Development and Flight Testing of Energy Management Algorithms for Small-Scale Sounding Rockets

Shannon Eilers*and Shane Robinson†

The development, implementation, and flight results for a navigation algorithm and an energy management system are presented. The navigation algorithm employs both a Kalman filter and an inertial navigation routine. The energy management system includes an asymptotic targeting algorithm and pneumatically deployed air brakes that deplete excess energy from the rocket in flight. These algorithms were developed to fulfill objectives in the NASA USLI competition. The energy management algorithm has been shown to successfully target an apogee altitude on two separate test flights.

I. Introduction

The USU Chimaera project has been involved in the design, construction, testing, and launch of experimental rockets for over ten years. However, for many of these years the Chimaera rocket team did not take a progressive technological path, instead centering on the design and construction of low-cost, low-tech hybrid rockets. Starting with Fall 2007, the Chimaera team shifted its sights from local rocket launches and competitions to the NASA University Student Launch Initiative (USLI). The NASA USLI is a competition that challenges university students to design, build and launch sounding rockets to a mile altitude with a scientific payload. This shift of focus allowed the Chimera rocket team to develop much needed testing and analysis infrastructure that allowed the creation of significantly more sophisticated rockets. In the 2007-2008 academic year, the USU team took home the grand prize in the USLI competition as well as receiving several awards including best presentations and documentation, best payload design, and best team spirit.

During the 2007-2008 academic year, design and analysis centered around an energy management algorithm. Part of the competition score is based upon a "closest to the mark" criterion so the team decided to design a payload that would maximize this part of the score. The design and implementation of a payload capable of accurate energy management required the development of sophisticated navigation and guidance algorithms. The design for the 2007-2008 academic year culminated in the construction of the 2008 USLI grand prize winning rocket, the Barracuda. Unfortunately, during the competition launch of the Barracuda, the avionics package failed and the target altitude was not reached. The 2008-2009 USU USLI team further revised the navigation, guidance, and energy management algorithms flown on the Barracuda and incorporated them into the design of the 2009 USU, USLI entry, the Pike. At the time of the submission of this paper, the navigation and energy management algorithms have been test flown twice, and have demonstrated reasonable accuracy in targeting a final altitude. The test flights have shown several small errors in the algorithm implementation, some of which still require solutions. However, in the words of Dr. Hugh L. Dryden, early NASA scientist and engineer, full-scale flight testing "is to separate the real from the imagined... to make known the overlooked and the unexpected." In that, the flight tests of the algorithms presented in this paper have done quite well.

II. Hardware Overview

The 2009 USU USLI entry, the Pike, is approximately 1.8 meters in length and weights about 9.3 kg without propellant. Aft-mounted air brakes are powered with a pneumatic actuator located in front of the motor assembly. This pneumatic actuator is powered by carbon dioxide and actuates the air brakes by

*Graduate Student, Department of Mechanical and Aerospace Engineering, 4130 Old Main Hill, AIAA Student Member
†Graduate Student, Department of Mechanical and Aerospace Engineering, 4130 Old Main Hill, AIAA Student Member

pulling four steel cables, each of which is connected to an air brake. Each air brake is 10 cm in length and the four air brakes encircle the 15 cm diameter airframe. The overall rocket configuration is illustrated in Fig. 1.

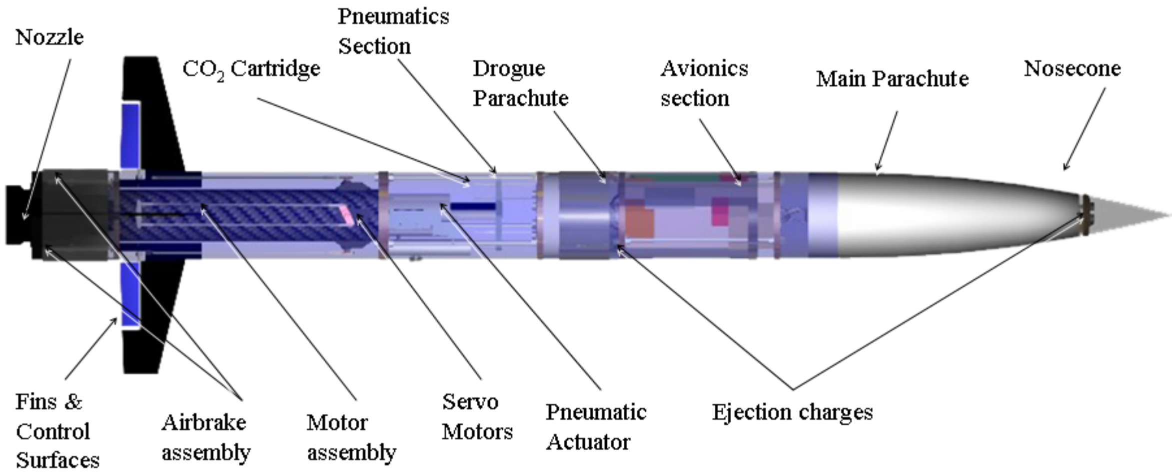

Figure 1. "The Pike" rocket configuration.

Inertial acceleration and attitude measurements are acquired through a Microstrain 3DM-GX2[a] which contains a triaxial accelerometer, triaxial rate gyro, and magnetometer. This sensor contains an internal filter which converges on a 3-axis orientation solution by combining the output of the rate gyros and the magnetometer. This feature allowed greater flexibility in the design of the navigation algorithms, as an attitude solution was provided directly from the IMU. Due to this feature, integration of the rotation rates was not required to provide an orientation solution.

The avionics package also includes a pressure based altimeter, a PerfectFlite MAWD[b]. Although this pressure sensor is not highly accurate, it is the sensor upon which the competition is judged. The error in this sensor required a significant amount of consideration in the design of the navigation and guidance algorithms. In effect, the guidance and navigation algorithms are not trying to predict the altitude and velocity of the rocket in reality, they are trying to converge on a state of the rocket that includes any bias error on the PerfectFlite altimeter.

III. Energy Management Algorithm Development

The physical mechanism for energy management on both the Pike and the Barracuda involved the actuation of pneumatic axi-symmetric air brakes to decrease airspeed at specific way points during flight. As air brakes obviously cannot add energy to the system, a prudent energy management design includes an asymptotic targeting algorithm so that the system will have excess energy until the final apogee altitude can be estimated with a high degree of accuracy. In addition, the physical air brake system on the Pike only allows a finite number of air brake deployments so these deployments must be chosen with care. An algorithm for propagating the state of the rocket from any point during the assent of the rocket to apogee must also be computed. Due to hardware performance constraints, this algorithm must also be written with computational efficiency as a major consideration.

The final algorithm flown on the Pike incorporates four trigger altitudes at which air brakes, if necessary, are deployed, and a corresponding target altitude for each trigger altitude way point. Altitude based triggers were chosen instead of time-based triggers because the fundamental target is altitude based and altitude triggers minimize sensitivity to rocket trajectories with a great deal of excess energy. For example, a rocket with a higher than expected total impulse will be traveling faster than expected and will have traveled higher than a nominal flight by the time a trigger time is reached. This limits the amount of time available for energy depletion through air brake deployments. Altitude-based triggers do not create this potential

[a]http://www.microstrain.com/3dm-gx2.aspx
[b]http://www.perfectflite.com

instability. The rocket guidance algorithm can be summarized in the following steps.

1. When the rocket's estimated altitude passes one of the altitude way points, the state of the rocket is propagated to apogee by integrating a simple ballistic trajectory with the current best estimate of brakes-closed drag coefficient to maximum altitude with a trapezoidal integrator. If this altitude is higher than the target altitude, air brakes are deployed. If the altitude is below the trigger altitude, the rocket proceeds to the next way point and air brakes are not deployed.

2. The excess altitude above the target altitude is used to estimate the excess energy in the state of the rocket. This energy is subtracted for the current velocity of the rocket and the rocket's state is propagated to apogee again. Energy lost due to drag is a strong function of velocity. This dependency will not allow this method to converge on the target altitude in a single iteration. Thus, the maximum altitude is propagated to apogee a third time to minimize this error. This process generates a trajectory that will hit the target altitude given the current state of the rocket. The desired kinetic energy of the rocket with respect to altitude is explicitly curve fit with a parabola. A new curve fit is reproduced every five iterations of the main software loop.

3. At every iteration of the flight computer loop, the current velocity of the rocket is compared to the desired kinetic energy. Once the altitude and kinetic energy of the rocket intersects the desired trajectory curve fit, the air brakes are retracted.

4. The rocket proceeds to the next trigger altitude.

Monte-Carlo analysis was employed to intelligently pick trigger and target altitudes. Trigger altitudes were dispersed such that an air brake deployment at the first trigger altitude would have sufficient time to decrease the energy of the rocket to the required level to meet the target altitude. The last trigger altitude was placed close to the final target altitude to improve the accuracy of the maximum altitude estimation at that deployment, but still be low enough to be able to impact the final altitude. The intermediate altitudes were then placed in between these two altitudes and adjusted through simple trial and error. After the trigger altitudes were picked, the corresponding target altitudes were sequentially adjusted using Monte-Carlo analysis. Motor thrust, aerodynamic coefficients, and atmospheric properties were varied between runs. The target altitude for a given trigger altitude was then picked such that the resulting three-sigma dispersion envelope at apogee after that deployment would remain above the final target altitude (one mile). This process was repeated in series for each of the four trigger altitudes. The trigger altitudes, target altitudes and simulated dispersions are shown in Fig. 2.

IV. Navigation Algorithm Development

In order to accurately propagate the current state of the rocket to apogee, current altitude, velocity, mass, and drag coefficient require accurate estimation. The Pike employs an inertial navigation algorithm during the thrusting portion of flight and a Kalman Filter for state estimation during the coasting portion.

The navigation state was chosen to incorporate enough information to allow accurate maximum altitude prediction while minimizing computation time. The resulting state vector is

$$\mathbf{x} = \begin{bmatrix} h \\ \mathbf{v} \\ \beta_c \\ \beta_o \end{bmatrix} \tag{1}$$

where

$$\begin{aligned} \beta_o &= \frac{C_{d_o} A_{ref}}{2m} \\ \beta_c &= \frac{C_{d_c} A_{ref}}{2m} \end{aligned} \tag{2}$$

and C_{d_o} and C_{d_c} are drag coefficients with the air brakes open and closed, respectively. The constants β_o and β_c are refereed to as ballistic constants for the rest of this paper.

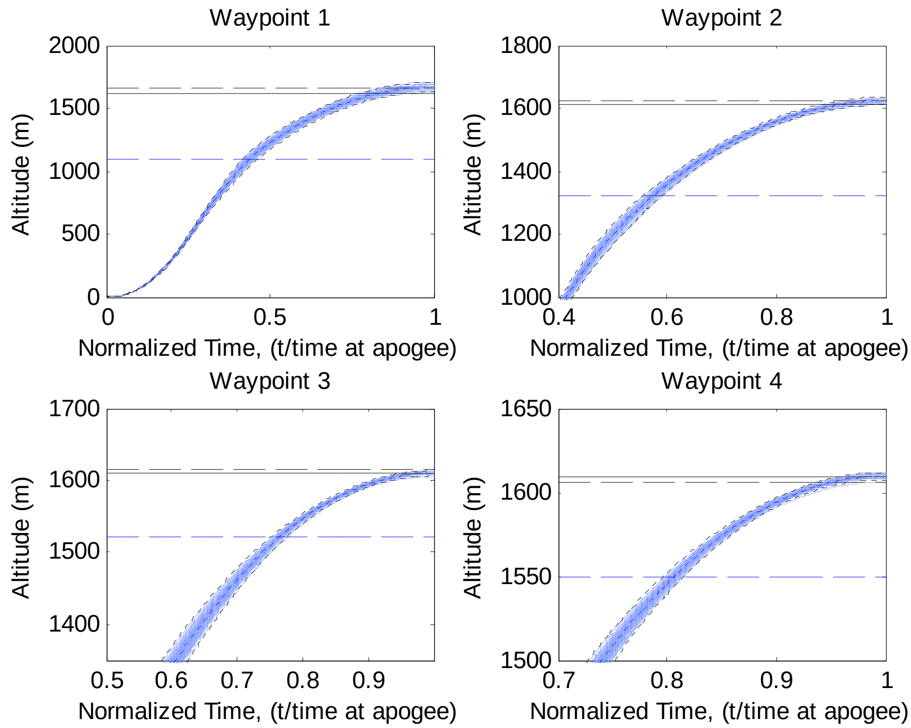

Figure 2. Trigger (blue dashed) and target altitude (black dashed) Monte-Carlo tuning process results. Three-sigma dispersion envelopes are shown in solid blue.

IV.A. Inertial Navigation Development

During the powered portion of flight, the creation of an accurate state model is difficult, as the variance in the rocket motor thrust would need to be accounted for. For this reason, a simple inertial navigation algorithm was employed until the coasting portion of flight began. The inertial navigation algorithm employs a simple reward-looking trapezoidal integration algorithm. The body-fixed accelerations generated by the IMU are transformed to an inertial NED coordinate frame with the direct orientation measurements supplied by the IMU. No integration of the body-fixed angular rates was necessary, as the IMU attitude solution was supported by both the internal rate gyroscopes and the magnetometer. Unfortunately, inertial navigation alone is not sufficient to either force the PerfectFlite to converge on the correct altitude or provide accurate ballistic constant estimation. Thus, development of another navigation routine was required.

IV.B. Kalman Filter Development

An extended Kalman filter was chosen to produce a navigation estimate after motor burnout. It should be noted that the rocket effectively incorporates two Kalman filter's, one for the air brakes-open portion of flight and one for when the air brakes are closed. Hence, the state vector in the filter update equations has five elements instead of the six that are contained in the main avionics algorithm loop. In the development of the Kalman filter, the rocket ballistic constants will be expressed without subscripts denoting air brake status. In the actual filter, these variables are swapped out appropriately after an air brake command is given by the flight computer.

IV.B.1. State and Covariance Propagation

To facilitate computational efficiency, reasonable effort must be given to the choice of a simple, yet sufficiently accurate state model. As the angle of attack of sounding rockets is generally very low and lift is often negligible, ballistic flight often provides sufficient accuracy for apogee prediction. If simple ballistic flight is assumed, the nonlinear state dynamics are

$$\dot{\mathbf{x}} = \begin{bmatrix} v_z \\ -\frac{1}{2}\frac{\rho\mathbf{v}|\mathbf{v}|C_d A_{ref}}{m} - g i_z \\ 0 \\ 0 \end{bmatrix} \tag{3}$$

These equations are integrated with a simple trapezoidal method to propagate between state updates.

For covariance propagation, the state equations must be linearized. The linearized state dynamics for ballistic rocket flight can be represented as

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} \tag{4}$$

where

$$\mathbf{F} = \begin{bmatrix} 0 & i_{\mathbf{z}}^{\mathbf{T}} & 0 \\ \mathbf{0} & -\rho\beta\left(\mathbf{v}i_{\mathbf{v}}^{\mathbf{T}} + |\mathbf{v}|\,\mathbf{I}\right) & -\rho\mathbf{v}\,|\mathbf{v}| \\ 0 & \mathbf{0} & 0 \end{bmatrix} \tag{5}$$

Although the air density, ρ is a function of altitude, the effect of this differential term is small and so it is neglected. However, for state propagation and measurement updates, the density is modeled as an exponential function of altitude of the form

$$\rho{=}\rho_0 e^{\lambda(h-h_0)} \tag{6}$$

where ρ_0 is the density at sea level and λ is a constant derived from temperature variations in the standard atmosphere.

The transformation matrix can be approximated with the first three terms in the series expansion,

$$\Phi = \mathbf{I} + \mathbf{F}dt + \mathbf{F}^2\frac{dt^2}{2} \tag{7}$$

The covariance is then propagated with the discrete Ricatti equation,[1]

$$\mathbf{P}_{i+1} = \Phi\mathbf{P}\Phi^{\mathbf{T}} + \mathbf{Q}_d \tag{8}$$

where the process noise, $\mathbf{Q}_d$, is

$$\mathbf{Q}_d = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.3^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.3^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & (3E-6)^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (5E-6)^2 \end{bmatrix} dt \tag{9}$$

The constant values in the process noise relation were tuned through Monte-Carlo simulation. Time tags for measurements and state propagation were received from the Microstrain IMU. The entire filter operated at approximately 30 Hz in flight conditions.

IV.B.2. State Update Equations

The Kalman filter has two measurement inputs, the PerfectFlite altitude measurement and the inertial measurement unit's accelerometer measurement. The incorporation of the PerfectFlite measurement is straightforward. The measurement partial for the altimeter update is simply

$$h_{PF} = \begin{bmatrix} 1 & \mathbf{0}_{1,3} & 0 \end{bmatrix} \tag{10}$$

As the PerfectFlite altimeter was the standard upon which altitude was judged, the measurement noise was adjusted so that the filter would converge upon the the PerfectFlite measurement as the estimated altitude approached the final target altitude. The PerfectFlite measurement noise was adjusted according to

$$R_{PF} = c_{PF}\left(h_{\mathbf{target}} - h\right)^2 \tag{11}$$

where c_{PF} was a constant adjusted through Monte Carlo simulation and $h_{\mathbf{target}}$ was one mile altitude.

The accelerometer measurements were given somewhat novel treatment in the Kalman filter to allow the direct estimation of drag coefficient and velocity. The measurement equation for the IMU measurement is a simple relation for the specific forces on the rocket in flight.

$$\mathbf{a}_{\mathrm{IMU}} = -\frac{1}{2}\frac{\rho\mathbf{v}\left|\mathbf{v}\right| C_d A_{ref}}{m} \tag{12}$$

This results in the measurement partial

$$h_a = \begin{bmatrix} 0 & -\rho\beta\left(\mathbf{v}i_{\mathbf{v}}^{\mathbf{T}} + \left|\mathbf{v}\right|\mathbf{I}\right) & -\rho\mathbf{v}\left|\mathbf{v}\right| \end{bmatrix} \tag{13}$$

The covariance matrix propagated during flight only includes the covariance of either the air brakes open or closed ballistic coefficient. To avoid complications in swapping covariance terms after each air brake deployment, the elements in the same row and column as the ballistic constant uncertainty are reset to zero after each air brake actuation. Clearly, information here is lost using this method, but the overall stability of the filter is not compromised by using erroneous covariance values.

V. Flight Test Results

The navigation and energy management algorithms have been flown in the final USLI competition in Huntsville, Alabama. The algorithm converged approximately on the correct altitude, unfortunately, the noise on the pressure based altimeter made convergence on the final altitude difficult. The PerfectFlite had approximately 30 meters of noise from peak to peak. The "competition altitude" was the maximum value reported by the PerfectFlite altimeter, which was therefore about 15 meters high. However, the filter and algorithm showed a surprising amount of resilience to this noise and the smoothed altitude is only approximately 6 meters low. The target altitude convergence for the competition launch is shown in Fig. 3. The on-board estimated altitude, vertical velocity, and ballistic constants are shown in Figures 4 through 6. The actual air brake deployments as photographed by an on-board camera are shown in Fig. 7.

VI. Conclusions and Future Work

Although the navigation and guidance algorithms flown on the Pike have demonstrated enough accuracy to be declared successful, further improvements could be made to increase accuracy in future flights. During the competition launch near Huntsville, Alabama, the PerfectFlite altimeter exhibited a noise level in excess of 30 meters from peak to peak. This noise level vastly exceeded the expected amount of noise on the signal, so the maximum value for altitude reported by the PerfectFlite altimeter exceeded the target altitude. Post flight processing of the data indicated that the noise spikes corresponded directly to periods when the avionics package was transmitting data to the ground. The PerfectFlite data and telemetry points are shown in Fig. 8. It is estimated that improvement in electromagnetic shielding of the altitude sensor would substantially reduce this noise level and improve the effectiveness of both the navigation and energy management algorithms.

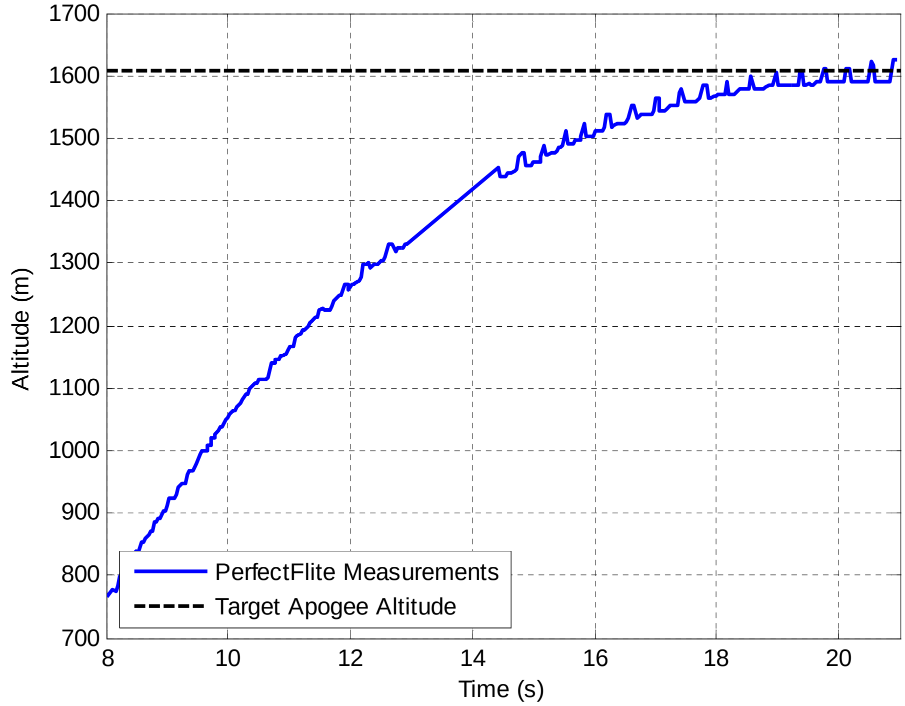

Figure 3. Target altitude convergence.

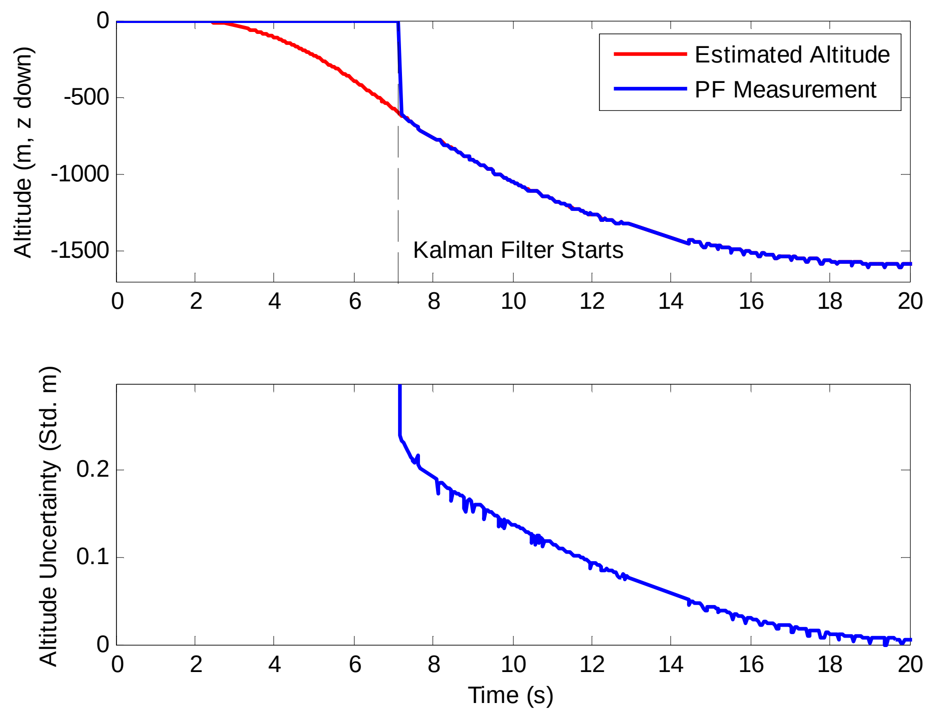

Figure 4. In-flight Kalman filter altitude estimation.

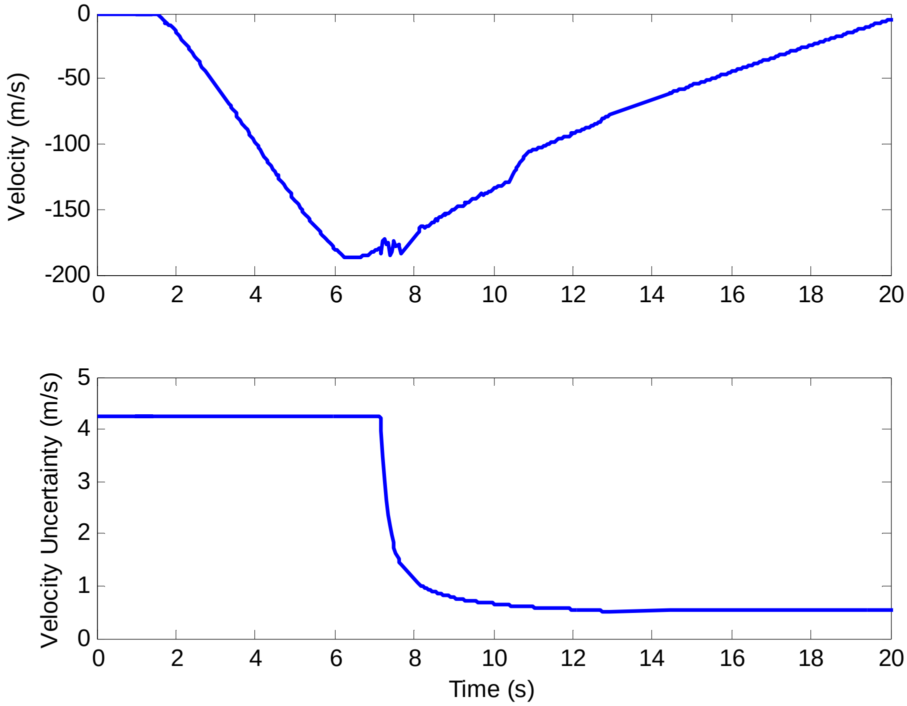

Figure 5. In-flight Kalman filter velocity estimation.

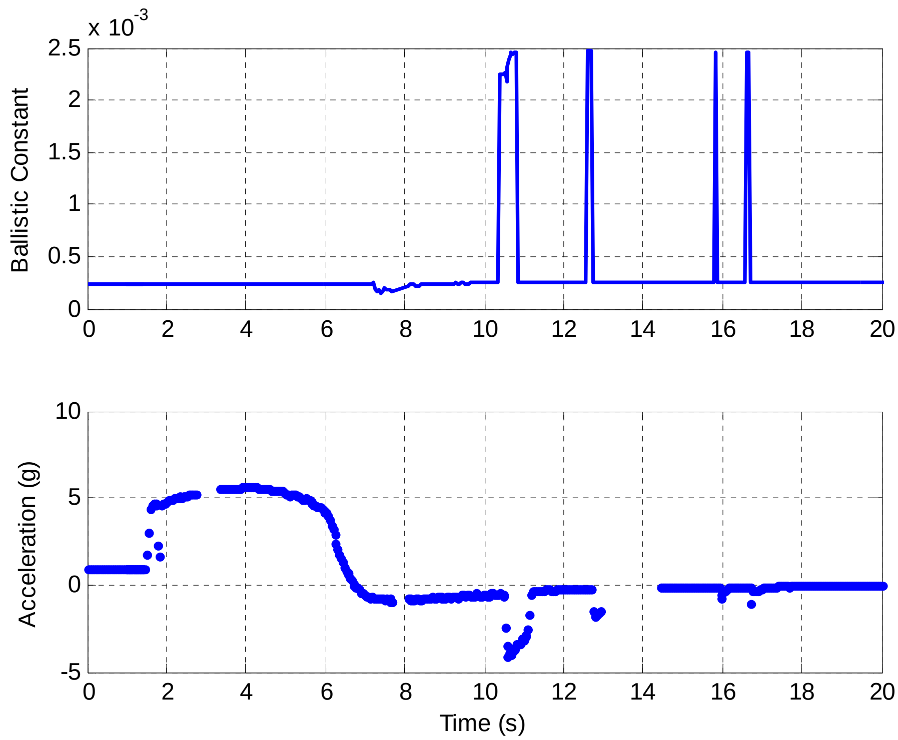

Figure 6. In-flight Kalman filter ballistic constant estimation (top) and acceleration (bottom).

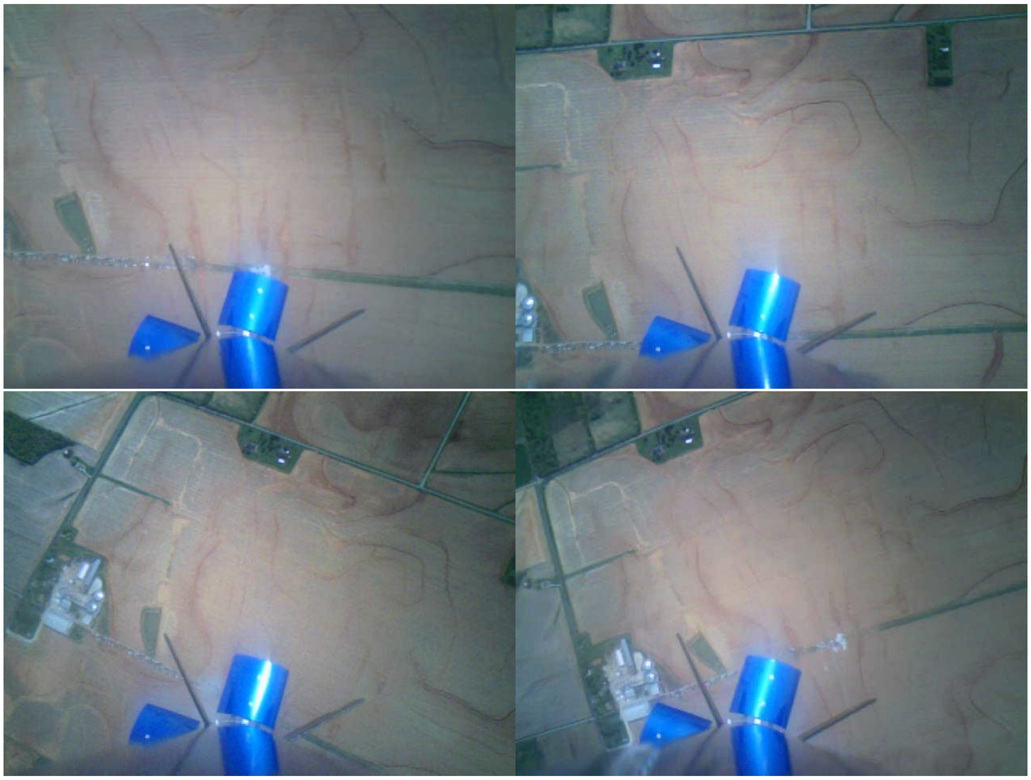

Figure 7. Air brake deployments (sequential from top left to bottom right).

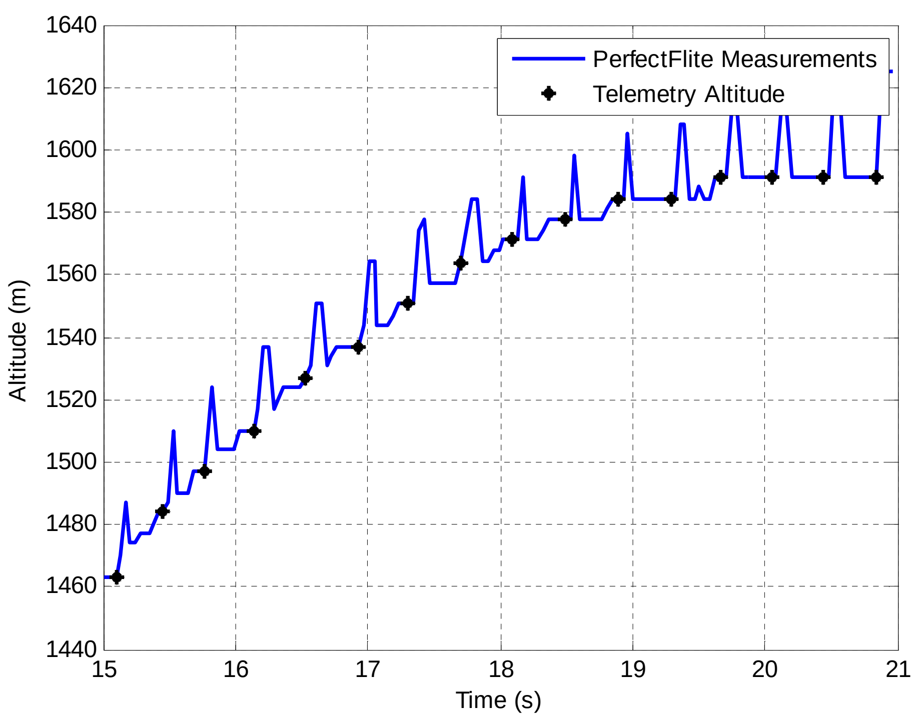

Figure 8. PerfectFlite altitude and telemetry points.

References

[1] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:35–45, 1960.