Three-Axes Attitude Determination and Control System Design for Low-Cost Micro-Satellites

Craig J. Van Beusekom, Ron Lisowski, Joe M. Fulton, Christophe Morand
USAF Academy Dept. of Astronautics
2354 Faculty Drive
USAF Academy CO 80841
719-333-4110
c03craig.vanbeusekom@usafa.edu, ron.lisowski@usafa.edu, joe.fulton@usafa.edu, christophe.morand@usafa.edu

Abstract—FalconSat-3 is a 25 kg microsatellite being designed and built by cadets at the USAF Academy as part of on-going scientific research sponsored by the Department of Defense. The biggest design challenge for this mission is the requirement for onboard 3-axis attitude determination and control (ADCS), a capability earlier USAFA-built satellites lacked. The added challenge is to develop such a system as part of a low-cost, undergraduate satellite program. This paper begins with a brief overview of the Academy satellite program. Specific ADCS requirements for the FalconSAT-3 mission are then presented, followed by a detailed discussion of the preliminary system design. This paper then focuses on the FalconSAT-3 method of attitude determination. Attitude conventions, satellite attitude motion, and extended Kalman filtering are presented in a format ideal for readers desiring an introduction to designing micro-satellite attitude determination. The paper concludes with preliminary simulation results, a discussion of the current program status, and a description of future work.

TABLE OF CONTENTS

1. INTRODUCTION

FalconSat-3 is the fourth microsatellite developed at the USAF Academy Small Satellite Research center. Prior satellites have conducted research such as GPS signal measurements outside the constellation and spacecraft wake charging studies. The flight model of the most recent satellite—FalconSAT-2—has been assembled and is currently awaiting launch. It carries a device to measure ionospheric plasma. None of these platforms have required three-axis attitude determination or attitude control to meet the mission requirements.

FalconSAT-3 will fly three new payloads for the Air Force. This ambitious mission will collect fundamental data on the low-Earth orbit plasma environment using two, revolutionary micro-sensors called FLAPS and PLANE as well as serve as a test bed for new class of micro-actuators using pulse plasma thrusters.

With only 25 cadets and a dozen faculty members who work part time with FalconSAT coupled with the small budget of the program, system design must concentrate on using commercial off-the-shelf (COTS) technology. The mission is all the more challenging given there is nearly 100% turnover of cadets every year. The program has a goal of designing and building satellites in an aggressive two-year cycle.

USAFA has developed a relationship with the Surrey Space Center and Surrey Satellite Technology Ltd in Guildford, UK. FalconSAT-2 will fly Surrey-developed systems for power, communications, and computing. FalconSAT-3 will add ADCS to that list. SNAP-1 (Surrey Nanosatellite Applications Platform) was launched June 28th, 2000. The mission was to develop nanosatellite architecture that could be adapted for future platforms. At an impressive 6.5 kg, SNAP-1 is currently the smallest satellite to have flown with 3-axis active attitude control. By using a miniaturized magnetometer, momentum wheel, and three magnetorquers, SNAP-1 was able to achieve less than 1° knowledge accuracy and 3° pointing accuracy.

---

[1] U.S. Government work not protected by U.S. copyright

[2] IEEEAC paper #1416, Updated December 10, 2002

	Magnetometer	Sun Sensors	Torqrods	Momentum Wheel	ADCS Module
Manufacturer	SSTL	AeroAstro	SSTL	SSTL	SSTL
Quantity	1	6	3	1	1
Type	Fluxgate	Four-Quadrant Photo-Diode	Nickel-alloy core	Brushless DC Motor	C515 CAN μController
Range	±60 μTesla	136° FOV	±0.127 Am^2	0-5000 rpm 0-0.01 Nms	-
Resolution/ Accuracy	±60 nT	1.0°	10 msec min. pulse	±5 rpm	-
Mass (gram)	117	36	75 each	80	300
Size (mm)	35x32x83	10xΦ9	165xΦ8	40xΦ47	168x122x20
Power (mW)	150	0	100	100-500	250

Table 1 - FalconSAT-3 Engineering Model ADCS Components

FalconSAT-3 will fly the same basic hardware as SNAP-1 with the addition of sun sensors. Nevertheless, FalconSAT-3 will have nearly four-times the mass and all-new attitude determination and control algorithms must be developed for the on-board computer. Undergraduates must develop an attitude determination scheme consistent with our software limitations and technical experience—which is precisely what this paper addresses.

2. FS3 ADCS System Design

FalconSAT-3 must exploit as much as possible the capabilities offered by the COTS Surrey hardware to meet its attitude determination and control requirements.

FalconSAT-3			
Launch:	Boeing Delta IV MLV05	Size	50 cm (20") cube
Adapter:	ESPA	Mass:	25 kg
Date:	Mar-06	Cost:	$1.75M
Delivery:	Sep-05	Inclination:	35°
Design Life:	6 months	Altitude	540 km

Table 2 - FalconSAT-3 Design Details

FalconSAT-3 Mission

FalconSAT-3 has three core mission requirements. First, it will carry three experimental payloads to conduct DoD research. For the cadets involved, the program will "establish and sustain a cadre of space professionals" by providing an opportunity for USAFA cadets to "learn space by doing space." Finally, the satellite will continue to validate subsystem design for future FalconSAT missions.

Plasma Local Noise Environment (PLANE)—The PLANE payload will carry two sensors. They will measure ambient plasma levels to help characterize the ionosphere. The spacecraft facet with the instruments must be kept in the ram (velocity-vector) direction.

Flap Plasma Spectrometer (FLAPS)—The FLAPS payload will provide scintillation data, differential in angle and energy, in order to gain a better understanding of the ionosphere and scintillation conditions. Both PLANE and FLAPS will provide valuable data that can be used to forecast ionospheric satellite communication and navigation outages caused by plasma. These temporary outages have adversely affected DoD missions around the world. The experiments and hardware were designed and built by cadets and faculty of the USAF Academy Department of Physics

Micro-Propulsion Attitude Control System (MPACS)—The MPACS payload will be the first on-orbit demonstration of a new class of reduced-sized micro pulsed plasma thrusters developed by the Air Force Research Laboratory for attitude control. Two of these tiny actuators can provide up to 25 μN of thrust each with 8-10W of power. The current design envisions four clusters of three micro-PPTs each, which will allow for 3-axis attitude control when placed at opposite corners of the spacecraft. AFRL is providing the MPACS hardware to USAFA.

MPACS will not be operating while the other scientific payloads are collecting data, because the plasma released in the exhaust plume is not differentiable from the ionospheric plasma and would contaminate the data. However, for that very reason the scientific instruments aboard FalconSAT-3 will be useful in confirming the current models of the vacuum exhaust plume of the micro-PPTs.

Engineering Model Integration during Spring 2003—The USAF Academy currently uses a design, test, and evaluation process that manufactures three models of each spacecraft—an engineering model, qualification model, and flight model. In addition to providing a thorough test campaign to evaluate the performance of spacecraft subsystems, this allows each graduating class of cadets to build a deliverable model. Integration for the FalconSAT-3 Engineering Model will commence in February 2003 with the test campaign sequence planned for April 2003. The goal for completing the final integration of the flight model will be summer 2004. Based on further analysis and engineering model results, the qualification and flight models of FalconSAT-3 may include sensors and actuators different from the engineering model.

ADCS Requirements

The driving requirements for FalconSAT-3 ADCS are given by FLAPS and MPACS. FLAPS requires 1° of attitude knowledge and 5° of attitude control. MPACS requires 1° of attitude knowledge as well—the better accuracy that can be provided, the more significant the scientific results.

There are no slew requirements for FalconSAT-3 once the spacecraft has attained a velocity-vector oriented attitude. The only need for the actuators is to offset disturbance torques such as aerodynamic drag, residual magnetic dipole, and gravity gradient. The maximum disturbance torques have been calculated as follows:

Disturbance Torques	Worst Case Magnitude
Magnetic Torque (D=0.10)	5.1468×10^{-6} Nm
Drag Torque (c_{pa}-c_m=1cm)	1.163×10^{-6} Nm
Gravity Gradient Torque (FS2 MoI)	0.1933×10^{-6} Nm
Solar Pressure Torque	0.00735×10^{-6} Nm

Table 3 – FalconSAT-3 Worst Cases Disturbance Torques

Currently, FalconSAT-3 is planning for a 540 km, 35°-inclination orbit based on an opportunity for launch with EELV Secondary Payload Adapter (ESPA) on a Boeing Delta IV rocket in late 2006. However, the experiments are still able to provide valuable scientific data at a variety of altitudes and inclinations, and thus there are no orbit requirements other than the satellite be launched into low-earth orbit. Ultimately, the orbit depends solely on any launch FalconSAT-3 can get as a secondary payload.

FalconSAT-3 ADCS Hardware

Magnetometer—A standard Fluxgate magnetometer will measure the magnetic field vector in the spacecraft body frame. The usefulness of the magnetometer also depends on the accuracy of the local magnetic field vector propagated by the OBC with the most current International Geomagnetic Reference Field.

Magnetorquers—Three magnetorquers will provide an external control torque for FalconSAT-3. The engineering model will be built with the SNAP-1 torque rods. As shown in Table 2, we calculated that the residual magnetic dipole disturbance torque is the most significant. Both magnetorquer torque and magnetic disturbance are based on Earth's local magnetic field. Therefore it is crucial that the magnitude of each torque rod dipole be significantly greater than our expected residual dipole, to ensure FalconSAT-3 is controllable. For this reason, USAFA had SSTL re-design the torque rods to provide approximately four times the torque as the rods on SNAP-1. FalconSAT-3 must also be engineered with great care to reduce the residual magnetic dipole to the lowest levels feasible. A relatively large residual dipole on SNAP-1 gave Surrey challenges when operating SNAP-1 [3], and USAFA hopes to avoid that situation.

The magnetometer measurements will be invalid during the magnetorquer firing. The magnetorquers will be on a five second cycle with up to four seconds of operation and then a one second pause for an updated magnetometer measurement.

Momentum Wheel—The momentum wheel will be placed in the pitch direction, to provide the satellite with inertial stiffness in the orbit normal direction. The wheel and magnetorquer combination will allow the satellite to provide control torques in any axis—something the magnetorquers cannot do alone. The wheel provides a momentum bias that is also necessary for attitude determination during eclipse.

Sun Sensors—The current design is to have six sun sensors providing azimuth and elevation angles for attitude determination during the sunlight portion of the orbit. Virginia-based AeroAstro manufactures Medium Sun Sensors achieving 1.0° accuracy. The sun sensors are the four-quadrant photo-diode set inside a conical aperture with a 136° field-of view. They are engineering to prevent the entry of stray light such as albedo from entering the sensor. Approximately the size of a US nickel—they only require power for the analog/digital conversion. They will be mounted flush with the six sides of the cubic spacecraft.

ADCS Module and On-Board Computer—FalconSAT-3 will utilize the same Surrey-built ADCS module as SNAP-1. It provides the analog/digital interface between the on-board computer, sensors, and actuators—with the exception of the AeroAstro sun sensors that have a separate A/D interface. The OBC will execute the attitude determination and control algorithms for the satellite.

3. Attitude Conventions and Definitions

Coordinate Systems

This document uses three coordinate systems—the body axis coordinate system, the local orbital coordinate system, and the inertial coordinate system. The body axis coordinate system (B) is defined with an origin at the center of mass and the $+X_B$ direction orthogonal to the +X facet of the satellite—the side with the scientific instruments that requires pointing in the velocity vector direction. The $+Y_B$ and $+Z_B$ are each orthogonal to those spacecraft facets, respectively.

The local orbital frame used (O) is also known as the tangent-normal-radial (TNR) frame. The origin is at the spacecraft's center of mass. The primary axis $+X_O$ is tangent to the orbit at the local horizontal. The second axis $+Y_O$ will be normal to the spacecraft's orbit. The third axis is the radial direction from the center of the earth to the spacecraft.

The inertial frame (I)—known as the earth-centered inertial coordinate system—is defined with the origin at the center of the earth, the primary axis $+X_I$ toward the first point of Aires, and the X_I-Y_I plane in the equatorial plane, and the $+Z_I$-axis through the north pole.

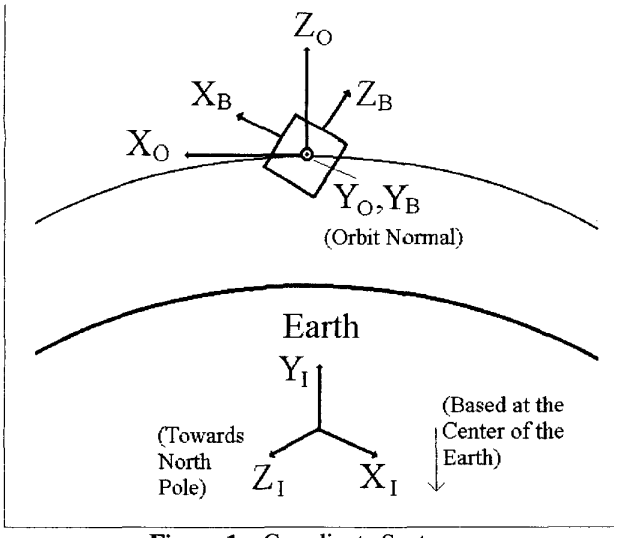

Figure 1 – Coordinate Systems

Earth Centered Inertial - Local Orbital Transformation

This transformation is used to transform a vector from the inertial to local orbital frames. The direction cosine matrix is created by augmenting three unit vectors (**U**, **V**, **W**) that form an orthonormal basis for the local orbit coordinate frame with respect to the inertial frame [7]. The vector **r** and **v** are the inertial-referenced position and velocity vectors. $\mathbf{C}^{O/}_{I}$ is a transformation matrix from inertial to local orbital.

$$\mathbf{C}^{O/}_{I} = \begin{bmatrix} U_x & U_y & U_z \\ V_x & V_y & V_z \\ W_x & W_y & W_y \end{bmatrix} \tag{1}$$

where

$$\begin{aligned} \mathbf{U} &= \begin{bmatrix} U_x & U_y & U_z \end{bmatrix}^T = \mathbf{V} \times \mathbf{W} \\ \mathbf{V} &= \begin{bmatrix} V_x & V_y & V_z \end{bmatrix}^T = \frac{\mathbf{r} \times \mathbf{v}}{|\mathbf{r} \times \mathbf{v}|} \\ \mathbf{W} &= \begin{bmatrix} W_x & W_y & W_z \end{bmatrix}^T = \frac{\mathbf{r}}{|\mathbf{r}|} \end{aligned} \tag{2}$$

Euler Angle Representation of Attitude

The attitude of a satellite is typically represented by Euler Angles. They represent an attitude change from the local orbit frame to the body axis frame. Several Euler angle rotations can be defined and used. FalconSat-3 will use the 1-2-3 transformation, the same as used by Steyn [1]. This represents a transformation first rotation of angle ϕ about the X_O-axis, then a transformation around the intermediate axis Y' by angle θ, and finally a rotation ψ around the new Z' axis. $\mathbf{C}^{B/}_{O}$ is a transformation matrix from the local orbital to body frames.

$$\mathbf{C}^{B/}_{O} = \begin{bmatrix} C\psi & S\psi & 0 \\ -S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & -S\theta \\ 0 & 1 & 0 \\ S\theta & 0 & C\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\phi & S\phi \\ 0 & -S\phi & C\phi \end{bmatrix} \tag{3}$$

Multiplying these three matrices together, we arrive at the direction cosine matrix (DCM).

$$\mathbf{C}^{B/}_{O} = \begin{bmatrix} C\psi C\theta & C\psi S\theta S\phi + S\psi C\phi & -C\psi S\theta C\phi + S\psi S\phi \\ -S\psi C\theta & -S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi + C\psi S\phi \\ S\theta & -C\theta S\phi & C\theta C\phi \end{bmatrix} \tag{4}$$

where C and S correspond to sine and cosine functions

Euler angles can be used to propagate orbit on board a satellite. However, when the time derivative of the DCM is taken, there is a singularity when pitch angle θ is equal to 0° or 180°. Another common way to represent attitude is with Euler symmetric parameters, also known as quaternions. In addition to being singularity free, no trigonometric functions are required when working with them—only additions and multiplications. Between any two coordinate systems, there exists a unique direction cosine matrix that will transform a vector from one coordinate to the other. Also, between any two attitude orientations there exists an Euler axis unit vector

$\mathbf{e} = [e_x \quad e_y \quad e_z]^T$ such that one orientation of angle Φ around the Euler axis will transform a vector from the first coordinate system to the second coordinate system [2]. With this Euler axis vector and the rotation angle Φ we can define the four-element quaternion vector:

$$\mathbf{q} = [q_1 \quad q_2 \quad q_3 \quad q_4]^T \tag{5}$$

where

$$q_1 = e_x \sin\frac{\Phi}{2}, \; q_2 = e_y \sin\frac{\Phi}{2}$$
$$q_3 = e_z \sin\frac{\Phi}{2}, \; q_4 = \cos\frac{\Phi}{2} \tag{6}$$

The following restriction exists on the quaternion. This can be confirmed with the above definitions.

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \tag{7}$$

The quaternion transformation matrix is defined as:

$$\mathbf{C}^{B/O} = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \tag{8}$$

This represents the exact same attitude transformation as the one defined in (3) and (4). Later on, we will find it necessary to differentiate this matrix with respect to each quaternion. These partial derivatives can be found in Appendix A. The opposite transformation—from the body axis coordinate system to the local orbital system—is simply the transpose.

$$\mathbf{C}^{O/B} = \left[\mathbf{C}^{B/O}\right]^{-1} = \left[\mathbf{C}^{B/O}\right]^T \tag{9}$$

If the Euler angle representation or the unique DCM matrix is known, the four quaternions can be found using the following relations where C_{ij} is the (i,j) component of the matrix $\mathbf{C}^{B/O}$ from (4) [2]. In the case that q_4 is equal to zero, a singularity exists in the calculation of the other elements of the quaternion. In that case, another set of equations can be derived from (8) to calculate the quaternions from the transformation matrix [2].

$$\begin{aligned} q_4 &= \frac{1}{2}[1 + C_{11} + C_{22} + C_{33}]^{0.5} \\ q_1 &= \frac{1}{4q_4}[C_{23} - C_{32}] \\ q_2 &= \frac{1}{4q_4}[C_{31} - C_{13}] \\ q_3 &= \frac{1}{4q_4}[C_{12} - C_{21}] \end{aligned} \tag{10}$$

Likewise, the three 1-2-3 Euler Angles are found from the transformation matrix (8) by the following relations [1]. (arctan 2 is the four quadrant arc-tangent function.)

$$\begin{aligned} \phi &= \arctan 2\left(\frac{-C_{32}}{C_{33}}\right) \\ \theta &= \arcsin(C_{31}) \\ \psi &= \arctan 2\left(\frac{-C_{21}}{C_{11}}\right) \end{aligned} \tag{11}$$

4. Equations of Motion

FalconSat-3's dynamics will be modeled using Euler's moment equation. Three body torques will be included in this analysis—the magnetorquer control torque, the momentum wheel angular acceleration torque, and an all-encompassing disturbance torque term. All variables are written in the body frame.

$$\mathbf{I}\dot{\boldsymbol{\omega}}_B^I = \mathbf{N}_{MT} + \mathbf{N}_D + \mathbf{N}_T - \boldsymbol{\omega}_B^I \times (\mathbf{I}\boldsymbol{\omega}_B^I + \mathbf{h}_W) + \dot{\mathbf{h}}_W \tag{12}$$

where,

$\mathbf{I}$ = the 3x3 moment of inertia tensor

$\mathbf{N}_{MT} = [N_{MT_x} \quad N_{MT_y} \quad N_{MT_z}]^T$ = magnetic torque vector

$\mathbf{N}_D$ = environmental disturbance torque vector

$\mathbf{N}_T$ = thruster control torque vector

$\boldsymbol{\omega}_B^I = [\omega_x \quad \omega_y \quad \omega_z]^T$ = inertially-referenced body angular rate written in the body frame

$\mathbf{h}_W = [0 \quad h_W \quad 0]^T$ = momentum wheel angular momentum

$\dot{\mathbf{h}}_W = [0 \quad \dot{h}_W \quad 0]^T$ = pitch momentum wheel torque

FalconSat-3 is expected to have a moment of inertia tensor with three large diagonal terms that are nearly the same and off-diagonal elements which are zero or very small. When estimating attitude, the actual moment of inertia will be used.

The attitude update of the satellite will be done using quaternions. The quaternion equation of motion is represented using the following set of differential equations represented in matrix form.

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2}\boldsymbol{\Omega}\mathbf{q} \tag{13}$$

with

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 & \omega_{oz} & -\omega_{oy} & \omega_{ox} \\ -\omega_{oz} & 0 & \omega_{ox} & \omega_{oy} \\ \omega_{oy} & -\omega_{ox} & 0 & \omega_{oz} \\ -\omega_{ox} & -\omega_{oy} & -\omega_{oz} & 0 \end{bmatrix} \tag{14}$$

and

$\boldsymbol{\omega}_B^O = \begin{bmatrix} \omega_{ox} & \omega_{oy} & \omega_{oz} \end{bmatrix}^T$ = local orbital referenced body angular rate written in the body frame

This vector can be found using the inertially referenced body rates and the orbital–referenced orbital rate vector by the following equations:

$$\boldsymbol{\omega}_B^O = \boldsymbol{\omega}_B^I - \mathbf{C}^{B/O}\boldsymbol{\omega}_0 \tag{15}$$

where

$\boldsymbol{\omega}_0 = \begin{bmatrix} 0 & \omega_0 & 0 \end{bmatrix}^T$ = inertially referenced orbital angular rate written in the orbital frame.

If we can assume that the satellite has a circular orbit, then we can find ω_0 with the equation for mean motion:

$$\omega_0 = \sqrt{\frac{GM}{a^3}} \tag{16}$$

where GM is the earth's gravitational parameter and a is the satellite's semi-major axis.

5. The Extended Kalman Filter

A Kalman filter is an optimal, recursive, data processing algorithm. It is a blend of statistics and matrix algebra where multivariable calculus is used to develop representations of non-linear systems such as spacecraft attitude. Vallado [5], Steyn [1], and Wertz [2] all address Kalman filtering more extensively than here.

The underlined description of the algorithm is very important. By data processing algorithm, it is meant that it is a set of equations used to transform a large amount of data into results. The filter is an algorithm. Optimal means that this particular solution or algorithm was developed through matrix algebra and calculus and results in the best interpretation of the noisy data. Recursive means that each step is based off variables from the previous step rather than all previous steps. The advantage of a Kalman lies with interpreting a continuous stream of noisy data. It stores information from previous measurements and information about the variability of the system in one small matrix. The extended Kalman filter is a variation of the traditional discrete Kalman filter that is more useful in extremely non-linear systems such as spacecraft attitude.

With any filter, we are estimating a state. In the case of FalconSat-3 attitude determination, the state we seek to estimate is the four elements of the quaternion and the three elements of the inertially referenced body angular rate.

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} & \boldsymbol{\omega}_B^I \end{bmatrix}^T = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & \omega_x & \omega_y & \omega_z \end{bmatrix}^T \tag{17}$$

Since we are using an extended Kalman filter, when we propagate forward to the next state, we can use a nonlinear state propagator between steps. This may include disturbance torques or the moment of inertia.

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \int_k^{k+1} f(\mathbf{x}_k, t_k)dt \tag{18}$$

We also assume that there exists a linearized model of the system dynamics that can relate the state estimate at one step to the state estimate at the next step. This requires us to make a few assumptions but we can represent the update from one step to the next with a matrix called the state transition matrix. The state vector $\mathbf{x}_k$ is assumed to be free of noise or error.

$$\mathbf{x}_{k+1} = \boldsymbol{\Phi}_k \mathbf{x}_k \tag{19}$$

At each time step, we take a measurement $\mathbf{z}_k$. There also must exist a matrix $\mathbf{H}_k$ that relates the state vector to the error-free value of the measurement.

$$\mathbf{x}_k = \mathbf{H}_k \mathbf{z}_k \tag{20}$$

To develop the filter, we assume that the measurement vector contains zero-mean uncorrelated white noise. Finally we define a residual matrix $\mathbf{b}_k$ that is the difference between the measurements and the expected value of a measurement at the predicted state.

$$\mathbf{b}_k = \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^- \tag{21}$$

The Extended Kalman Filter Algorithm

1. Calculation of the Kalman Gain

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \tag{22a}$$

2. State Update Estimate

$$\Delta\hat{\mathbf{x}}_k = \mathbf{K}_k \mathbf{b}_k \tag{22b}$$

3. Error Covariance Estimate

$$\mathbf{P}_k = (\mathbf{1}_7 - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^- \tag{22c}$$

4. State Estimate

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \Delta\hat{\mathbf{x}}_k \tag{22d}$$

5. Project ahead (Predict the next step's estimate and error covariance matrix)

$$\hat{\mathbf{x}}_{k+1}^- = \hat{\mathbf{x}}_k + \int_k^{k+1} f(\hat{\mathbf{x}}_k, t_k)dt \tag{22e}$$

$$\Delta\hat{\mathbf{x}}_{k+1}^- = 0 \tag{22f}$$

$$\mathbf{P}_{k+1}^- = \mathbf{\Phi}_k\mathbf{P}_k\mathbf{\Phi}_k^T + \mathbf{Q}_k \tag{22g}$$

This process is repeated for each step (each measurement from the sensors). For FalconSAT-3, this will be every 5 seconds.

Further elaboration of Kalman Filter Variables

At each step t_k:

$\hat{\mathbf{x}}_k$ = The calculated state estimate for this step (using measurement data from this step)

$\hat{\mathbf{x}}_k^-$ = The predicted state estimate from the previous step. The superscript is read as "super-minus."

$\mathbf{x}$ = The actual value of the state. We never know this value but seek to estimate it.

$\mathbf{P}_k^-$ = Predicted error covariance matrix from the previous step

$\mathbf{P}_k$ = Calculated error covariance matrix

$\mathbf{H}_k$ = Measurement matrix.

$\mathbf{b}_k$ = Measurement Residual vector

$\int_k^{k+1} f(\hat{\mathbf{x}}_k, t_k)dt$ = *Nonlinear State Propagation Term*—This term represents the numerical integration of the nonlinear propagator. If all the assumptions that we made for the state transition matrix were valid, it could be used to propagate the state. However, in a real system we can model known nonlinear effects such as disturbance torques and a non-principal axis moment of inertia. These two factors are what makes the nonlinear state propagation different than the state transition matrix.

$\mathbf{\Phi}_k$ = *State Transition Matrix* (7 x 7 matrix)—The state transition matrix or system matrix, represents the ideal linearized relationship between the state estimation at one measurement and the state estimation at the next measurement. Its formulation is based on the dynamics of the satellite. It must be calculated at each time step from the quaternions, the Euler rates, the Δt between measurements, and other factors such as the momentum wheel speed.

$\mathbf{P}_k$ = *Error Covariance Matrix* (7 x 7 matrix)—The error covariance matrix is the filter's way of estimating the variability of its previous predictions. The diagonal element stores the estimated variance in each state estimate (i.e. the estimated variance $\sigma_{q1}^2 \ldots \sigma_{\omega z}^2$ of our estimates for $q_1 \ldots \omega_z$). The non-diagonal elements store the estimated covariance between different elements of the state. This is the matrix variable that "stores" information about all previous measurements. When the filter receives several measurements that are very close to what it expected to receive, the values in the covariance matrix are likely to be relatively small. If the filter receives a measurement that very different than what it expected, it will respond much differently than if the covariance values had been relatively high. Smaller covariance matrices will make the filter put more faith in outlying measurements.

$\mathbf{K}_k$ = *Kalman Gain Matrix* (7 x 3 matrix)—The Kalman gain matrix—also known as the blending factor—is the optimal solution to blend the data from the previous estimate of the current state ($\hat{\mathbf{x}}_k^-$) and the current measurement ($\mathbf{z}_k$). A new Kalman gain matrix is calculated at each step. The optimization derivation can be seen in Vallado [5]. Its values are not intuitive and the best appreciation comes from looking at the actual filter algorithm itself.

$\mathbf{R}_k$ = *Observation Noise Matrix* (3 x 3 matrix)—The observation noise variance matrix is a diagonal matrix with values that are the engineer's estimated variance of the sensor measurements. Even though it is denoted as R_k, the observation noise matrix during our attitude determination is constant, so it can also be denoted as R.

$\mathbf{Q}_k$ = *Process Noise Matrix* (7 x 7 matrix)—The process noise covariance matrix estimates the total variance caused by the factors not modeled in the state transition matrix. For attitude determination, this includes disturbance torques, uncertainties in the hardware, and non-symmetric moments of inertia.

6. EKF APPLICATION TO FALCONSAT-3

FalconSat-3 State Transition Matrix

The first matrix that we need to define is the state transition matrix, which relates one estimated state to the next, as in (19). The next step is equal to the previous step plus the

time rate of change of the state vector—a function $F(\mathbf{x}) = \dot{\mathbf{x}}$. Since $F(\mathbf{x})$ is a function of all seven state variables, we can get the matrix form of this function by differentiating with respect to each state variable.

$$\begin{aligned}\mathbf{\Phi}_k &\approx \mathbf{1}_{7\text{x}7} + \frac{\partial F(\mathbf{x})}{\partial \mathbf{x}}\Delta t = \mathbf{1}_{7\text{x}7} + \frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}}\Delta t \\ &= \mathbf{1}_{7\text{x}7} + \begin{bmatrix} \frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} & \frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I} \\ \frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \mathbf{q}} & \frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \boldsymbol{\omega}_B^I} \end{bmatrix}\Delta t = \mathbf{1}_{7\text{x}7} + \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix}\Delta t\end{aligned} \tag{23}$$

Recall equation (13)—the quaternion rate equation. This equation is the first four rows of $F(\mathbf{x})$.

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & \omega_{oz} & -\omega_{oy} & \omega_{ox} \\ -\omega_{oz} & 0 & \omega_{ox} & \omega_{oy} \\ \omega_{oy} & -\omega_{ox} & 0 & \omega_{oz} \\ -\omega_{ox} & -\omega_{oy} & -\omega_{oz} & 0 \end{bmatrix}\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \frac{1}{2}\mathbf{\Omega q} \tag{24}$$

We will find it convenient to define another matrix corresponding to the same differential equation.

$$\mathbf{\Lambda} = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \tag{25}$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{\Omega q} = \frac{1}{2}\mathbf{\Lambda}\boldsymbol{\omega}_B^O \tag{26}$$

Recall that the equation for the attitude motion of FalconSAT-3 is:

$$\mathbf{I}\dot{\boldsymbol{\omega}}_B^I = \mathbf{N}_{MT} + \mathbf{N}_D + \mathbf{N}_T - \boldsymbol{\omega}_B^I \times (\mathbf{I}\boldsymbol{\omega}_B^I + \mathbf{h}_W) + \dot{\mathbf{h}}_W \tag{27}$$

The $\mathbf{N}_D$ term of the equation of motion includes all the non-linearities that the satellite experiences. We cannot easily predict most of these torques and others produce coupled, non-linear differential equations that cannot be put into matrix form. When developing the state transition matrix, we will assume that this term is zero. However, when propagating the attitude we will include these factors. Also recall that we are only dealing with one momentum wheel aligned with the spacecraft body frame y-axis. That simplifies the equation further.

To model the dynamics of the satellite for the Kalman filter, we will assume that the principal axes of the satellite are the X_B, Y_B, and Z_B axes and that the moments of inertia around each axis are identical.

$$\mathbf{I} = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_1 & 0 \\ 0 & 0 & I_1 \end{bmatrix} \tag{28}$$

With the above assumptions about the dynamics, equation (27) reduces to:

$$\begin{aligned} I_1\dot{\omega}_x &= N_{MT_X} + N_{T_X} + h_W\omega_z \\ I_1\dot{\omega}_y &= N_{MT_Y} + N_{T_Y} + \dot{h}_{W_Y} \\ I_1\dot{\omega}_z &= N_{MT_Z} + N_{T_Z} - h_W\omega_x \end{aligned} \tag{29}$$

Thus, with equations (26) and (29), we have defined $\dot{\mathbf{x}}$ and we can derive the four component matrices of $\mathbf{\Phi}$--F_{11}, F_{12}, F_{21}, and F_{22}, according to equation (23).

4 x 4 F_{11} Matrix Computation

Using equation (24) with substitutions from equations (26) and (15), the F_{11} matrix takes shape as follows [7]:

$$\begin{aligned} \frac{\partial \dot{\mathbf{q}}}{\partial q_i} &= \frac{1}{2}\mathbf{\Omega}\frac{\partial \mathbf{q}}{\partial q_i} + \frac{1}{2}\frac{\partial \mathbf{\Omega}}{\partial q_i}\mathbf{q} \\ \frac{\partial \dot{\mathbf{q}}}{\partial q_i} &= \frac{1}{2}\mathbf{\Omega}\frac{\partial \mathbf{q}}{\partial q_i} + \frac{1}{2}\mathbf{\Lambda}\frac{\partial \boldsymbol{\omega}_B^O}{\partial q_i} \\ \frac{\partial \dot{\mathbf{q}}}{\partial q_i} &= \frac{1}{2}\mathbf{\Omega}\frac{\partial \mathbf{q}}{\partial q_i} + \frac{1}{2}\mathbf{\Lambda}\frac{\partial}{\partial q_i}\left(\boldsymbol{\omega}_B^I - \mathbf{C}^{B/O}\boldsymbol{\omega}_O\right) \\ \frac{\partial \dot{\mathbf{q}}}{\partial q_i} &= \frac{1}{2}\mathbf{\Omega}\frac{\partial \mathbf{q}}{\partial q_i} - \frac{1}{2}\mathbf{\Lambda}\frac{\partial \mathbf{C}^{B/O}}{\partial q_i}\boldsymbol{\omega}_O \end{aligned} \tag{30}$$

Thus the matrix is computed by evaluating

$$\mathbf{F}_{11} = \frac{1}{2}\mathbf{\Omega} + \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 \end{bmatrix} \tag{31}$$

$$\text{where } \gamma_i = -\frac{1}{2}\mathbf{\Lambda}\frac{\partial \mathbf{C}^{B/O}}{\partial q_i}\boldsymbol{\omega}_O \tag{32}$$

4 x 3 F_{12} Matrix Computation

The partial derivative of $\dot{\mathbf{q}}$ with respect to $\boldsymbol{\omega}_B^I$ can be derived as follows. Again, it requires a substitution of equation (15) [7].

$$\begin{aligned} \frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I} &= \frac{\partial}{\partial \boldsymbol{\omega}_B^I}\left(\frac{1}{2}\mathbf{\Omega q}\right) = \frac{\partial}{\partial \boldsymbol{\omega}_B^I}\left(\frac{1}{2}\mathbf{\Lambda}\boldsymbol{\omega}_B^O\right) \\ \frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I} &= \frac{1}{2}\mathbf{\Lambda}\frac{\partial \boldsymbol{\omega}_B^O}{\partial \boldsymbol{\omega}_B^I} = \frac{1}{2}\mathbf{\Lambda}\frac{\partial}{\partial \boldsymbol{\omega}_B^I}\left(\boldsymbol{\omega}_B^I - \mathbf{C}^{B/O}\boldsymbol{\omega}_O\right) \\ \frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I} &= \frac{1}{2}\mathbf{\Lambda}\frac{\partial \boldsymbol{\omega}_B^I}{\partial \boldsymbol{\omega}_B^I} = \frac{1}{2}\mathbf{\Lambda} \end{aligned} \tag{33}$$

3 x 4 F_{21} Matrix Computation

Recall the equation for $\dot{\boldsymbol{\omega}}$ from (29). The only term on the

right side of the equation that is dependent on the attitude is the magnetorquer control torque. Thus the matrix F_{21} can be computed by evaluating:

$$\mathbf{F}_{21} = \left[\frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{q}}\right] = \frac{1}{I_1}\left[\frac{\partial \mathbf{N}_{MT}}{\partial \mathbf{q}}\right] \tag{34}$$

Note that if the magnetorquers are not firing, it will reduce to:

$$\mathbf{F}_{21} = \left[\mathbf{0}_{3x4}\right] \tag{35}$$

However, if the magnetorquers are actively correcting attitude, we must take into account how the magnetorquer torque varies with respect to each quaternion

$$\mathbf{N}_{MT} = \mathbf{M}_j \times \mathbf{B}_B \tag{36}$$

$$\mathbf{B}_B = \mathbf{C}^{B/O}\mathbf{C}^{O/I}\mathbf{B}_I \tag{37}$$

It is convenient to use the $[\mathbf{M}\times]$ cross product operator when computing this value. Note that m_x, m_y, and m_z are the magnetic dipoles of each torque rod while that torque rod is firing.

$$[\mathbf{M}\times] = \begin{bmatrix} 0 & -m_z & m_y \\ m_z & 0 & -m_x \\ -m_y & m_x & 0 \end{bmatrix} \tag{38}$$

$$\mathbf{N}_{MT} = [\mathbf{M}\times]\mathbf{C}^{B/O}\mathbf{C}^{O/I}\mathbf{B}_I \tag{39}$$

Now we can differentiate this value with respect to each quaternion.

$$\frac{\partial \mathbf{N}_{MT}}{\partial \mathbf{q}} = [\mathbf{M}\times]\frac{\partial \mathbf{C}^{B/O}}{\partial \mathbf{q}}\mathbf{C}^{O/I}\mathbf{B}_I \tag{40}$$

The resulting F21 matrix can be computed as follows:

$$\mathbf{F}_{21} = \left[\frac{\partial \dot{\boldsymbol{\omega}}}{\partial q_1} \quad \frac{\partial \dot{\boldsymbol{\omega}}}{\partial q_2} \quad \frac{\partial \dot{\boldsymbol{\omega}}}{\partial q_3} \quad \frac{\partial \dot{\boldsymbol{\omega}}}{\partial q_4}\right] \tag{41}$$

$$\text{where } \frac{\partial \dot{\boldsymbol{\omega}}}{\partial q_i} = \frac{1}{I_1}[\mathbf{M}\times]\frac{\partial \mathbf{C}^{B/O}}{\partial q_i}\mathbf{C}^{O/I}\mathbf{B}_I \tag{42}$$

3 x 3 F_{22} Matrix Computation

Using equation (29), the partial derivative F_{22} matrix can be found as follows.

$$\mathbf{F}_{22} = \begin{bmatrix} 0 & 0 & h_W/I_1 \\ 0 & 0 & 0 \\ -h_W/I_1 & 0 & 0 \end{bmatrix} \tag{43}$$

Magnetometer Observation Equation

The expected magnetometer observation is as follows, with **n** as a zero-mean white noise vector.

$$\hat{\mathbf{B}}_B = \mathbf{C}^{B/O}\mathbf{C}^{O/I}\hat{\mathbf{B}}_I + \mathbf{n} \tag{44}$$

To find the observation equation for the magnetometer, we need to differentiate this with respect to the state vector.

$$\mathbf{H}_m = \left[\frac{\partial \hat{\mathbf{B}}_B}{\partial \mathbf{q}} \quad \frac{\partial \hat{\mathbf{B}}_B}{\partial \omega_{BY}}\right] \tag{45}$$

$$\mathbf{H}_m = \left[\frac{\partial \mathbf{C}^{B/O}}{\partial q_1}\hat{\mathbf{B}}_O \quad \frac{\partial \mathbf{C}^{B/O}}{\partial q_2}\hat{\mathbf{B}}_O \quad \frac{\partial \mathbf{C}^{B/O}}{\partial q_3}\hat{\mathbf{B}}_O \quad \frac{\partial \mathbf{C}^{B/O}}{\partial q_4}\hat{\mathbf{B}}_O \quad \mathbf{0}_{3x3}\right]$$

$$\text{where } \hat{\mathbf{B}}_O = \mathbf{C}^{O/I}\hat{\mathbf{B}}_I \tag{46}$$

Sun Sensor Observation Equation

The azimuth and elevation measurements from the sun sensor can be converted into a body-frame sun vector. The sun sensor observation equation, therefore, is identical to the formulation of the magnetometer observation equation above.

$$\mathbf{H}_s = \left[\frac{\partial \mathbf{C}^{B/O}}{\partial q_1}\hat{\mathbf{S}}_O \quad \frac{\partial \mathbf{C}^{B/O}}{\partial q_2}\hat{\mathbf{S}}_O \quad \frac{\partial \mathbf{C}^{B/O}}{\partial q_3}\hat{\mathbf{S}}_O \quad \frac{\partial \mathbf{C}^{B/O}}{\partial q_4}\hat{\mathbf{S}}_O \quad \mathbf{0}_{3x3}\right]$$

$$\text{where } \hat{\mathbf{S}}_O = \mathbf{C}^{O/I}\hat{\mathbf{S}}_I = \text{the local orbital sun vector} \tag{47}$$

If the filter is running with just the magnetic field or sun vector measurements, then the measurement $\mathbf{z}_k$ is equal to that vector and $\mathbf{H} = \mathbf{H}_m$ or $\mathbf{H}_s$. However, if there are both magnetometer and sun sensor measurements available, then both the measurements and observation matrices need to be augmented.

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_m \\ \mathbf{H}_s \end{bmatrix} \text{ and } \mathbf{z}_k = \begin{bmatrix} \hat{\mathbf{B}}_{B-measured} \\ \hat{\mathbf{S}}_{B-measured} \end{bmatrix} \tag{48}$$

Process Noise Covariance Matrix

The process noise covariance matrix is used to represent the uncertainty in the linearized model of the system and how that uncertainty is correlated between the states. Based on the design of a Kalman filter and probability definitions, the process noise covariance matrix **Q** for step k is defined as follows [6]:

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1},\tau)\mathbf{G}(\tau)E\left[\mathbf{w}(\tau)\mathbf{w}(\alpha)^T\right]\mathbf{G}(\alpha)^T\mathbf{\Phi}(t_{k+1},\alpha)^T d\alpha d\tau \tag{49}$$

For our state vector and process, this reduces to the following equation, where F_{12} and F_{22} are parts of the state transition matrix defined earlier. The full derivation is found in Appendix B.

$$\mathbf{Q}_k = \mathbf{Q}_1\Delta t + \mathbf{Q}_2\frac{(\Delta t)^2}{2} + \mathbf{Q}_3\frac{(\Delta t)^3}{3} \tag{50}$$

where:

$$\mathbf{Q}_1 = \begin{bmatrix} \mathbf{0}_{4x4} & \mathbf{0}_{4x3} \\ \mathbf{0}_{3x4} & \mathbf{S} \end{bmatrix} \tag{51}$$

$$\mathbf{Q}_2 = \begin{bmatrix} \mathbf{0}_{4x4} & \mathbf{F}_{12}\mathbf{S} \\ \mathbf{S}\mathbf{F}_{12}^T & \mathbf{S}\mathbf{F}_{22}^T + \mathbf{F}_{22}\mathbf{S} \end{bmatrix} \tag{52}$$

$$\mathbf{Q}_3 = \begin{bmatrix} \mathbf{F}_{12}\mathbf{S}\mathbf{F}_{12}^T & \mathbf{F}_{12}\mathbf{S}\mathbf{F}_{22}^T \\ \mathbf{F}_{22}\mathbf{S}\mathbf{F}_{12}^T & \mathbf{F}_{22}\mathbf{S}\mathbf{F}_{22}^T \end{bmatrix} \tag{53}$$

$$\mathbf{S} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_x^2 & 0 \\ 0 & 0 & \sigma_x^2 \end{bmatrix} \tag{54}$$

The variances $\sigma_x^2 \ldots \sigma_x^2$ are an estimated variance of the body rates. This is not a straightforward quantity and best determined by computer simulation.

7. RESULTS

At the time of this paper's writing, only initial simulation results were available. Before attitude determination is complete for FalconSAT-3, there will likely be many more weeks of testing and tuning. Nevertheless, the results are very optimistic and demonstrate the Extended Kalman Filter method introduced in previous sections.

These simulation results were all completed during sun-lit portions of orbit, when both the sun sensors and magnetometer were providing data to the filter. The simulation propagates the actual orbit of FalconSAT-3 and calculates the sun vectors. The local magnetic field is calculated using the International Geomagnetic Reference Field (IGRF) for the year 2000. The IGRF is a standard model of the earth's magnetic field and is the one typically used on satellites such as FalconSAT-3. The IGRF allows the user to calculate an estimate of the local magnetic field from spherical harmonics and the current satellite and position vector in the inertial frame.

The following graph demonstrates how the filter will typically converge to the correct attitude from a very poor initial guess. All simulation measurements have the actual noise that we would expect from our sun sensors and magnetometers.

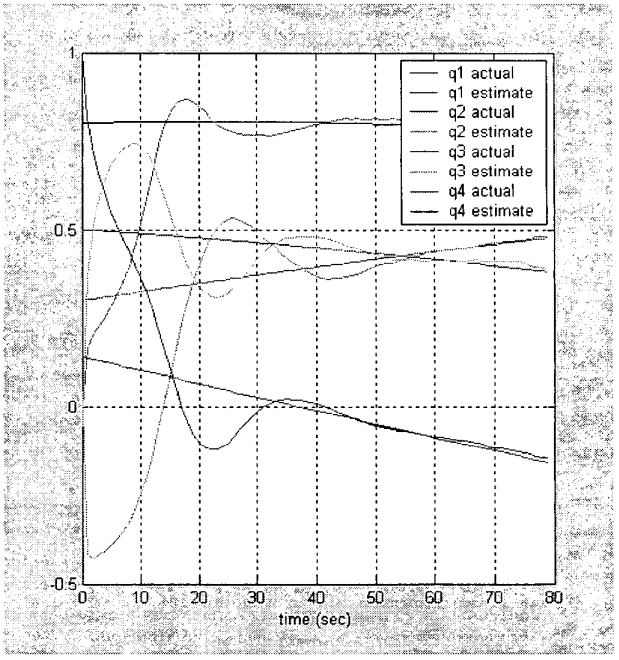

Figure 2 – Convergence of Quaternions

As shown in Figure 2, when given noisy data, the filter often tends to oscillate before finally converging on the actual quaternion. The following figure shows how the absolute error of each quaternion decreases from the initial guess until convergence.

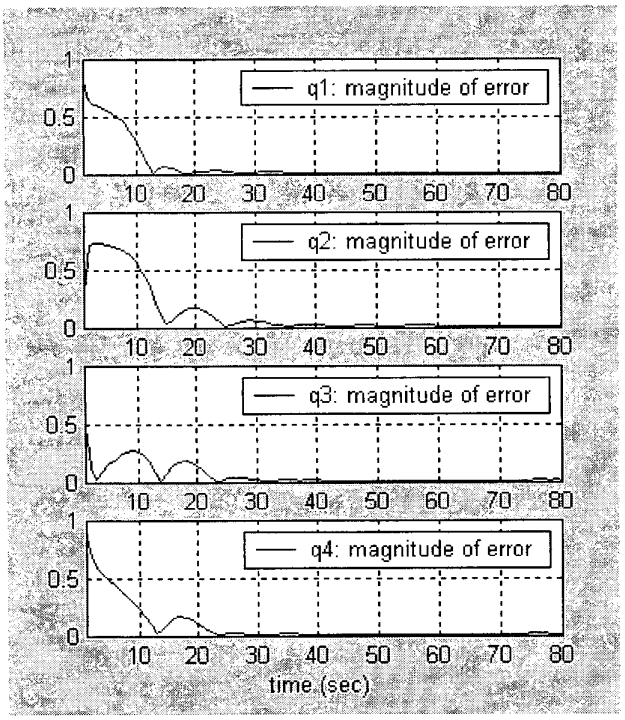

Figure 3 – Error of Quaternions versus Time

When the quaternion estimate converges to the actual value, the filter finds an ever smaller difference between its estimated measurement and the actual measurement.

Preliminary Results of Tuning the Filter

As mentioned, engineers typically have to vary different parameters using simulations in order to obtain the optimal Kalman filter that minimizes attitude knowledge error and maximizes the capability of the satellite hardware. Typically these include the process noise matrix and the observation noise matrix—and the relative observation noise of different measurements (i.e. sun sensors and magnetometer). Figures 4, 5, & 6 demonstrate how the variation of the **S** matrix from (54) affects the behavior of the filter. The **S** matrix is used to compute the process noise matrix **Q**.

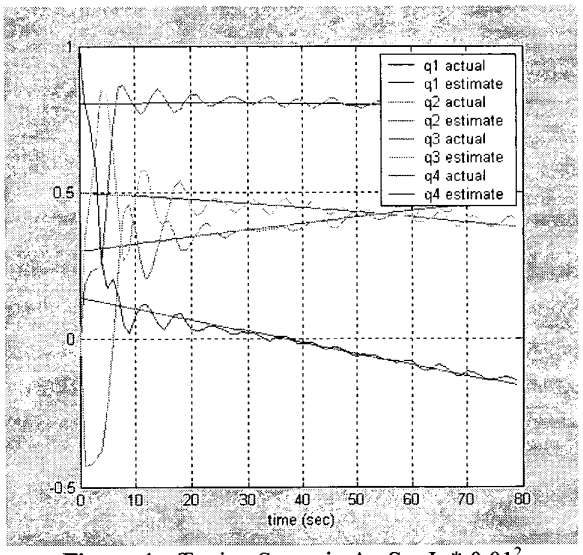

Figure 4 – Tuning Scenario A: $S = I_3 * 0.01^2$

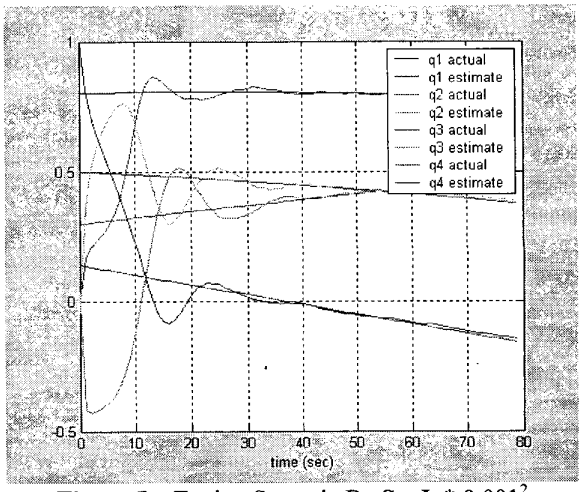

Figure 5 – Tuning Scenario B: $S = I_3 * 0.001^2$

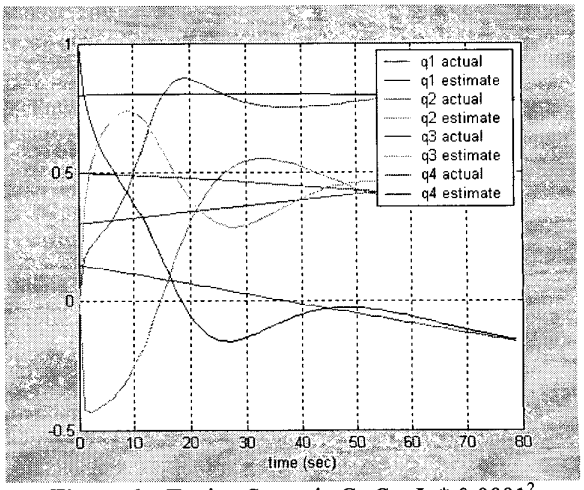

Figure 6 – Tuning Scenario C: $S = I_3 * 0.0001^2$

Over the short time period covered here, it is very easy to see how the changing one parameter affects the filter's performance. Scenario B is certainly an advantage over scenario A. However, with scenario C it appears that you may be worse off than scenario B because it takes so long to converge. With a small S and thus small a Q matrix in the third scenario, the filter may become unresponsive to new measurements that reflect something such as an un-modeled disturbance torque.

Meaning of the Results Thus Far

The results contained above demonstrate that the theory developed in this paper does actually result in a filter that can converge to the correct attitude. With the sun-sensor-magnetometer filter, we have addressed a wide variety of initial conditions and the filter always converges to the correct attitude. Thus, we are confident that the sun sensor-magnetometer filter will provide FalconSAT-3 with robust attitude determination for 2/3 of the orbit.

The magnetometer-momentum bias filter has shown promise as well. However, we are able to come up with certain cases where the filter will not converge or not converge to the correct attitude. Currently, we are doing a barrage of tests to quantify just how robust this filter is and under what conditions will it converge. Perhaps it just needs an accurate initial guess to converge (which can be provided by the sun sensor-magnetometer filter as the satellite travels into eclipse.)

There are other changes to the simulation we will make to better approximate the actual conditions that our filter will operate in. First and foremost, the simulated motion of the satellite so far has been free of disturbance and control torques. Those must be added to achieve the best tuning of the filter. Also, FalconSAT-3's on-board computer will run the attitude determination filter at step size of 5 second. All of the previous simulations have had a step size of 1 second.

These changes and others are the next additions to the simulation, necessary to achieve the optimal attitude determination for FalconSAT-3.

8. Conclusions

After introducing the FalconSAT-3 attitude determination and control system design, this paper produces a relatively straightforward and proven method of satellite attitude determination—relying on a magnetometer, sun sensors, and an angular momentum bias. It also lays the foundation for further research in FalconSAT-3 attitude control.

An important result was our inclusion of the momentum wheel dynamics in our full-spectrum attitude extended Kalman filter. Attitude determination is not possible during eclipse without this. The sun sensor provides a margin of redundancy during the sunlit portions of the orbit. The other alternative would to add an Earth or star sensor—but the additional mass, volume, and budget required would be inconsistent with our requirements for the FalconSAT-3 program.

Significance of Work

As commercial, civil, and defense organizations find the advantages of using low-cost microsatellites to do the job of traditionally larger and more complex satellites, it is essential to provide capabilities such as three-axis attitude determination and control for a fraction of the cost, mass, volume of larger ADCS systems. The capability of such a system has already been demonstrated by Surrey Satellite Technology Ltd.'s SNAP-1 nanosatellite.

The USAF Academy FalsoSAT-3 mission will also demonstrate the capability of undergraduate students in designing complex spacecraft navigation and control systems.

Current Status of FalconSAT-3

On December 10, 2002, FalconSAT-3 held its conceptual design review at the USAF Academy. Representatives from funding sources, the secondary launch provider, and other stakeholders attended. Engineering model integration will commence as scheduled in February 2003.

Future Work

The immediate task is to further refine the Kalman filter. By running simulations and characterizing the data, we can refine our estimate of the observation and process covariance matrix. We will also be able to evaluate the effectiveness of the assumptions made in developing the Kalman filter and troubleshoot any inconsistencies. Later, the primary author and FalconSAT-3 ADCS team will begin to look at specific attitude control strategies and algorithms. Ultimately the torch will have to passed to the next class of cadets in the late spring.

References

[1] W.H. Steyn. *A Multi-Mode Attitude Determination and Control System for Small Satellites.* PhD Dissertation. University of Stellenbosch, South Africa, 1995.

[2] J.R. Wertz, *Spacecraft Attitude Determination and Control,* Astrophysics and Space Science Library, Vol. 73, D. Reidel Publishing Company, 1978 (reprint 1986).

[3] W.H. Steyn and Y. Hashida, "In-Orbit Performance of the 3-axis Stabilised SNAP-1 Nanosatellite," *Proceedings of the 15th annual AIAA/USU Conference on Small Satellites,* August 2001.

[4] W.H. Steyn, Y. Hashida, and V. Lappas "An Attitude Control System and Commissioning Results of the SNAP-1 Nanosatellite," *Proceedings of the 14th annual AIAA/USU*

Conference on Small Satellites, August 2000.

[5] David A. Vallado, *Fundamentals of Astrodynamics and Applications 2nd Ed,* Space Technology Library, Microcosm Press, 2001.

[6] The Analytic Sciences Corporation, *A short course on Kalman Filter Application and Theory,* course notes, 1971.

[7] Y. Hashida "ADCS Design for Future UoSAT Standard Platform," Surrey Space Centre, Guildford, UK, July 1997.

Cadet First Class Craig Van Beusekom *is a senior cadet at the USAF Academy. He is an Astronautical Engineering major and has been working with the USAFA Small Satellite Program since April 2002. For five weeks last summer, he worked with SSTL Staff at Surrey Space Center in Guildford, UK.*

Ron Lisowski *is a professor in the USAF Academy Department of Astronautics. A 1969 graduate of the USAF Academy, he has a PhD in Aeronautics and Astronautics from Illinois. He is currently serving on the ASEE Aerospace Division's Board of Directors and is an Associate Fellow in AIAA, where he has served on the Space Systems, the Guidance, Navigation and Control, and currently, the Astrodynamics Technical Committees.*

Capt Joe Fulton *is an assistant professor in the USAF Academy Department of Astronautics. He has just been selected by the Air Force to earn his PhD in Astronautical Engineering.*

Maj Christophe Morand *is an officer in the French Air Force, currently assigned to the USAF Academy and is a faculty member of the FalconSAT team.*

APPENDIX A: PARTIAL DERIVATIVES OF DCM

It is necessary to take the partial derivative of equation (8) with respect to each quaternion.

$$\frac{\partial \mathbf{C}}{\partial q_1} = 2\begin{bmatrix} q_1 & q_2 & q_3 \\ q_2 & -q_1 & q_4 \\ q_3 & -q_4 & -q_1 \end{bmatrix} \tag{55a}$$

$$\frac{\partial \mathbf{C}}{\partial q_2} = 2\begin{bmatrix} -q_2 & q_1 & -q_4 \\ q_1 & q_2 & q_3 \\ q_4 & q_3 & q_2 \end{bmatrix} \tag{55b}$$

$$\frac{\partial \mathbf{C}}{\partial q_3} = 2\begin{bmatrix} -q_3 & q_4 & q_1 \\ -q_4 & -q_3 & q_3 \\ q_1 & q_1 & q_3 \end{bmatrix} \tag{55c}$$

$$\frac{\partial \mathbf{C}}{\partial q_4} = 2\begin{bmatrix} q_4 & q_3 & -q_2 \\ -q_3 & q_4 & q_1 \\ q_2 & -q_1 & q_4 \end{bmatrix} \tag{55d}$$

APPENDIX B: FS3 PROCESS NOISE DERIVATION

Let $\mathbf{w}(\tau)$ and $\mathbf{w}(\alpha)$ be two three-dimensional column vector representing the body rates from (29) at two different instants α and τ between t_k and t_{k+1}. The process noise covariance matrix **Q** for step k is defined as follows [6]:

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1},\tau)\mathbf{G}(\tau)E\left[\mathbf{w}(\tau)\mathbf{w}(\alpha)^T\right]\mathbf{G}(\alpha)^T\,\mathbf{\Phi}(t_{k+1},\alpha)^T\,d\alpha d\tau \tag{56}$$

where

$$\mathbf{G} = \begin{bmatrix}\mathbf{0}_{3x4} & \mathbf{1}_3\end{bmatrix}^T$$

$$\mathbf{\Phi}(t_{k+1},\tau) = \left(\mathbf{1}_{7x7} + \begin{bmatrix}\mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22}\end{bmatrix}\tau\right) \text{ from (23)}$$

Since $\Delta t = t_{k+1} - t_k$, equation (56) can also be represented as [7]:

$$\mathbf{Q}_k = \int_0^{\Delta t}\int_0^{\Delta t} \mathbf{\Phi}(t_{k+1},\tau)\mathbf{G}(\tau)E\left[\mathbf{w}(\tau)\mathbf{w}(\alpha)^T\right]\mathbf{G}(\alpha)^T\,\mathbf{\Phi}(t_{k+1},\alpha)^T\,d\alpha d\tau \tag{57}$$

and we have

$$E\left[\mathbf{w}(\tau)\mathbf{w}(\alpha)^T\right] = \begin{bmatrix}\sigma_x^2\delta(\tau-\alpha) & 0 & 0 \\ 0 & \sigma_y^2\delta(\tau-\alpha) & 0 \\ 0 & 0 & \sigma_z^2\delta(\tau-\alpha)\end{bmatrix} \tag{58}$$

where δ is the dirac delta function and $\delta(0) = 1$. The dirac delta function properties can be used to perform the integration over α, leaving us with the following:

$$\mathbf{Q}_k = \int_0^{\Delta t} \mathbf{\Phi}(t_{k+1},\tau)\mathbf{G}(\tau)E\left[\mathbf{w}(\tau)\mathbf{w}(\tau)^T\right]\mathbf{G}(\tau)^T\,\mathbf{\Phi}(t_{k+1},\tau)^T\,d\tau \tag{59}$$

$$E\left[\mathbf{w}(\tau)\mathbf{w}(\tau)^T\right] = \begin{bmatrix}\sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2\end{bmatrix} = \mathbf{S} \tag{60}$$

By using the definitions for **G** and **Φ**, we arrive at the following integral to compute $\mathbf{Q}_k$:

$$\mathbf{Q}_k = \int_0^{\Delta t}\left(\mathbf{1}_{7x7} + \begin{bmatrix}\mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22}\end{bmatrix}\tau\right)\begin{pmatrix}\mathbf{0}_{4x4} & \mathbf{0}_{4x3} \\ \mathbf{0}_{3x4} & \mathbf{S}\end{pmatrix}\left(\mathbf{1}_{7x7} + \begin{bmatrix}\mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{21} & \mathbf{F}_{22}\end{bmatrix}^T\tau\right)d\tau \tag{61}$$

By multiplying through and collecting like terms, this

reduces to:

$$\mathbf{Q}_k = \int_0^{\Delta t} \left(\mathbf{Q}_1 + \mathbf{Q}_2 \tau + \mathbf{Q}_3 \tau^2 \right) d\tau \tag{62}$$

where Q_1, Q_2, Q_3 are the matrices previously defined in equations (50), (51), and (52). With a simple integration, this becomes what we saw before:

$$\mathbf{Q}_k = \mathbf{Q}_1 \Delta t + \mathbf{Q}_2 \frac{(\Delta t)^2}{2} + \mathbf{Q}_3 \frac{(\Delta t)^3}{3} \tag{63}$$