{k-1} \tag{1}$$

$$P_k^- = F P_{k-1}^+ F^T + Q \tag{2}$$

$$\tilde{y}_k = z_k - H x_k^- \tag{3}$$

$$K_k = P_k^- H^T (R + H P_k^- H^T)^{-1} \tag{4}$$

$$x_k^+ = x_k^- + K_k \tilde{y}_k \tag{5}$$

$$P_k^+ = (I - K_k H) P_k^- \tag{6}$$

Equations 1-2 describe the prediction step, and 3-6 describe the update step. These equations are taken from a chapter of *Introduction and Implementations of the Kalman Filter* [12].

2. Flight Data Analysis Methods

2.1 Rockets Flown

Two rockets were flown to gather data for this analysis. The first is Travel Request Denied (TRD), which was flown in August of 2021 and April of 2022. The second is 'Carby,' name derived from it's carbon fiber body. Carby was flown in September and November of 2021. Though the rockets were generally the same between flights, the exact geometry was not. Carby flew with two different sets of fins on its two flights, and the TRD April flight included a large protrusion. Additionally, electronics flown differed between flights.

2.2 Electronic Boards Flown

Several electronic boards were flown, each of which having different sensors. The main board is the Redshift Data Acquisition board (DAQ) v2, for which credit goes to Ben Helfrich and Johnny Cao, among other members of the avionics team. This board has an MS5607 barometer, a nine-degrees-of-freedom inertial measurement unit (IMU), three 24 bit analog to digital converter (ADC) channels, and four thermocouple reader channels. The thermocouples were intended to measure heating of the nosecone tip during a supersonic flight. This board logs at over 100 Hz, typically around 110Hz. All of the flights in this paper were equipped with a pitot tube to measure stagnation pressure at the nosecone tip, which separates them from many other Project Redshift launches in the past two years. A pressure transducer read the pitot tube's pressure and output an analog signal, logged by the DAQ board. A second pressure transducer measured static pressure, also logged by the DAQ board.

Easyminis are a commercial altimeter board made by Altus Metrum. Project Redshift typically uses a redundant pair of Easyminis to control a rocket's parachute deployment. These boards have an MS5607 barometer, which measures atmospheric pressure. These boards are kept in a rocket's electronics bay, which has several small holes to allow the pressure inside the bay to equalize with the pressure outside, as the rocket goes up. Easyminis log at 100 Hz during ascent. Telemetrums are very similar to Easyminis, with the addition of a single-direction accelerometer, and GPS and radio capabilities. These were flown on some flights in order to ensure the rocket could be located.

Another in-house Printed Control Board (PCB) from Project Redshift is the Flight Control Board (FCB), which was flown on some of the flights. This board has two barometers, two nine-degrees-of-freedom IMU's, GPS and radio, among other functionality. The measurements from this board can be used as a backup and a sanity check for other readings.

One final board was flown, called the Haaky-DAQ. This was a more makeshift version of the above DAQ board, with similar sensors. The recent electronics chip shortage meant that making more Redshift DAQ's wasn't possible, and so this board was a workaround. Due to the commercial breakout-boards used for sensors, it logs at a slower frequency, typically 15-20 Hz.

2.2.1 Temporal Alignment Between Boards

Each board tracks time precisely, but those times need to be matched at one point in order to be comparable between boards. Temporal alignment, or lack thereof, has the potential to be a defining error when combining data between boards. The moment the rocket launches is the most distinctive feature of most flights, so that is a good place to align times. The change in pressure over the first tenth of a second of launch is small and sometimes within the noise of the barometer. When boards have accelerometers, the start of launch is more distinctive - however it is not necessarily the same as the barometer's detected launch. Over four flights this difference was no more than one or two hundredths of a second.

2.3 Weather Data and Atmospheric Model

Weather data was used for calibration, and as a third known thermodynamic property. All of the rockets for this analysis were launched at the Champlain Region Model Rocketry Club launch site in St. Albans, Vermont. Weather data was taken from `timeanddate.com` for St. Albans Bay State Park [11]. This data was compared with reported weather data from the Plattsburgh International

Airport [6]. The pre-launch average barometer pressure consistently agreed closer with the weather data from St. Albans Bay State Park, with an average offset of -207 Pascals compared to +579 Pascals using the Plattsburgh International Airport data. Therefore, the `timeanddate.com` weather data was used for all flights.

Barometric pressure, temperature, and relative humidity at ground level were found for each launch, and fed into a simple atmospheric model. The temperature at higher altitudes was calculated using equations from a NASA atmospheric model [1]. This same model was used to convert between pressure and altitude. Based on temperature, the saturation pressure of water can be found using the steam tables, data taken from *Fundamentals of Engineering Thermodynamics* [19]. The partial pressure of water vapor can be found using equation 7 , where P_v is vapor pressure of water, ω is relative humidity, and P_s is the saturation pressure.

$$P_v = \omega * P_s \tag{7}$$

The partial pressure of water vapor was assumed to decay with altitude at the same rate as total air pressure. This isn't necessarily a good assumption, but lacking higher fidelity weather data it suffices. The partial pressure of air can be found by taking the difference between absolute pressure and water vapor pressure. The Dalton model can be used in conjunction with the ideal gas equation of state, Eq. 8, to find the total fluid density. ρ refers to density, P to pressure, M to the equivalent molar mass of air, R to the universal gas constant and T to temperature. Accurate density is necessary to calculate the rocket's drag coefficient.

$$\rho = \frac{PM}{RT} \tag{8}$$

2.4 Sensors and Calibration Procedures

Most sensors were not calibrated before flight, so in-situ offsets were important. All gyroscopes were zeroed during an averaging period while the rocket was stationary on the pad, to eliminate the zero-bias errors common in Micro-Electromechanical System (MEMS) gyroscopes. The FCB's gyroscope had a factor of four gain error due to a bug in the offloader code. This has since been fixed, but all FCB flights prior to March 15th, 2022 were adjusted for this factor.

2.4.1 Pressures

The MS5607 barometers were offset such that a pre-launch average value would match the barometric pressure given by weather data. The pressure transducers used for the pitot tube and static port are an interesting case - the sensors themselves are well calibrated, but the ADC on the DAQ has a substantial gain error. Calibration with sensors connected to the DAQ revealed a 5% gain error, consistent between two different pressure transducers. This issue was traced to an ambiguous component datasheet. This error is evident in the pre-launch values of static and stagnation pressure in the August and November flights. The pressures over the rocket's ascent are plotted in figure 1, as measured by different sources for the Carby November flight.

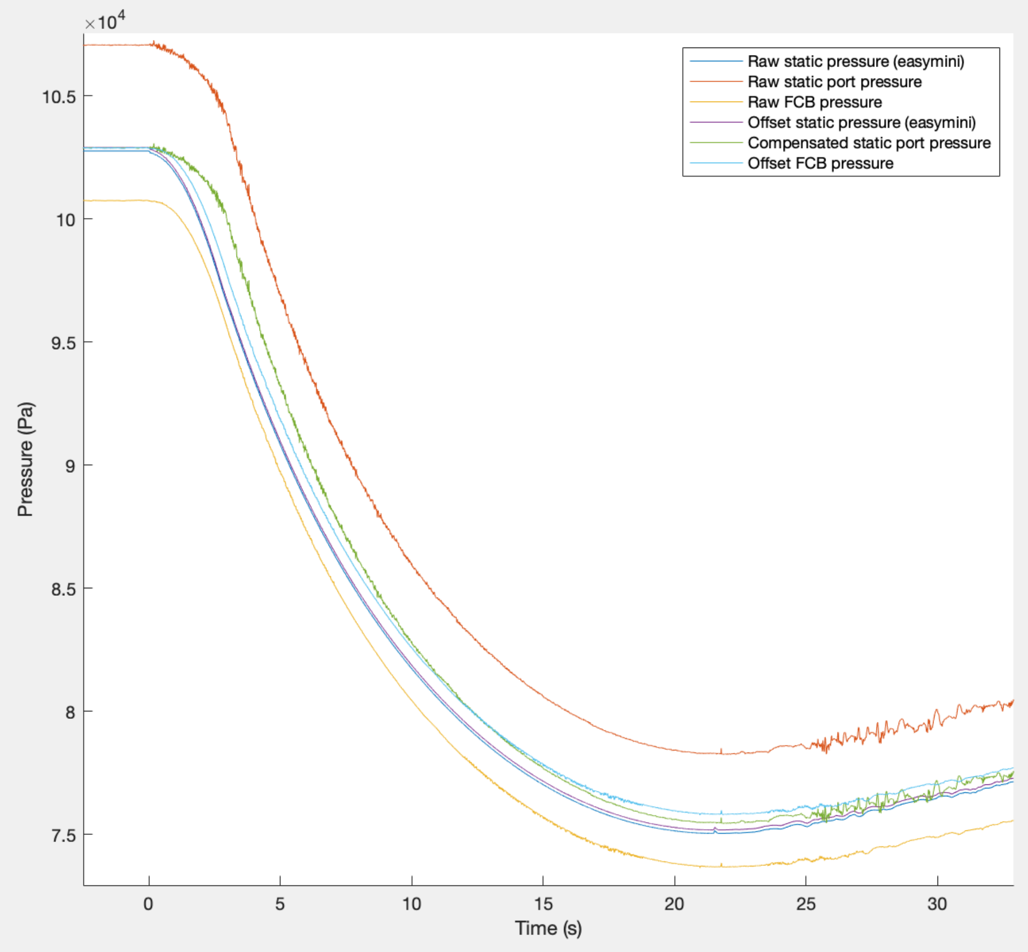

Figure 1 – Different pressure readings during ascent, Carby November 2021

The September Carby flight is slightly different, in that the pre-launch gain errors for static and stagnation pressure are 1.0327 and 1.0410 respectively. This is substantially different from the 5% error found in calibration. One sanity check for pressure calibration is looking at the static and stagnation pressure readings at apogee. Because the velocity is low, and stagnation pressure scales with velocity squared, even if the rocket has some velocity at apogee, the stagnation pressure should be close to the static pressure. The pressures at apogee match much closer for the September Carby flight when the pressures are scaled by the average pre-launch gain error instead of 5%, with differences of 100-150 Pascals instead of 1400-2100 Pascals. Therefore this in-situ calculated gain error was used to compensate the data for this flight, rather than the standard 5%. For the other flights, the in-situ calculated gain errors are close to 5% and the pressures match well at apogee. Calibration curves for the two pressure transducers while connected to the DAQ are included in Appendix A, figures 23 and 24. A calibration curve for an MS5607 barometer is also included, figure 25.

An additional takeaway from figure 1 is the delays of different pressure measurements. The static port, measured by a pressure transducer, is among the slowest due to its small diameter and large volume. The DAQ's pressure readings also lag behind, because the DAQ was in the relatively sealed nosecone. The Easymini's equalize the fastest, as might be expected from being in a tube with several holes.

2.4.2 Accelerometer

Accelerometers were not aclibrated before flight. For some flights there are differences between pre-launch readings and known gravity. This leaves a couple options to correct the pre-launch acceleration. The first option is a simple offset so that the y direction (upwards) reads the magnitude of gravity. However, this assumes that the rocket is perfectly vertical before launch, when in reality it is often tilted. This method would add an offset even to a calibrated sensor, if it were off of vertical at an angle - thus in most cases, this method would overestimate the required offset. Since the rocket is likely within ten degrees of vertical before launch, the small angle assumption holds, which reduces the issues with the simple offset method. An alternative method is to scale the accelerometer's pre-launch readings so they equal the magnitude of gravity, but there is not enough evidence to suggest a gain error.

Looking at the magnitudes of the pre-launch accelerometer readings across multiple flights, the total magnitude is not significantly closer to known gravity than the vertical-axis reading. Based on this, and in the absence of full IMU calibration, the simple offset in the vertical axis was applied for all flights. One potential source of error is the possible difference in orientation between the inertial measurement unit on a PCB and the rocket's axes. This is thought to be minor. Figure 26 in Appendix A shows the correlation between axial acceleration measured by the FCB and by the DAQ board. This plot is limited to the coast region of the November Carby flight. These are in strong agreement (0.38% different), which is evidence against a gain error in either sensor.

2.4.3 Magnetometer

Magnetometers measure the local magnetic flux vector in three directions, and can be used to ground attitude estimates and compensate for drift in integrating gyroscopes. However, magnetometers are very susceptible to two types of error - hard iron and soft iron. These are caused by offsets in the sensor itself, metallic objects and magnetic field sources near the sensor. The standard calibration procedure is to rotate the sensor in all directions. The measurements can then be offset and stretched to make them a sphere with zero origin. This calibration procedure was not been performed before any of AeroNU's flights. However, during descent under parachute, rockets typically exhibit chaotic motion. The raw magnetometer readings during ascent and descent are plotted in figure 2 for the November 2021 Carby flight. While these readings include some noise, they do provide good coverage over the ellipsoid necessary for calibration.

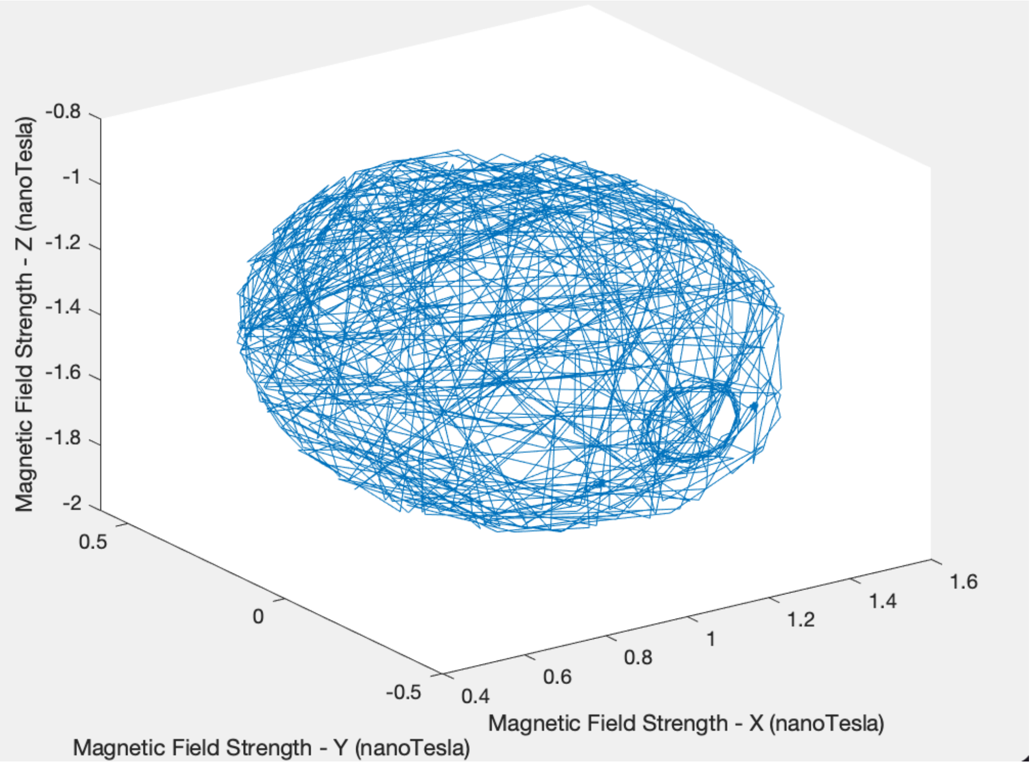

Figure 2 – Raw magnetometer readings during ascent and descent, Carby November 2021

This data was calibrated using the magcal() function from MatLab, which generates an offset vector and gain matrix in order to best fit the input magnetometer data to a sphere [17]. Calibrated magnetometer readings for the FCB and DAQ are plotted in figure 3 below, during the ascent of the November Carby flight. The two disagree, despite showing similar trends. This is possibly due to poor calibration of the FCB, because it's raw magnetometer readings cover much less of the required ellipsoid than the DAQ's readings. Calibrated magnetometer data can be used for angular sensor fusion. Results from algorithms incorporating magnetometer data are shown in section 2.7

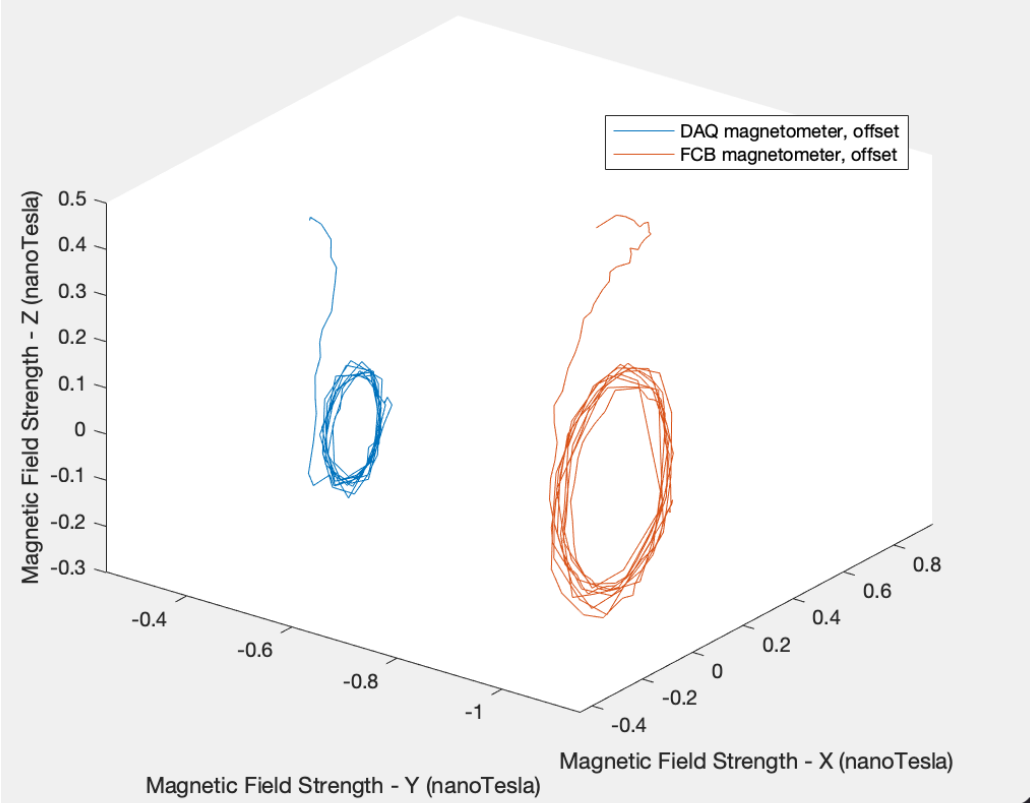

Figure 3 – Calibrated FCB and DAQ magnetometer readings during ascent, Carby November 2021

2.5 Coast Phase Analysis

This section will describe the theory behind pitot tube data analysis, and why it is primarily centered around one phase of a flight. First, an overview of the phases of a rocket's flight: motor burn, coast, and descent.

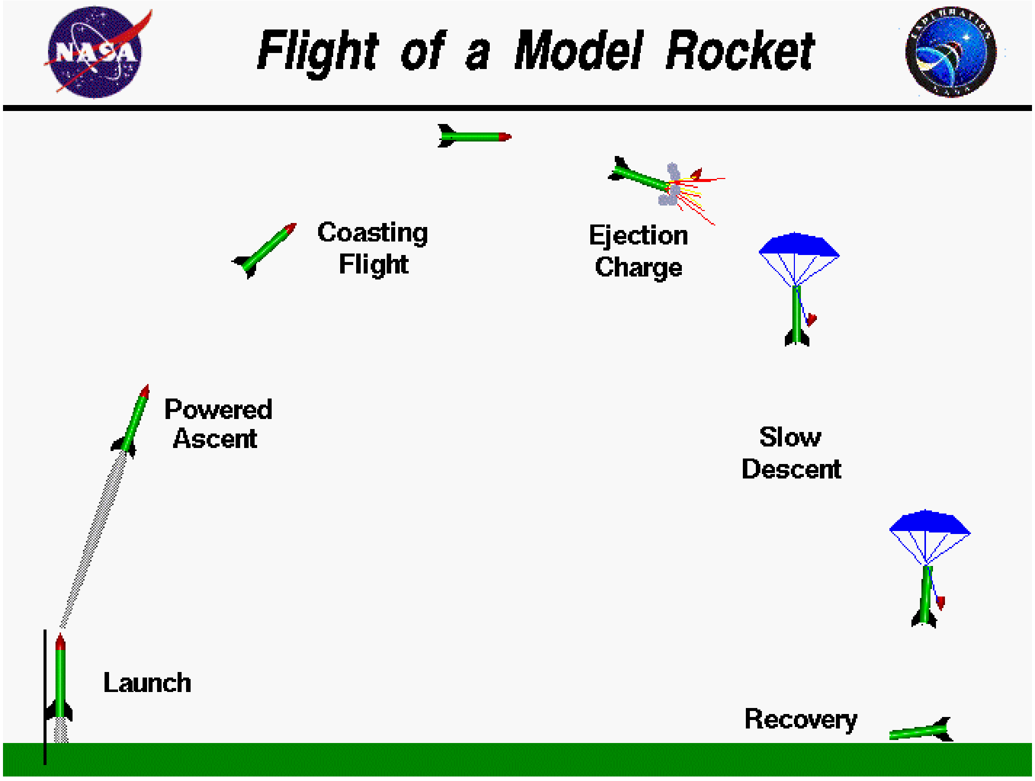

Figure 4 – Rocket Flight Overview

Figure 4 above was taken from NASA with permission from their media guidelines [2]. The powered ascent phase is when the motor is propelling the rocket upward. The descent phase is when the rocket is falling under a parachute. The coast phase is the intermediate time, after motor burn out but before parachute deployment. This is an ideal time to collect data on the rocket's aerodynamics, because aerodynamic forces and gravity are the only forces acting on the rocket. Only the aerodynamic forces are measured by the onboard IMU, because the IMU itself is accelerating due to gravity.

2.5.1 Mass

If a rocket's mass is known, Newton's second law $F = ma$ can be used to calculate the aerodynamic force. Each rocket is weighed before flight as part of a safety check, so the wet mass is known. Calculating the rocket's dry mass requires the propellant mass. This can be taken from websites that give the manufacturer's specification, such as Chris's Supplies [9]. In one case, the propellant mass for a rocket was taken from the motor database of an open source rocket design software, OpenRocket [20]. The dry mass number that is critical in the coast phase analysis has more error than one may expect, due to data relying on manufacturer-provided information, and because the motor's smoke grain burns during coast. Both acceleration and force were assumed to be axial along the rocket; off-axial forces were neglected in this analysis. The velocity over time is necessary to make the drag force over time meaningful.

2.5.2 Pitot Tube Aerodynamics

The velocity can be calculated using either of two equations, depending on whether or not the fluid is assumed to be incompressible. The compressible relation was used, since it holds for incompressible flow as well. These equations are taken from Anderson's *Fundamentals of Aerodynamics* [15]. In these equations P_t is total pressure measured by the pitot tube, P_s is the static pressure, γ is the specific heat ratio, and M_1 is the free-stream Mach number.

$$v = \sqrt{\frac{2(P_t - P_s)}{\rho}} \tag{9}$$

$$M_1^2 = \frac{2}{\gamma - 1}[\frac{P_t}{P_s} - 1]^{\frac{\gamma-1}{\gamma}} \tag{10}$$

The local speed of sound, a, can be calculated by using Eq. 11 below, where R is the specific gas constant, and T is the temperature in Kelvin. This can be combined with the compressible pitot tube equation and $v = Ma$ to calculate the rocket's velocity.

$$a = \sqrt{\gamma RT} \tag{11}$$

With drag force and velocity known, the instantaneous drag coefficient (CD) can be calculated via equation 12. A_ref is the reference area, taken as a rocket's cross sectional area.

$$F_D = \frac{1}{2}\rho A_{ref} C_D v^2 \tag{12}$$

2.6 Filtering

Flight data was fed into two different filters in order to evaluate the pitot tube. The first was a simple two-state Kalman Filter (KF), the second being a three-state Extended Kalman Filter (EKF). An overview of the general Kalman Filter algorithm is given in the introduction — this section will describe the specific filters used here. The notation is consistent with the introduction, and the conventions of *Introduction to Kalman Filter and Its Applications*.

2.6.1 Kalman Filter

The basic Kalman Filter is one dimensional and has states of altitude and vertical velocity. It only measures the altitude, and it uses the measured acceleration offset by gravity as a control input. The measurement vector is z and the state prediction dynamics matrix is F. The barometer-measured altitude is y_b. This is laid out in the equations below.

$$x = \begin{bmatrix} y \\ v \end{bmatrix} \tag{13}$$

$$u = \begin{bmatrix} a \end{bmatrix} \tag{14}$$

$$z = \begin{bmatrix} y_b \end{bmatrix} \tag{15}$$

$$F = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \tag{16}$$

$$B = \begin{bmatrix} 0 \\ dt \end{bmatrix} \tag{17}$$

The B matrix describes how the control input affects the states during the prediction step. The acceleration affects the velocity by a factor of dt. One variation of this filter includes a $\frac{1}{2}dt^2$ for the acceleration directly affecting altitude. This does make this aspect of the filter second order accurate as it simplifies to using a second order Runge-Kutta scheme. However other parts of the filter are first order accurate. It is not generally helpful to have mixed orders of accuracy as the lowest order will determine the overall accuracy. Comparing the filter's results with and without this term shows little difference, as expected for a small (hundredth of a second) time step. If the data was much lower frequency, a higher order time stepping scheme may be more important.

The equations for the standard operation of a standard Kalman Filter in the prediction and update steps are taken from a chapter in *Introduction and Implementations of the Kalman Filter* [12]. The measurement covariance was taken from the sensor datasheet, and process covariances were chosen somewhat arbitrarily. The performance of the filter was shown to be relatively insensitive to these values.

This filter works well, but relies on assuming a small angle between the rocket and vertical, and assuming good accuracy of the accelerometer. Because the accelerometer readings are fed in directly as the control input, their uncertainty is not included in the filter's error estimates. The other issue with this filter is that it only models velocity, so it's not necessarily accurate as a way to calculate velocity. This is seen as expected in the filter's covariance matrix having a large term for the velocity uncertainty. This filter is valid for all phases of the rocket's flight, from before launch to after landing. Plots of altitude, velocity, acceleration, and altitude measurement residual for this filter are shown in figures 5 and 6 below. The details of this filter's results, including larger plots and comparison to pitot tube data will be discussed in section 3.6One takeaway from this filter is that the assumption of being near vertical may be poor, especially as the rocket approaches apogee. This was the motivation behind the Extended Kalman Filter — to incorporate a tilt angle.

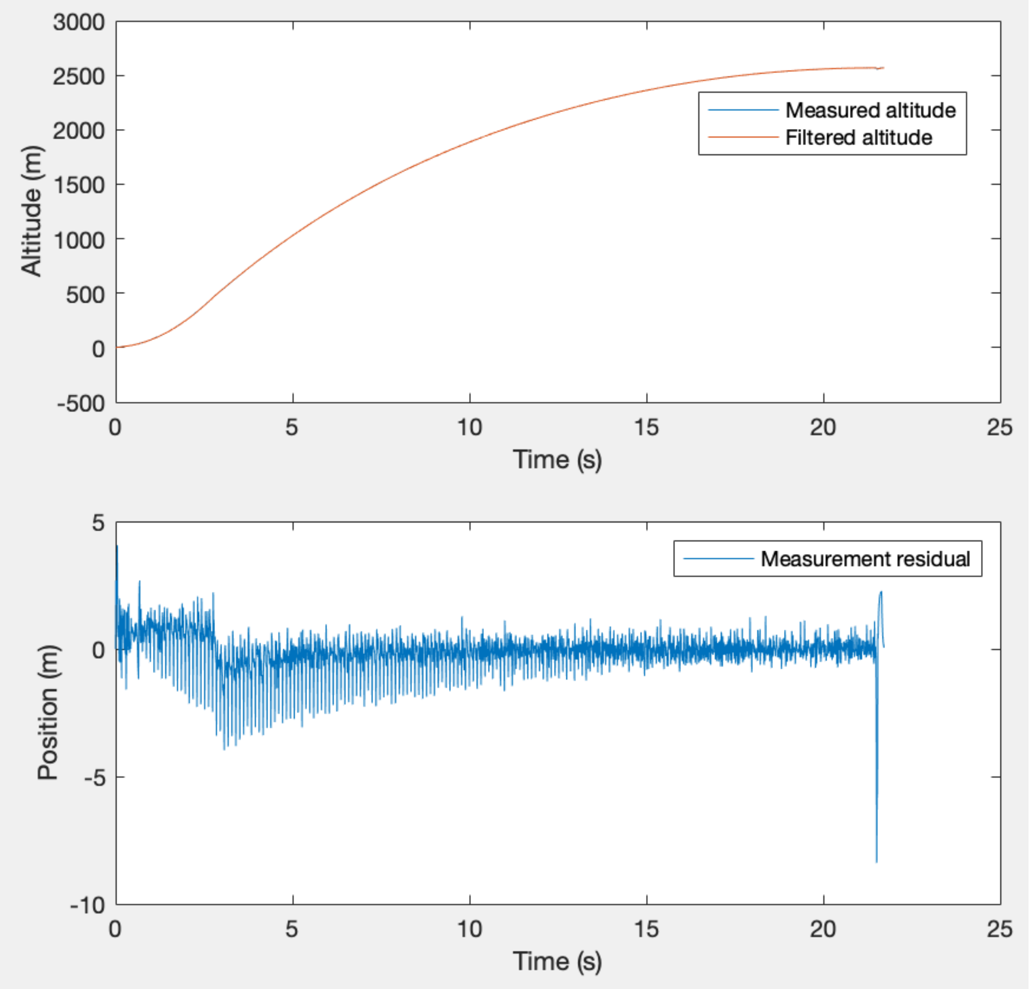

Figure 5 – Kalman Filter altitude during ascent, Carby November 2021

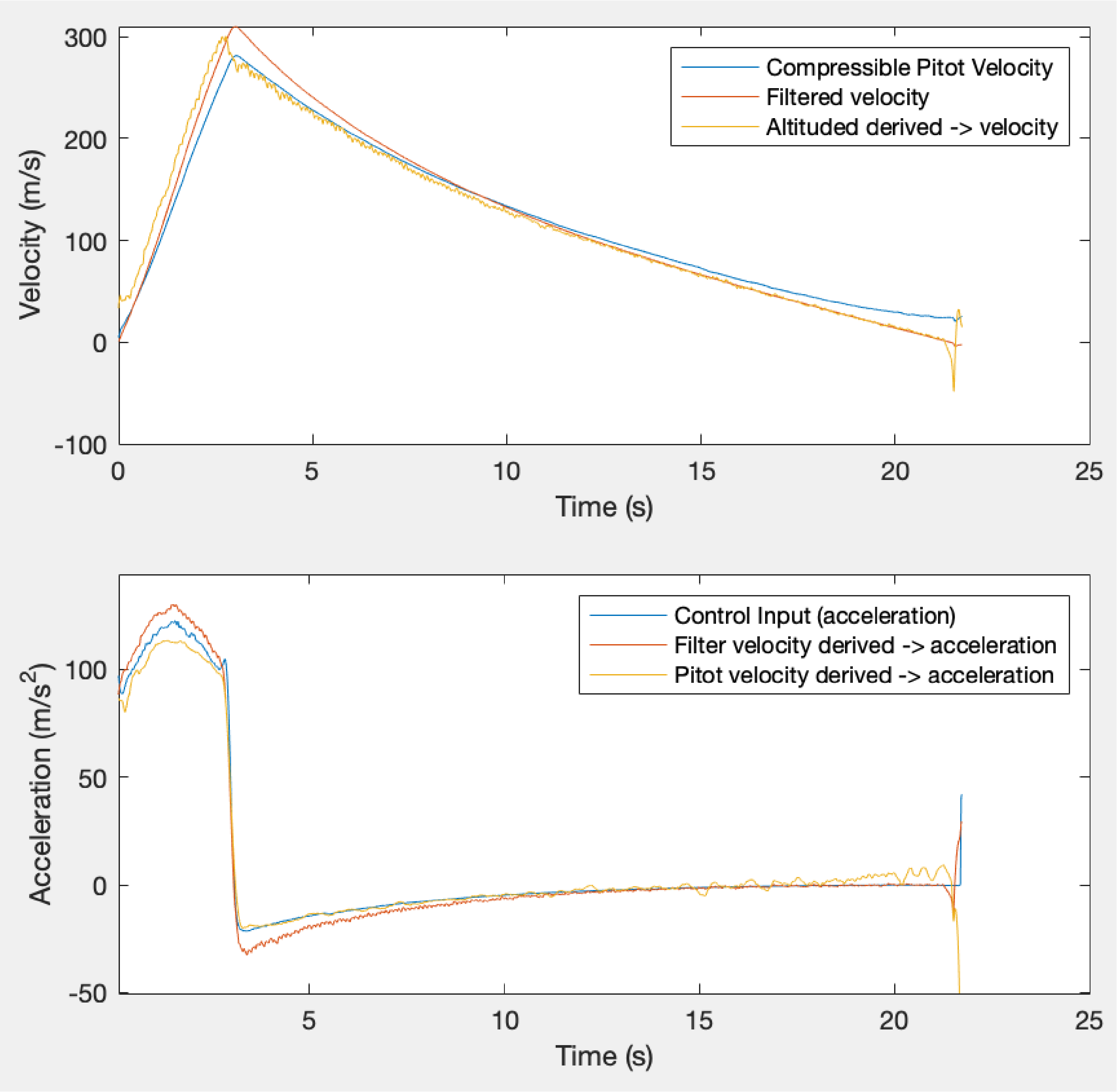

Figure 6 – Kalman Filter velocity and acceleration during ascent, Carby November 2021

2.6.2 Extended Kalman Filter

The Extended Kalman Filter eliminates assumptions with regards to angle by incorporating a tilt angle. For the purpose of simplicity and because of the method chosen, this angle is computed separately, outside of the filter, as described in section 2.7 This filter has states for altitude, vertical velocity, and vertical acceleration. It measures altitude from a barometer, velocity from the pitot tube and acceleration from the IMU. Modeling acceleration as a state variable predicted by the velocity and drag coefficient allows the IMU's acceleration to be used as a measurement instead of as a control input. This means that the uncertainty in that measurement can be tracked, instead of assumed accurate. This method does require that the filter be given a drag coefficient as input.

All states are defined as being vertical, and positive upward. Equations converting between the filter's states and corresponding quantities along the rocket's axis are shown below, where v_a and a_a are rocket-aligned. α is the rocket's angle off of vertical. These same equations can be applied to the Kalman filter, to transform its states into rocket aligned quantities. This is done when comparing the two filters and when using the Kalman filter's results.

$$v_a = \frac{v}{cos(\alpha)} \tag{18}$$

$$a_a = \frac{a}{cos(\alpha)} \tag{19}$$

The discretized system update equations incorporating the tilt angle are shown below. The acceleration prediction is done using the drag equation 12. Density at each time step is based on the current altitude / pressure, using the simple atmosphere model. The drag force is axial and relies on axial velocity. Plugging in and simplifying to find the vertical component of acceleration using equations 18 and 19 above yields the equation below. This filter has no control inputs and includes gravity in the nonlinear state update function.

$$y_{k+1} = y_k + v_k * dt \tag{20}$$

$$v_{k+1} = v_k + a_k * dt \tag{21}$$

$$a_{k+1} = -\frac{\frac{1}{2} * \rho * A_{ref} * C_D * v_k^2}{\cos(\alpha_k)} - g \tag{22}$$

The measurement vector is shown below, where y_b is the barometer-measured altitude, v_p is the pitot-tube-measured velocity, and a_{IMU} is the axial acceleration measured by the IMU. With the pitch angle incorporated to transform the measurements to the filter's vertical reference frame, the matrix mapping a state to the predicted measurements becomes the identity matrix.

$$z = \begin{bmatrix} y_b \\ v_p * cos(\alpha) \\ a_{IMU} * cos(\alpha) - g \end{bmatrix} \tag{23}$$

The Extended Kalman Filter algorithm requires Jacobian matrices that make local linear approximations of the system and measurement dynamics. The Jacobian for the process is shown below. Since the measurement dynamics are already linear, linearization is not necessary.

$$F_k = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & dt \\ 0 & -\frac{\rho * A_{ref} * C_D * v_k}{\cos(\alpha_k)} & 0 \end{bmatrix} \tag{24}$$

As before, the equations for the operation of an Extended Kalman Filter in the prediction and update steps are taken from a chapter of *Introduction and Implementations of the Kalman Filter* [12]. The measurement covariances were taken from the sensor datasheet, and process covariances were chosen somewhat arbitrarily. The performance of the filter was shown to be relatively insensitive to these values. This filter is only valid during the coast phase, as it does not model the acceleration due to motor burn. Filter values are initialized to their corresponding measurements. Plots of altitude, velocity, and acceleration for this filter, along with residuals for each are shown in figures 7-9 below, for the November 2021 Carby flight.

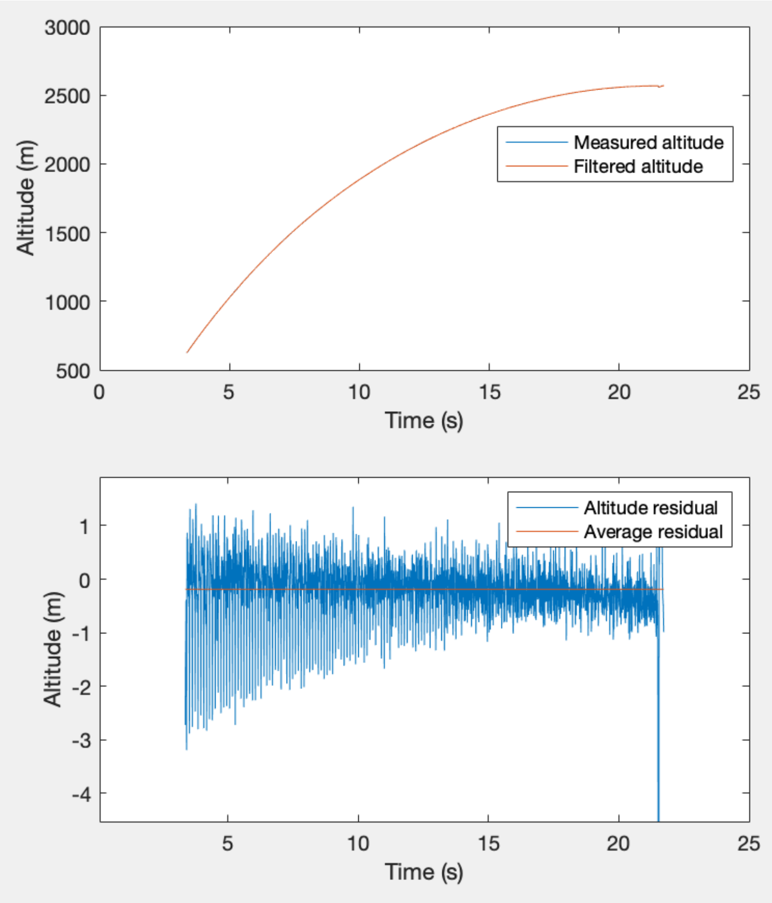

Figure 7 – Extended Kalman Filter Altitude during ascent, Carby November 2021

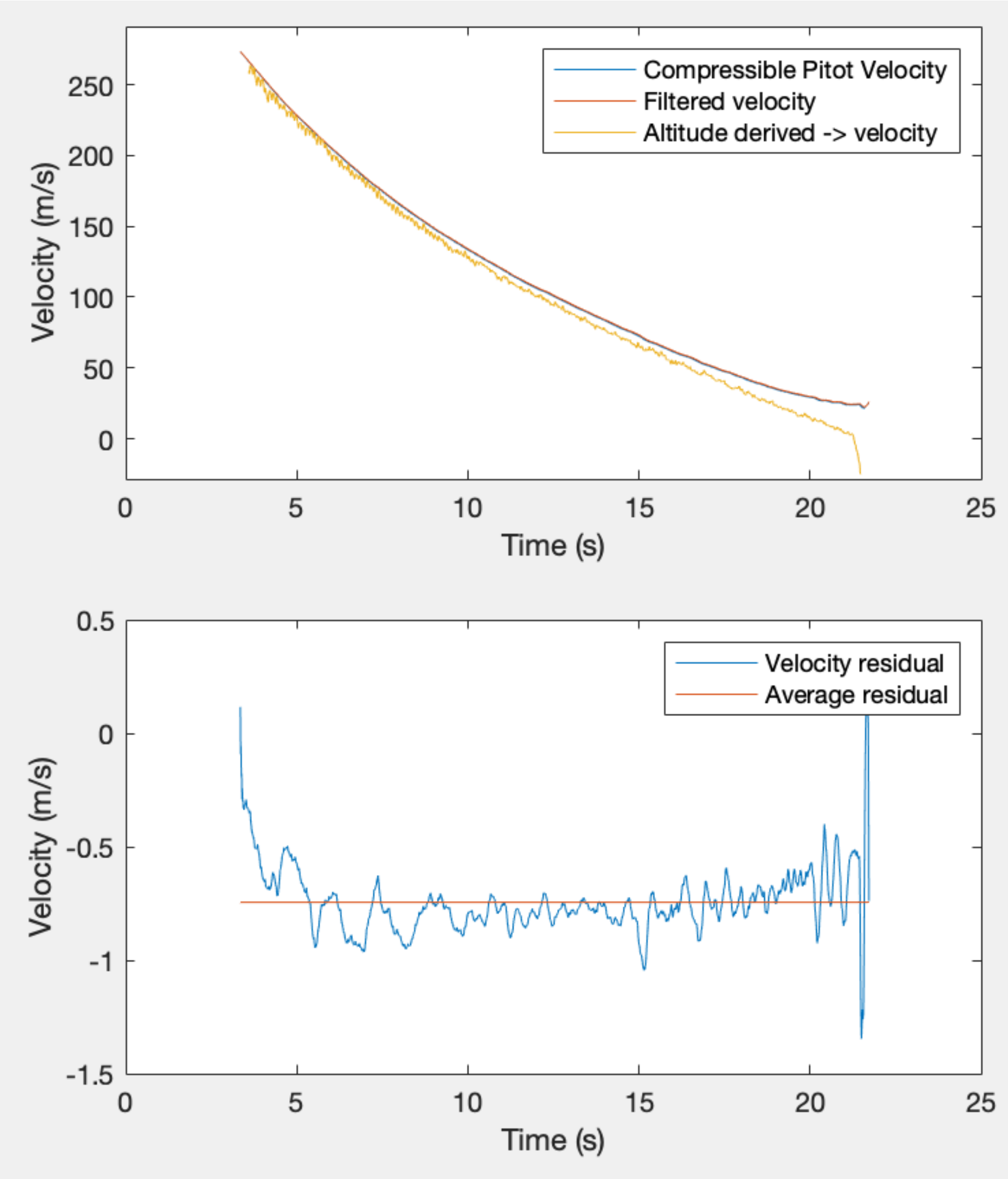

Figure 8 – Extended Kalman Filter Velocity during ascent, Carby November 2021

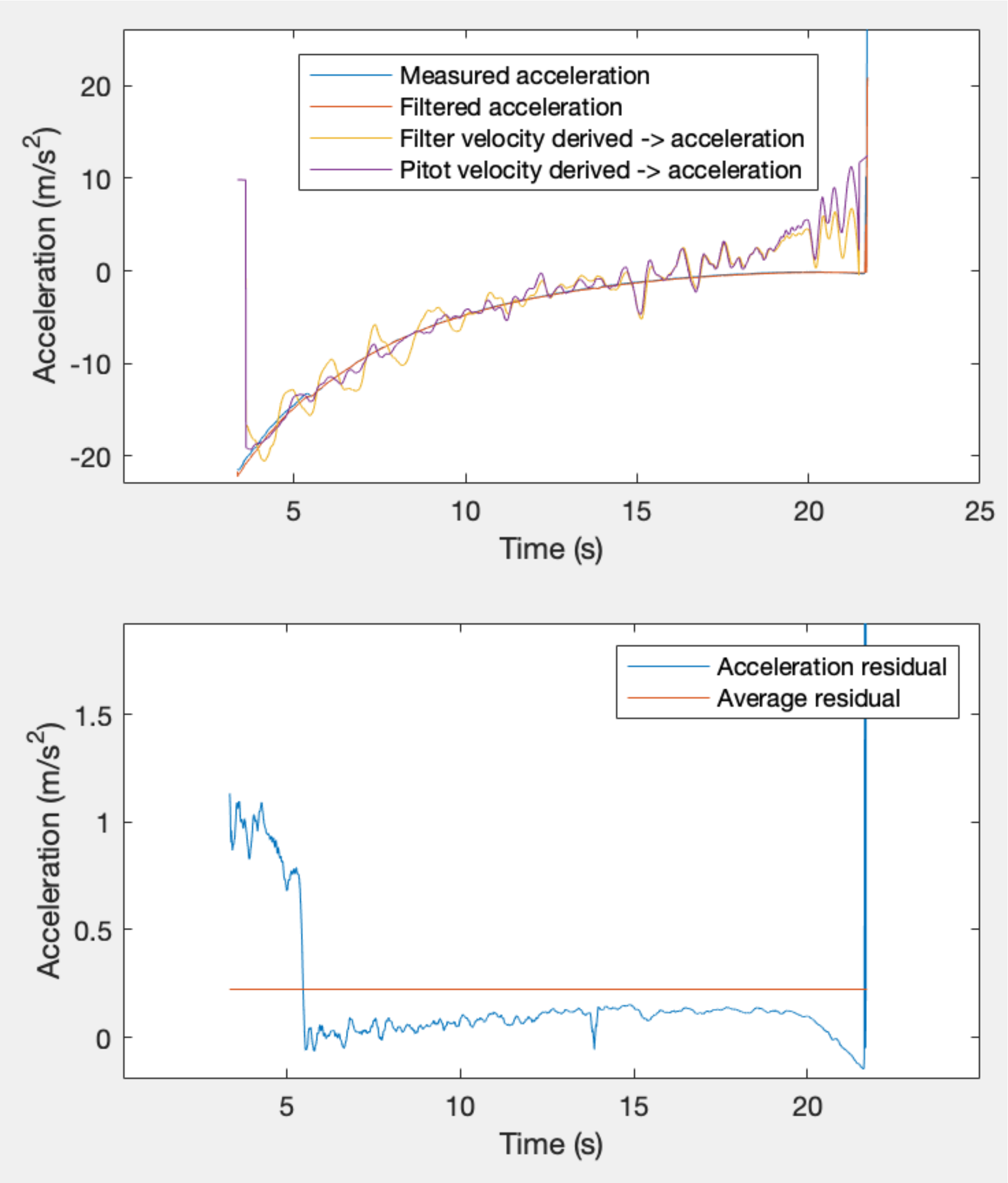

Figure 9 – Extended Kalman Filter Acceleration during ascent, Carby November 2021

The altitude residual over time is similar to the Kalman Filter's, and is near-zero with some noise. However, the velocity residual is more substantially offset from zero, compared with the noise. This is probably reflective of the altitude's derivative disagreeing with the pitot tube velocity. The acceleration residual starts substantially off before moving close to zero. This is unique to this flight, and is due to the discontinuous change in IMU readings about 5.5 seconds into flight.

2.7 Gyroscope Rate Processing

2.7.1 Gyroscope Rate Integration

A variety of methods employed to calculate the tilt angle that was fed into the Extended Kalman Filter. The simplest of which is integration of the roll rates measured by the IMU's gyroscopes. The quaternion math behind this integration was taken from the AHRS python library `https://ahrs.readthedocs.io/en/latest/filters/angular.html` and implemented in MatLab [21]. The results have been verified against the library's outputs natively in python. This integration was fed an initial tilt angle computed using the IMU measurements of the gravity vector while the rocket was on the pad. The math for that is trigonometry, and is explained at `https://www.digikey.com/en/articles/using-an-accelerometer-for-inclination-sensing` [10]. For cases where gyroscope data was not available, the rocket was treated as being at its initial tilt angle for the duration of the flight. This is not a very good assumption, and reduces the usefulness of incorporating tilt angle. The gyroscope integration returned angular position as a quaternion array. This was used to calculate an array of rotation matrices, then multiplied by the initial tilt vector. Trigonometry was used to calculate the pitch angle over time from the rotated vectors.

2.7.2 MatLab's imufilter()

A second way to find angular position is MatLab's imufilter(). This takes in gyroscope data and accelerometer data, and assumes that the gravity vector is measured by the IMU. Rockets are in freefall for the relevant portions of flight, so onboard IMU's do not measure the gravity vector. However, by passing the filter zeros for accelerometer readings, the filter essentially reduces to integration of gyroscope rates.

2.7.3 Matlab's ahrsfilter()

A third way to find angular position is MatLab's ahrsfilter(), which takes in magnetometer data in addition to accelerometer and gyroscope data. This makes the same assumption about measuring the gravity vector, so again zeros are input for the accelerometer data. Both this filter and imufilter() return filtered angular rates, in addition to the orientation. This can be used as a means of comparison

2.7.4 Comparison of Methods

Figure 10 below shows the FCB and DAQ gyroscope rates, for each of the above filters, in addition to the raw data, for the November Carby flight. The filtered angular rates are close to the raw measurements, as expected. The FCB's readings are close the the DAQ's in Y and Z, but are different in X.

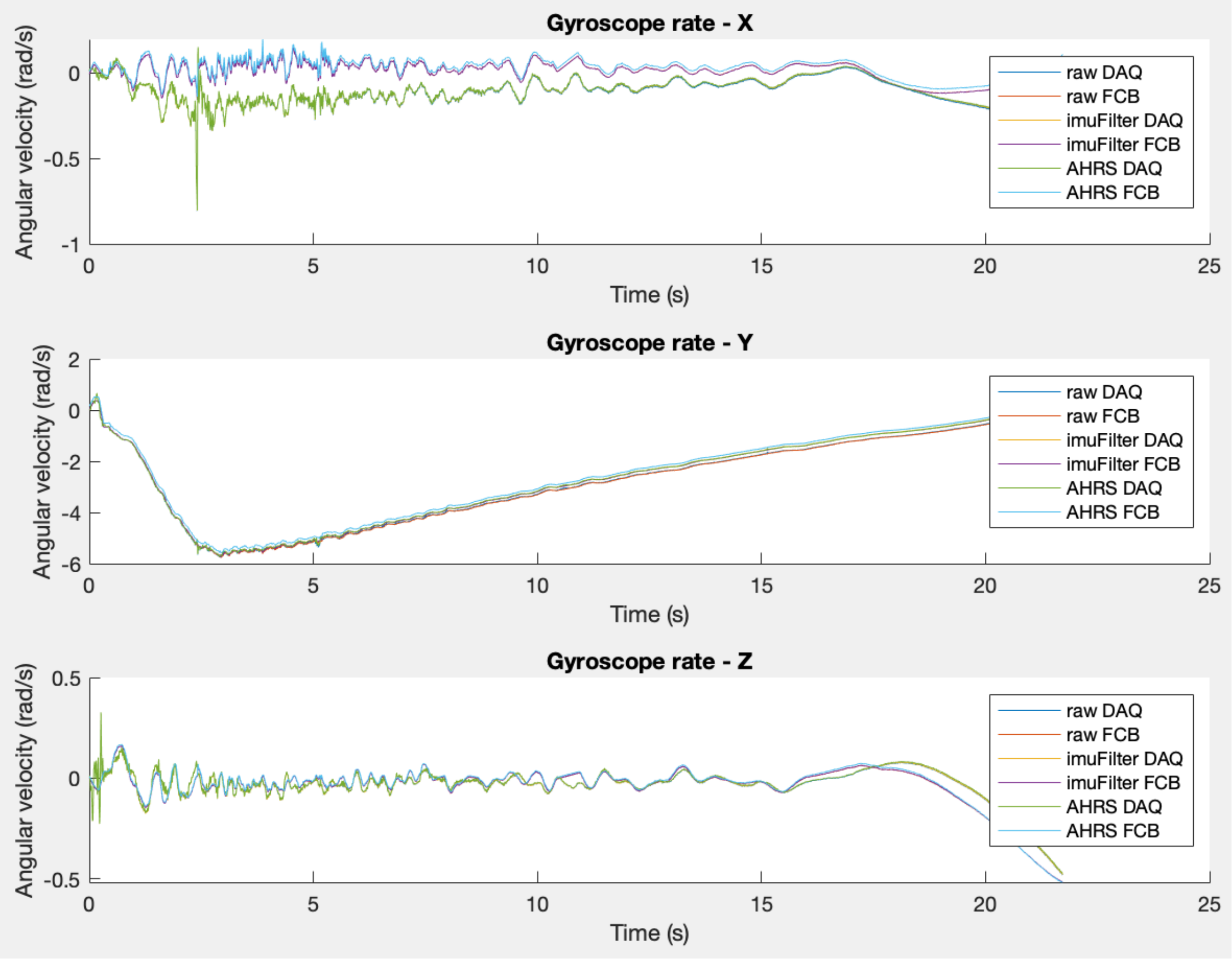

Figure 10 – Gyroscope rates during ascent, Carby November 2021

Figure 11 below shows the calculated tilt angle during ascent of the November Carby flight, using each of the above methods. Most of these methods are strongly in agreement, except the ahrsfilter() for the FCB. This disagreement is likely due to the magnetometer calibration for the FCB being questionable for this flight. The agreement of the other methods verifies the simple gyroscope integration algorithm as sufficient. The agreement of the ahrsfilter() for the DAQ verifies the in-flight magnetometer calibration as viable, in some cases. The integration of rates should show a growing error compared to the ahrs attitude estimation. This data is evidence that 20 seconds of a rocket's flight is insufficient time for that error to accumulate significantly. The average of the tilt angles computed from the FCB's and the DAQ's integrated rates was used for the purposes of the Extended Kalman Filter.

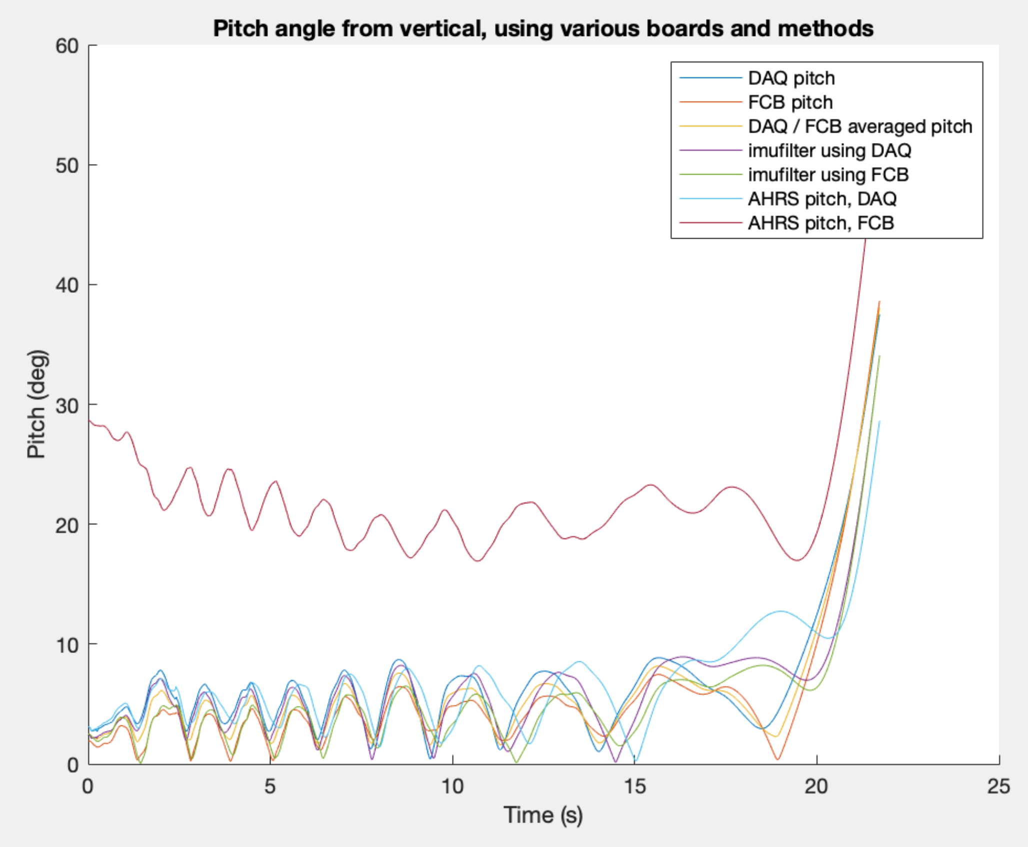

Figure 11 – Tilt angle during ascent, Carby November 2021

2.8 Rolling-Average Numerical Derivative

Filtered states that aren't tied to a measurement are not physically meaningful. Because of how measurements are fed back to influence states, the velocity returned by the Kalman Filter is not necessarily the actual velocity of the rocket. It is not necessarily the derivative of the rocket's altitude. This motivated performing numerical derivation on certain quantities to get numbers that are directly physical. Since numerical derivation can be very noisy if the initial signal is not smooth, a rolling average method was employed. In some cases, simple rolling averages (no derivation) were used in order to smooth noisy data. The normal drawback of moving averages is that they lag behind the true measurement, but this is only when the average is performed live, as data is coming in. When all of the data is known and is being analyzed after the fact, a centralized method can use points both ahead and behind the point of interest, reducing or eliminating the delay.

3. Flight Data Results

This section will provide an overview of the four flights, including plots comparing measurement and filter results for each. The altitude plots are included in Appendix A because generally the filters agree with the measurements in that.

3.1 TRD August 21st, 2021

TRD was flown on August 21st, 2021 with a Redshift DAQ v2, two Easyminis, and an FCB. The FCB did not log data due to neglecting to turn it on, and the DAQ did not log gyroscope rates. The rocket successfully captured 360°camera footage, viewable here: `https://www.youtube.com/watch?v=_V8L9K-EFNQ`. Yes it did land in a corn field, yes the author spent several hours in said corn field looking for the rocket.

Plots of the Kalman Filter and Extended Kalman Filter variables are shown below for powered ascent and coast. The variables agree where they are expected to - the filter states are close to the measurements they stem from. The Kalman Filter's velocity disagrees with the others. The EKF velocity's derivative agrees with the pitot tube velocity's derivative, and both agree closely with the measured acceleration. The derived altitude agrees with none of the other velocity measurements, thought it approaches the KF's velocity in the second half of the coast phase.

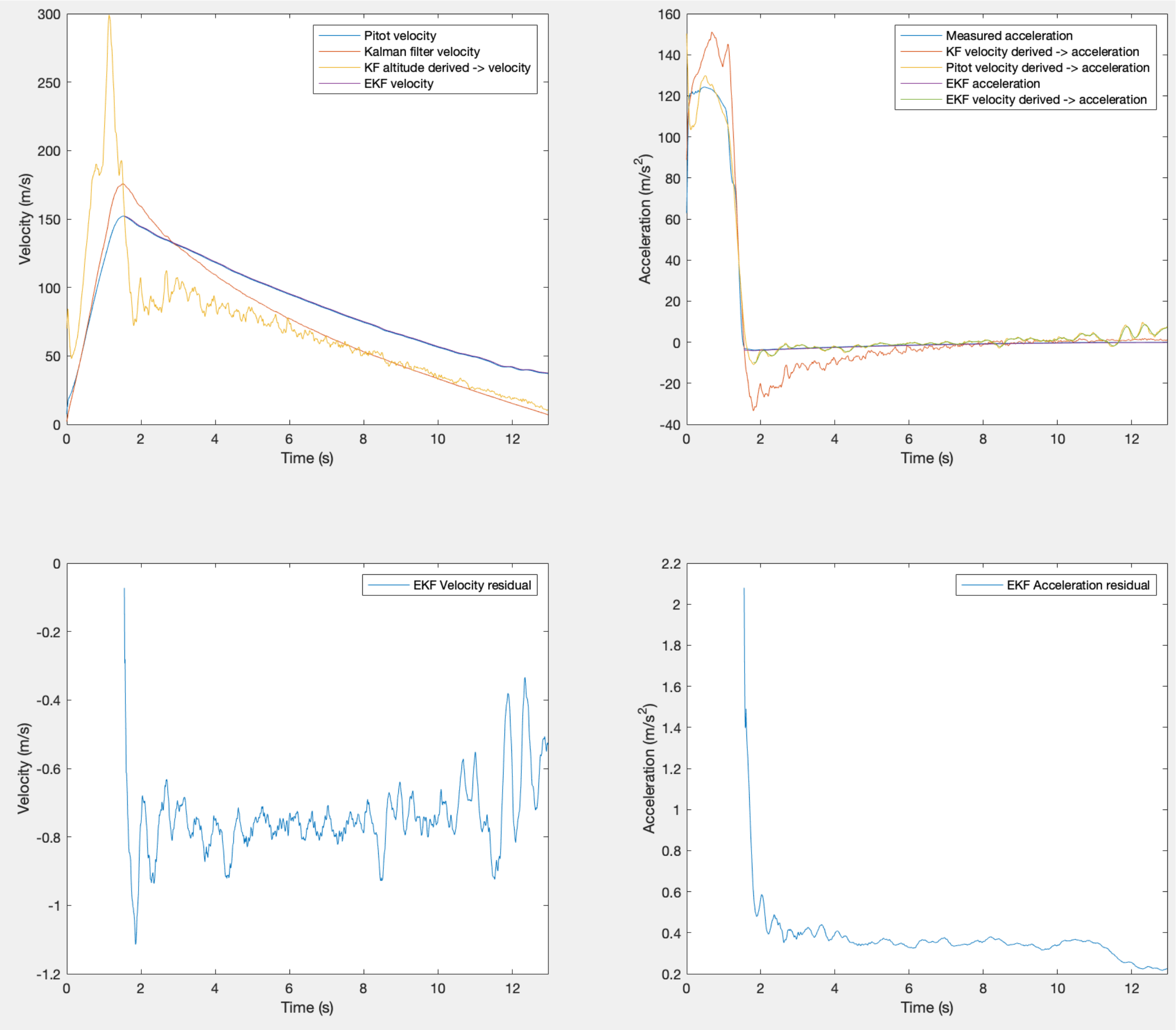

Figure 12 – Filter variables during ascent, TRD August 2021

3.2 Carby September 19th, 2021

Carby was flown on September 19th, 2021 with a Redshift DAQ v2, two Easyminis, a Telemetrum, and an FCB. The FCB did not log data due to last minute software changes, and the DAQ did not log gyroscope rates. The rocket successfully captured 360°camera footage, viewable here: `https://www.youtube.com/watch?v=AcEfYzD447Y`.

Plots of the Kalman Filter and Extended Kalman Filter variables are shown below for powered ascent and coast. The trends in variable discrepancies are similar to those in the TRD August flight, albeit with variables matching closer. These trends will be discussed in section 3.6

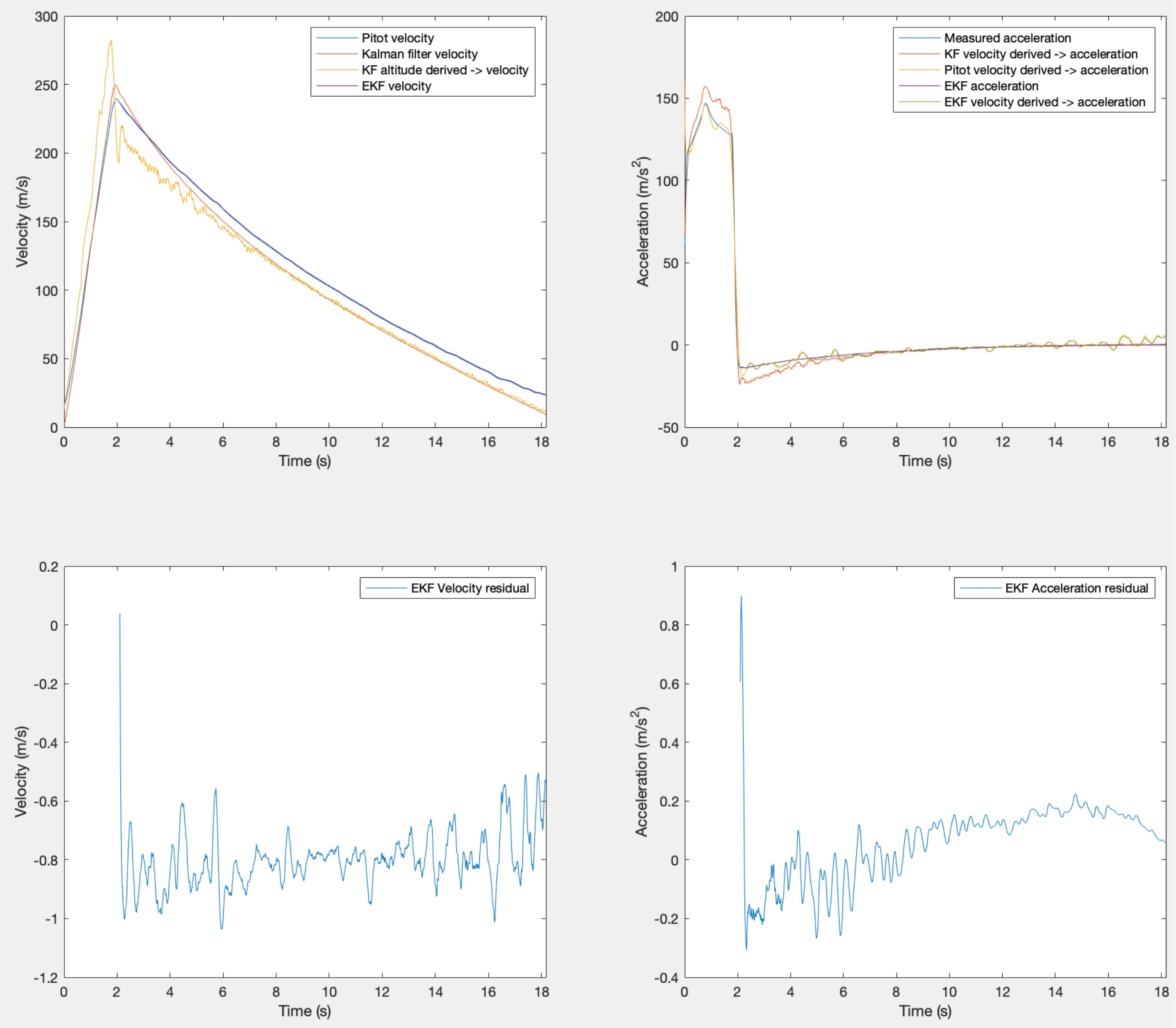

Figure 13 – Filter variables during ascent, Carby September 2021

3.3 Carby November 20th, 2021

Carby was flown on November 20th, 2021 with a Redshift DAQ v2, two Easyminis, a Telemetrum, and an FCB. The FCB successfully logged data, and the DAQ successfully logged gyroscope rates. The rocket successfully captured 360°camera footage, viewable here: `https://www.youtube.com/watch?v=xKTLUiodUv0`. Pay no attention to where the rocket landed. The author may or may not have been involved with the recovery effort.

The tilt angle obtained using multiple methods was plotted previously in figure 11 and is not repeated here. For the purpose of the EKF, the average of the DAQ's integrated tilt angle and the FCB's integrated tilt angle was used.

Plots of the Kalman Filter and Extended Kalman Filter variables are shown below for powered ascent and coast. The trends in variable discrepancies are similar to those in the Carby September flight. These trends will be discussed in section 3.6

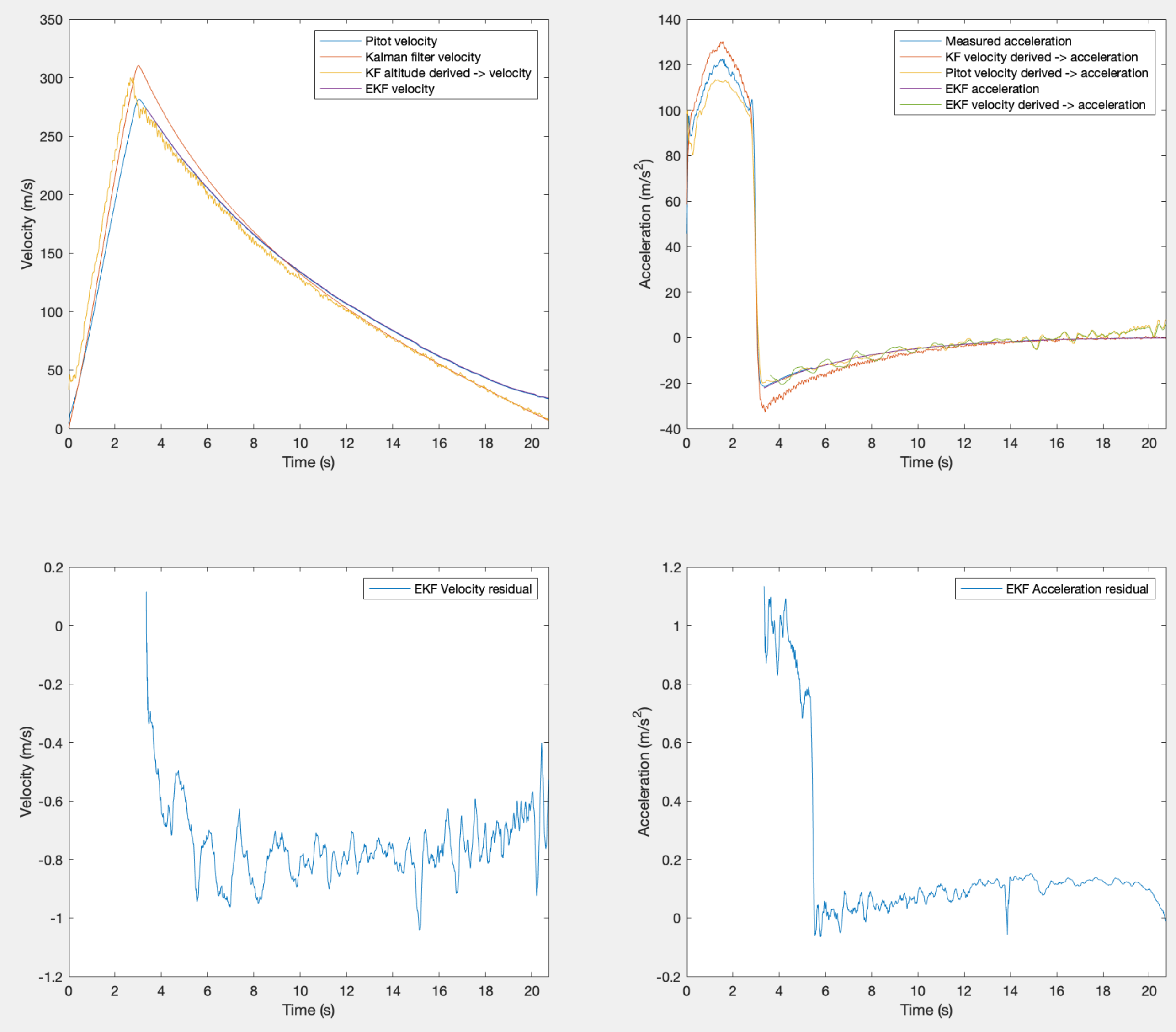

Figure 14 – Filter variables during ascent, Carby November 2021

3.4 TRD April 2nd, 2021

TRD was flown on April 2nd, 2021 with a Haaky DAQ, two Easyminis, a Telemetrum, and an FCB. The FCB successfully logged data, and the Haaky DAQ successfully logged gyroscope rates. The rocket successfully captured 360°camera footage, viewable here: `https://www.youtube.com/watch?v=gjtVOzEQucE`.

The calibrated magnetometer data from this flight is shown in figure 31 in Appendix A. The data does not cover the ellipsoid , thus it is assumed to be poorly calibrated. The tilt angle obtained using multiple methods is plotted in figure 32 in Appendix A. As expected from the imperfect magnetometer calibration, the ahrsfilter() tilt angle is substantially off from the gyroscope integrated tilt angle. For the purpose of the EKF, the DAQ's integrated tilt angle was used.

Plots of the Kalman Filter and Extended Kalman Filter variables are shown below for powered ascent and coast. The biggest discrepancies are caused by the lack of sensor resolution, but the common trends of discrepancies are still present. These trends will be discussed in section 3.6

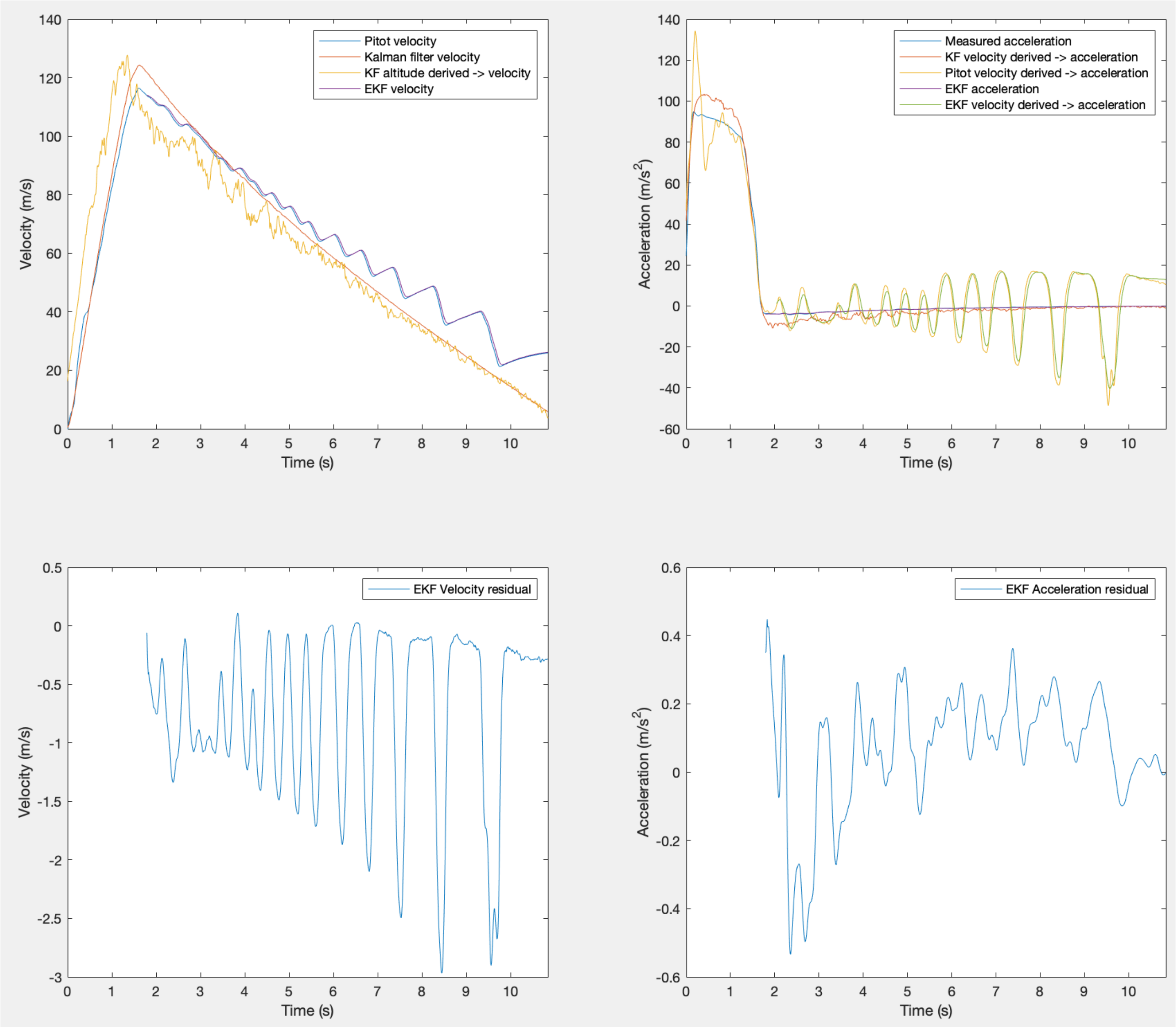

Figure 15 – Filter variables during ascent, TRD April 2022

3.5 Other Launches

TRD was launched for the first time in February of 2021 with a Redshift DAQ v2, one easymini, one stratologger, and an FCB. The pitot tube for this flight was not fully sealed, causing the measured stagnation pressure to be off from the true value. A Redshift DAQ v2 with accompanying pitot tube system was also flown on a club member's certification rocket in the spring of 2021, but the pitot tube for that flight was also leaky, and the DAQ did not log acceleration in the rocket's axis. TRD was launched again in June of 2021 with the DAQ and pitot tube system, however the DAQ did not log data due to an issue with flashing the board. A rocket named Beanboozler was flown in spring 2022 with a HaakyDAQ, however due to a resolution limitation, the pitot tube data was not useful. Overall, the author's takeaway is that electronic boards often take a launch or two to iron out hardware and software issues before they are fully functional. Performing testing with sensors connected to the data acquisition board can be very valuable, in addition to testing sensors independently.

3.6 Data Summary and Analysis

A consistent trend across all flights is that the pitot tube velocity is below the filtered velocity at motor burn out, then above the filtered velocity near apogee. Another interesting discrepancy is that the velocity derived from the filtered altitude in general does not agree with either the filtered velocity, or the pitot tube's velocity. It consistently overestimates velocity before motor burnout, then underestimates afterward. There are many possible theories for why these could be happening. First is that the pitot tube and static pressure have time constants on equalizing to the instantaneous pressures, so their readings lag behind. And this is definitely true, to an extent - both can be thought of as second order systems with some time constant of equalization. However, if this were the case, the dynamic pressure (and pitot tube velocity) would be expected to continue increasing after the time when the filter reaches maximum velocity. This is not the case.

The pitot tube velocity and filtered velocity both reach their maximum values remarkably close to the same time, within a hundredth of a second for both Carby flights and the August TRD flight. This is well within the margin of error for temporal alignment between boards. The rocket's maximum velocity is directly related to when the motor stops firing, reflected in a sharp change in the rocket's acceleration from positive to negative. This acceleration is fed into the Kalman filter's velocity estimation. The temporal alignment between pitot tube and filtered maximum velocity is physically meaningful as the pitot tube reading a maximum velocity at motor burnout. This is the expected behavior, thus the delay theory alone does not explain the discrepancy.

Another possibility is that the pitot tube was slowly leaking. Though all the pitot tubes were tested and confirmed to seal, this test was done by applying vacuum pressure rather than positive pressure. This could explain the pitot tube being lower at burnout, but it does not account for the pitot tube readings staying high near apogee. Another possibility is imperfect offsets and sensitivities for the sensors. These could contribute to the pitot tube readings near apogee, but this would have the opposite effect at burnout.

Another theory is that the pitch angle is contributing to the discrepancies, despite being accounted for in some flights. This may partially explain the difference in measurements near apogee, where the pitot tube velocity is above the filtered velocity, but it does not explain the filter's overestimate of maximum velocity.

It is unfortunate that (excluding the Haaky-DAQ flight) only the November Carby flight has gyroscope data, and that data agrees most closely between pitot tube and filter. The author suspects that other flights like the August TRD flight pitched over significantly more, however there is no way to evaluate this without gyroscope data. The flights without gyroscope data do have magnetometer readings. Unfortunately a magnetometer alone is not enough to determine heading, as magnetometer readings only have two true degrees of freedom - the third axis is driven by the other two.

It could be that all of the above behaviors - equalization time, a slow leak, imperfect sensors, and the rocket pitching over - combine and cancel each other such that it behaves as observed.

Deriving the pitot tube velocity should return the acceleration measured by the IMU regardless of pitch angle, because both are aligned with the rocket's axis. Using the rolling average derivative, the acceleration derived from the pitot tube's velocity is plotted against the IMU's measured acceleration in figure 16. This plot is for the coast phase only, for the Carby November flight. It is cutoff four seconds before apogee because at that point the velocity is small and the derived acceleration is noisy. The best fit trendline shows a 3.36% scale factor — in light of noisy data this is strong agreement. This comparison is valid in the boost and coast phases of flight, and a similar graph including powered ascent is shown in Appendix A, figure 33. The best fit for that is slope 0.9440, indicating a larger error that seems to be due to transient effects. Overall this agreement is a point in favor of the pitot tube's accuracy, especially because the coast phase is the region of interest for analysis.

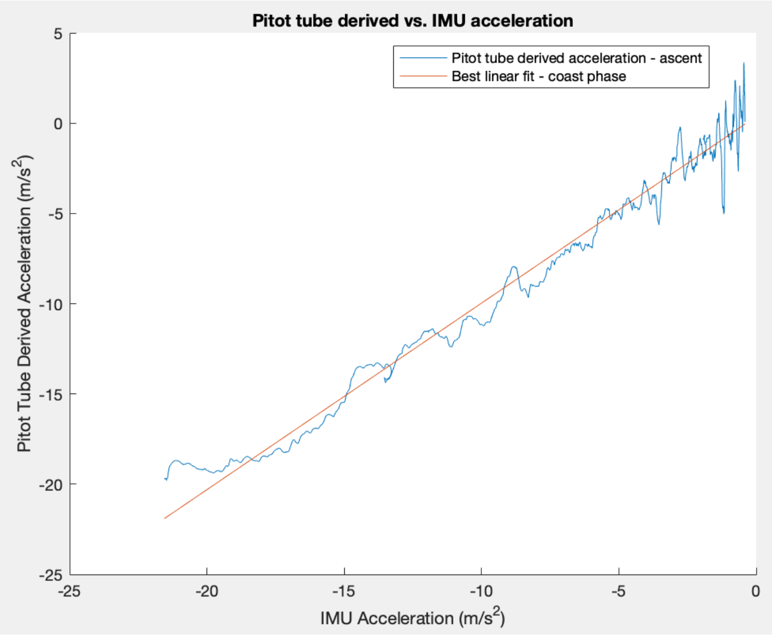

Figure 16 – Pitot tube derived acceleration vs. IMU acceleration during coast, Carby November 2021

3.7 Data Selection for Drag Coefficient Calculation

When trying to pull a drag coefficient from the flight data available, there are a couple of choices. First is which period to sample at. In general the calculated drag coefficients were near constant for 5-10 seconds after motor burnout, before substantially changing. This time period is where velocity and acceleration are highest, so the sensor errors are least important. The exact period chosen was slightly different for each flight, for example it was chosen as after the accelerometer discontinuity in the November Carby flight data.

The next choice is what acceleration and velocity calculations to use. There are many potential combinations, and many are equivalent. I will discuss three here, focused on differentiating between data coming from different sensors. First is the most directly measured - using the pitot tube velocity and IMU acceleration. A similar calculation uses the EKF's velocity and acceleration. These will not be exactly equal, but will be close and act similar to a moving average. Second is using only data from the pitot tube — directly for velocity and its derivative acceleration. The third is using no data from the pitot tube — using the Kalman filter's altitude derived for velocity, and IMU for acceleration. One could use the Kalman filter's velocity, but it generally agrees less closely with the pitot tube velocity, and is less physical than deriving altitude.

The drag coefficients using different calculation methods are plotted for each flight in figures 17 - 20 below. The average of each drag coefficient is plotted as a horizontal line in the same color.

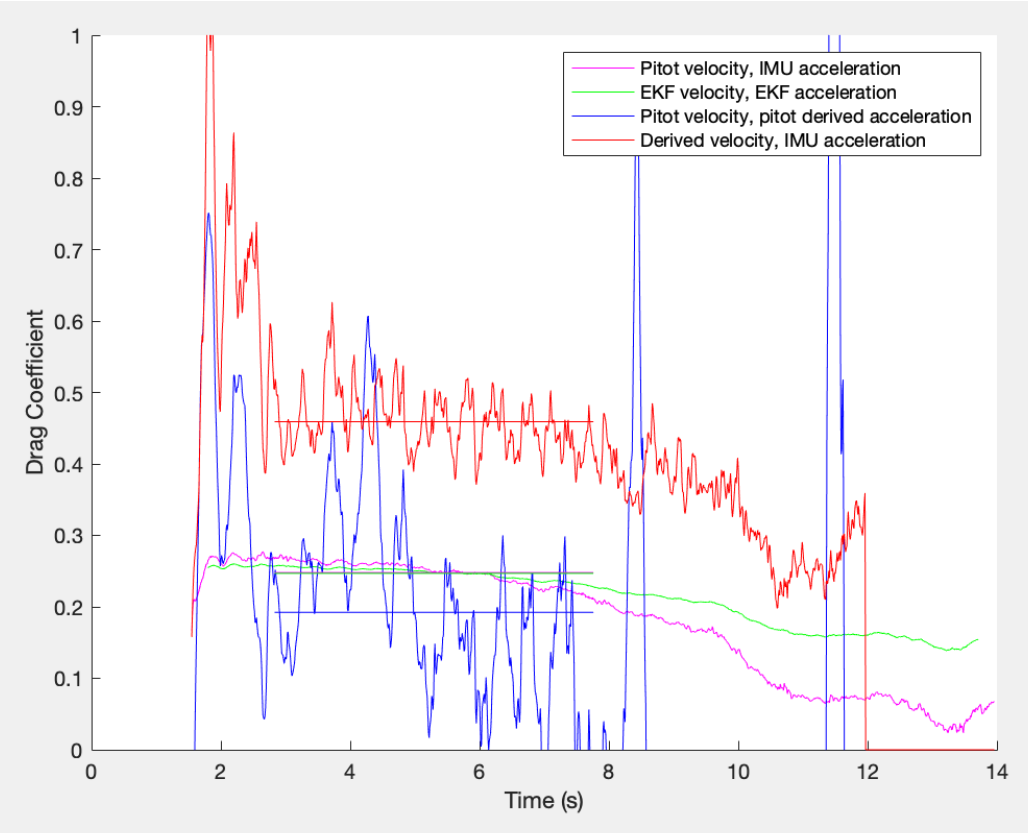

Figure 17 – Drag coefficients during coast, TRD August 2021

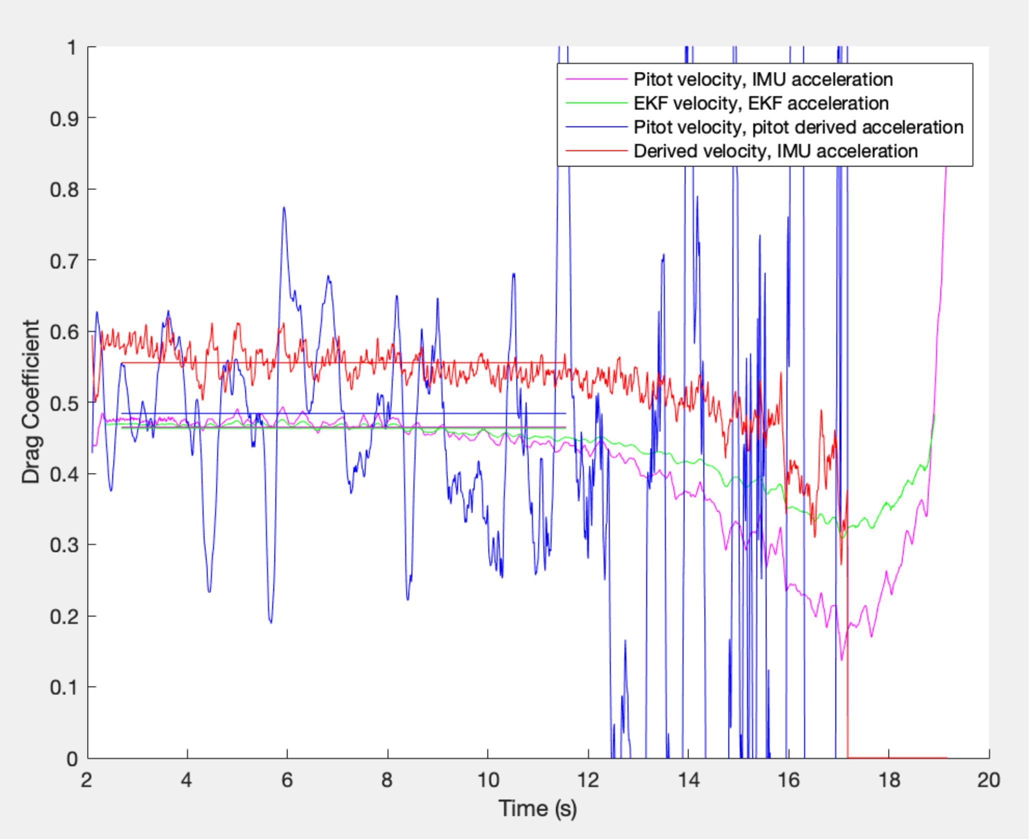

Figure 18 – Drag coefficients during coast, Carby September 2021

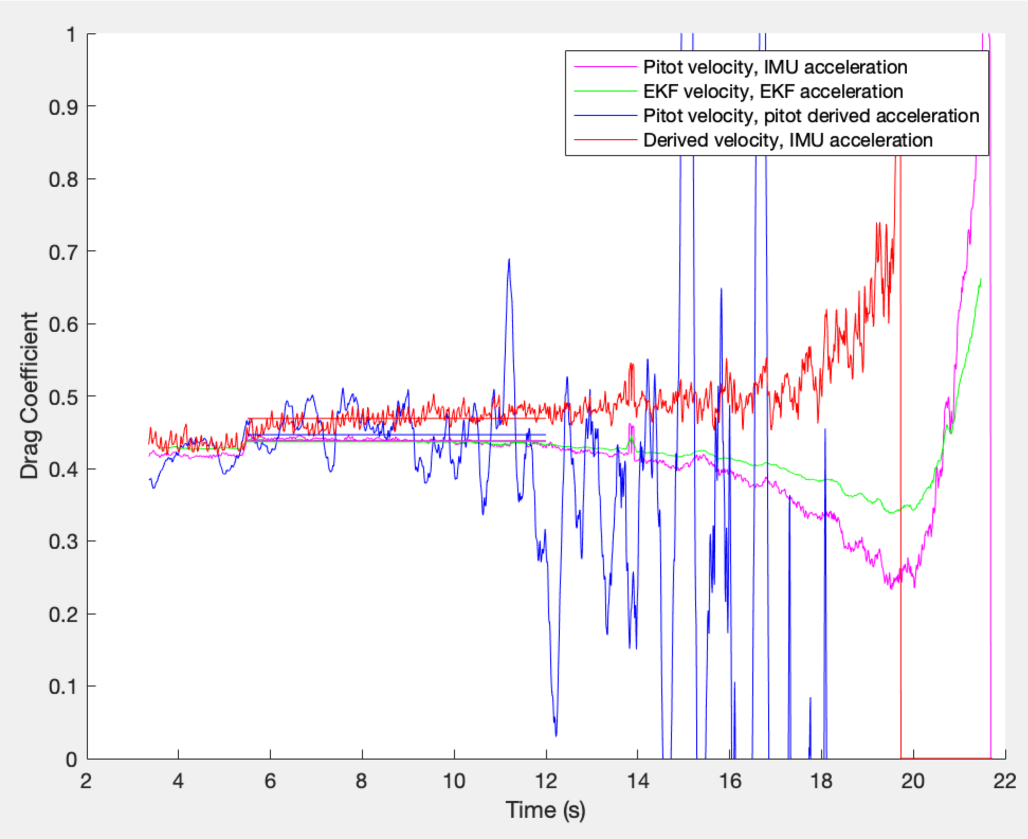

Figure 19 – Drag coefficients during coast, Carby November 2021

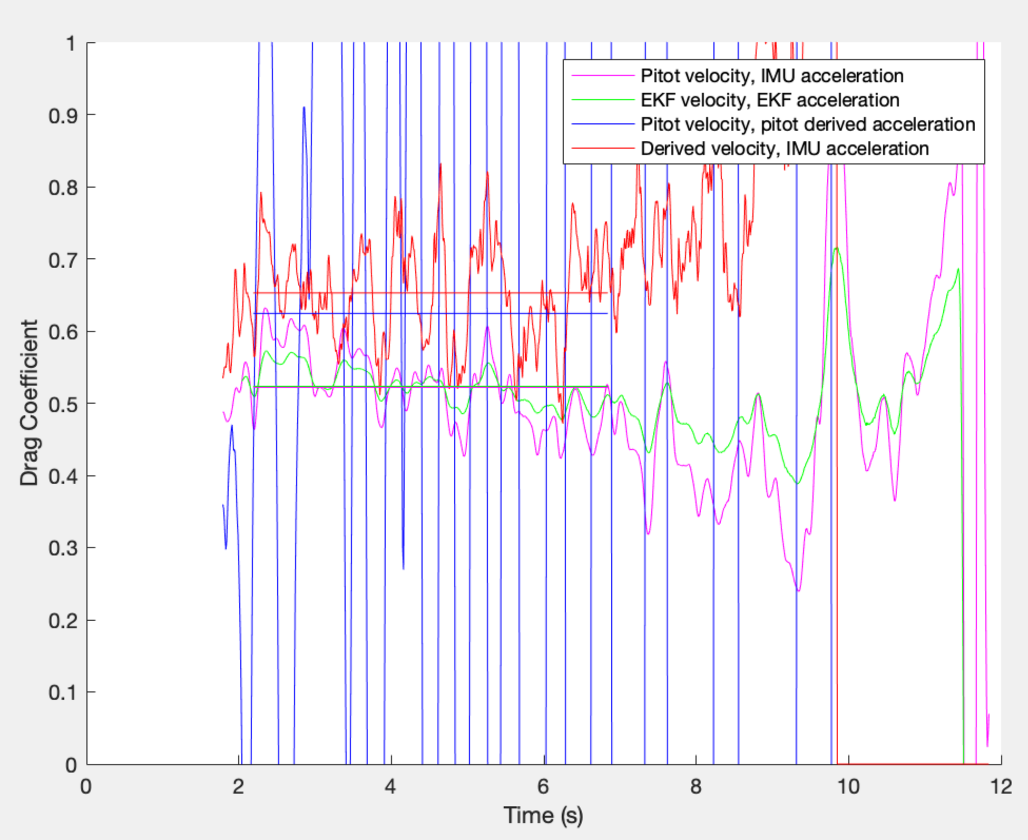

Figure 20 – Drag coefficients during coast, TRD April 2022

One may note that the Extended Kalman Filter necessarily inputs a drag coefficient — the average

CD using the pitot tube and IMU was used. Thus, the EKF based drag coefficients tend to be close to the drag coefficients using the data directly. If a CD different than the directly measured value is input to the filter, the EKF value tends to be around halfway between the measured value and the input value. This confirms that the CD is a good fit for the model. Future work may examine this relationship more closely. The four drag coefficients for each flight and their source for velocity and acceleration are listed in tables 1-4 below. For the flights that have been simulated, the drag coefficient from CFD is included in the caption.

Velocity Source	Acceleration Source	Calculated CD
Pitot tube	IMU	0.2480
EKF	EKF	0.2469
Pitot tube	Pitot tube derived	0.1923
Altitude derived	IMU	0.4591

Table 1 – CD Results TRD August 2021, Simulated CD = 0.29

Velocity Source	Acceleration Source	Calculated CD
Pitot tube	IMU	0.4648
EKF	EKF	0.4631
Pitot tube	Pitot tube derived	0.4841
Altitude derived	IMU	0.5553

Table 2 – CD Results Carby September 2021

Velocity Source	Acceleration Source	Calculated CD
Pitot tube	IMU	0.4384
EKF	EKF	0.4369
Pitot tube	Pitot tube derived	0.4464
Altitude derived	IMU	0.4691

Table 3 – CD Results Carby November 2021, Simulated CD = 0.44

Velocity Source	Acceleration Source	Calculated CD
Pitot tube	IMU	0.5219
EKF	EKF	0.5234
Pitot tube	Pitot tube derived	0.6242
Altitude derived	IMU	0.6528

Table 4 – CD Results TRD April 2022

In general the directly calculated CD's first decrease, then increase greatly — the EKF CD's follow similar trends but more muted. The EKF based CD is also less affected by the IMU discontinuity in the November Carby flight. The pitot tube only CD's tend to be closer than the altituded derived CD's. For the two Carby flights, the pitot tube only CD's agree within 2% of those calculated using the pitot tube and IMU. The drag coefficients based on derived altitude tend to be higher than the direct measurements, due to underestimation of the velocity. There is no good explanation for this discrepancy, but it is much more apparent for certain flights than others. For example, the two Carby flights' drag coefficients all line up much closer than those from TRD. The drag coefficients using anything from the pitot tube on the TRD April flight are very noisy, and in that case the altitude derived CD should be more reliable. It is a wonder that the pitot tube only CD and the altitude derived CD for that flight are close, perhaps merely a coincidence.

4. Computational Fluid Dynamics Efforts

The author has examined two of the rockets under consideration through a series of CFD calculations. The first is the TRD rocket flown August 21st 2021. The CFD for that case was steady, incompressible and used the $k-\omega$ SST turbulence model. The author performed an extensive sensitivity analysis of that case and found the drag coefficient to be independent of mesh level (beyond a critical point), domain size, free stream turbulence parameters, and inflation layer parameters. Drag coefficient was found to change no more than 1-2% with moderate changes in input parameters. Surface roughness is an important factor, so a proper measurement of that is key to an accurate drag coefficient. Both velocity and density affect the drag coefficient, but no more than a few percent over the range of interest. That analysis found a drag coefficient of approximately 0.29 [16].

This section will detail similar efforts for the Carby rocket, in the launch configuration from the November 20th, 2021 flight. The rocket was flown with different fins in September, and that configuration has not been simulated. With the relative sensitivity of different factors mapped out, the current analysis need not be so robust. A wide range of mesh levels are tested. This case is also steady and uses the same turbulence model, but is compressible.

4.1 Geometry

The geometry used for this analysis is the Carby rocket, a 4 inch diameter, 8 ft., 9in. tall carbon fiber rocket built by project Redshift in 2021. A $\frac{1}{3}$ symmetry domain is employed, as Carby is a three-finned rocket. Some key simplifications were used to make this geometry tractable. These include the neglect of screws, rail buttons, small changes in diameter at the transparent section, and various holes in the rocket. Importantly all of these simplifications should lead to a rocket geometry that produces less drag than the real rocket. This rocket geometry was subtracted from a larger body to form the computational domain, then subdivided into smaller regions, as seen in figure 21.

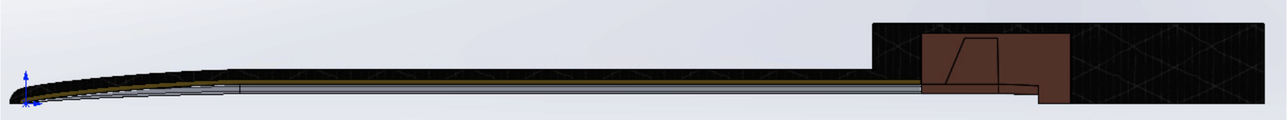

Figure 21 – Decomposed computational domain, side view, zoomed near to rocket

The intent of these regions was to control the local mesh size in order to resolve key flow features, such as pressure and velocity gradients, which are expected to be highest close to the rocket and behind it. The regions allow the mesh to be higher locally near the rocket, while maintaining a large enough buffer for mesh size to transition without adding error.

4.1.1 Boundary Conditions

A pressure-far-field boundary condition was used for the front face of the domain, with the Mach number specified as 0.65. This evaluates to 225.6 m/s for the purpose of calculating drag coefficient, using 347m/s as the speed of sound. Pressure-far-field boundaries were used for the outside surface and the back of the domain. The incompressible TRD case examined by the author had slight issues with domain independence due to the boundary conditions causing constriction and an artificial local velocity increase. When using the ideal gas model for density, as in this case, the pressure-far-field boundary condition is available, which can be an inlet or outlet, eliminating the constriction. The surface of the rocket was modeled using a no-slip boundary with surface roughness set to a measured value for unpainted carbon fiber rocket tubes.

Slip wall (zero shear stress) boundary conditions were used for the sides of the domain. The better practice would be to use periodic boundaries for the sides of the domain, however due to inflexibility in Ansys this wasn't possible. There should not be any real periodic behavior of information being transported from one side of the domain to the other, so the slip wall condition is approximately equivalent. An alternative option was using far field boundaries for the sides, but running a brief comparison showed a major difference. The drag coefficient was approximately 60% higher with the fair field boundary when compared to an otherwise identical case with a slip boundary. This comparison was done at a mesh level deemed sufficient, as discussed in section 4.2 One possible reason for this discrepancy is that the far field boundary enables the possibility of flow out of the domain at the sides, which violates periodicity. A second possibility is that Ansys may be imposing a velocity on the boundary, close to the no-slip wall of the rocket, leading to increased shear stress. For these reasons, a slip wall was used for all cases.

4.1.2 Turbulence Model

The turbulence model used for all simulations is the latest version of the $k-\omega$ SST model, originally proposed in 1994 by Menter. This model uses a blending function to combine Wilcox's $k-\omega$ model in the logarithmic boundary layer region with the standard $k-\varepsilon$ model in free shear regions. This model also formulates eddy viscosity to account for the transport of turbulent stress, hence the name Shear Stress Transport (SST). This allows the model to predict flow separation in adverse pressure gradient flows, while avoiding the free-stream turbulence property dependence that characterizes the $k-\omega$ model [18, 14]. The exact version of this model and equations used are described in the Ansys Fluent theory guide [4]. The turbulent intensity at the velocity inlet was specified as 0.5% and turbulent viscosity ratio as 2. These values were taken from an Ansys Innovation course [3] on external aerodynamic CFD.

Large Eddy Simulation (LES) was considered for these simulations, as it could resolve certain flow features like vortices better. However, the rule of thumb is that an accurate LES simulation should resolve 80% of the turbulent kinetic energy, which requires a very fine mesh in most of the domain. Considering the size of the rocket in question, LES would need be very computationally intensive, making it beyond the current scope. Use of an Unsteady Reynolds-Averaged Navier Stokes (URANS) model was also considered for this analysis, however a pseudo-transient steady model was found to be sufficient.

4.2 Meshing

The meshes for this analysis were generated using poly-hexcore cells in Fluent Meshing (Watertight method), adding local sizing controls based on the subdivided domain geometry. This meant that the orthogonal quality was near perfect in the center of flow regions, far from boundaries, but lower near the rocket geometry.

4.2.1 Inflation Layers

Inflation layers were employed in order to fully resolve the boundary layer, an important part of predicting aerodynamic behavior. The parameters of the inflation layers were calculated using an online tool, setting the first cell height to approximately $y+=1$ [5]. The Reynolds number (Re) is defined below, where U is the velocity, L is the length of the rocket from tip to tail and μ is the viscosity. For this case at a baseline of Mach 0.65, the Reynolds number has a value around $3.6*10^7$, in the turbulent regime for external flows [15].

$$Re = \frac{\rho U L}{\mu}$$

4.2.2 Mesh Study

A mesh refinement study was performed to determine if the results were independent of the mesh density. The cell size was reduced in each region by the same global factor each refinement. Cell count scales roughly with length cubed, so the global factor was varied somewhat to compensate. For the first couple refinements this factor was 0.6 (i.e. length at level 2 = 60% of length at level 1), then increased to 0.8 and finally 0.9. This resulted in a range of cell counts from 66 thousand to 25 million. The global growth rate was decreased from 1.05 at the coarse meshes to 1.01 at the fine meshes in order to improve mesh quality. Another mesh factor is the cells per gap, which was increased from one at the coarsest mesh to five at the finer meshes. This is why the cell count grows somewhat discontinuously.

The results of this study are shown in table 5 below. Employing the rule of thumb that for a mesh-independent CFD case, the key results shouldn't meaningfully change over two subsequent mesh refinements. This case meets that requirement at mesh level 8 (11 million cell case), beyond which there are 3 refinements without more than 2% change. The percent difference reported is in comparison to the drag coefficient of the highest cell count case.

Growth Rate	Refinement Factor	Cell Count	Min. Orthogonal Quality	CD	CL	% Diff
1.05	N/A	66,169	0.00245	0.774	0.633	76.7
1.05	0.6	113,746	0.00447	0.648	0.671	47.9
1.05	0.6	168,137	0.00598	0.569	0.742	30.0
1.035	0.6	409,613	0.0129	0.487	0.722	11.1
1.035	0.6	726,184	0.0117	0.454	0.726	3.67
1.025	0.8	1,524,837	0.0224	0.451	0.734	2.94
1.025	0.8	2,053,258	0.0281	0.458	0.737	4.51
1.01	0.8	11,008,733	0.0516	0.446	0.758	1.79
1.01	0.8	14,387,272	0.0526	0.442	0.779	0.90
1.01	0.8	19,246,623	0.0612	0.443	0.805	1.05
1.01	0.8	25,475,736	0.0662	0.438	0.837	0

Table 5 – Mesh study results

The results converge quickly up to about one million cells (10^6), then converge slightly further after that. This can be seen more clearly in figure 22 below. However, the results for the lift coefficient, in this case positive radially inward, converge much less cleanly. Though this is a potential red flag that not all of the flow behavior is being resolved, the clean convergence of the drag coefficient reduces this concern.

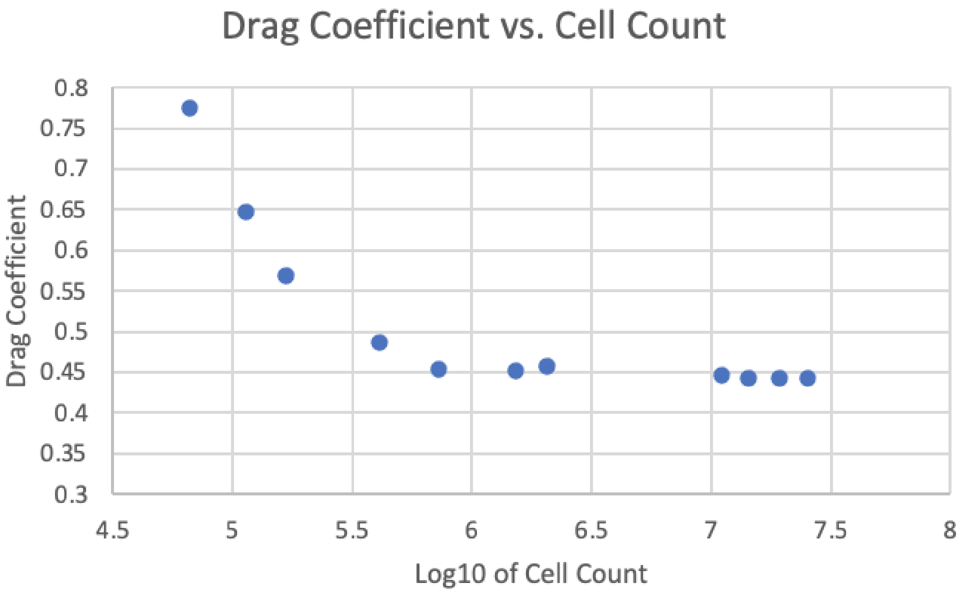

Figure 22 – Drag coefficient as a function of cell count

Though the convergence is not perfect and there is a slight trend even at the higher mesh levels, it is small enough to conclude that the results are sufficiently mesh independent. The 11 million cell case was chosen as the default case. Though some of the cases had a minimum orthogonal quality below 0.05, the number of cells with orthogonal quality below that threshold was typically on the order of 1,000 cells. In a mesh with millions of cells, these cells are not expected to have a huge impact on the overall results.

All CFD cases were run on the Massachusetts Green High Performance Computing Center (MGHPCC). All cases except the coarsest were above the student cell count limit on Ansys - these were made possible by the sponsored license that ANSYS Inc. grants AeroNU.

5. Conclusion

This paper presented a detailed review of data from four flights of instrumented rockets, and accompanying CFD calculations. Processing the data involved aligning data from different boards, and performing post-facto calibration of IMU's and pressure sensors. An in-flight magnetometer calibration was shown to perform well in one case. A simple atmospheric model was employed to account for varying density based on altitude and relative humidity. All rockets were equipped with a pitot tube, so compressible subsonic fluid relations were used to calculate the rocket's velocity. This was compared to the velocity based on derived altitude, and velocity returned by a Kalman Filter. The Kalman Filter's estimates deviated, in some cases substantially, from the pitot tube's readings, so an Extended Kalman Filter was employed to compensate for the rocket's tilt angle. Unfortunately the launch that showed the largest deviation did not have gyroscope data, and all other flights' tilt angles were small (<10°). Many possible sources of the deviations were examined, and each can individually explain a discrepancy, but fails to explain the whole behavior.

Drag coefficients based on the original measurements, the EKF filter values, the pitot tube alone, and the altitude derived were compared. The direct measurements and EKF CD's agree as expected, and for two flights the pitot tube alone agrees closely as well. The altitude derived drag coefficients were generally higher than the others, because of underestimation of the velocity.

Computational Fluid Dynamics calculations were performed for the August TRD and November Carby flights. These used a $\frac{1}{3}$ symmetry domain, mesh refinement regions, the $k-\omega$ SST turbulence model, and a pseudo-transient solver. Both were shown to be sufficiently insensitive to mesh level. For the August TRD flight, the CFD drag coefficient was substantially off (17%) from the calculated CD. However, this is the flight with the largest discrepancy between different CD calculation methods. For the November Carby flight, the CFD drag coefficient was almost coincidentally close to the calculated CD (.7%). This is well within the margin of error of the sensors and CFD. The multiple methods for calculating drag coefficient agree most closely for this flight.

In light of the presented data, trends and discrepancies, the CFD models seem as trustworthy as the experimental data. Having exhausted all reasonable ideas to explain the consistent trend of discrepancies in multiple flights of data, I invite the reader to suggest possible alternative theories.

6. Contact Author Email Address

The author is happy to answer questions directed to the following email address.
mailto: kierulf.b@northeastern.edu

7. Acknowledgements

I would like to thank my fellow students in the Aerospace Club for their efforts in building rockets and helping to collect data, and for letting me analyze that data in a paper. I would like to thank Northeastern University for the funding it provides to the Aerospace Club. I would like to thank Ansys and my primary contact there, Ryan Gordon, for granting AeroNU a sponsored academic license. This license was critical to running the simulations in this paper. I would also like to thank the research computing staff at Northeastern for their help in using Discovery. Finally I'd like to show my gratitude to Professor Andrew Gouldstone, the Aerospace Club's faculty advisor, who was a great help in accessing the above resources.

8. Copyright Statement

References

[1] Earth Atmosphere Model, Metric Units. Available at https://www.grc.nasa.gov/www/k-12/airplane/atmosmet.html, 2021. Accessed 05/22/22.

[2] Flight of a Model, Rocket. Available at https://www.grc.nasa.gov/www/k-12/rocket/rktflight.html, 2021. Accessed 05/22/22.

[3] Ansys innovation courses. Available at https://courses.ansys.com, 2022. Accessed 02/26/22.

[4] Fluent theory guide. Available via Ansys Customer Portal https://support.ansys.com/Home/HomePage, 2022. Accessed 02/26/22.

[5] Inflation layers calculator. Available at https://www.fluidmechanics101.com/pages/tools.html, 2022. Accessed 02/26/22.

[6] Plattsburgh, NY Weather History. Available at https://www.wunderground.com/history/daily/us/ny/plattsburgh/KPBG/date/2021-8-21, 2021. Accessed 05/22/22.

[7] B. Bollermann. Study of 30 km to 200 km meteorological rocket sounding systems. Technical Report CR-1790, Space Data Corporation, 1331 South 26th Str, Phoenix, Arizona 85034, May 1971.

[8] Thomas Chateuvert, Jonathan Bédard, Alexandre Rivard, Edouard Demers, Maxime Guillemette, Louis-Philippe Drolet, and Jean-Christophe Blais. High v: Team 42 project technical report to the 2018 spaceport america cup. (Unpublished), 2018.

[9] Chris's Supplies Rocketry. Available at https://csrocketry.com/rocket-motors.html, 2022. Accessed 05/22/22.

[10] Christopher J. Fisher. Using an accelerometer for inclination sensing. Available at https://www.digikey.com/en/articles/using-an-accelerometer-for-inclination-sensing, 2011. Accessed 05/22/22.

[11] Past Weather in Saint Albans Bay State Park, Vermont, USA. Available at https://www.timeanddate.com/weather/@5240573/historic?month=4&year=2022, 2022. Accessed 05/22/22.

[12] Felix Govaers. *Introduction and Implementations of the Kalman Filter*. IntechOpen, Rijeka, 2019.

[13] Kristoffer Hammargren. Aerodynamics modeling of sounding rockets. Master's thesis, Luleå, Sweden, 2018.

[14] Hrvoje Jasak. *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows*. PhD thesis, London, UK, 1996.

[15] Jr. John D. Anderson. *Fundamentals of Aerodynamics*. McGraw Hill, fifth edition, 2011.

[16] Bjorn Kierulf. Sensitivity analysis and verification of cfd methods for an external aerodynamic incompressible case. Blacksburg, Virginia, 2022. AIAA.

[17] Mathworks: Sensor fusion and tracking toolbok. Available at https://www.mathworks.com/help/fusion/, 2022. Accessed 05/22/22.

[18] F. R. Menter. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA Journal*, 32(8):1598–1605, August 1994.

[19] Daisie D. Boettner Margaret B. Bailey Michael J. Moran, Howard N. Shapiro. *Fundamentals of Engineering Thermodynamics*. Wiley, eighth edition, 2014.

[20] Sampo Niskanen. Openrocket. Available at https://openrocket.info/documentation.html, 2022. Accessed 05/22/22.

[21] Ahrs: Attitude and heading reference systems. Available at https://ahrs.readthedocs.io/en/latest/index.html, 2022. Accessed 05/22/22.

9. Appendix A: Noncritical Plots

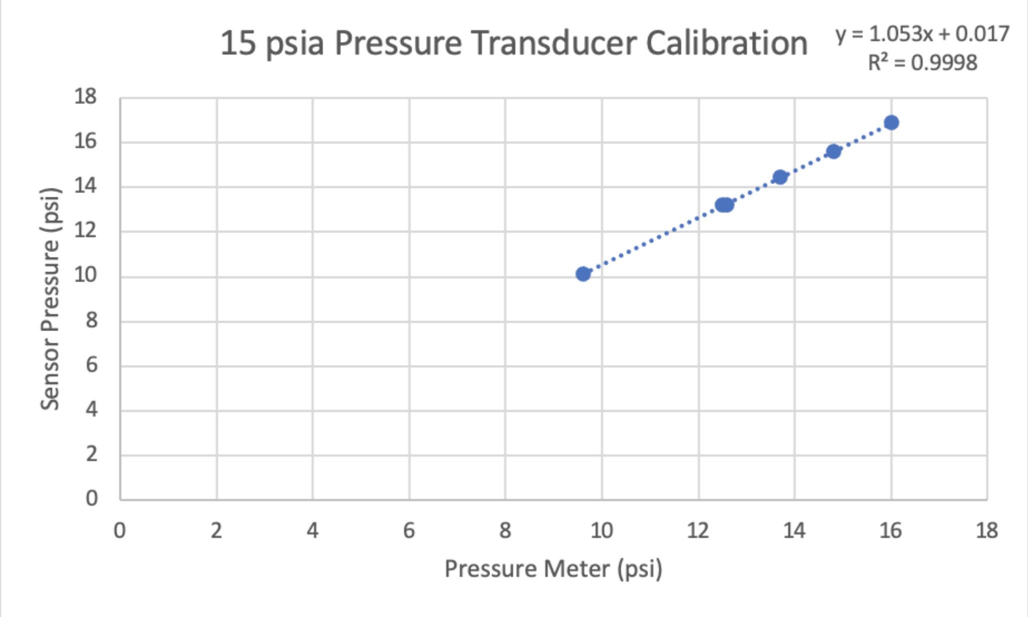

Figure 23 – Pressure transducer (15 psia) calibration, connected to DAQ v2

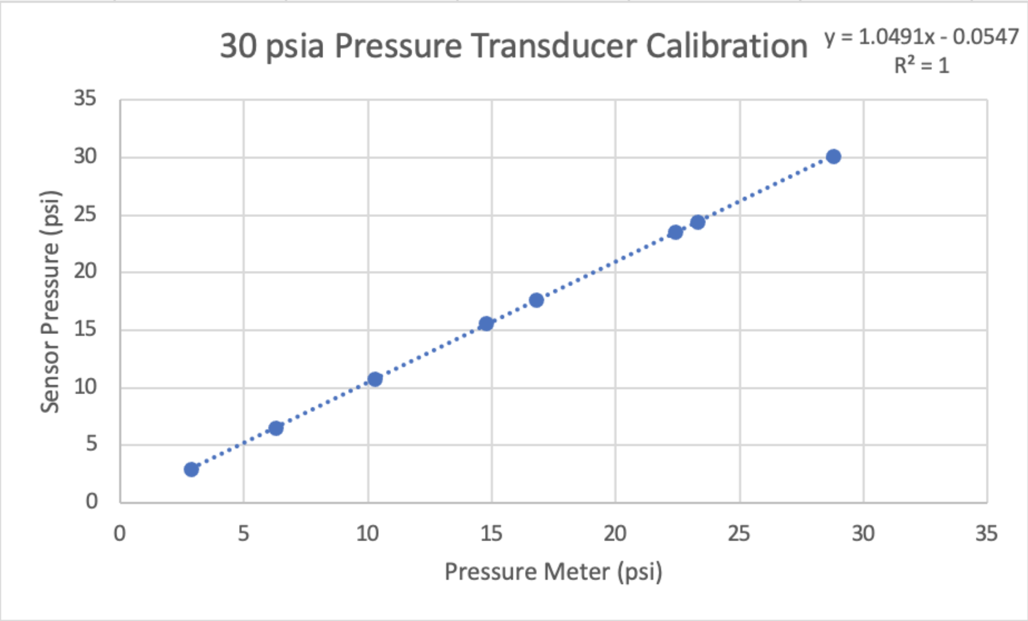

Figure 24 – Pressure transducer (30 psia) calibration, connected to DAQ v2

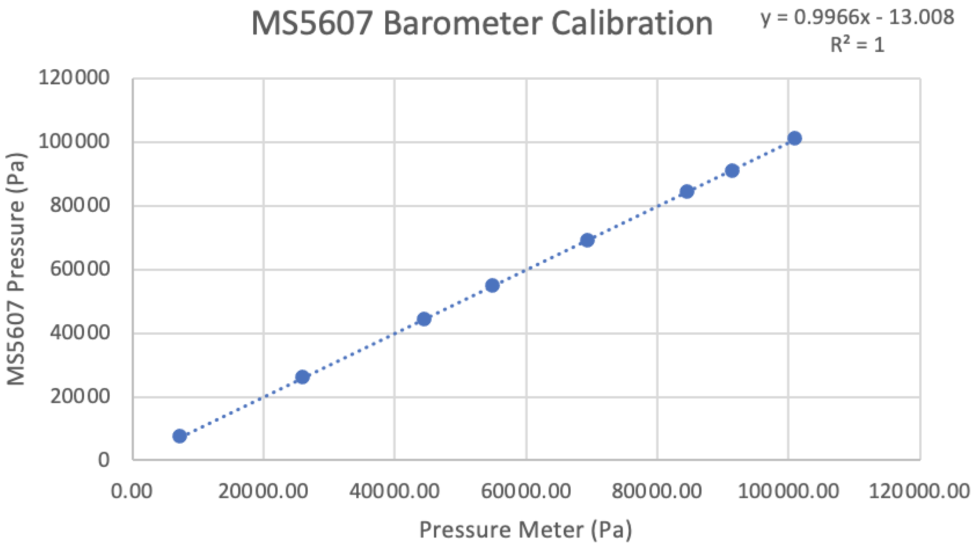

Figure 25 – MS5607 barometer calibration

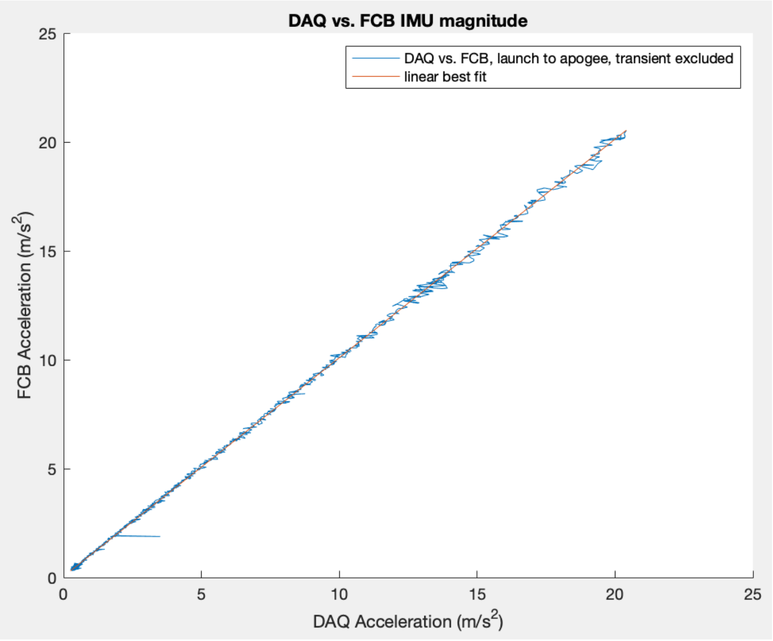

Figure 26 – DAQ and FCB accelerometer magnitudes during coast, Carby November 2021

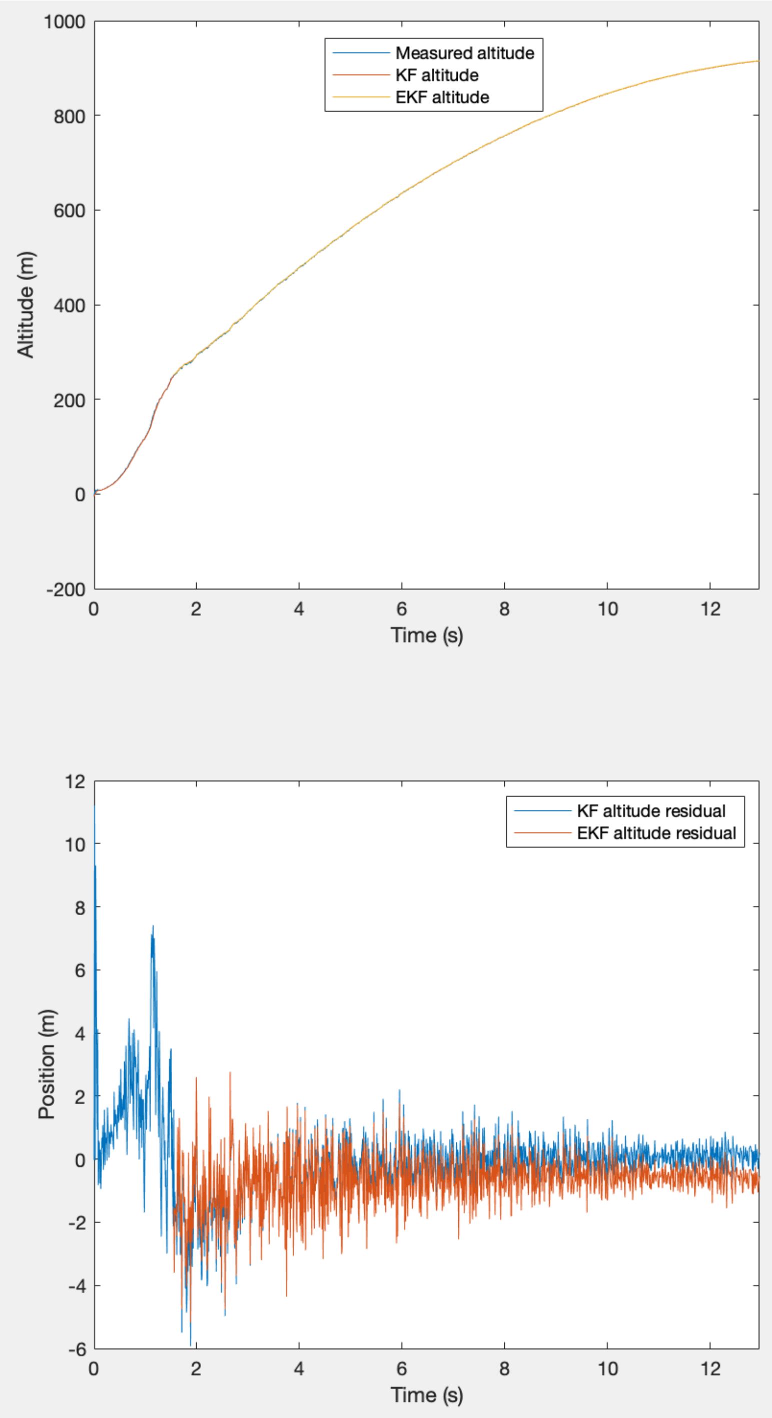

Figure 27 – Filter variables during ascent, TRD August 2021

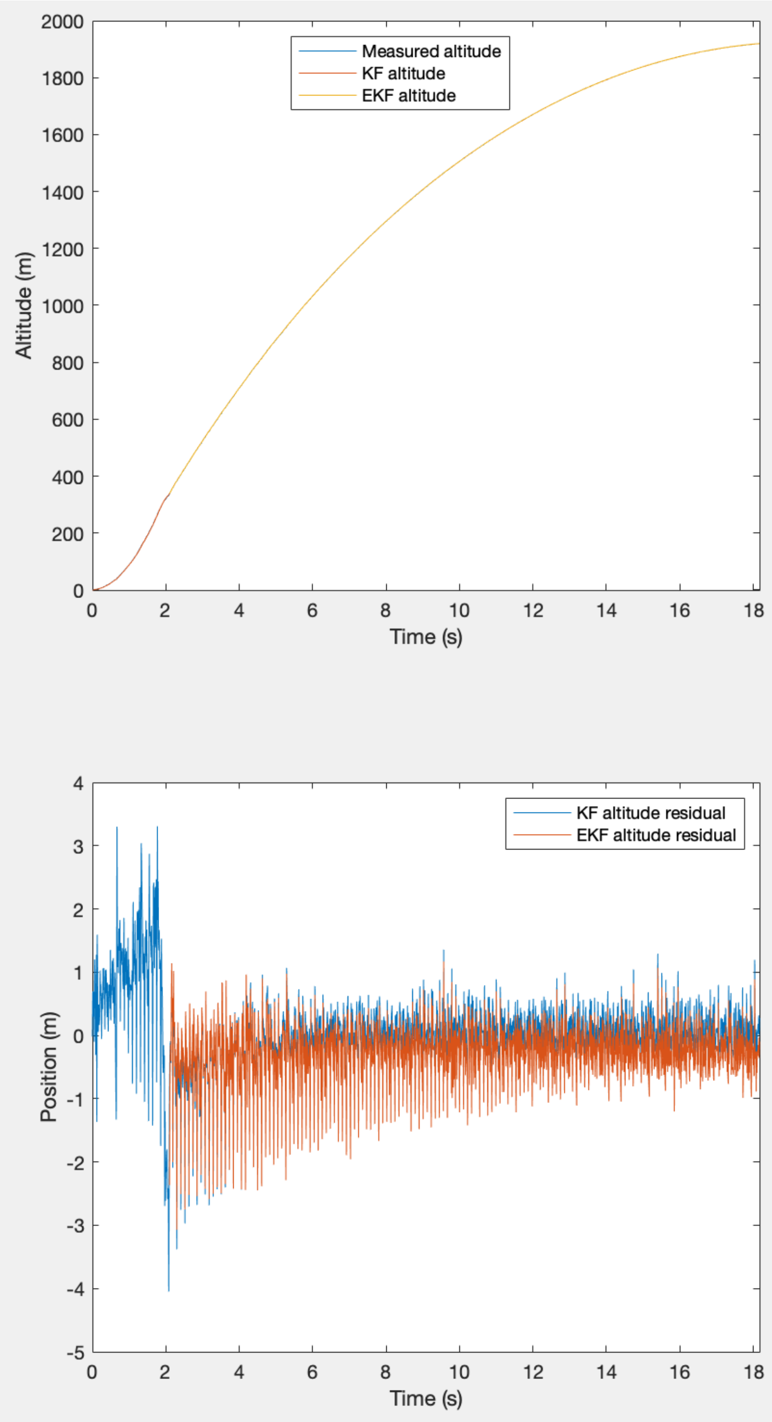

Figure 28 – Filter variables during ascent, Carby September 2021

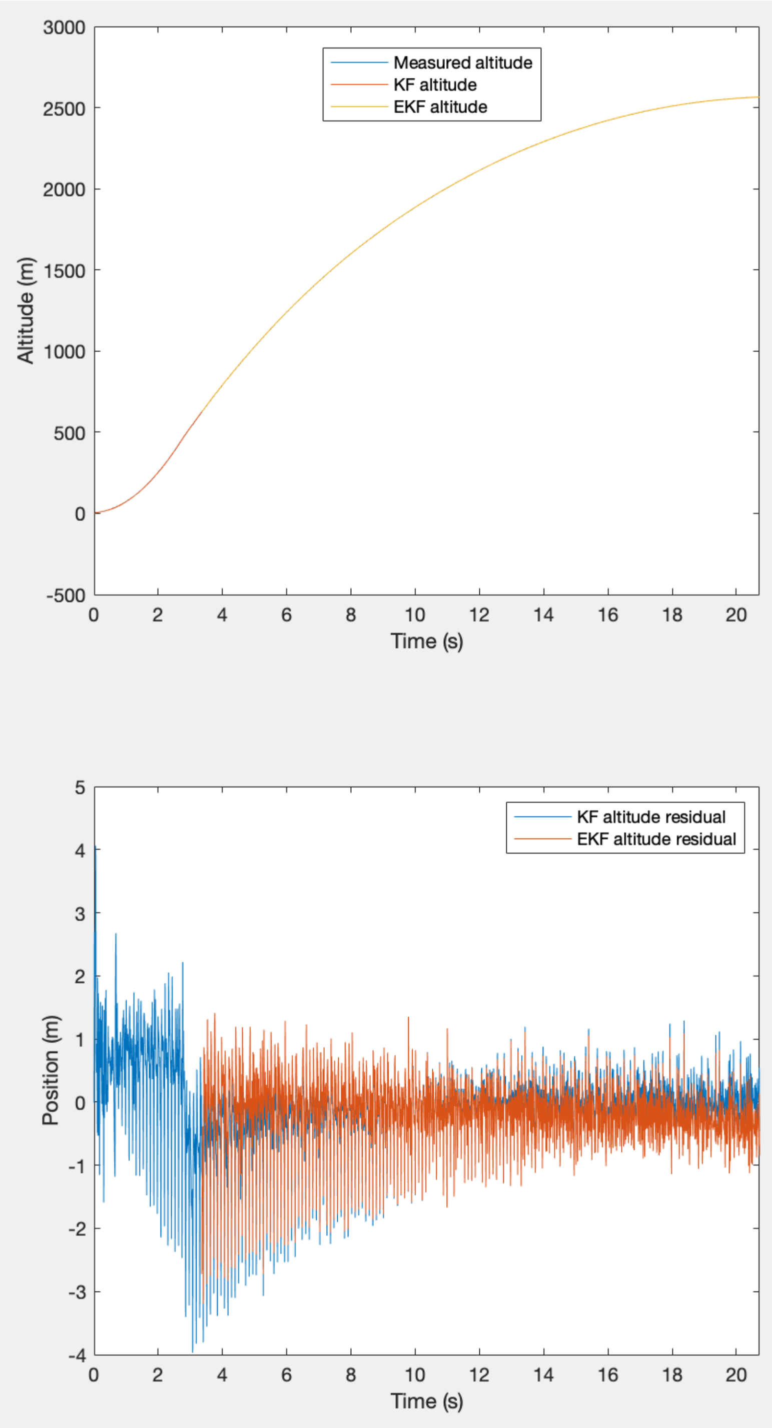

Figure 29 – Filter variables during ascent, Carby November 2021

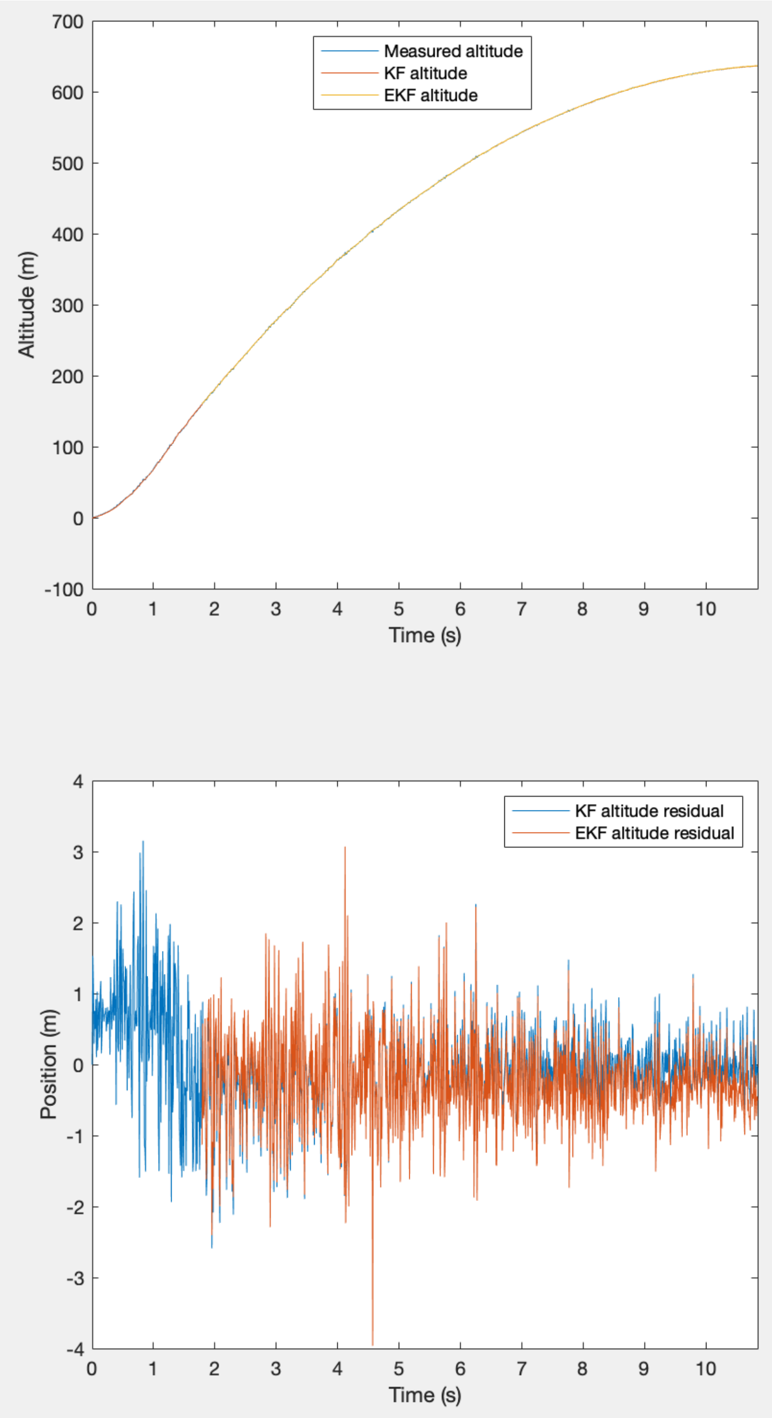

Figure 30 – Filter variables during ascent, TRD April 2022

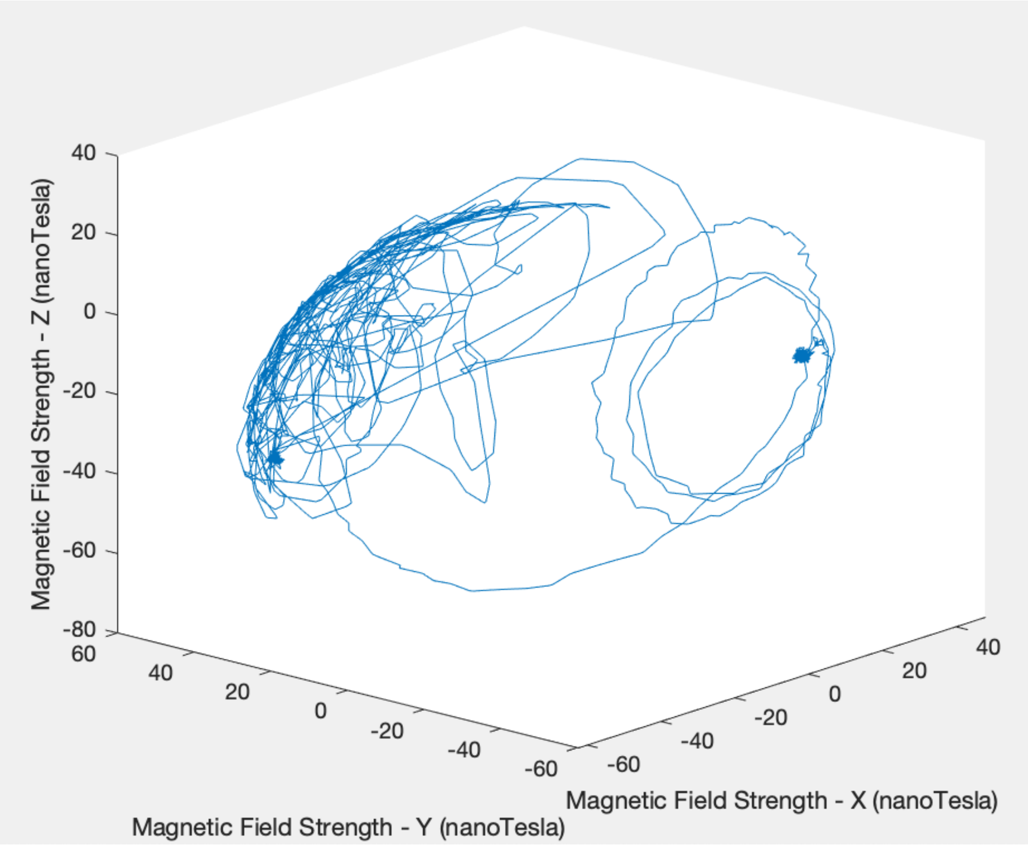

Figure 31 – Corrected magnetometer readings during ascent and descent, TRD April 2022

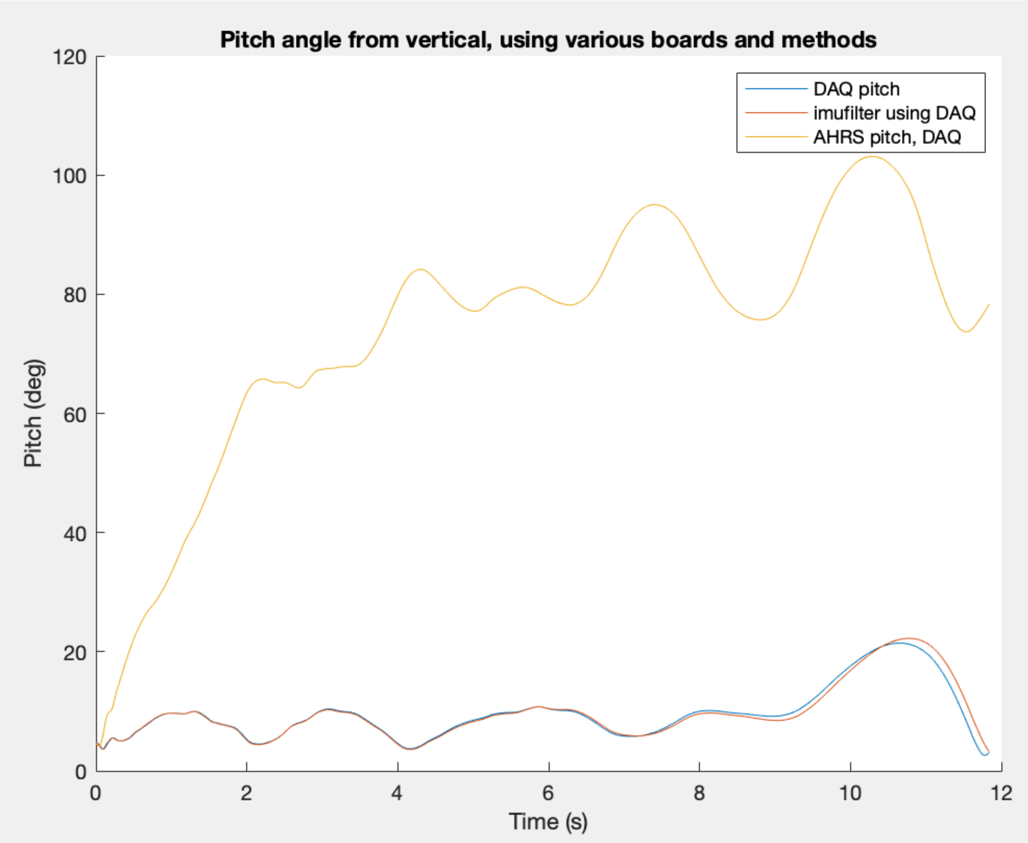

Figure 32 – Tilt angle during during ascent, TRD April 2022

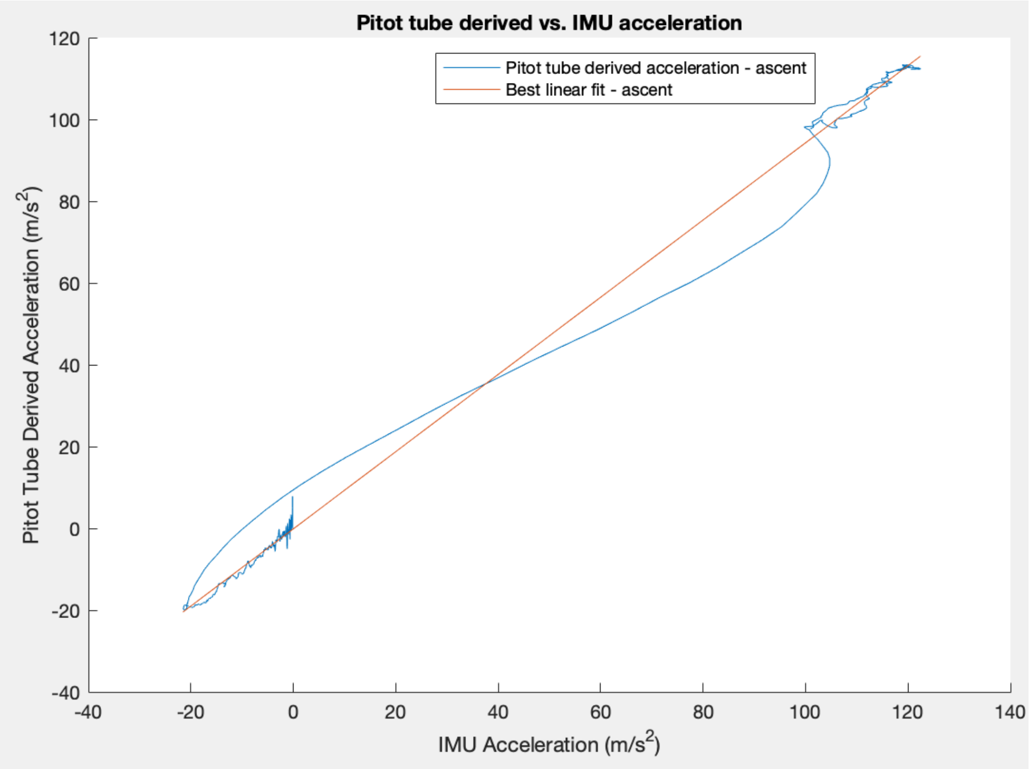

Figure 33 – Pitot tube derived acceleration vs. IMU acceleration during ascent, Carby November 2021

10. Appendix B: Ansys Workflow and Automation

I have found that having a workflow for CFD can be the difference between easily running studies and getting stuck in the setup phase. Therefore for the benefit of posterity I will present an overview of my methods. Geometry preparation was done in SolidWorks, creating a clean single body for the rocket with the origin at the nosecone tip. I created a single body for the surrounding domain and bodies for each refinement region. This was imported to Ansys Fluent's Watertight meshing workflow, and the refinement regions were specified as Bodies of Influence. The mesh was transferred to Ansys Fluent for setup: solver models, boundary conditions and reference values were set, and report were defined for drag, lift, and moment coefficients. The lift and moment coefficients are not strictly necessary but I have found them to be useful in evaluating whether symmetry is modeled correctly. I initialized the case using the hybrid method, then using Fluent's full multigrid initialization. I have found this method to more likely yield a stable calculation, especially in compressible and supersonic cases where divergence is common. I then ran the case fully steady first, then switching to pseudo-transient once it had stabilized. In some cases, I employed solution steering to assist in early convergence.

10.1 Automation Using Journals and Python

I have found that a primary time sink in running CFD cases is performing a mesh study. Ansys has a feature for scripting called Journals, that can reduce this time sink substantially. Journals can be recorded, eliminating the need to write them by hand. I recorded a journal file for setting mesh size parameters and generating a mesh, which output a text file. I replaced the mesh size numbers in the text file with recognizable strings, e.g. !globalSize!, then wrote a python script to read in mesh size parameters from an excel file and replace the strings in the text file. This allowed the easy creation of journal files that could be run individually to mesh each refinement level. Similar methods can be used to automate other steps of the process. I like to duplicate the base module within workbench and have one module for each mesh level - this can be done automatically. Journal files can be recorded for opening a mesh level, initializing and solving it, easing the burden on the user.

10.2 High Performance Computing Automation

The setup and journal file recording described above, as well as a solve with a very coarse mesh was done on a local computer. The workbench file and all associated files were then transferred to a remote server. From there, almost all interaction with the file was done using Journal scripts, eliminating interaction with the GUI. Further automation at the server level is a work in progress. My end goal is to run a python script on the server that then runs Ansys Workbench from the command line with automatically created journal files.