

PROCEEDINGS OF SPIE

[SPIDigitalLibrary.org/conference-proceedings-of-spie](https://spiedigitallibrary.org/conference-proceedings-of-spie)

Flight performance of the attitude control system of the balloon experimental twin telescope for infrared interferometry (BETTII)

Jordi Vila Hernandez de Lorenzo, Stephen A. Rinehart, Maxime Rizzo, Marc Casalprim, Steve Maher, et al.

Jordi Vila Hernandez de Lorenzo, Stephen A. Rinehart, Maxime Rizzo, Marc Casalprim, Steve Maher, Arnab Dhabal, George Nehmetallah, D. J. Fixsen, Todd J. Veach, Fernando Esteves, "Flight performance of the attitude control system of the balloon experimental twin telescope for infrared interferometry (BETTII)," Proc. SPIE 10700, Ground-based and Airborne Telescopes VII, 1070067 (6 July 2018); doi: 10.1117/12.2312659

SPIE.

Event: SPIE Astronomical Telescopes + Instrumentation, 2018, Austin, Texas, United States

Flight performance of the Attitude Control System of the Balloon Experimental Twin Telescope for Infrared Interferometry (BETTII)

Jordi Vila Hernandez de Lorenzo^{a,b}, Stephen A. Rinehart^a, Maxime Rizzo^{a,d}, Marc Casalprim^{a,b,g}, Steve Maher^{a,c}, Arnab Dhabal^e, George Nehmetallah^b, D. J. Fixsen^e, Todd J. Veach^f, and Fernando Esteves^b

^aNASA Goddard Space Flight Center

^bEECS Department, The Catholic University of America

^cScience Systems and Applications, Inc.

^dNASA Postdoctoral Fellow

^eUniversity of Maryland

^fSouthwest Research Institute

^gSUPAERO/Polytechnic University of Catalonia

ABSTRACT

The Balloon Experimental Twin Telescope for Infrared Interferometry (BETTII) is an 8-meter baseline far-infrared interferometer designed to fly on a high altitude balloon. BETTII uses a double-Fourier Michelson interferometer to simultaneously obtain spatial and spectral information on science targets; the long baseline provides subarcsecond angular resolution, a capability unmatched by any other far-infrared facilities. BETTII had its first successful engineering flight in June 2017. The pointing loop on BETTII is based on an Extended Kalman Filter, which uses different sensors and actuators to keep the telescope pointed at the desired target star. In order to achieve high precision pointing, we use an embedded Field-programmable gate array (FPGA) that combines the gyroscope and star cameras information to generate a pointing solution every 10 milliseconds.

The BETTII control system serves a critical function in making interferometric observations possible. This paper discusses the design and implementation of the BETTII control system and presents engineering data of the attitude control system from our pre-flight tests at the Columbia Scientific Balloon Facility (CSBF) and data from our first 12-hour flight from Palestine, TX. This includes pointing performance of the Kalman Filter estimator in the RA, DEC and ROLL Equatorial Coordinate System as well as the payload's attitude behavior when switching between the different modes we implemented: Safe, Brake, Slew, Track and Acquire. These modes are part of the procedure to point the telescope to a desired target. We discuss the performance of the payload's control system in each of these modes and present data showing how the azimuth actuators follow the position and velocity profiles calculated by the flight computers.

Keywords: Balloon, Telescope, Kalman Filter, Attitude control, Gyroscope, Star Camera

1. INTRODUCTION

Space-based interferometric missions are crucial to the future of high resolution astronomy at infrared wavelengths^{1,2} and BETTII is a balloon-borne pathfinder for future missions. The instrument is designed to achieve a spatial resolution of $\sim 1''$ and a spectral resolution $\lambda/\Delta\lambda$ of up to 100 at Far Infrared (FIR) wavelengths (30 to 90 μm). Balloon experiments usually fly between

Author contact: Jordi Vila Hernandez de Lorenzo, jordi.vilahernandezdelorenzo@nasa.gov

Ground-based and Airborne Telescopes VII, edited by Heather K. Marshall,
Jason Spyromilio, Proc. of SPIE Vol. 10700, 1070067 · © 2018 SPIE
CCC code: 0277-786X/18/\$18 · doi: 10.1117/12.2312659

35 and 42km, above the 99.5% of the air in the atmosphere, which make them particularly suited for studying the universe at infrared, far-infrared and sub-millimeter wavelengths.

To obtain high angular resolution, BETTII collects light in two separate 50-cm apertures. Separate telescope assemblies compress the beam and deliver it to the cryogenic double-Fourier instrument. Details of the BETTII design can be found in Rinehart et al. (2014), and in additional papers on the BETTII subsystems.³⁻⁸ To make these observations effective, however, BETTII requires accurate pointing of the gondola, and coalignment of the two interferometric arms.

The goal is to combine two beams of light reflected off from two spatially separated mirrors located at the edges of the truss. The beams reflect on the siderostat mirrors and are redirected to the telescope assemblies located in the center of the truss. Figure 1 shows the telescope assemblies that consist of 3 powered mirrors and a folding flat and provide a 20:1 compression ratio of the beams.⁴ After the telescope assemblies, the light passes through a set of mirrors that adjust the phase difference between the beams and center the targets on the science detectors. The science sensors are located inside the cryogenic instrument, also called the dewar. Inside the dewar we use several refrigeration stages of Helium and Nitrogen in order to cool down the Transition-Edge Sensor (TES) detectors at temperatures of a $\sim 400\text{mK}$. At these low temperatures we are able to minimize the thermal noise and achieve good signal to noise ratio in the far-infrared spectrum that we are observing.

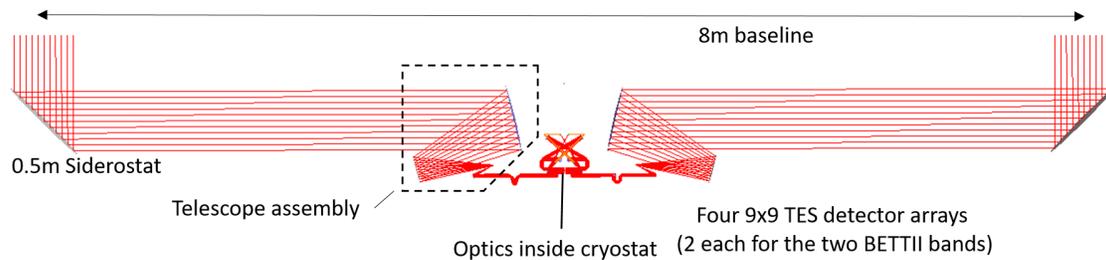


Figure 1: Optical path of the beams from the siderostat mirrors to the detectors.

To achieve these pointing requirements the control system consists of two stages; the coarse and the fine pointing. During the coarse pointing stage, the goal is to track the target in elevation and cross elevation within $\pm 15''$; the fine-pointing system can then engage to center the target on the detector using a Tip/Tilt mechanism at the entrance of the cryogenic chamber. Offset information extracted from a dedicated near-infrared tracking detector is then used both as feedback for the Tip/Tilt stage and for the coarse pointing. The coarse pointing adjustment helps maintain the tip/tilt stages near the center of their range of motion. The control system uses multiple operating modes to slew the telescope, acquire the target, and then center the target on the far-infrared detector arrays. The control loop is implemented on a field-programmable gate array (FPGA) embedded on a cRIO device. The estimation loop runs at 100 Hz and uses a set of tunable parameters, such as PID controller gains, that can be adjusted during flight. The system is also designed to be autonomous, once a target has been identified.

In the following section, we briefly describe the sensors and actuators that are involved in BETTII's control loop. A much more thorough description is offered in Rizzo (2016).⁴

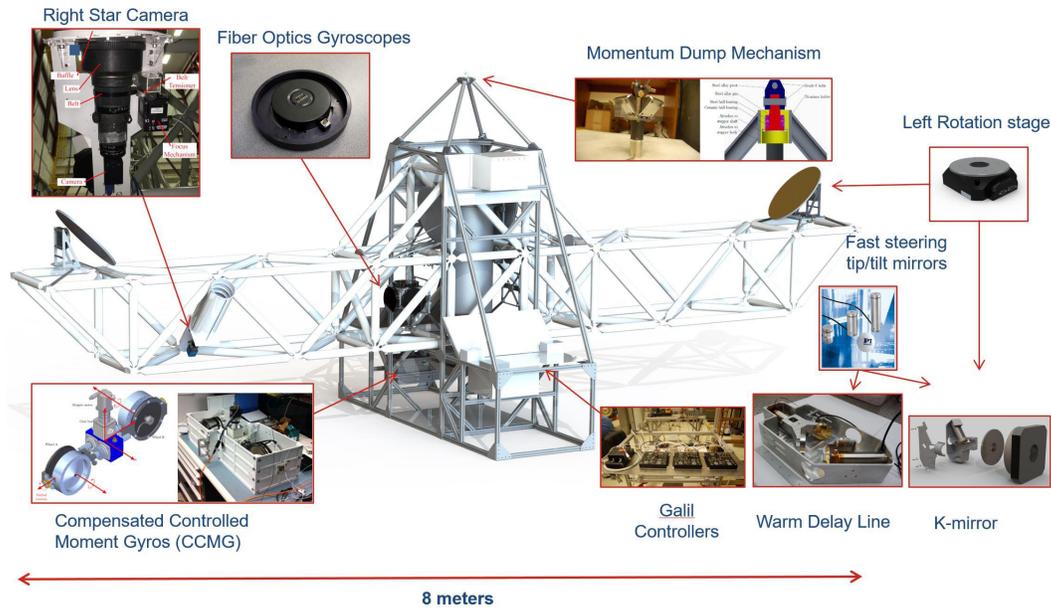


Figure 2: BETTII structural model showing the location of the different sensors and actuators,⁹ which are summarized below.

Sensors

Three sensing subsystems are involved in order to determine the pointing quaternion of the telescope at every stage of the flight.

- Star cameras:** The attitude control system of BETTII has two star cameras, located approximately in the center of each arm. The sensor is a Sony Pregius IMX174 CMOS sensor with 1920×1200 pixels and a pixel scale of $4.02''/\text{pixel}$. The star cameras capture an image of a patch of the sky of approximately $2.14^\circ \times 1.34^\circ$. Once the image is captured, it is sent to one of the flight computers via a USB connection. That computer runs a C++ script that contains the finder software developed by Astrometry.net, which processes the image and matches the blobs to a stars in a catalog. If a solution is found, the script returns the pointing solution of the star camera vector (RA,DEC,ROLL). For convenience, we convert that solution to an attitude quaternion and feed it back to the control system along with the time stamp corresponding to when the image was taken.
- Magnetometer:** The Honeywell Microelectronics Magnetometer (HMR3300-D00-232) provides another coarse attitude determination, used when star camera information is not available. For example, it can be used to prevent the payload to accidentally pointing towards the Sun. Combining the information provided by the magnetometer and the UTC time provided by the GPS we are able to determine a pointing solution (RA,DEC,ROLL) accurate to 1° , which is enough to protect the optics and keep the telescope pointed away from the sun.
- Gyroscopes:** Three Optolink SRS-2000 gyroscopes are used to measure the angular velocity. These gyroscopes are exceptionally good, with very low random walk and bias drifts;⁴ integration of the gyroscope angular velocity measurement to obtain a position estimate over one hour would have a standard deviation of 2 arcseconds. The bandwidth is 50Hz with readout of up to 2kHz. Additionally, built-in thermal regulation maintains the temperature within a certain range for optimal performance.

Actuators

Several actuators are needed for the coarse pointing loop, fine pointing loop, as well as for control of the optical path difference between the two arms of the interferometer.

- **Compensated Controlled Moment Gyroscopes Wheels:** The CCMG controls the payload's azimuth and consists of two counter-rotating reaction wheels spinning at 50 revolutions per second, providing a stored angular momentum of $M_{CCMG} = 20.8$ N.m.s. These wheels are mounted on a gimbal which is actuated with a stepper motor through a gear box. With this system, the projection angle of the wheel's combined momentum vector can be changed (from positive z axis to negative z axis). The rate of change of the wheel's angle determines the torque applied to the payload, with maximum torque available when the wheel's spin axis is in the horizontal plane. The wheel spin rate is kept constant throughout all operations in order to minimize power usage and freeze any undesired beat frequencies between the wheels as much as possible.
- **Momentum Dump:** The function of the Momentum Dump motor is to maintain the wheels angle as close as possible to zero, thus, maximizing the available torque at any given time. This mechanism is attached to the balloon through a 100 foot ladder.
- **Siderostat mirrors:** The elevation control is provided by Griffin Motion rotation stages upon which two siderostat mirrors are mounted. The rotary encoder used on the motor that moves the mirrors has 26 bits of resolution, which corresponds to $0.019''$ of rotation precision.
- **Delay Lines:** The Kalman Filter estimator constantly monitors the attitude error in elevation and cross elevation, and calculates the phase error introduced in the optical path from gondola mis-pointing. The Warm Delay Line (WDL), a linear stage mounted in one arm of the interferometer, compensates for this phase error by changing the optical path length of its arm. A second linear mechanism, the Cold Delay Line (CDL), is located inside the instrument; it has a pair of rooftop mirrors that travel back and forth at constant speed in order to provide the lag-spectra used to recover the scientific information.

2. QUATERNIONS AND KALMAN FILTER

2.1 Quaternions

BETTII has several reference frames that determine a unique attitude pointing. The coordinate systems relevant to the pointing control system are shown in Figure 3.

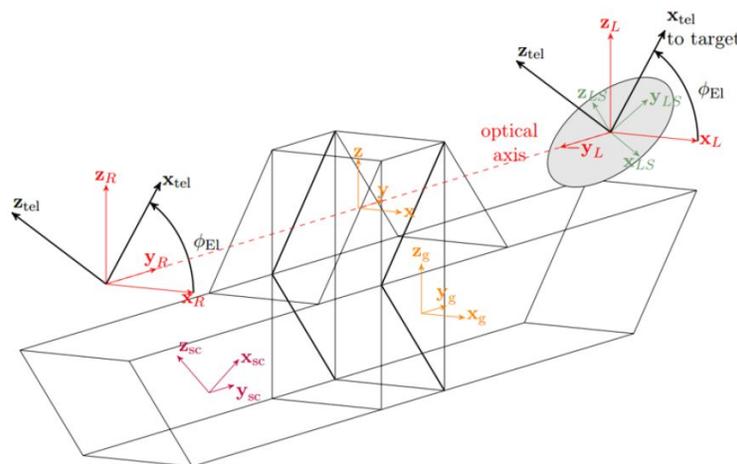


Figure 3: The subscripts sc, g, tel, L, R and LS indicate the star camera, gyroscope, telescope, left, right and left siderostat reference frames, respectively. The gyroscope reference frame is nominally aligned with the gondola reference frame, which has no subscript.⁴

To represent the attitudes and rotations between the different reference frames we use unit quaternions. Quaternions are similar to complex numbers, but with a real part q_r and three imaginary parts q_i, q_j, q_k . A unit quaternion has unit norm and can be parametrized by an angle θ describing the angle of rotation around an axis defined by the vector $\mathbf{k} = [k_x, k_y, k_z]^T$:

$$\bar{q} = \begin{bmatrix} q_i \\ q_j \\ q_k \\ q_r \end{bmatrix} = \begin{bmatrix} k_x \sin \theta/2 \\ k_y \sin \theta/2 \\ k_z \sin \theta/2 \\ \cos \theta/2 \end{bmatrix} \quad (1)$$

The notation ${}^G_I \bar{q}$ represents a quaternion describing the rotation between the inertial reference frame I and the gondola reference frame G . In this manner, the quaternion ${}^{sc} \bar{q}$ describes the rotation applied to the gondola reference frame to become the right star camera reference frame. The rotations are applied using the quaternion multiplication. In this work we do not use the Hamilton multiplication but the "natural order" multiplication, in accordance with the JPL proposed Standard Conventions,^{10,11} which define a multiplication operator \otimes :

$$\begin{aligned} \bar{p} \otimes \bar{q} &= (p_r + p_i \mathbf{i} + p_j \mathbf{j} + p_k \mathbf{k}) \times (q_r + q_i \mathbf{i} + q_j \mathbf{j} + q_k \mathbf{k}) \\ &= (p_r q_r - p_i q_i - p_j q_j - p_k q_k) \\ &\quad + (p_r q_i + p_i q_r - p_j q_k + p_k q_j) \mathbf{i} \\ &\quad + (p_r q_j + p_j q_r - p_k q_i + p_i q_k) \mathbf{j} \\ &\quad + (p_r q_k + p_k q_r - p_i q_j + p_j q_i) \mathbf{k} \end{aligned} \quad (2)$$

Using this convention, we can entirely describe the telescope's attitude as its rotation quaternion from the inertial reference frame, ${}^T_I \bar{q}$. We can obtain this quaternion by multiplying the quaternion obtained from the star camera software, which provides the quaternion between the inertial reference frame and the star camera ${}^S_I \bar{q}^{meas}$, and rotating it by the quaternion between the star camera axis and the gondola axis ${}^G_S \bar{q}$, and then by the quaternion representing the elevation angle of the siderostat mirrors, ${}^T_G \mathbf{q}$. The chain of rotations is as follows (note the order of the quaternion multiplications):

$${}^T_I \bar{q}^{meas} = {}^T_G \bar{q} \otimes {}^G_S \bar{q} \otimes {}^S_I \bar{q}^{meas}. \quad (3)$$

2.2 State equations

At each cycle of the control loop, that is every 10ms, the control system determines a unique attitude quaternion ${}^G_I \bar{q}$ describing the inertial attitude of the gondola. We use the star camera quaternion information when available, and integrate the gyroscopes velocities $\boldsymbol{\omega}(t)$ when the star camera information is not available. This makes it very important both to have accurate gyroscope measurements during each control loop cycle, and to have accurate estimates of the intrinsic biases of all three gyroscopes $\mathbf{b}(t)$. A Kalman filter is used to obtain the estimate of these values and weigh the different sensors information based on their associated inherent noise. The seven-dimensional state vector of our system is defined by the four components of the attitude quaternion and the biases of the three gyroscopes:

$$\mathbf{x}(t) = \begin{bmatrix} {}^G_I \bar{q}(t) \\ \mathbf{b}(t) \end{bmatrix}, \quad (4)$$

The evolution of the state vector is defined by:

$${}^G_I \dot{\bar{q}}(t) = \frac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}(t)) {}^G_I \bar{q}(t), \quad (5)$$

$$\dot{\mathbf{b}}(t) = \mathbf{n}_b(t), \quad (6)$$

where

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} = \begin{bmatrix} [\hat{\boldsymbol{\omega}}_{\times}] & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix}, \quad (7)$$

with $\boldsymbol{\omega} = \boldsymbol{\omega}^{meas} - \mathbf{b} - \mathbf{n}_g$. Here \mathbf{n}_g is the noise measured by the three gyroscopes and \mathbf{n}_b is the bias instability. These two noises are considered white and Gaussian with variances σ_g^2 and σ_b^2 respectively. However, the noise values are unknown so we use an estimate of the state vector $\hat{\mathbf{x}} = [{}^G_I \hat{q}(t), \hat{\mathbf{b}}(t)]^T$ that follows the same dynamic equations but with the noises equal to zero. When dealing with propagation of quaternions, the Linear Kalman filter faces some numerical complications. To avoid this problem, a multiplicative Extended Kalman filter is implemented by using small angle approximation of the quaternions.¹² This modified Kalman filter has a new six-dimensional state vector called the error vector $\tilde{\mathbf{x}} = [\delta\boldsymbol{\theta}, \Delta\mathbf{b}]^T$, where $\Delta\mathbf{b} = \mathbf{b} - \hat{\mathbf{b}}$ and $\delta\boldsymbol{\theta}$ are the three angular errors between the true and estimated attitude quaternions taken directly from the difference quaternion ${}^G_I \delta\bar{q} = {}^G_I \bar{q} \otimes {}^G_I \hat{q}^{-1} \approx [\frac{1}{2}\delta\boldsymbol{\theta}, 1]^T$. With these new variables we can write our gyroscope model as:

$$\boldsymbol{\omega}^{meas} = \boldsymbol{\omega} + \mathbf{b} + \mathbf{n}_g, \quad (8)$$

$$\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}^{meas} - \hat{\mathbf{b}}, \quad (9)$$

$$\boldsymbol{\omega}^{true} = \hat{\boldsymbol{\omega}} - \mathbf{n}_g - \Delta\mathbf{b}, \quad (10)$$

The resulting continuous state equations from this new error state vector are:¹²

$$\dot{\delta\boldsymbol{\theta}} = -\hat{\boldsymbol{\omega}} \times \delta\boldsymbol{\theta} - \Delta\mathbf{b} - \mathbf{n}_g, \quad (11)$$

$$\Delta\dot{\mathbf{b}} = \dot{\mathbf{b}} - \dot{\hat{\mathbf{b}}} = \mathbf{n}_b, \quad (12)$$

Combining these results we may write the error state equation as

$$\dot{\tilde{\mathbf{x}}} = \begin{bmatrix} \dot{\delta\boldsymbol{\theta}} \\ \Delta\dot{\mathbf{b}} \end{bmatrix} = \mathbf{F} \begin{bmatrix} \delta\boldsymbol{\theta} \\ \Delta\mathbf{b} \end{bmatrix} + \mathbf{G} \begin{bmatrix} \mathbf{n}_g \\ \mathbf{n}_b \end{bmatrix}, \quad (13)$$

with

$$\mathbf{F} = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}_{\times}] & -\mathbf{I}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}, \quad (14)$$

and

$$\mathbf{G} = \begin{bmatrix} -\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix}, \quad (15)$$

where $[\hat{\boldsymbol{\omega}}_{\times}]$ is the skew-symmetric matrix made out of the elements of $\hat{\boldsymbol{\omega}}$, $[\hat{\boldsymbol{\omega}}_{\times}]\delta\boldsymbol{\theta} = \hat{\boldsymbol{\omega}} \times \delta\boldsymbol{\theta}$. To obtain a discrete transition matrix Φ_k of the process, we can integrate the state equation between the times t_{k-1} and $t_k = t_{k-1} + \Delta t$:

$$\Phi_k = \Phi(t_k, t_{k+1}) = e^{\mathbf{F}\Delta t} = \mathbf{I}_{6 \times 6} + \mathbf{F}\Delta t + \frac{1}{2!}\mathbf{F}^2\Delta t^2 + \dots = \begin{bmatrix} \Theta_k & \Psi_k \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix}, \quad (16)$$

with $\Theta_k \sim \mathbf{I}_{3 \times 3} - \Delta t[\hat{\boldsymbol{\omega}}_{\times}] + \frac{\Delta t^2}{2}[\hat{\boldsymbol{\omega}}_{\times}]^2$ and $\Psi_k \sim -\mathbf{I}_{3 \times 3}\Delta t + \frac{\Delta t^2}{2}[\hat{\boldsymbol{\omega}}_{\times}]$. The full details of these calculations are given elsewhere.¹² In the final implementation of the Kalman filter, we use the first order terms of the Θ_k and Ψ_k expressions.

2.3 Kalman filter

Prediction

In summary, assuming we receive a measurement ω_k^{meas} and we have an estimate of the quaternion $\hat{q}_{k-1|k-1}$ and the bias $\hat{\mathbf{b}}_{k-1|k-1}$ at time step $k-1$, the propagation equations for our estimations are:

$$\hat{\mathbf{b}}_{k|k-1} = \hat{\mathbf{b}}_{k-1|k-1}, \quad (17)$$

$$\hat{\omega}_{k|k-1} = \omega_k^{meas} - \hat{\mathbf{b}}_{k|k-1}, \quad (18)$$

$$\hat{q}_{k|k-1} = \exp\left(\frac{1}{2}\Omega(\hat{\omega}_{k|k-1})\Delta t\right) \hat{q}_{k-1|k-1}, \quad (19)$$

The propagation of the covariance matrix $\mathbf{P}_{k-1|k-1}$ of the error vector depends on the covariance matrix of the process noise \mathbf{Q}_d and the state transition matrix Φ_k . The matrix Φ_k is computed using $\hat{\omega}_{k|k-1}$ in the expressions of Θ_k and Ψ_k . The noise covariance matrix \mathbf{Q}_d is considered constant and has the following theoretical expression:¹²

$$\mathbf{Q}_d \sim \begin{bmatrix} \sigma_g^2 \Delta t \mathbf{I}_{3 \times 3} & -\sigma_b^2 \frac{\Delta t^2}{2} \mathbf{I}_{3 \times 3} \\ -\sigma_b^2 \frac{\Delta t^2}{2} \mathbf{I}_{3 \times 3} & \sigma_b^2 \Delta t \mathbf{I}_{3 \times 3}, \end{bmatrix} \quad (20)$$

where $\sigma_g = 1.5 \times 10^{-7} \text{ rad}/\sqrt{s}$ represents the uncertainty related to the ARW noise and $\sigma_b = 1.8 \times 10^{-6} \text{ rad s}^{-3/2}$ the bias instability, obtained from the gyroscope's manufacturer and remeasured in Rizzo et al. (2016).⁴ In practice, the best results are not achieved with the previous values. An empiric tuning of σ_g and σ_b is performed once the whole control system is functional. According to the Kalman filter equations, the matrix $\mathbf{P}_{k|k}$ propagation is computed:

$$\mathbf{P}_{k|k-1} = \Phi_k \mathbf{P}_{k-1|k-1} \Phi_k^T + \mathbf{Q}_d. \quad (21)$$

Update

The update phase of the Kalman filter occurs when a star camera measurement is available. We will use the difference between our prediction $\hat{q}_{k|k-1}$ and the new star camera measurement \bar{q}_k^{meas} . This difference is called innovation, \mathbf{z}_k , and it will be equal to the angular part $\delta\theta_k^{meas}$ of the difference quaternion $\delta\bar{q}^{meas} = \bar{q}_k^{meas} \otimes \hat{q}_{k|k-1}^{-1}$. The measurement model will be as follows:

$$\mathbf{z}_k = \delta\theta_k^{meas} \approx \mathbf{H}\tilde{\mathbf{x}}_k + \mathbf{n}_k^{meas}, \quad (22)$$

where $\mathbf{H} = [\mathbf{I}_{3 \times 3} \quad \mathbf{0}_{3 \times 3}]$ and \mathbf{n}_k^{meas} is the star camera measurement noise, characterized by a covariance matrix \mathbf{R}_k . Then, the covariance matrix of the innovation is $\mathbf{S}_k = \mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^T + \mathbf{R}_k$ and the Kalman gain results $\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^T\mathbf{S}_k^{-1}$. Once we calculate the previous matrices, we can start the update:

$$\tilde{\mathbf{x}}_{k|k} = \mathbf{K}_k \mathbf{z}_k = \begin{bmatrix} \delta\theta \\ \Delta\mathbf{b} \end{bmatrix} = \begin{bmatrix} 2\mathbf{v} \\ \Delta\mathbf{b} \end{bmatrix}, \quad (23)$$

$$\delta\bar{q} = \begin{cases} \begin{bmatrix} \mathbf{v} \\ \sqrt{1 - \mathbf{v}^T \mathbf{v}} \end{bmatrix}, & \text{if } \mathbf{v}^T \mathbf{v} \leq 1 \\ \frac{1}{\sqrt{1 + \mathbf{v}^T \mathbf{v}}} \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}, & \text{otherwise,} \end{cases} \quad (24)$$

$$\hat{q}_{k|k} = \delta\bar{q} \otimes \hat{q}_{k|k-1}, \quad (25)$$

$$\hat{\mathbf{b}}_{k|k} = \hat{\mathbf{b}}_{k|k-1} + \Delta\mathbf{b}, \quad (26)$$

$$\hat{\omega}_{k|k} = \hat{\omega}_k^{meas} - \hat{\mathbf{b}}_{k|k}, \quad (27)$$

$$\mathbf{P}_{k|k} = (\mathbf{I}_{3 \times 3} - \mathbf{K}_k \mathbf{H}) \mathbf{P}_{k|k-1} (\mathbf{I}_{3 \times 3} - \mathbf{K}_k \mathbf{H})^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T. \quad (28)$$

The control loop cycles through these steps, propagating the estimated states until a new measurement is received and the update is performed again. However, there exists a delay of a few seconds between the trigger of the star cameras and the reception of the solution. To solve this issue, the \hat{q}_k^{meas} measurement is propagated until the current time step k , using propagation matrices calculated since the instant $k-N$ where the image was taken. In case the star cameras are not able to provide an absolute position solution, a less accurate absolute position measurement is available combining the magnetometer and GPS data.

3. FLIGHT AND TESTING DATA

Prior to flight, a detailed plan was developed for the sequential activation of payload components. During ascent, as each component was turned on, diagnostics were performed to verify that they were active and were providing appropriate telemetry. This was aided by a sophisticated ground software package that provided the ability to simultaneously monitor nearly every system on the payload. By the time BETTII achieved float altitude, we were able to verify that all systems were on, and we began exercising all of the individual system components. Over the course of the night, we exercised system components and had a number of significant successes, including the Kalman Filter estimator, and a couple issues, one of them concerning the Galil microcontroller used to control the CCMG reaction wheels. We established that the cause of the problem was that the Galil lacked an Internet Protocol (IP) address. Despite this, with the momentum dump alone, we were able to verify the accuracy of the coarse pointing loop. In this section we present flight data of the Kalman Filter estimator and flight and test data of the performance of the control system.

3.1 Performance of the Kalman Filter Estimator

The three main components on the payload that allow the proper estimation of the pointing position at every loop cycle are the flight computers, the gyroscopes and the star cameras. The first two worked seamlessly all throughout flight, however during ascent to 135,000 feet we had some difficulties resolving the first star camera images into pointing solutions, mainly because the CCMG wheels were not spinning due to the IP address assignment problem and because the turbulences did not allow the star camera to take pictures that were sharp enough for the finder software. Once we reached float altitude we were able to reduce the inertial azimuth velocity of the payload, run the auto-focus mechanisms and find the first star camera solutions that determined for the first time the inertial attitude of BETTII during flight. Throughout flight we updated the Kalman Filter state quaternion with the star camera solutions that were being found. During the 12-hour flight there were periods of time as long as twenty minutes where we had no star camera solutions; during these periods, we were still able to predict our state quaternion using a previous star camera measurements and the integrated gyroscope velocities. Subsequent star camera images received after integrating the gyroscopes for extended periods of time, corroborated that the Kalman Filter prediction was accurate throughout the flight. Figures 4, 5 and 6 show results of the performance of the attitude estimator. The data corresponds to the period from 02:09AM to 4:13AM, where we obtained a total of 50 star camera solutions.

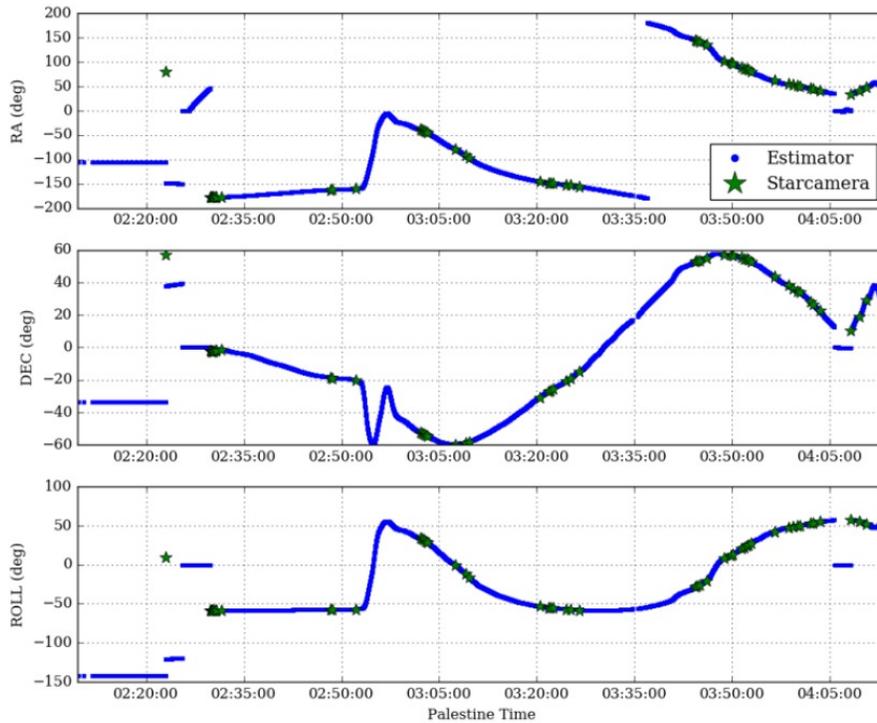


Figure 4: Evolution of the estimated attitude angles in the inertial reference frame (blue) overlapped with the star camera measurements received (green).¹³

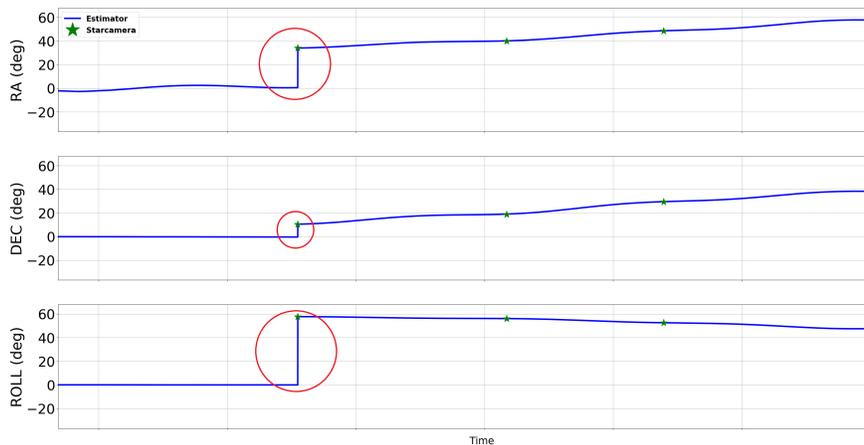


Figure 5: When the estimator was restarted the RA, DEC and ROLL were set to zero until we received a new star camera solution. The first star camera solution that was found was directly assigned to the current position. We can see that in this figure, where the star camera solution, denoted with a green star, perfectly overlaps with current position estimator in blue for all three angles.



Figure 6: This zoomed in portion of the image shows that after a long period of time (over 20 minutes) without star camera solutions and just using the gyroscopes information, the RA of the estimator was only off by $\sim 155.23^\circ - 155.225^\circ = 0.005^\circ$ or $18''$. Notice how the estimator (blue) corrects the position weighting in the latest star camera pointing solution.

3.2 Execution time of the control loop

BETTII has two on-board computers. The first (called "Ford") operates a low-latency Linux kernel and is used to store all the data, process the up/down telemetry,¹⁴ acquire star camera images, solve for inertial attitude, and process the science detector and HIRG frames. During flight, Ford's CPU registered internal temperatures as low as 0°C , whereas during ascent, the external temperatures were as low as -60°C . The second computer (called "Boop") is a National Instrument cRIO- system that features a reprogrammable FPGA chip in addition to a dual-core real-time operating system (RT). The FPGA receives all the input sensor signals (Star camera solutions, Gyroscopes, Magnetometer, Motor Encoders and H4RG angle sensor information and outputs the signals to the actuators (Rotation stages, CCMG Wheels, Momentum Dump Motor, Tip/Tilt mechanism and Delay Lines). Simultaneously, the code on the RT uses the sensors information to implement the Kalman Filter estimation and provides the output commands to each of the actuators.

NI LabView is the software language to the system and is used to program both the FPGA and the real-time control software. In this section we show some of the most relevant plots related to the timing on the software loops that are running on the cRIO; the RT and the FPGA.

Although the use of quaternions simplifies the mathematical complexity of the Kalman Filter, the flight computers still have to perform multiple computations per second. In order to monitor the computer resource usage of each group of mathematical computations we divided the 10ms control loop into fifteen different sequential sections. This programming technique can be achieved in LabVIEW using Flat Sequence Structures which consist of one or more subdiagrams, or frames, that execute sequentially. The use of these structures ensures that a subdiagram executes after of before another subdiagram and maintains a sequential data flow.

The BETTII control loop is actually split into many subloops but here we focus on three of them; two on the FPGA and one on the RT. These control loops are operated at a nominal frequency of 100.16 Hz, which is the speed at which we read out the gyroscopes and issue new commands to the actuators. Using a robust synchronization scheme slaved to our system's master clock, we can align the various pieces of information with very high accuracy. The diagram in Figure 7 shows the three loops that we describe in this section; Boop FPGA Trigger and read sensors, Boop FPGA Apply commands and Boop RT Mode manager & estimator.

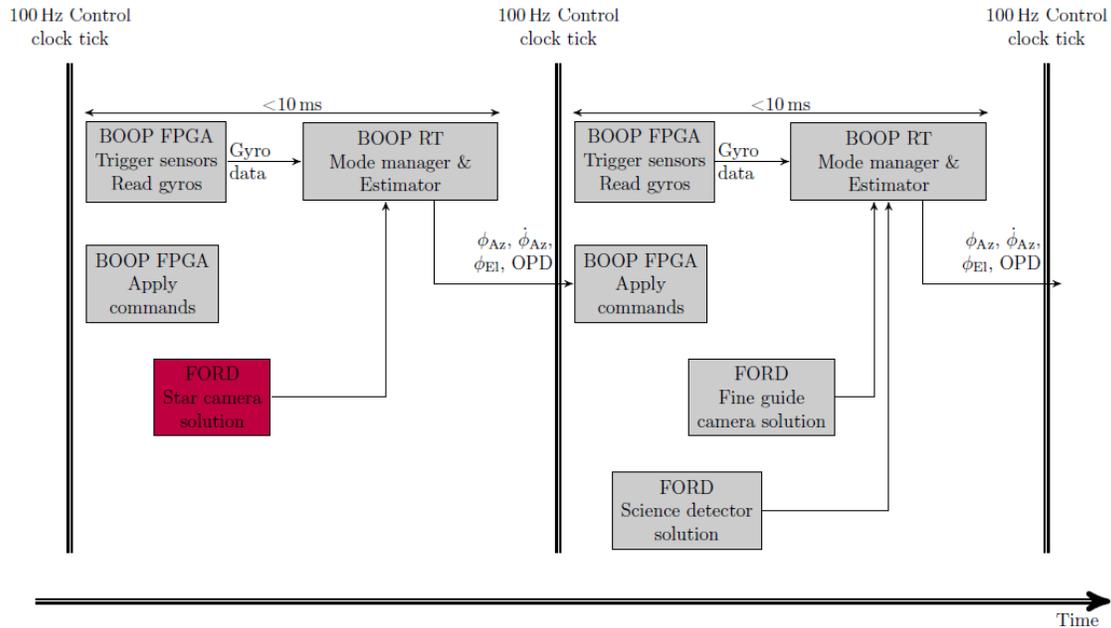


Figure 7: Three loops of the main control software. Right at each clock tick, the gyros are triggered in Boop FPGA and read out. Once the sensors are triggered and read out, it triggers the loop in Boop RT, which has less than 10 ms to complete and send new commands back to the Boop FPGA Apply commands loop. These commands are not applied by the actuators until the next clock tick. This has the advantage to completely lock all moving parts of our payload to our heartbeat. The commands sent use information from the previous loop. This constant lag is preferred, as opposed to a scenario where the commands are applied as soon as possible, which would lead a variable lag which would depend on the processing time within Boop RT.⁴

3.2.1 Real Time Operating System loop execution time

The fifteen subdiagrams in which the real time operating system control loop is divided have a myriad of functions but they are organized in a logic sequence. For instance, there are some subdiagrams that are solely dedicated to telemetry and communications between Boop and Ford, using Zero MQ, a high performance messaging system, other subdiagrams just read the sensors information from the FPGA loops and others are dedicated to single steps of the Kalman Filter. In fact, the most resource-consuming subdiagrams are dedicated to the Kalman Filter operations. For example, the Update step entails a high computational load inverting matrices to find the optimal solution when a new absolute measurement is received from the star camera. Figures 8, 9 and 10 show metrics of the computing performance of each of these control loops and some of their subdiagrams. Notice that starting at 3.30 am we started having spikes in the duration of every subdiagram on the RT loop. The source of this noise is still undetermined, but the problem was resolved by restarting Boop.

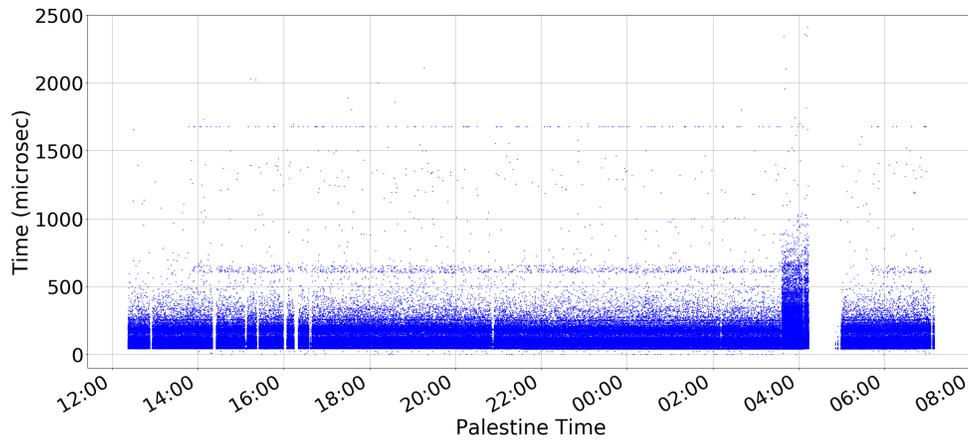


Figure 8: Total execution time, in microseconds, of the "update" step on the BOOP RT Mode manager & estimator loop. The "update" step on the Kalman filter is performed whenever we receive a new star camera solution.

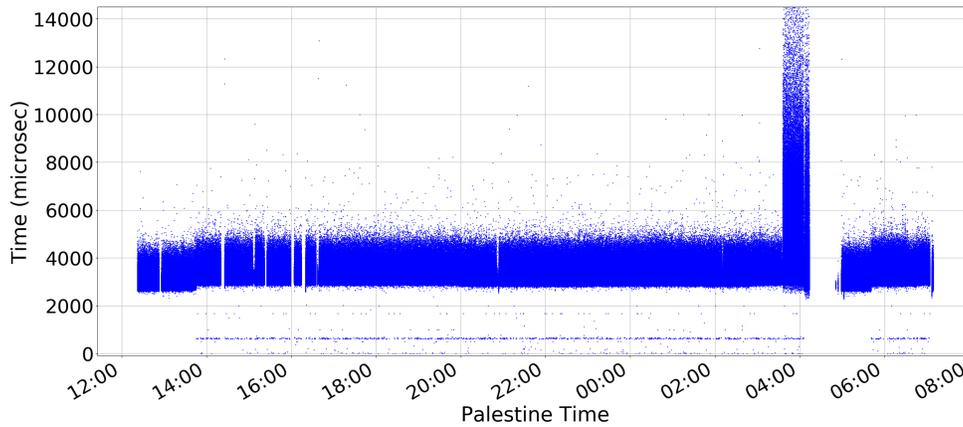


Figure 9: Total loop duration, in microseconds, of the BOOP RT Mode manager & estimator control loop, well below 10ms almost all of the time.

In order to keep track of the performance of the RT estimator loop that implements the Kalman Filter we incremented a counter variable every time we missed a control loop timestamps, that is, every time we couldn't finish all our computations within the stipulated 10ms time frame. This is shown in Figure 10.

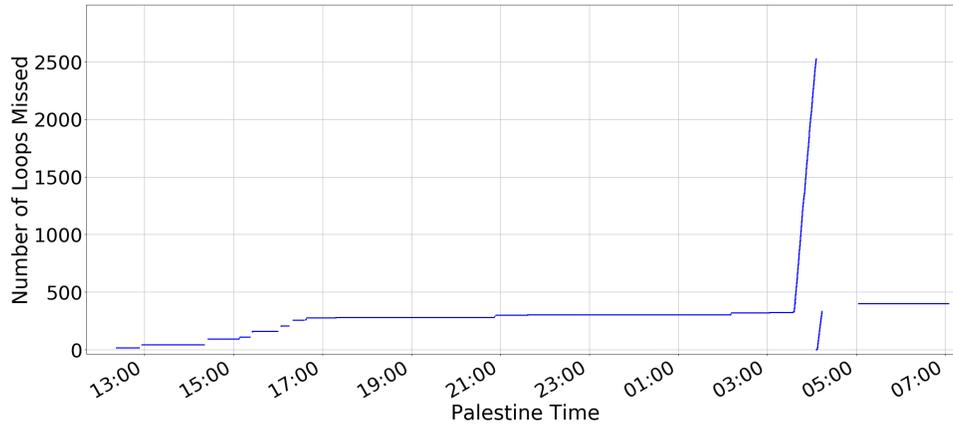


Figure 10: Number of loops missed during flight. That is, total number of loops where the Boop RT Mode manager and estimator control loop did not complete in less than 10ms. The system was reset at 4am after observing a sharp increase in the number of loops (2526 loops) that took more than 10ms to execute.

It is important to notice that before the execution time spike at 3.30am, we had executed the loop 100 times per second over 14 hours straight. During this period of time we just missed around 400 loops; $\sim 0.0127\%$ of the total number of control loops executed during that time. Even in those instances, the estimator is able to quickly catch up with gyroscope information from the following loop and apply the correct commands to the actuators. Nevertheless, before flight the code was optimized so that the loop execution time averaged below 5ms and almost never exceeded 10ms. The number of loops missed is an excellent metric to quantify this performance.

3.2.2 FPGA Loops execution times

The FPGA is more deterministic than the RT. The physical circuits that execute all the instructions are defined by hardware routings so there is little to no variability in execution time for a given set of computations. For this reason, the timing results are much more consistent than on the RT. For instance, the time elapsed to trigger the gyroscopes was constant at 30 microseconds during flight. This digital Transistor to Transistor Logic (TTL) signal is completely internal to the FPGA, thus is deterministic. Because generating this signal does not involve establishing communication protocols with the gyroscopes, its execution time remained constant throughout flight to within 30 microseconds. However, the execution times are not 100% pre-determined for the complete loop because in some subdiagrams, the FPGA inputs and outputs (I/O) ports interact with a set of controllers that run real time, non-deterministic, operating systems. Although every I/O instruction on the FPGA loops has a preset timeout, buffers and read/write commands to external devices can slightly delay the execution time of a section of the control loop. While the time between the start of two consecutive executions of the Boop FPGA "Apply command" block (see Figure 7) remained constant at 9984 microseconds, some of the sequences of that loop did not have a constant duration throughout flight. The variable execution time of some of the subsections of the loop is shown in Figure 11.

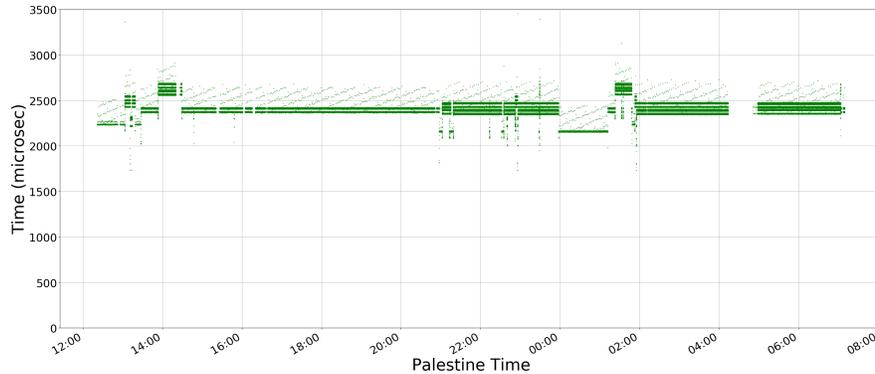


Figure 11: Execution time of the subdiagram of the Apply PIDs control loop where the FPGA reads and writes to the Galil controllers I/O ports. The execution time of all the loops stayed below 10000 microseconds.

Thus, if the execution times of some subdiagrams of the loop fluctuated, the total execution time of the loop had to fluctuate too. Figure 12 show that it never exceeded 4000 microseconds, which is well below the maximum allotted time of 9984 microseconds.

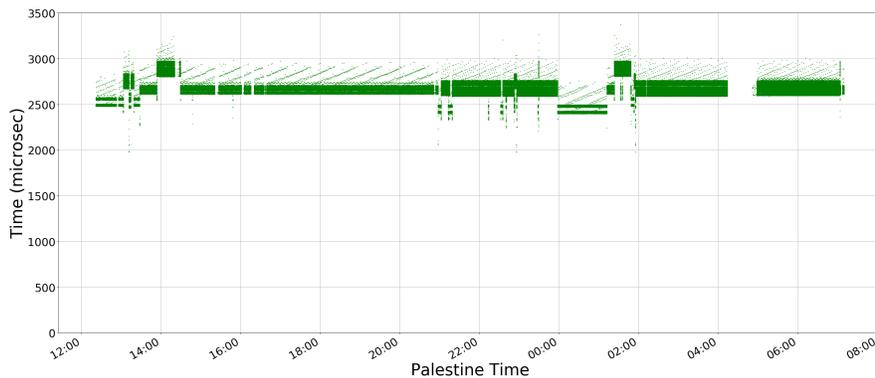


Figure 12: Total execution time of the Apply PIDs control loop. The variable execution time of some of the subsections of the loop (shown in Figure 11) also made the total execution time of the Apply PIDs control loop variable. B

Because the subdiagram corresponding to Figure 11 was the only one in the loop with a variable execution time, the total execution time of the loop shown in Figure 12 has the exact same print. In fact, it just adds a constant time value that corresponds to the aggregated execution time of the rest of the subdiagrams in the loop. The variable execution a loop never delayed the start of the following loop. That is, all control loops during flight started exactly 9984 microsecond apart.

Constantly keeping up with the required execution times of all the control loops involved in the BETTII control system is crucial for smooth functioning of all flight operations that eventually lead and enable science data capture.

3.3 Azimuth pointing and mode transitions

During flight, the payload experiences different environmental conditions, which require a different set of sensors and actuators to be active at each stage of the flight. This section summarizes the flight procedure from the launchpad to observation at 135,000 feet and presents some data about the pointing of BETTII during flight and testing at the Columbia Scientific Balloon Facility (CSBF) in Palestine, TX.

While rolling to the launchpad, during balloon inflation, and during ascent, BETTII is in SAFE mode. No commands are given to motors and all the PID gains of the control system are set to zero, allowing free movement of the gondola as it ascends for ~ 2 hours. As the payload ascends, turbulence begins to die down, and the payload is put into BRAKE mode. This reduces the gondola's inertial velocity, enabling the star cameras to obtain unblurred images and hence determine the gondola's current pointing. The gyroscopes are used to monitor the braking process. Figure 13 shows the three gyroscopes velocities during flight and how we reduced the amplitude of the azimuth velocity by two orders of magnitude while in Brake mode. Due to the fact that the CCMG wheels were not spinning we couldn't achieve optimal behavior, but we were able to brake enough to stabilize the payload.

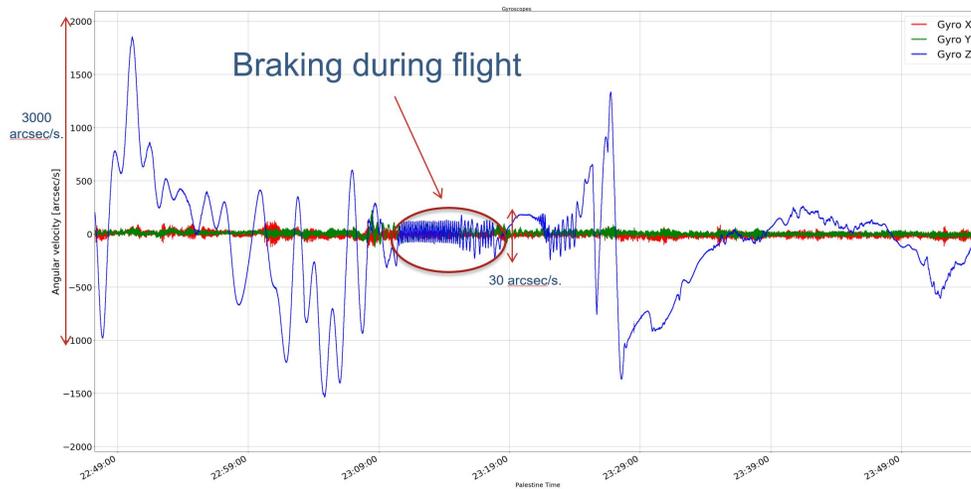


Figure 13: This figure shows the transition from Safe mode to Brake mode, where we decreased the amplitude of the oscillations of the inertial azimuth velocity from $3000''s^{-1}$ to $30''s^{-1}$,

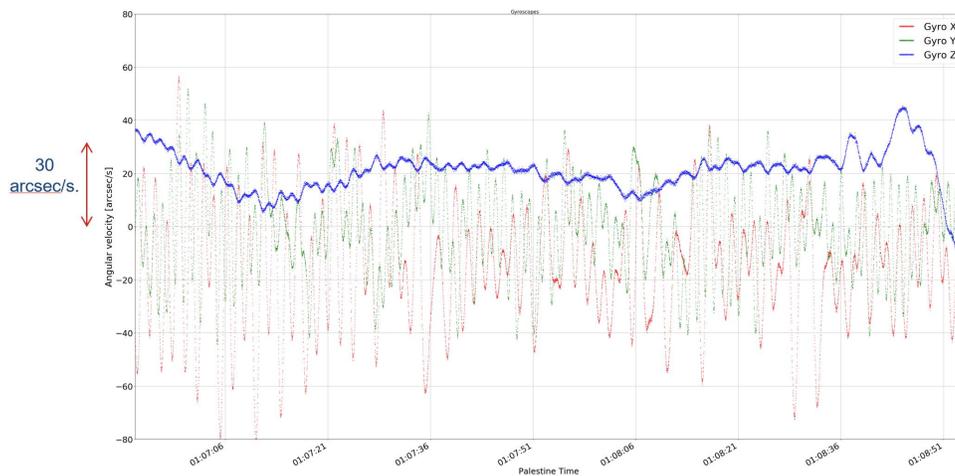


Figure 14: Braking during another stage of flight just with the momentum dump motor. An inertial azimuth velocity of $30''s^{-1}$ is enough to find star camera solutions and to determine absolute pointing.

Once we have determined our pointing position we can start the observation procedure. A new target in RA and DEC is set from the ground and transmitted up to the flight computers, causing the system to enter SLEW mode. This creates a profile of desired azimuth position and velocity as a function of time for each step of the control loop (every 10ms). The software commands the reaction wheels to turn the payload about its z axis following these predetermined velocity and position profile points. These profiles create a smooth transition from any random position to the desired azimuth angle of the target that we want to observe, which could be as far as 180 degrees away from the current orientation of the payload in either direction (clockwise or counterclockwise). This profile is implemented to avoid a large jump in the error signal that drives the azimuth control loop. The profile generates a velocity and position setpoint for each step of the control loop, which are fed every 10ms to the proper actuators; the stepper motors that control the rotation axis of the CCMG wheels and the momentum dump (azimuth control), and the rotation stage angle (elevation control). While the maximum azimuth acceleration of the payload is a constraint given by the inertia, the total mass distribution of the payload (around 1 ton) and the capabilities of the actuators, the maximum slew cruise velocity (V_{max}) is a parameter that we can adjust from the ground and defaults to 2000 arcseconds/s. The following graph is an example of a calculated slew profile (velocity and position), where P_{target} stands for the azimuth position of the target that we want to observe. For the control system, we determine that the target to be observed is always at $0''$ azimuth, thus, the azimuth position is always determined respect to the target. In the graph, P_o is our current azimuth position (which is negative in this example). In order to reach our target, in this example we will move the payload in a clockwise direction around its z axis (see Figure 15).

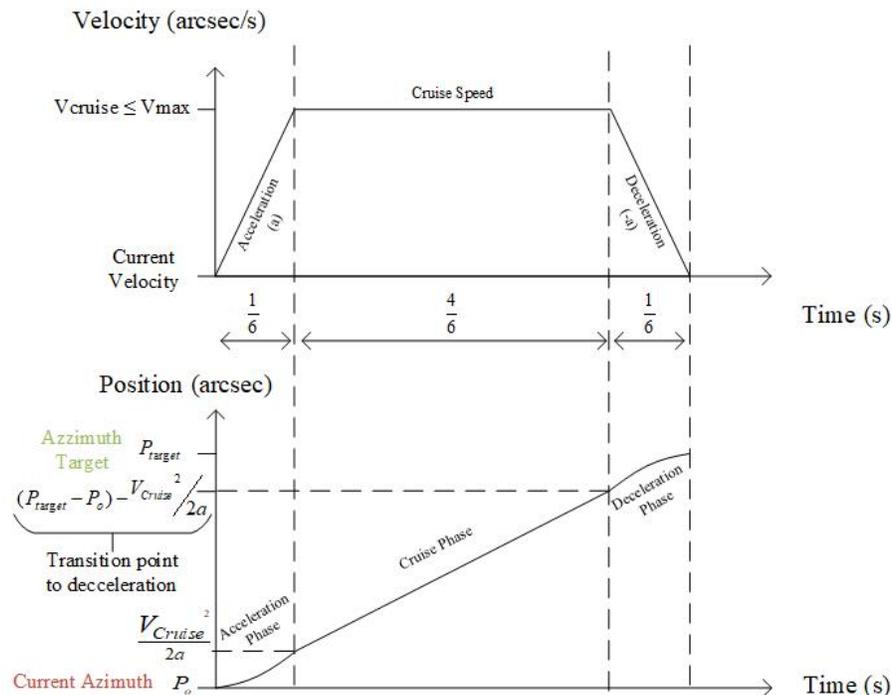


Figure 15: The Slew profile in velocity and position always has an acceleration stage which corresponds to $\frac{1}{6}$ th of the total slew time, a constant cruise period lasting $\frac{4}{6}$ th of the slew time and a deceleration phase taking the remaining $\frac{1}{6}$ th of the total time. This diagram shows a positive cruise velocity but it could be negative if the azimuth projection of the target star is to the right of the current azimuth pointing.

Figures 16 and 17 show data measurements of the azimuth and velocity setpoints (red) overlapped with the actual azimuth position and velocity of the payload during our tests before launch. These tests were used to tune up the gains of the control system. The background colors represent the State of the payload at each control loop cycle and have been extracted from the telemetry data on the estimator control loop.

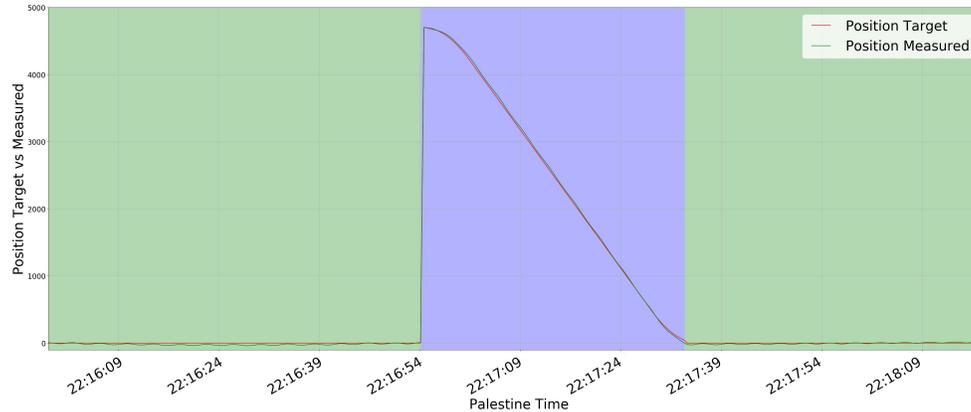


Figure 16: Evolution of the azimuth position during a slew profile. The azimuth of the target star we want to observe is always 0, thus when we updated the target setpoint, our current location jumped to 4800'' and the system automatically transitioned from safe mode to track (green background to blue background). The green line corresponds to the azimuth position of the payload whereas the red line corresponds to the set points generated by the slew profile algorithm. The payload perfectly followed the predetermined slew profile during the acceleration, cruise and deceleration phases.

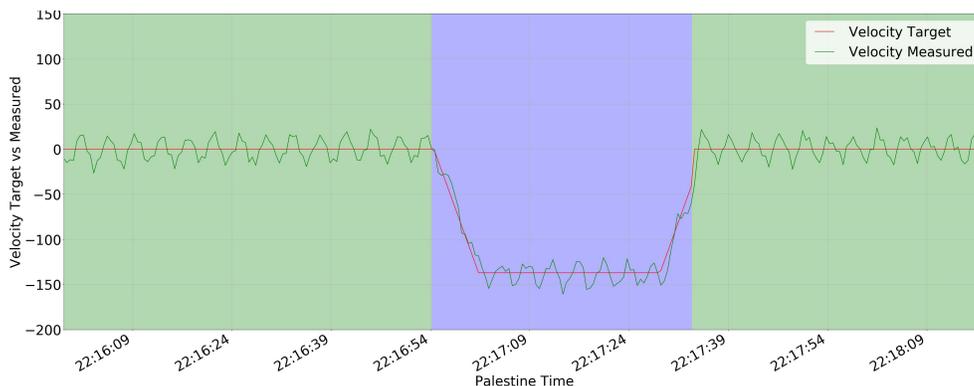


Figure 17: Evolution of the azimuth velocity during the same slew profile as in Figure 16. The red line denotes the setpoints whereas the green line is the actual azimuth velocity of the payload measured by the gyroscopes. We can clearly appreciate the acceleration, cruise and deceleration phases taking $\frac{1}{6}$ th, $\frac{4}{6}$ ths and $\frac{1}{6}$ th, respectively, of the total slew time. As shown in Figure 16, at the beginning of the slew we are 4800'' away from our target. Thus, the slew velocity has to be negative in order to reach the 0'' azimuth position corresponding to the target.

At the same time that the payload moves in azimuth, the control loop also commands the rotation stages that control the telescopes' elevation. The control in elevation and control in azimuth are entirely decoupled but occur simultaneously. In fact, before the slew profile is completed, the

system calculates the target's elevation -for the desired azimuth angle- and moves the siderostat mirrors at the proper elevation. This will allow BETTII to start the observations as soon as the azimuth slew profile is completed.

When the deceleration phase is complete and the azimuth of the target has been reached, the system switches to TRACK mode. This mode tries to maintain control of the telescope within $\pm 15''$ of the observed target. The TRACK mode requires small azimuth and elevation corrections. For this reason the PID gains that drive the actuators are completely different from the gains used during Slew mode. While in TRACK mode, the error signal fed to the PID control loop is the difference between the azimuth position of the target and the current azimuth angle of the payload. If the magnitude of this error signal gets above a certain threshold (i.e., due to an unexpected perturbation caused by atmospheric conditions) the system automatically switches back to SLEW mode and approaches the target gradually, following, once again, a smooth slew profile that keeps the error signal of the PIDs constant within a range.

Finally, for each of the two arms, we need to acquire a guide star onto the near-infrared angle tracking camera. This requires a fast imaging capability and a fast-steering tip/tilt correction mechanism to freeze the motion of the sky on the guide camera. This is ACQUIRE mode. Two images of the sky are captured by the detector, one from each arm; the guide star is located in each image, and the tip/tilt mechanisms are actuated to center this star onto a location of the detector that corresponds to maximum overlap at the science detectors. Once the star is centered onto that location, the window size of the camera decreases, and the acquisition speed increases. When on target, two subarrays of 3535 pixels each are read out at 50 Hz. When this acquisition speed is reached BETTII enters Locked mode. In both Acquire and Locked mode, the position of the two Tip/Tilt platforms contains information on the overall mispointing of the optical train: when the actuators are both off in the same direction with respect to their nominal position, it means that the entire truss is off the guide star by this amount. When available, this information is used by the estimator along with the gyroscope and star camera information to compute the best possible attitude estimate. Since the tip/tilt is tied to the actual optics train, its information is heavily weighted compared to the other sensors such as the star camera or the gyroscopes.

The red background on Figure 18 indicates the payload was in Track and Acquire mode. The figure shows how we followed the star in the sky, oscillating both in position around 0 arcseconds, which correspond, as always, to the azimuth projection angle of the target in the gondola reference frame. This data corresponds to the pre-flight tests carried out in Palestine, Tx.



Figure 18: Azimuth position of the payload while in acquire mode (green line) with respect to our target (red line at $0''$). An amplitude of $15''$ is enough to center the target star into our science detectors using the tip/tilt mechanism.

4. CONCLUSION

As an Engineering Flight, BETTII's maiden voyage was very successful; we were able to demonstrate the functionality of nearly every system. BETTII flew the night of June 8th 2017 and operated for nearly 12 hours before shutdown at 7 AM on June 9. Unfortunately, when BETTII was released from the balloon around 8 AM, an anomaly occurred resulting in the separation of the payload from the parachute. BETTII fell 136,000 feet in 12 minutes, and the loss was near-total. A NASA investigation post-flight showed that this problem occurred because the steel pin that attached the gondola to the payload had snapped; additional discussion of this can be found in two other papers presented at this meeting (Rinehart et al. and Veach et al.). Fortunately, of the ten solid state drives on the payload, eight were recovered. Data was stored in mirrored solid-state drives, and all (over a terabyte) of the flight data was recovered, including extensive engineering and monitoring data.

In this paper, we have covered elements of the control system that operated successfully on flight. This includes the sensors and actuators that form the control system of BETTII, and the ability of the control loop software to provide closed-loop pointing control in all modes of operation. We also demonstrated that at low pressure and temperatures, the electronics and the flight computers behaved as expected and that the software performance was near-perfect all throughout flight.

The same team of scientists and engineers that built BETTII have started to develop BETTII2, building directly upon the development of the BETTII experiment. By incorporating lessons learned from BETTII, BETTII2 will have improved scientific capabilities. The successful BETTII flight took a significant step forward towards the system-level demonstration needed to pave the way for future space-based interferometers; BETTII2 will complete this demonstration and will provide valuable experience needed to develop the techniques for working with interferometric data. BETTII2 could fly as early as summer 2020.

REFERENCES

- [1] Leisawitz, D., Baryshev, A., Griffin, M. J., Helmich, F. P., Ivison, R. J., Rinehart, S. A., Savini, G., and Shibai, H., “Advancing toward far-infrared interferometry in space through coordinated international efforts,” (2013).
- [2] Kouveliotou, C., Agol, E., Batalha, N., Bean, J., Bentz, M., Cornish, N., Dressler, A., Figueroa-Feliciano, E., Gaudi, S., Guyon, O., Hartmann, D., Kalirai, J., Niemack, M., Ozel, F., Reynolds, C., Roberge, A., Straughn, K. S. A., Weinberg, D., and Zmuidzinas, J., “Enduring Quests-Daring Visions (NASA Astrophysics in the Next Three Decades),” *ArXiv e-prints* (Jan. 2014).
- [3] Rinehart, S. A., Rizzo, M., Benford, D. J., Fixsen, D. J., Veach, T. J., Dhabal, A., Leisawitz, D. T., Mundy, L. G., Silverberg, R. F., Barry, R. K., Staguhn, J. G., Barclay, R., Mentzell, J. E., Griffin, M., Ade, P. A. R., Pascale, E., Klemencic, G., Savini, G., and Juanola-Parramon, R., “The Balloon Experimental Twin Telescope for Infrared Interferometry (BETTII): An Experiment for High Angular Resolution in the Far-Infrared,” *Publications of the ASP* **126**, 660–673 (July 2014).
- [4] Rizzo, M. J., *BETTII: A pathfinder for high angular resolution observations of star-forming regions in the far-infrared*, PhD thesis, University of Maryland, College Park (Jan. 2016).
- [5] Rizzo, M. J., Mundy, L. G., Dhabal, A., Fixsen, D. J., Rinehart, S. A., Benford, D. J., Leisawitz, D., Silverberg, R., Veach, T., and Juanola-Parramon, R., “Far-Infrared Double-Fourier Interferometers and their Spectral Sensitivity,” *Publications of the ASP* **127**, 1045–1060 (Oct. 2015).
- [6] Dhabal, A., Rinehart, S., Rizzo, M. J., Mundy, L., Sampler, H., Juanola-Parramon, R., Veach, T., Fixsen, D., and Vila, J., S. R., “Optics alignment of a balloon-borne far-infrared interferometer BETTII,” in [*Optical and Infrared Interferometry IV*], *Proceedings of the SPIE* **9907** (2017).
- [7] Dhabal, A., Rinehart, S., Rizzo, M. J., and Mundy, L., “Optics of the Balloon Experimental Twin Telescope for Infrared Interferometry (BETTII): delay lines and alignment,” in [*Optical and Infrared Interferometry IV*], *Proceedings of the SPIE* **9907** (2016).
- [8] Veach, T. J., Rinehart, S. A., Mentzell, J. E., Silverberg, R. F., Fixsen, D. J., Rizzo, M. J., Dhabal, A., Gibbons, C. E., and Benford, D. J., “The balloon experimental twin telescope for infrared interferometry (BETTII): optical design,” in [*Optical and Infrared Interferometry IV*], *Proceedings of the SPIE* **9146**, 91462H (July 2014).
- [9] Vila Hernandez de Lorenzo, J., Nehmetallah, G., and Rinehart, S., “Attitude Control System for a Balloon Based Telescope,” in [*Proceedings Volume 10646, Signal Processing, Sensor/Information Fusion, and Target Recognition XXVII*], *Proceedings of the SPIE* (Apr. 2018).
- [10] Breckenridge, W. G., “Quaternions - Proposed Standard Conventions. ech. rep. INTEROFFICE MEMORANDUM IOM 343-79-1199. NASA-JPL,” (1999).
- [11] Caley, A., “The Collected Mathematical Papers of Arthur Cayley,”
- [12] Trawny, N. and Roumeliotis, S., “Indirect Kalman filter for 3D attitude estimation,” in [*In: University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep 2, (cit. on pp. 13, 14)*], *Proceedings of the SPIE* (20015).
- [13] Casalprim Torres, M., “Control System of the Balloon Experimental Twin Telescope for Infrared Interferometry,” in [*ISAE-Supaero and Polytechnic University of Catalonia*], (2017).
- [14] Vila Hernandez de Lorenzo, J., “Telemetry System of the Ballon Experimental Twin Telescope for Infrared Interferometry,” ISAE-Supaero and Polytechnic University of Catalonia (2014).