

# The Model and Control Development for an Autonomous Parafoil Return Vehicle

Eric Dickey<sup>1</sup>  
Matt Fetig  
Kenneth Harcsztark  
Mitchell Harris  
Daniel Hood  
John Lee  
Thanh Nguyen  
Aaron Okken  
Mark Schwartz  
Christopher Sprague

*University of Colorado Boulder, Boulder, CO, 80303*

**In an effort to design a vehicle that recovers high altitude balloon launched scientific payloads for the University of Colorado Space Grant Consortium, it was decided that an autonomous parafoil would be the most effective way to accomplish this goal. Currently, no commercial products exist for the strict purpose of autonomously controlling a parafoil; therefore, one had to be designed. This paper details the theory and means of designing and verifying a six degree of freedom model that will be used in the development of a parafoil control system and autopilot for the return of high altitude balloon launched payloads.**

## Nomenclature

$\bar{c}$	=	mean geometric chord
$\bar{d}$	=	control flap width
$\delta_a$	=	asymmetric control deflection
$\delta_{bias}$	=	control bias
$g$	=	gravitational force
$I_x$	=	moment of inertia
$I_{xl}$	=	inverse moment of inertia
$I_{xy}$	=	cross product of inertia
$I_{xyl}$	=	inverse cross product of inertia
$m_t$	=	total system mass
$p, q, r$	=	roll, pitch, and yaw rates
$\dot{p}, \dot{q}, \dot{r}$	=	roll, pitch, and yaw accelerations
$x, y, z$	=	position vector components
$\dot{x}, \dot{y}, \dot{z}$	=	velocity vector components
$u, v, w$	=	velocity components acting in the positive $x, y,$ and $z$ directions, respectively
$V_a$	=	total linear velocity

## I. Introduction

**I**N 2003, the Colorado Space Grant Consortium, along with the University of Colorado's Aerospace Engineering program, started a class called "Gateway to Space" where university students had the opportunity to build and launch BalloonSats. Each semester about 10 BalloonSats are launched by weather balloon in cooperation with Edge of Space Sciences to an altitude of up to 100,000 feet. The BalloonSats are mostly 5"x5"x5" cubes weighing a maximum of 1.5 pounds a piece and carry out a number of scientific experiments as designed by the students. These experiments range from taking pictures of the ascent and decent to creating a temperature profile of the excursion.

---

<sup>1</sup> All authors: Students, Aerospace Engineering Sciences Department, 429 UCB, no AIAA membership.

When the weather balloon bursts at 100,000 feet, a parachute deploys and the BalloonSats return to earth in an uncontrolled manner, resulting in drifts of up to 50 miles away from the launch site. In addition, the uncontrolled parachute may land in areas that are dangerous or hard to reach, potentially making the recovery of the BalloonSats impossible.

In the 2006-07 school year, the Peregrine Return Vehicle (PRV) team first took the challenge of trying to return the BalloonSats to the launch site. Due to weight restrictions, however, they were limited to returning the BalloonSats to a location that was within a quarter mile of a road, which would greatly ease the recovery effort. The PRV team succeeded in constructing a blended wing body glider that successfully demonstrated the ability to fly without user input in several hot air balloon tests, but during the test up to 100,000 ft, the cutaway device to separate the vehicle from the balloon broke and was unable to attempt an autonomous return. Along with the failure of the cutaway device the aircraft was unstable with the payloads on board and was forced to be tested empty. The High Altitude Research Return Vehicle (HARRV) team will take on the task of trying to solve this unique problem with a parafoil. But before this task can be carried out, a theoretical model around which an autopilot could be constructed must first be verified.

## II. Aerodynamic Model

This section describes the predictive control model, based on the Costello and Slegers<sup>1</sup> model predictive control strategy, developed from a six degree of freedom (DOF) model for a parafoil and payload system. A non-linear six DOF model that accurately predicts the flight dynamics of the parafoil and payload system is used for the basis of a reduced-order, linear model. This reduced-order model requires five aerodynamic coefficients:  $C_{l\phi}$ ,  $C_{l\psi}$ ,  $C_{l\delta_a}$ ,  $C_{nr}$ ,  $C_{n\delta_a}$ , and a constant bias term  $\delta_{bias}$  that are identified by system identification. Using the method developed by Costello and Slegers, recursive weighted least squares estimation is used to identify the five aerodynamic terms and the bias term. Through flight tests of the parafoil and payload system the reduced-order, linear model and the identified terms will be verified by demonstrating that the reduced model can accurately model the parafoil and payload system. Once verified, a directional control system that can track points in the x-y plane will be implemented.

The parafoil and payload bay are modeled dynamically using a six DOF model composed of three Euler orientation angles and three inertial position components. The six DOF model assumes that the fully inflated parafoil, support lines, and control lines are rigid. In addition, the parafoil, payload, and connection point are assumed to be coupled, allowing the system only three degrees of translation and three degrees of rotation. The parafoil and payload system is controlled by left and right trailing edge control surfaces (known as brakes) that effectively increase the drag on their respective sides of the system; deflecting one control surface creates yawing and rolling moments, while deflecting both control surfaces creates a pitch moment. This is accomplished using two wench servos that shorten the brakes lines, thereby deflecting the brakes. The kinematic equations for the motion of the parafoil and payload system are given by Eqs. (1) and (2).

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = S^T \begin{pmatrix} u \\ v \\ w \end{pmatrix} \quad (1)$$

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & s_{\phi} t_{\theta} & c_{\phi} t_{\theta} \\ 0 & c_{\phi} & -s_{\phi} \\ 0 & s_{\phi} \sec \theta & c_{\phi} \sec \theta \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \quad (2)$$

The matrix  $S$  is the transformation matrix from the inertial reference frame to the parafoil body-axis frame and is given Eq. (3).

$$S = \begin{pmatrix} c_{\psi} c_{\theta} & s_{\psi} c_{\theta} & -s_{\theta} \\ c_{\psi} s_{\theta} s_{\phi} - s_{\psi} c_{\phi} & s_{\psi} s_{\theta} s_{\phi} + c_{\psi} c_{\phi} & c_{\theta} s_{\phi} \\ c_{\psi} s_{\theta} c_{\phi} + s_{\psi} s_{\phi} & s_{\psi} s_{\theta} c_{\phi} - c_{\psi} s_{\phi} & c_{\theta} c_{\phi} \end{pmatrix} \quad (3)$$

where  $c$  and  $s$  are shorthand notation for the trigonometric functions cosine and sine, respectively.

The dynamic equations of motion, which include the aerodynamic forces and aerodynamic moments acting around the system center of mass, are given by Eqs. (4) and (5). These equations are derived from the momentum conservation laws along with the external forces and moments acting on the vehicle.

$$\begin{Bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{Bmatrix} = \frac{1}{m_t} \left[ \frac{1}{2} \rho S V_a (C_{D0} + C_{D\alpha^2} \alpha^2 + C_{L\delta_a} \delta_a) \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} + m_t \mathbf{g} \begin{Bmatrix} -s_\Theta \\ c_\Theta s_\Phi \\ c_\Phi \Theta \end{Bmatrix} \right] - T S_w T^T \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} \quad (4)$$

$$\begin{Bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{Bmatrix} = \begin{Bmatrix} I_{xI} & 0 & I_{xzI} \\ 0 & I_y & 0 \\ I_{xzI} & 0 & I_{zI} \end{Bmatrix} \left[ \frac{1}{2} \rho S \bar{c} V_a^2 \begin{Bmatrix} C_{l\phi} \Phi + \frac{C_{lp} \bar{c} p}{2V_{air}} + \frac{C_{l\delta_a} \delta_a}{d} \\ C_{m0} + C_{m\alpha} + \frac{C_{mq} \bar{c} q}{2V_{air}} \\ \frac{C_{nr} \bar{c} r}{2V_{air}} + \frac{C_{n\delta_a} \delta_a}{d} \end{Bmatrix} - \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{Bmatrix} I_x & 0 & I_{xz} \\ 0 & I_y & 0 \\ I_{xz} & 0 & I_z \end{Bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \right] \quad (5)$$

In order to utilize predictive control, the non-linear model given by Eqs. (1) to (5) must be linearized. We can see in Eq. (1) that the yaw angle is related to the inertial velocities and simply linearizing would constrain the yaw to small angles. From observing typical parafoil flights, it is concluded that a small yaw angle constraint is not acceptable to model the expected turn dynamics. By assuming a constant velocity  $V_a$  and steady state conditions, however, a reduced-order, linearized model may be formed. This is done by using the rolling and pitching equations,  $[\dot{\phi} \ \dot{\psi} \ \dot{p} \ \dot{r}]^T$ , given by Eqs. (2) and (5). The reduced-state, linear model is given in Eqs. (6) and (7).

$$s_0 = \begin{Bmatrix} \Phi_0 \\ \Theta_0 \\ p_0 \\ q_0 \\ r_0 \\ \delta_{a0} \end{Bmatrix} = \begin{Bmatrix} 0 \\ \Theta_0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad (6)$$

$$\begin{Bmatrix} \dot{\phi}^* \\ \dot{\psi}^* \\ \dot{p}^* \\ \dot{r}^* \end{Bmatrix} = \begin{Bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{c_{\Theta_0}} \\ \frac{1}{2} \rho S \bar{c} V_{air}^2 C_{l\phi} I_{xI} & 0 & \frac{1}{4} \rho S \bar{c} V_{air}^2 C_{lp} I_{xI} & \frac{1}{4} \rho S \bar{c} V_{air}^2 C_{nr} I_{xzI} \\ \frac{1}{2} \rho S \bar{c} V_{air}^2 C_{l\phi} I_{zI} & 0 & \frac{1}{4} \rho S \bar{c} V_{air}^2 C_{lp} I_{zI} & \frac{1}{4} \rho S \bar{c} V_{air}^2 C_{nr} I_{zI} \end{Bmatrix} \begin{Bmatrix} \phi^* \\ \psi^* \\ p^* \\ r^* \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ \frac{\rho S \bar{c} V_{air}^2 (C_{l\delta_a} I_{xI} + C_{n\delta_a} I_{xzI})}{2d} \\ \frac{\rho S \bar{c} V_{air}^2 (C_{l\delta_a} I_{zI} + C_{n\delta_a} I_{zI})}{2d} \end{Bmatrix} \{\delta_a^*\} + \begin{Bmatrix} 0 \\ 0 \\ \frac{\rho S \bar{c} V_{air}^2 I_{xzI}}{d} \\ \frac{\rho S \bar{c} V_{air}^2 I_{zI}}{d} \end{Bmatrix} \{\delta_{bias}\} \quad (7)$$

Using model predictive control, a control law can be developed from the reduced-order, linearized state space system. Predictive control offers a robust method to deal with uncertainties or disturbances in the system. The formulation of the state space system is shown in Eq. (8).

$$\begin{Bmatrix} \mathbf{x}_{k+1} \\ \mathbf{y}_k \end{Bmatrix} = \begin{Bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{Bmatrix} \begin{Bmatrix} \mathbf{x}_k \\ \mathbf{u}_k \end{Bmatrix} + \begin{Bmatrix} \mathbf{D} \\ \mathbf{0} \end{Bmatrix} \quad (8)$$

The theory for model predictive control can be found in its entirety in Ikonen and Najim.<sup>3</sup> Predictive control functions by calculating the estimated future control sequence across a finite set of future time instances called the prediction horizon,  $H_p$ . The state space equations can be reformulated to calculate the predicted states across  $H_p$  as shown in Eq. (9).

$$\hat{\mathbf{Y}}_{k+1} = \mathbf{K}_{CA} \mathbf{x}_k + \mathbf{K}_{CAB} \mathbf{u}_k + \mathbf{K}_{CAD} \quad (9)$$

where

$$\mathbf{K}_{CA} = \begin{bmatrix} \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{H_p} \end{bmatrix} \quad (10)$$

$$\mathbf{K}_{CAB} = \begin{bmatrix} \mathbf{CB} & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{CA}^{H_p-1} \mathbf{B} & \dots & \mathbf{CB} \end{bmatrix} \quad (11)$$

$$\mathbf{K}_{CAD} = \begin{bmatrix} \mathbf{CD} \\ \mathbf{CAD} + \mathbf{CD} \\ \mathbf{CA}^2\mathbf{D} + \mathbf{CAD} + \mathbf{CD} \\ \vdots \\ \mathbf{CA}^{H_p-1}\mathbf{D} + \mathbf{CA}^{H_p-2} + \dots + \mathbf{CD} \end{bmatrix} \quad (12)$$

From the predicted model and assuming that a desired trajectory  $\mathbf{W}$  is known, an estimated error from the desired trajectory can be found through Eq. (6). By using the estimated error over the entire time set of  $H_p$ , a quadratic cost function can be formed as shown in Eq. (14).

$$e_k = w_k - y_k \quad (13)$$

$$J = (\mathbf{W} - \hat{\mathbf{Y}})^T (\mathbf{W} - \hat{\mathbf{Y}}) + \mathbf{U}^T \mathbf{R} \mathbf{U} \quad (14)$$

where  $\mathbf{R} = \mathbf{rI}$ .

Minimizing the cost function yields the optimal predictive controller input  $\mathbf{U}$ . The solution to minimizing the cost function is shown in Eq. (15).

$$\mathbf{U} = \mathbf{K}(\mathbf{W} - \mathbf{K}_{CA}\mathbf{x}_k - \mathbf{K}_{CAD1}) \quad (15)$$

where

$$\mathbf{K} = (\mathbf{K}_{CAB}^T \mathbf{K}_{CAB} + \mathbf{R})^{-1} \mathbf{K}_{CAB}^T$$

Since the controller updates the prediction at every instant, only the first control input in the control horizon is of interest. Therefore, Eq. (15) can be reformed into Eq. (16) to extract just the next control input and reduce the on-board computational overhead.

$$u_k = \mathbf{K}_1(\mathbf{w}_{k+1} - \mathbf{K}_{CA}\mathbf{x}_k - \mathbf{K}_{CAD1}) \quad (16)$$

where  $\mathbf{K}_1$  is the first row of  $\mathbf{K}$ .

The state space system will be identified before flight, so the gain matrices  $\mathbf{K}_1$  and  $\mathbf{K}_{CA}$  are computed beforehand to implement the controller in the flight control system.

### III. Vehicle

#### A. Structure

##### 1) Purpose and Refinement

The initial purpose of the main structure – not including the parafoil – was to hold the electronics, backup parachute, release mechanism, parafoil, harness, and five cubesat payloads. However, after the conclusion of the design process in January 2008, it was decided that the initial project goals had to be descoped, and the new goal focused purely on validating an aerodynamic model of the vehicle. This inevitably removed nearly all structural requirements, including weight and payload integration. As a result, the structure devolved into a simple box structure that will be easy and inexpensive to re-manufacture if the structure were ever to be destroyed. This is realized to be a significant advantage because of the time limitation placed on the team.

##### 2) Components

The only components of the final structure are the backup parachute and the wooden box with plexiglass top, which are described in more detail below.

**Backup parachute:** This parachute is a typical dome-type chute and is unlike the main parafoil. It is required for any moderately high-altitude model verification drop test to avoid having it freefall in the case of a parafoil failure.

**Wood box with plexiglass top:** This is the structure that holds all of the electronics, backup parachute, etc. The top of the box is plexiglass to allow the on-board GPS unit to transmit to orbiting satellites.

##### 3) Contributions to Goals

The most obvious contribution this subsystem makes is that it provides the capability of holding and containing all other subsystems so that they can make measurements and validate the model. It also has the added benefits of protecting the other subsystems from damage upon landing, protecting the other subsystems from adverse weather conditions, being sturdy and durable, and, most importantly, being very stable in flight.

## **B. Electronics**

### *1) Primary Functional Requirements*

The primary purpose of the electronics is to allow for the verification of our six degree of freedom model. Several functions must be performed in order to complete this task. First, the electronics must maintain physical control over the vehicle. Two winch servos are used to control the parafoil, which in turn must be controlled by the electronics. Additionally, the capability must exist to initiate deployment of the vehicle's backup parachute if necessary. Next, the electronics must take physical data from a variety of sensors for use in our model verification. This data must be easily offloaded from the system for post-flight analysis. Finally, it is desirable to be able to override autonomous operation from the ground in the event of an emergency, so there must be a means to communicate with the vehicle wirelessly. In the event that our secondary objective of a guidance, navigation, and control (GNC) system is pursued, the electronics must also be capable of determining the absolute position of the vehicle.

### *2) Description of Subsystem Design*

The core of the electronics system is a 32-bit ARM7 microcontroller. It was decided that a 32-bit microcontroller (as opposed to the simpler 8-bit) would be used because a wider data path would allow for much faster real-time data processing. Although most, if not all, of the system verification data will be analyzed on a PC after each flight, implementing a guidance system will likely require some real-time data filtering as well as 32-bit floating point calculations. One megabyte of SRAM is available for temporary data buffering.

A variety of analog sensors will be used for model verification, including MEMS accelerometers (3 axis), MEMS rate gyros (3 axis), static and dynamic pressure, and temperature. Two 16-bit, six-channel analog to digital converters allow for the simultaneous sampling of up to 12 analog sensors at a rate of 250,000 samples per second per channel. A digital compass and an embedded GPS receiver are included to allow for a GNC system to be implemented if time allows.

While SRAM can be used for temporarily storing data, all data sampled will also be logged to an SD memory card in FAT16 format. The SD card will also be used as the primary means of user input via configuration files. The servo control signals from the microcontroller will be multiplexed with those from a commercial RC receiver, allowing for both autonomous and manual control. The igniter for the backup parachute ejection charge can be fired using a built-in high current transistor switch capable of delivering up to 5 amps for at least one second. Communication with the ground station will take place over a 1 watt, 900 MHz serial RF modem, which can communicate at ranges of at least 10 miles.

The electronics will be built on two vertically stacked, four-layer PCB's. Surface mount components will be used wherever possible to save board space. Separate analog and digital ground planes will exist, as well as a separate power plane for each voltage required by system components. Analog power for the sensors and analog to digital converters will be locally regulated.

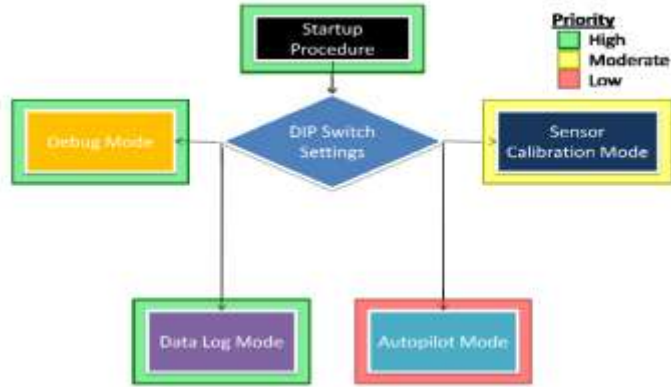
## **IV. Software**

Figure 1 below shows the top level software flow for the system. The priorities listed here were determined by the mode's role in the goals of the project. The autopilot mode is the least important as it is only necessary for the stretch goals of this project. The sensor calibration mode was also determined to be of lower priority because the calibration could be done manually with an SD card. The rest of the modes are of high priority because they are important to ensure proper functionality of the electronics and software package.

Sensor calibration mode will most likely be done manually at first, but in order to make autonomous flight feasible, the system must have a simple, easy-to-use way of calibrating the sensors before each flight. It is believed that with this mode, due to the sensors' good linearity, the sensors will be capable of very precise measurements. The idea of this mode is that the user can use the RF modem to select a test to be performed, such as spinning the board at a constant speed or acceleration. After selecting a test, the user will select a time to delay before sampling, which is an estimate of the time that it will take for the user to set up the test and have the constant input applied.

The software will sleep for this amount of time, then collect a pre-defined length of data. After that, it will filter the results and fit a trend. Finally, it will record the configuration constant to non-volatile flash memory.

For the goal of this mission to succeed in system identification, the software and hardware must be capable of recording data from all of the sensors to an SD card. This requires that two core drivers be written; SPI and FAT16. If either of these drivers does not work efficiently, there are 2 options: the data can be forwarded through UART to a base station to record the data or a SparkFun Logomatic data logger can be attached to the serial port and log the data. The data logger runs as a dual buffered system. This will allow the system to continue logging data while the interrupt-driven SPI driver uploads the data to an SD card.



**Figure 1. Software Flow Diagram**

The data logger runs as a dual buffered system. This will allow the system to continue logging data while the interrupt-driven SPI driver uploads the data to an SD card.

## V. Testing

The tests for the model verification is broken down into two parts. The first part is strictly to prepare for the system ID tests. The second part is the system ID test itself after all preparations have been made.

### A. Testing that Leads to System ID

In order to begin system ID, we must first test our vehicle for basic functionality. These functions include deployment of the parafoil, parafoil controllability, deployment of the backup parachute, freefall velocity of the vehicle with the backup parachute deployed, passive stability of the parafoil, sensor performance, and data logging. These functionality tests are vital to carry out before the system ID because the vehicle must be proven to survive high altitude testing and take good data for the system ID.

We must make sure that the parafoil is in a passively stable configuration. This is important for the system ID because we need a stable steady state flight as a baseline for the controlled flight. The test is performed by dropping the open parafoil from the top of a building and observing the flight path. Once the flight path of the parafoil is analyzed, we can then adjust the line trim if it is necessary. This test is repeated until we have a vehicle that travels straight at a constant pitch angle.

Sensing and data logging are also important to test in preparation for system ID. We will take each sensor and use them to measure a known quantity. Then we evaluate the accuracy and repeat the measurements for precision. With this information, we can calibrate the sensors. After this is completed, we will connect the sensors to the data logger and check to see if the data logger is functioning properly.

For the higher altitude tests that are required for system ID, we need to make sure that the parafoil may deploy properly on its own. The reason for this is because we are not allowed to release a pre-deployed parafoil from a hot air balloon or an airplane. To preserve our vehicle and test prototypes, we will attach dead weight to the parafoil with its deployment mechanism. Then we will drop test it and activate the deployment mechanism.

To mitigate the possibility of a parafoil system failure (either during flight or during the deployment), the vehicle will have a backup parachute system that needs to be tested. The deployment of the backup parachute will be tested in the same way that the deployment of the parafoil will be tested. We must also test the freefall velocity of the deployed backup parachute so that the vehicle will land safely in the event of a parafoil failure. To do this test, we will first throw the deployed backup parachute off of a building with dead weights. Then we will record the velocity of the vehicle with a Doppler gun.

In order for the vehicle to do an actual high altitude test, the vehicle must be able to fly autonomously. A controllability test must first be performed to see whether the vehicle will respond to inputs and perform the actual request. It is important to see whether an input is so strong that the vehicle spins out of control or stalls. On the other hand, it is equally important to make sure that the input is not so weak that the vehicle does not show any significant response. To test that the vehicle is indeed controllable, it will be drop tested and maneuvered from a low attitude with the control servos. Potentiometers will be mounted onto the servos during the tests to record inputs from the servos. Once dropped, the vehicle will receive an input and the response will be observed. The observation will be qualitatively analyzed and the test will be repeated as necessary.

When all of these tests are completed, the vehicle will be ready for the system ID tests.

## B. Parameter Estimation with Recursive Weighted Least-Squares Estimator

System ID consists of developing the control derivatives of the parafoil/payload model. The complete vehicle must be deployed at a test altitude of about 12,000 feet with all of the electronics and software ready to make measurements and record data. The vehicle will be flying autonomously with control deflections based on pre-programmed inputs. The inputs are timed, sinusoidal signals that tell the servos to control the control lines so that the parafoil produces significantly different rolls and yaws. In the mean time, the flight data due to the inputs are recorded onto two removable storage drives for post-processing. The outputs include angular rates in the principle axes (roll and yaw) as well as GPS data (position and time). The data will go through post-processing in order to give meaningful aerodynamic properties such as  $C_L$  and  $C_D$ , and then the recursive weighted least-squares estimator is implemented to estimate the five aerodynamic coefficients,  $C_{l\phi}$ ,  $C_{lp}$ ,  $C_{nr}$ ,  $C_{l\delta_a}$ , and  $C_{n\delta_a}$ , and the constant turning bias term,  $\delta_{bias}$ . These are necessary to create a reduced-order, linearized model. The estimator functions by using the angular rate measurements at each instant,  $k$ , from the roll and yaw gyros,  $z_k$ , to estimate the parameters of interest. The measurements have zero mean noise,  $n_k$ , which is accounted for by adding the assumed noise to the measurement. The system to be estimated is described by Eq. (17).<sup>1,3</sup>

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x} + \mathbf{n}_k \quad (17)$$

where

$$\mathbf{x} = \{C_{l\phi} \ C_{lp} \ C_{nr} \ C_{l\delta_a} \ C_{n\delta_a} \ \delta_{bias}\}^T$$

The matrix  $\mathbf{H}_k$  is the linear relationship between  $\mathbf{x}$  and  $\mathbf{z}_k$ , and is found via the linearization of  $\dot{p}$  and  $\dot{r}$  given in Eq. (7).

$$\mathbf{H}_k = \frac{\rho S b v^2}{2} \begin{bmatrix} I_{XXI} \phi_k & \frac{I_{XXI}(bp_k)}{2V} & \frac{I_{XXI}\delta_a}{d} & \frac{I_{XZI}br_k}{2V} & \frac{I_{XZI}\delta_{a_k}}{d} & \frac{I_{XZI}}{d} \\ I_{XZI} \phi_k & \frac{I_{XZI}bp_k}{2V} & \frac{I_{XZI}\delta_a}{d} & \frac{I_{ZZI}br_k}{2V} & \frac{I_{ZZI}\delta_{a_k}}{d} & \frac{I_{ZZI}}{d} \end{bmatrix} \quad (18)$$

The recursive weighted least-squares method for Eq. (17) is given in Eq. (19).<sup>3</sup>

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{P}_k \mathbf{H}_k^T \mathbf{Q}^{-1} (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k-1}) \quad (19)$$

The matrix  $\mathbf{P}_k$  is the diagonal error covariance matrix and is recursively updated by Eq. (20).

$$\mathbf{P}_k = \mathbf{P}_{k-1} - \mathbf{P}_{k-1} \mathbf{H}_k^T (\mathbf{Q} + \mathbf{H}_k \mathbf{P}_{k-1} \mathbf{H}_k^T)^{-1} \mathbf{H}_k \mathbf{P}_{k-1} \quad (20)$$

The matrix  $\mathbf{Q}$  is the noise covariance matrix that is fixed, and is 2x2 diagonal matrix that determined by the measurement noise,  $\mathbf{n}_k$ . The noise covariance matrix is computed in Eq. (21).

$$E\{\{\mathbf{n}_i \mathbf{n}_i^T\}\} = \begin{cases} 0, & i \neq j \\ Q, & i = j \end{cases} \quad (21)$$

The algorithm is started by taking an initial guess for  $\mathbf{x}$  of  $\{-0.1, -0.5, 0.1, -0.1, 0.1, 0\}$ , and initialize  $\mathbf{P}$  to 0.05 on the diagonal.<sup>1</sup>

## VI. Conclusion

In order to develop an autopilot to control a parafoil vehicle, it is necessary to perform the aforementioned tests and verify the six degree of freedom model. Once the theoretical model is verified, an autopilot can be designed based on the model. After system ID is completed, the aerodynamic properties of this unique parafoil may be determined, and then this model can be applied to problems other than that of returning payloads from high altitude balloon launches.

## References

<sup>1</sup>Costello, Mark and Slegers, Nathan, “Model Predictive Control of a Parafoil and Payload System,” *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, AIAA, Providence, Rhode Island, 2004.

<sup>2</sup>DeRusso, P.M., et al., *State Variables for Engineers*, 2<sup>nd</sup> ed., John Wiley and Sons, Inc.-New York City, New York, 1998.

<sup>3</sup>Ikonen, Enso and Najim, Kaddour, *Advanced Process Identification and Control*, Marcel Dekker, Inc.-New York City, New York, 2002.