

Improving Angles-Only Navigation Performance by Selecting Sufficiently Accurate Accelerometers

Jason Schmidt
Student-Utah State University
8458 Sun Valley Dr, Sandy UT; 435-512-5740
j.schmi@yahoo.com

Dr. David Geller
Professor-Utah State University
1400 Old Main Hill Logan, Utah; 435-797-2952
dgeller@engineering.usu.edu

Frank Chavez
Research Engineer
Air Force Research Lab, Space Vehicles Directorate

In order to make future satellites both smaller and smarter, more navigation information must be extracted from simpler, smaller sensors. One of the simplest sensors is an optical or infrared camera. With a camera, a satellite can track a second satellite located within its field-of-view. This simple measurement is the foundation of angles-only navigation. By its very nature, angles only navigation cannot determine the relative range to an object. Even as the dynamics associated with orbital rendezvous and proximity operations unfold, the relative range will generally remain unobservable. In this paper we confirm that an angles only navigation system can observe range if small maneuvers can be executed, and we show that the level of accelerometer accuracy determines how well the range can be observed.

I. Nomenclature

Accents, subscripts and superscripts

$$\begin{aligned}\text{True Value} &= \bar{o} \\ \text{Estimated Value} &= \hat{o} \\ \text{Measured Value} &= \tilde{o} \\ \text{Previous Time step Estimate} &= o^- \\ \text{Updated Estimate} &= o^+ \\ \text{Chaser camera frame} &= o^{cam} \\ \text{Chaser body frame} &= o^b \\ \text{Chaser values} &= o_c \\ \text{Target values} &= o_t \\ \text{Cross Product Matrix Form} &= [\bar{o} \times]\end{aligned}$$

Errors and noise

$$\begin{aligned}\text{Measurement Error } (v) &= \tilde{o} - \bar{o} \\ \text{Residual Error } (e) &= \tilde{o} - \hat{o} \\ \text{Misalignment } (\varepsilon) &= \text{vector of small angles} \\ \text{Bias } (\beta) &= \text{vector} \\ \text{Process Noise } (w) &= \text{continuous white noise} \\ \text{Measurement Noise } (\eta) &= \text{continuous white noise}\end{aligned}$$

II. Introduction

The purpose of this study is to show how accelerometer accuracy affects the performance of a Kalman filter processing angles-only measurements during orbital rendezvous. While angle measurements can be obtained from other types of sensors (Lidar or Radar for example), the camera has the additional advantage of being entirely passive.

By its very nature, an angle measurement contains no range information. If angle measurements are made to multiple objects with known locations that lie in different directions this limitation can be overcome. When the

angle measurements are made to a single object, such as during orbital rendezvous, the lack of range information can become a serious limitation.

Various strategies have been developed to overcome this limitation. Two of the most successful techniques are taking “apparent diameter” measurements, and performing translational maneuvers.¹ Both methods have their disadvantages. Apparent diameter measurements are range limited based on the resolution of the camera and require prior knowledge of the satellite being observed. Translational maneuvers consume fuel and lower the life of the satellite. In this paper the translational burn method will be considered.

Lingering Questions on Angles-Only Navigation

Angles-only navigation (AON) has been researched for use on spacecraft since the days of Gemini,² and proved useful for orbital rendezvous during the Apollo years. It was successfully implemented autonomously on Deep Space I.³ AON is useful because it does not require heavy and expensive sensors like radar. Instead, an expensive radar is replaced by a video camera and occasional maneuvers.

However, there are still serious questions about AON use during orbital rendezvous. For distances greater than 100km, Gauss’ method may be an option to determine range.⁴ However, at closer distances, Gauss’ method breaks down. This is because whole families of trajectories will exhibit nearly identical line-of-sight (LOS) measurement histories as seen in figure 1. They only differ in their range component. If this motion is linearized using CW equations, then the LOS measurement histories will truly be identical. Without a unique solution for the given measurements, the relative state of the chaser will remain unobservable.

The behavior of Kalman filters when processing LOS measurements has been analyzed well studied.^{1,5} It has been shown that the component of the covariance matrix parallel to the LOS vector will grow almost without bound unless maneuvers that change future measurements are performed. Within a short time, state estimation can return bogus results (See figure 8).

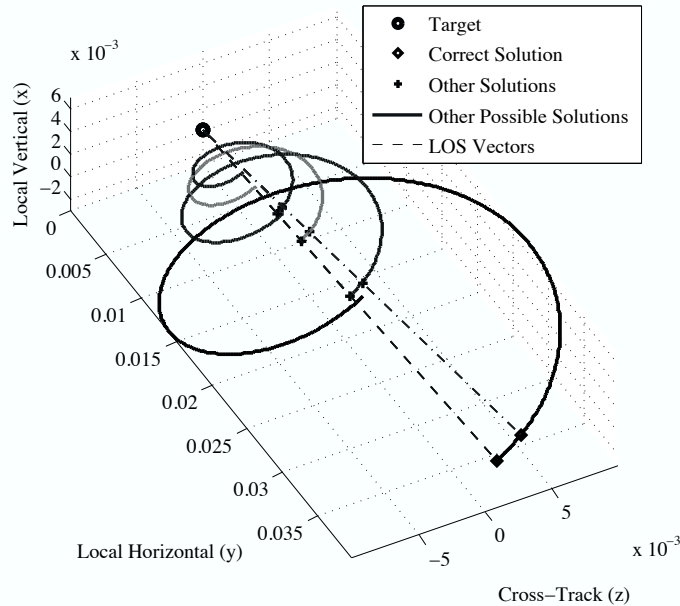


Figure 1. A family of relative-motion trajectories exhibiting identical line-of-sight measurement histories.

Observability Burns

In order to estimate range, satellites using AON for orbital rendezvous must make maneuvers in order to observe relative range. As Figure 2 shows, only one range will satisfy the known change in position δr resulting from a known acceleration delivered by the spacecraft's thrusters as long as δr is not parallel to $r_{nominal}$. The change in the LOS vector for a given δr is known as the observability angle (θ)⁶. These LOS measurements are processed in an EKF for state estimation.

The difference between the actual θ and the calculated $\hat{\theta}$ is a function of accelerometer and image centroiding errors. As shown in Figure 3, if the calculated observability angle is too small, then the Kalman filter may tend to ignore it, because it falls within noise of the system. Thus, accelerometer performance may have a tremendous impact on the ability of the navigation filter to estimate relative range.

Kalman Filter

The Kalman filter was first developed by R.E. Kalman⁷ and has been improved and expanded over the years to

process all sorts of measurements. A Kalman filter essentially propagates an estimated state (\bar{x}) and the covariance (P_x) of that state in real time. When measurements are made, the estimated state is improved and the size of the covariance decreases.

When angle measurements are processed in a Kalman filter, no information can be gleaned parallel to the LOS vector. Applying the filter in a rendezvous situation with no observability burns will result in a very long, skinny covariance ellipse. As a result, extremely large and very small values will be contained in the same covariance matrix. Within a short time numerical issues can crop up during the measurement update portion of the Kalman filter algorithm.

To help overcome this issue, the Kalman filter will be reformulated to update the square root of the covariance matrix. This formulation is known as a square-root Kalman filter, and has the significant advantage of making large numbers smaller and small numbers larger—making the filter much more robust when propagating a long, skinny covariance ellipse.⁸ The specific formulation the square root extended Kalman filter (SREKF) is adapted from Tapley⁹ (See figure 4).

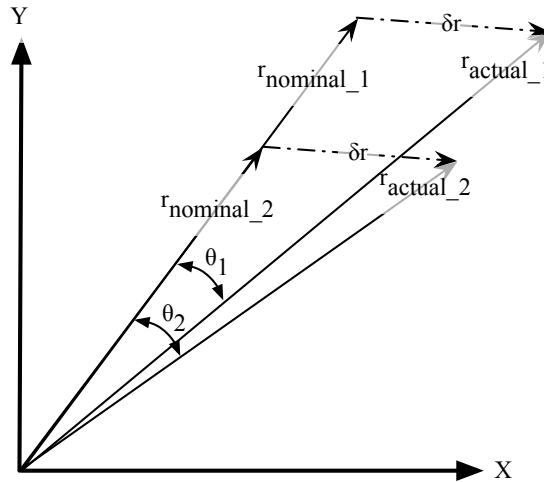


Figure 2. Change in observability angle as a result of δr produced by a known acceleration

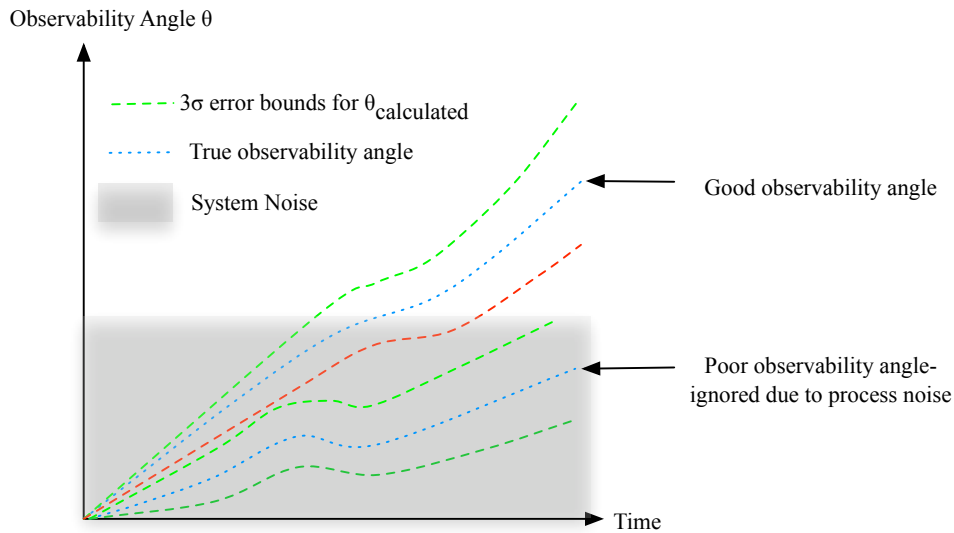


Figure 3. Observability angle as a function of time

III. Methods

A satellite rendezvous simulation created in Matlab/Simulink is used. It includes noise and bias on sensors and actuators. Vehicle dynamics, environmental models, sensor models, and actuator models already exists as Simulink blocks. Only the navigation algorithms was developed for this research.

Essentials include a satellite with three sensors—accelerometers, line-of-sight camera, and star-camera. Actuators include thrusters and momentum wheels. This satellite is modeled with familiar Euler equation rigid-body dynamics and J_2 gravity with noise. The flight software guidance is waypoint driven for trans-

lation and target tracking for attitude. Attitude control is obtained with a phase-plane controller for thrusters and a PID controller for momentum Wheels. Translational control leverages a PD controller for station keeping, and Clohessy-Wiltshire (CW) equations targeting for transfers. Navigation is detailed in Section III.A.

III.A. Kalman Filter Development

A Square-Root EKF is used for navigation. The operations of the square-root filter is distinguished from the standard filter as seen in figure 4. The filter has been formulated to process line-of-sight measurements. The filter assumes J_2 gravitational dynamics and uses onboard

accelerometers and star-tracker measurements to directly propagate the states. The filter runs onboard a chaser spacecraft and will estimate the inertial position and ve-

locity of both the chaser and a target vehicle that it is tracking. The filter will also estimate bias and misalignment of the accelerometer and cameras.

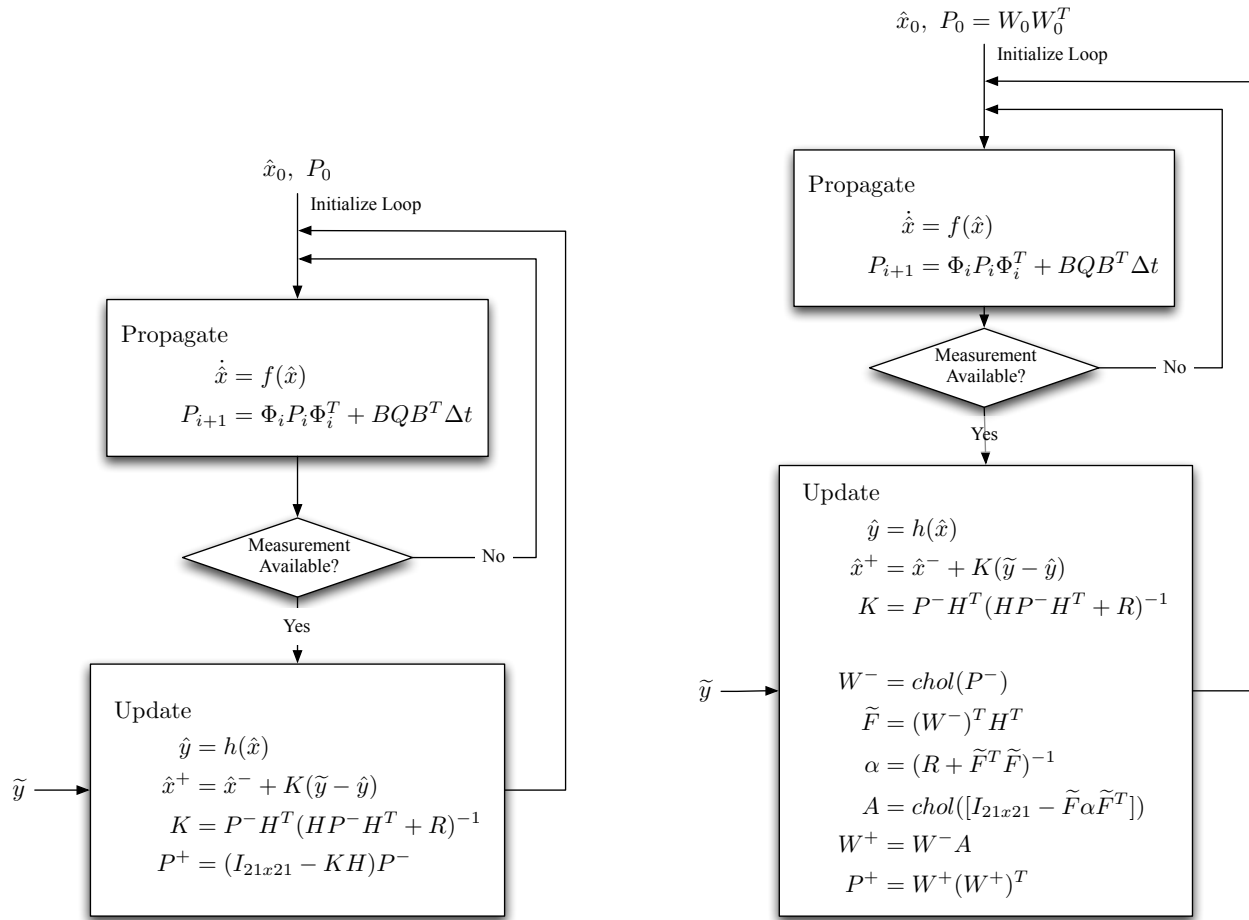


Figure 4. Square-Root Extended Kalman Filter (SREKF) and Extended Kalman Filter (EKF) flow charts (See section III.B)

III.A.1. Filter Design Model

This is the navigation system designer's "working model." This model accounts for the position and velocity of the chaser and target vehicles, misalignment and measurement noise on the accelerometers and the optical camera, and bias on the accelerometers. It uses process noise in the vehicle acceleration channel to account for unmodeled effects like drag and solar pressure.

The only measurements that will be processed in the

Kalman filter directly will be LOS and accelerometer measurements. Accelerometers will be used to propagate position and velocity states directly. As a result, the accelerometer measurement noise (η_c) will be treated like a process noise.

The filter design model can be written as:

$$\begin{cases} \dot{\bar{x}} = f(\bar{x}) + B\bar{w} \\ \bar{y} = h(\bar{x}) + \eta_{cam} \end{cases} \quad (1)$$

$$\dot{\bar{x}} = \begin{bmatrix} \dot{\bar{r}}_c \\ \dot{\bar{v}}_c \\ \dot{\bar{r}}_t \\ \dot{\bar{v}}_t \\ \dot{\bar{\beta}} \\ \dot{\bar{\epsilon}}_{acc} \\ \dot{\bar{\epsilon}}_{cam} \end{bmatrix}_{21 \times 1} \quad f(\bar{x}) = \begin{bmatrix} \bar{v}_c \\ g(\bar{r}_c) + T_{b \rightarrow I}[I_{3 \times 3} + [\bar{\epsilon}_{acc} \times]][\tilde{a} - \tilde{\beta}] \\ \bar{v}_t \\ g(\bar{r}_t) \\ -\tilde{\beta}/\tau_{acc\beta} \\ -\bar{\epsilon}_{acc}/\tau_{acc} \\ -\bar{\epsilon}_{cam}/\tau_{cam} \end{bmatrix}_{21 \times 1} \quad (2)$$

$$g(\bar{r}) = -\mu \frac{\bar{r}}{|\bar{r}|^3} - \mu \frac{J_2 R_e^2}{2|\bar{r}|^5} \{6(\bar{r} \cdot \bar{n})\bar{n} + 3\bar{r} - 15(\bar{i}_r \cdot \bar{n})^2 \bar{r}\}$$

$$\bar{w} = \begin{bmatrix} \bar{w}_c \\ \bar{\eta}_c \\ \bar{w}_t \\ \bar{w}_{acc\beta} \\ \bar{w}_{acc} \\ \bar{w}_{cam} \end{bmatrix}_{18 \times 1} \quad B = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ I_{3 \times 3} & -T_{b \rightarrow I}[I_{3 \times 3} + [\bar{\epsilon}_{acc} \times]] & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}_{21 \times 18}$$

The measurement equation is detailed below. See figure 5 for an explanation of the azimuth and elevation angle values.

$$h(\bar{x}) = \begin{bmatrix} \tan(az) \\ \tan(el) \end{bmatrix} = \begin{bmatrix} R_z^{cam}/R_x^{cam} \\ R_y^{cam}/R_x^{cam} \end{bmatrix} \quad (3)$$

$$\bar{r}_{rel}^{cam} = [I_{3 \times 3} - [\bar{\epsilon}_{cam} \times]](T_{I \rightarrow b}(\bar{r}_t - \bar{r}_c) - r_{cam}^b) = \begin{bmatrix} R_x^{cam} & R_y^{cam} & R_z^{cam} \end{bmatrix}^T \quad (4)$$

$$T_{I \rightarrow b}(\bar{q}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}_{3 \times 3} \quad (5)$$

where

$$\bar{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{4 \times 1} = \begin{bmatrix} \cos(\theta/2) \\ \hat{u} \sin(\theta/2) \end{bmatrix} \quad (6)$$

\hat{u} = unit vector defining axis of rotation

θ = angle of rotation in radians

INERTIAL-TO-BODY TRANSFORMATION MATRIX
The transformation matrix comes directly from the star camera, which returns a quaternion measurement (\bar{q}):

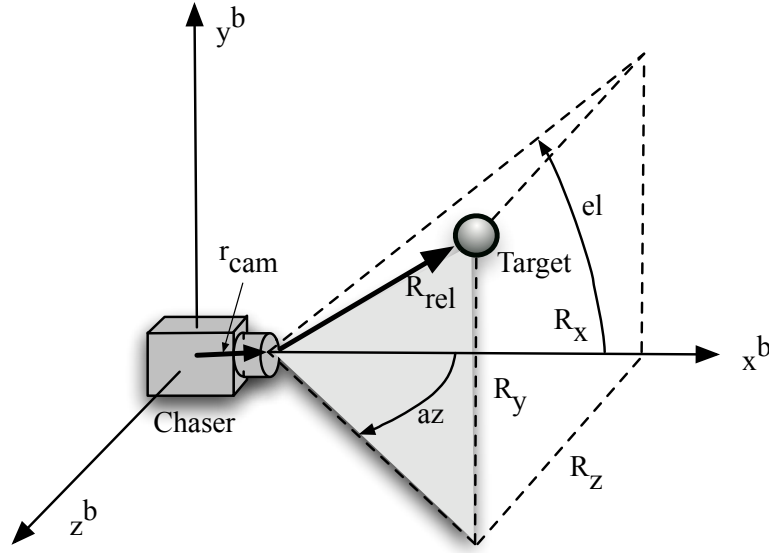


Figure 5. Azimuth and Elevation Measurements in Chaser Body Frame

STATE AND STATE COVARIANCE PROPAGATION EQUATIONS These are the equations the filter will use to propagate the estimated states and covariance

$$\begin{cases} \dot{\bar{x}} = f(\bar{x}) \\ P_{i+1} = \Phi_i P_i \Phi_i^T + B Q B^T \delta t \end{cases} \quad (7)$$

The strength of the process noise Q is related to the process noise \bar{w} by the relationship:

$$Q \delta(t' - t) = E[\bar{w}(t') \bar{w}(t)^T] \quad (8)$$

Where the E operator is the expected value, and $\delta(t' - t)$ is the Dirac delta function.

Q may be represented as a matrix as shown below.

$$Q = \begin{bmatrix} Q_{w_c} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & Q_{\eta_c} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & Q_{w_t} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & Q_{w_{acc\beta}} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & Q_{w_{acc}} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & Q_{w_{cam}} \end{bmatrix}_{18 \times 18} \quad (9)$$

The values in these submatrices usually come from the hardware specification, while Q_{ω_c} and Q_{ω_t} are functions of unmodeled accelerations of the chaser and target which would include J_{3+} gravity effects, drag, solar radiation forces, etc.

The linearization of the dynamics equation is needed to solve for the transition matrix Φ . The system dynamics equation ($f(\bar{x})$) is linearized as follows:

$$F = \left. \frac{\partial f(\bar{x})}{\partial \bar{x}} \right|_{\bar{x}} = \begin{bmatrix} \frac{\partial f(\bar{x})}{\partial \bar{r}_c} & \frac{\partial f(\bar{x})}{\partial \bar{v}_c} & \frac{\partial f(\bar{x})}{\partial \bar{r}_t} & \frac{\partial f(\bar{x})}{\partial \bar{v}_t} & \frac{\partial f(\bar{x})}{\partial \bar{\beta}} & \frac{\partial f(\bar{x})}{\partial \bar{\epsilon}_{acc}} & \frac{\partial f(\bar{x})}{\partial \bar{\epsilon}_{cam}} \end{bmatrix}_{21 \times 21} \quad (10)$$

Each element of the above matrix is a column of partials as seen below.

$$F = \begin{bmatrix} \frac{\partial \dot{\bar{r}}_c}{\partial \bar{r}_c} & \frac{\partial \dot{\bar{r}}_c}{\partial \bar{v}_c} & \frac{\partial \dot{\bar{r}}_c}{\partial \bar{r}_t} & \frac{\partial \dot{\bar{r}}_c}{\partial \bar{v}_t} & \frac{\partial \dot{\bar{r}}_c}{\partial \bar{\beta}} & \frac{\partial \dot{\bar{r}}_c}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \dot{\bar{r}}_c}{\partial \bar{\epsilon}_{cam}} \\ \frac{\partial \dot{\bar{v}}_c}{\partial \bar{r}_c} & \frac{\partial \dot{\bar{v}}_c}{\partial \bar{v}_c} & \frac{\partial \dot{\bar{v}}_c}{\partial \bar{r}_t} & \frac{\partial \dot{\bar{v}}_c}{\partial \bar{v}_t} & \frac{\partial \dot{\bar{v}}_c}{\partial \bar{\beta}} & \frac{\partial \dot{\bar{v}}_c}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \dot{\bar{v}}_c}{\partial \bar{\epsilon}_{cam}} \\ \frac{\partial \dot{\bar{r}}_t}{\partial \bar{r}_c} & \frac{\partial \dot{\bar{r}}_t}{\partial \bar{v}_c} & \frac{\partial \dot{\bar{r}}_t}{\partial \bar{r}_t} & \frac{\partial \dot{\bar{r}}_t}{\partial \bar{v}_t} & \frac{\partial \dot{\bar{r}}_t}{\partial \bar{\beta}} & \frac{\partial \dot{\bar{r}}_t}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \dot{\bar{r}}_t}{\partial \bar{\epsilon}_{cam}} \\ \frac{\partial \dot{\bar{v}}_t}{\partial \bar{r}_c} & \frac{\partial \dot{\bar{v}}_t}{\partial \bar{v}_c} & \frac{\partial \dot{\bar{v}}_t}{\partial \bar{r}_t} & \frac{\partial \dot{\bar{v}}_t}{\partial \bar{v}_t} & \frac{\partial \dot{\bar{v}}_t}{\partial \bar{\beta}} & \frac{\partial \dot{\bar{v}}_t}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \dot{\bar{v}}_t}{\partial \bar{\epsilon}_{cam}} \\ \frac{\partial \dot{\bar{\beta}}}{\partial \bar{r}_c} & \frac{\partial \dot{\bar{\beta}}}{\partial \bar{v}_c} & \frac{\partial \dot{\bar{\beta}}}{\partial \bar{r}_t} & \frac{\partial \dot{\bar{\beta}}}{\partial \bar{v}_t} & \frac{\partial \dot{\bar{\beta}}}{\partial \bar{\beta}} & \frac{\partial \dot{\bar{\beta}}}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \dot{\bar{\beta}}}{\partial \bar{\epsilon}_{cam}} \\ \frac{\partial \dot{\bar{\epsilon}}_{acc}}{\partial \bar{r}_c} & \frac{\partial \dot{\bar{\epsilon}}_{acc}}{\partial \bar{v}_c} & \frac{\partial \dot{\bar{\epsilon}}_{acc}}{\partial \bar{r}_t} & \frac{\partial \dot{\bar{\epsilon}}_{acc}}{\partial \bar{v}_t} & \frac{\partial \dot{\bar{\epsilon}}_{acc}}{\partial \bar{\beta}} & \frac{\partial \dot{\bar{\epsilon}}_{acc}}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \dot{\bar{\epsilon}}_{acc}}{\partial \bar{\epsilon}_{cam}} \\ \frac{\partial \dot{\bar{\epsilon}}_{cam}}{\partial \bar{r}_c} & \frac{\partial \dot{\bar{\epsilon}}_{cam}}{\partial \bar{v}_c} & \frac{\partial \dot{\bar{\epsilon}}_{cam}}{\partial \bar{r}_t} & \frac{\partial \dot{\bar{\epsilon}}_{cam}}{\partial \bar{v}_t} & \frac{\partial \dot{\bar{\epsilon}}_{cam}}{\partial \bar{\beta}} & \frac{\partial \dot{\bar{\epsilon}}_{cam}}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \dot{\bar{\epsilon}}_{cam}}{\partial \bar{\epsilon}_{cam}} \end{bmatrix}_{21 \times 21} \quad (11)$$

Many of these partial derivatives are zero, resulting in the following.

$$F = \begin{bmatrix} 0_{3 \times 3} & \partial \dot{\bar{r}}_c / \partial \bar{v}_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ \partial \dot{\bar{v}}_c / \partial \bar{r}_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \partial \dot{\bar{v}}_c / \partial \bar{\beta} & \partial \dot{\bar{v}}_c / \partial \bar{\epsilon}_{acc} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \partial \dot{\bar{r}}_t / \partial \bar{v}_t & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & \partial \dot{\bar{v}}_t / \partial \bar{r}_t & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \partial \dot{\bar{\beta}} / \partial \bar{\beta} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \partial \dot{\bar{\epsilon}}_{acc} / \partial \bar{\epsilon}_{acc} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \partial \dot{\bar{\epsilon}}_{cam} / \partial \bar{\epsilon}_{cam} \end{bmatrix}_{21 \times 21} \quad (12)$$

The individual partial derivatives are evaluated as follows:

$$\begin{aligned} \partial \dot{\bar{r}}_c / \partial \bar{v}_c &= I_{3 \times 3} \\ \partial \dot{\bar{v}}_c / \partial \bar{r}_c &= A_1(\bar{r}_c) + A_2(\bar{r}_c) + A_3(\bar{r}_c) \\ \partial \dot{\bar{v}}_c / \partial \bar{\beta} &= -T_{b \rightarrow I} [I_{3 \times 3} + [\bar{\epsilon}_{acc} \times]] \\ \partial \dot{\bar{v}}_c / \partial \bar{\epsilon}_{acc} &= T_{b \rightarrow I} [[\bar{\beta} \times] - [\tilde{a} \times]] \\ \partial \dot{\bar{r}}_t / \partial \bar{v}_t &= I_{3 \times 3} \\ \partial \dot{\bar{v}}_t / \partial \bar{r}_t &= A_1(\bar{r}_t) + A_2(\bar{r}_t) + A_3(\bar{r}_t) \\ \partial \dot{\bar{\beta}} / \partial \bar{\beta} &= \frac{-1}{\tau_{acc\beta}} I_{3 \times 3} \\ \partial \dot{\bar{\epsilon}}_{acc} / \partial \bar{\epsilon}_{acc} &= \frac{-1}{\tau_{acc}} I_{3 \times 3} \\ \partial \dot{\bar{\epsilon}}_{cam} / \partial \bar{\epsilon}_{cam} &= \frac{-1}{\tau_{cam}} I_{3 \times 3} \end{aligned} \quad (13)$$

where

$$\begin{aligned} A_1(\bar{r}) &= \frac{-\mu}{|\bar{r}|^3} (I_{3 \times 3} - 3\bar{i}_r \bar{i}_r^T) \\ A_2(\bar{r}) &= \frac{-3\mu J_2 R_e^2 \bar{n}}{|\bar{r}|^5} (\bar{n}^T I_{3 \times 3} - 5(\bar{i}_r^T \bar{n}) \bar{i}_r^T) - \frac{3\mu J_2 R_e^2}{2|\bar{r}|^5} (I_{3 \times 3} - 5\bar{i}_r \bar{i}_r^T) \\ A_3(\bar{r}) &= \frac{7.5\mu J_2 R_e^2}{|\bar{r}|^5} \times \\ &\quad \left[(\bar{i}_r^T \bar{n})^2 I_{3 \times 3} - 5(\bar{i}_r^T \bar{n})^2 \bar{i}_r \bar{i}_r^T + \frac{\bar{r}_r^T \bar{n} \bar{n}^T I_{3 \times 3}}{|\bar{r}|} (I_{3 \times 3} - \bar{i}_r \bar{i}_r^T) \right] \end{aligned} \quad (14)$$

In the Kalman filter the estimated states (\hat{x}) will be used to evaluate these partial derivatives.

Once F is computed, the transition matrix is given by:

$$\Phi = e^{Fdt} = I + Fdt + \frac{F^2 dt^2}{2!} + \frac{F^3 dt^3}{3!} + \dots \quad (15)$$

III.B. State and State Covariance Update Equation

The update equation is where the standard and square-root Kalman filters differ. A comparison of the two update algorithms may be seen in figure 4.

The standard update equation works well on long wordlength machines, but can cause the covariance matrix to lose positive definite nature during the update step when the covariance matrix is ill-conditioned.⁸ The square-root Kalman filter update equations are considerably more complex than the standard update equations, but they improve numerical accuracy and maintain the positive definite nature of the covariance matrix. The square root algorithm used in this paper was adopted from Tapley and Maybeck.^{8,9} The matrix square-root was calculated by way of Cholesky decomposition, which is referred to in figure 4 as ‘‘chol.’’ Cholesky decomposition results in a lower or upper tri-

angular matrix where each is the complex transpose of the other. For the lower triangular case, the result times its transpose results in the original matrix. Cholesky decomposition is among the most stable of matrix operations.¹⁰

$$R = chol(P) \quad (16)$$

$$RR^T = P \quad (17)$$

III.B.1. Linearization of Measurement Equation

The linearization of the measurement equation results in the measurement sensitivity matrix (H), which is required to solve for the Kalman gain (K).

Note that the H matrix will take the general form:

$$H = \left. \frac{\partial h(\bar{x})}{\partial \bar{x}} \right|_{\hat{x}} = \left. \frac{\partial h(\bar{x})}{\partial \bar{R}_{rel}^b} \frac{\partial \bar{R}_{rel}^b}{\partial \bar{x}} \right|_{\hat{x}} \quad (18)$$

$$\frac{\partial h(\bar{x})}{\partial \bar{R}_{rel}^b} \begin{bmatrix} \frac{\partial \bar{R}_{rel}^b}{\partial \bar{r}_c} & \frac{\partial \bar{R}_{rel}^b}{\partial \bar{v}_c} & \frac{\partial \bar{R}_{rel}^b}{\partial \bar{r}_t} & \frac{\partial \bar{R}_{rel}^b}{\partial \bar{v}_t} & \frac{\partial \bar{R}_{rel}^b}{\partial \beta} & \frac{\partial \bar{R}_{rel}^b}{\partial \bar{\epsilon}_{acc}} & \frac{\partial \bar{R}_{rel}^b}{\partial \bar{\epsilon}_{cam}} \end{bmatrix}$$

The angle measurement is not a function of acceleration bias (β), accelerometer misalignment ($\bar{\epsilon}_{acc}$), or vehicle

velocities. So the measurement sensitivity matrix becomes:

$$H = \frac{\partial h(\bar{x})}{\partial \bar{R}_{rel}^b} \begin{bmatrix} \frac{\partial \bar{R}_{rel}^b}{\partial \bar{r}_c} & 0 & \frac{\partial \bar{R}_{rel}^b}{\partial \bar{r}_t} & 0 & 0 & 0 & \frac{\partial \bar{R}_{rel}^b}{\partial \bar{\epsilon}_{cam}} \end{bmatrix}_{2 \times 21} \quad (19)$$

The individual partials are:

$$\frac{\partial h(\bar{x})}{\partial \bar{R}_{rel}^b} = \begin{bmatrix} \frac{-R_z}{R_x^2} & 0 & \frac{1}{R_x} \\ \frac{-R_y}{R_x^2} & \frac{1}{R_x} & 0 \end{bmatrix}_{2 \times 3}$$

$$\frac{\partial \bar{R}_{rel}^b}{\partial \bar{r}_c} = -[I_{3 \times 3} - [\bar{\epsilon}_{cam} \times]] T_{I \rightarrow b} / \quad (20)$$

$$\frac{\partial \bar{R}_{rel}^b}{\partial \bar{r}_t} = [I_{3 \times 3} - [\bar{\epsilon}_{cam} \times]] T_{I \rightarrow b}$$

$$\frac{\partial \bar{R}_{rel}^b}{\partial \bar{\epsilon}_{cam}} = [(T_{I \rightarrow b}(\bar{r}_t - \bar{r}_c) - r_{cam}^b) \times]$$

IV. Model Implementation

The simulation tool of choice is Matlab/Simulink. See Figure 6 for an example of this simulation. The values of the variables in the equations in section III are found in Table 1 through 5. It is important to note that key noise parameters and initial covariances must be matched as good as possible to the “real” values in the simulation in order for the filter to perform as expected.

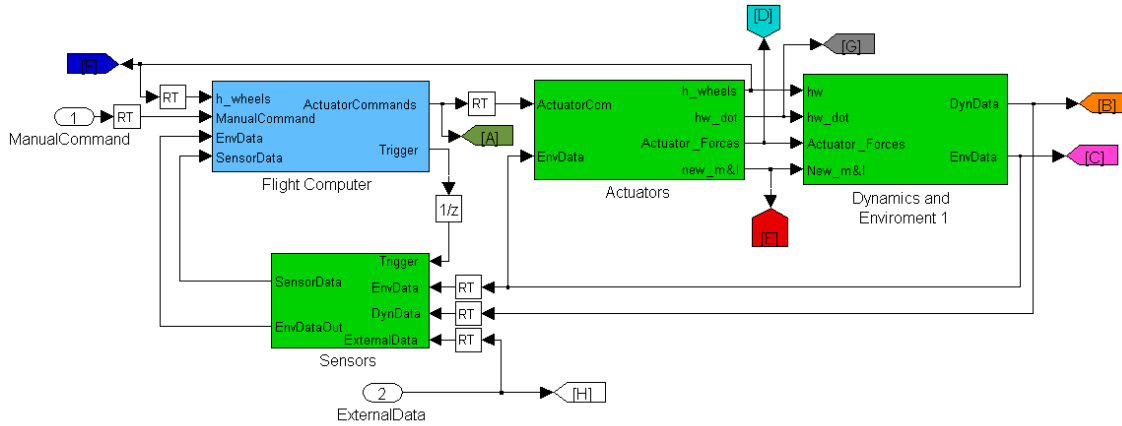


Figure 6. Example of simulation model implemented in Simulink.

IV.A. Noise and Time Constants

The noise strengths and time constants are the nominal values. The accelerometer measurement noise on the

chaser was varied in order to produce the results found in section V.

Table 1. Process Noise, Measurement Noise, and ECRV time contents

Parameter	Description	Value in Simulation	Value in Filter	Units
Q_{ω_c}	Strength of Process Noise due to random accelerations on Chaser	0	10^{-18}	$\frac{km^2}{s^3}$
Q_{η_c}	Strength of Process Noise due to accelerometer measurement noise being processed directly in propagator	10^{-12}	10^{-12}	$\frac{km^2}{s^3}$
Q_{ω_t}	Strength of Process noise due to random accelerations on Target	0	10^{-18}	$\frac{km^2}{s^3}$
$Q_{\omega_{acc\beta}}$	Strength of Process noise on accelerometer bias	10^{-20}	10^{-20}	$\frac{km^2}{s^3}$
$Q_{\omega_{acc}}$	Strength of Process noise on accelerometer misalignment	0	10^{-20}	$\frac{rad^2}{s}$
$Q_{\omega_{cam}}$	Strength of Process noise on camera misalignment	0	5×10^{-13}	$\frac{rad^2}{s}$
R	Variance of Measurement Noise on Angle Measurements	10^{-6}	$\begin{bmatrix} 10^{-6} & 0 \\ 0 & 10^{-6} \end{bmatrix}$	rad^2
$\tau_{acc\beta}$	ECRV time constant for accelerometer bias	10^6	10^6	sec
τ_{acc}	ECRV time constant for accelerometer misalignment	∞	10^6	sec
τ_{cam}	ECRV time constant for camera misalignment	∞	10^6	sec

IV.B. Initial Conditions

Table 2 dictates the value of the initial covariance of that variable.

Note that the components of the initial conditions are all 3x1 vectors. The value of the standard deviation in ta-

Table 2. State Initial Condition

State	Description	Standard Deviation	Value in Filter	Units
\bar{r}_c	Chaser Position in ECI Frame	10^{-2}	$\begin{bmatrix} x & 5114.067 \\ y & -3998.013 \\ z & -335.012 \end{bmatrix}$	km
\bar{v}_c	Chaser Velocity in ECI Frame	10^{-5}	$\begin{bmatrix} x & 4.8199 \\ y & 6.1714 \\ z & 0.07119 \end{bmatrix}$	km/s
\bar{r}_t	Target Position in ECI Frame	10^{-2}	$\begin{bmatrix} x & 5114.3258 \\ y & -3997.682 \\ z & -335.0156 \end{bmatrix}$	km
\bar{v}_t	Target Velocity in ECI Frame	10^{-5}	$\begin{bmatrix} x & 4.81954 \\ y & 6.171718 \\ z & -.071167 \end{bmatrix}$	km/s
$\bar{\beta}$	Bias on accelerometers	10^{-20}	0	km/s ²
$\bar{\epsilon}_{acc}$	Misalignment on accelerometers	10^{-10}	0	rad
$\bar{\epsilon}_{cam}$	Misalignment on camera	10^{-4}	0	rad

Table 3. Covariance Initial Conditions

Component of Covariance	Value	Units
$P_{r_c r_c}$	$(10^{-2})^2 \times I_{3 \times 3}$	km^2
$P_{v_c v_c}$	$(10^{-5})^2 \times I_{3 \times 3}$	km^2/s^2
$P_{r_i r_i}$	$(10^{-2})^2 \times I_{3 \times 3}$	km^2
$P_{v_i v_i}$	$(10^{-5})^2 \times I_{3 \times 3}$	km^2/s^2
$P_{\beta \beta}$	$10^{-20} \times I_{3 \times 3}$	km^2/s^4
$P_{\epsilon_{acc} \epsilon_{acc}}$	$10^{-20} \times I_{3 \times 3}$	rad^2
$P_{\epsilon_{cam} \epsilon_{cam}}$	$10^{-8} \times I_{3 \times 3}$	rad^2

Table 4. Navigation filter constants

Symbol	Definition	Constant and Units
μ	Gravitation Constant	$398600.4415 km^3/s^2$
J_2	Second order gravitation parameter	0.0010826269
R_e	Radius of the Earth	6378.1367 km
\bar{n}	Constant for calculations	[0 0 1]
dt	Propagation step size	0.25 sec

Table 5. Navigation filter inputs

Symbol	Definition	Units
\tilde{a}	measured acceleration from accelerometers	(km/s^2)
\tilde{q}	measured attitude from star camera	unit quaternion
\tilde{y}	Angle measurements to Target Satellite	unitless 2D vector
$T_{b \rightarrow I}$	transformation matrix (body to inertial)	See section

V. Analysis and Results

V.A. Observability Maneuvers Performance

In order to judge the effect of accelerometer accuracy on Kalman filter performance, the simulation was run for the flight path seen in figure 7. In brief, the satellite performs no maneuvers for the first 500 seconds. After this it burns positively and negatively in the crosstrack direction successively every 75 seconds. The importance of these maneuvers is emphasized by examining figures 8 and 9, which shows navigation performance with and without accelerometers.

The effects of using poor, average, and good accelerometers on navigation performance is shown in figures 10 through 12 and may be contrasted with the “perfect accelerometer” results found in figure 9. These results were generated with a square-root Kalman filter with all values defined as seen in section IV, except for the accelerometer accuracies, which are defined in table 6.

A couple of other methods were investigated as well. The first is to improve state estimation by increasing the

magnitude of the thrust maneuvers so that they are more easily distinguished from accelerometer noise and error. The second method is to ignore the accelerometer measurements when a burn is not being performed. These two cases are shown in figures 13 and 14 for average accelerometers.

The final plot, figure 15, shows results when LN-200 type accelerometers are used. These commercial, off-the-shelf accelerometers have a acceleration noise variance of $3 \times 10^{-13} km^2/s^4$, slightly better than the average case, and a bias with a standard deviation of $3 \times 10^{-6} km/s^2$, much worse than the $10^{-20} km/s^2$ used for all the other runs.¹¹

	Variance of Acceleration Measurement
Perfect	$0 km^2/s^4$
Good	$10^{-14} km^2/s^4$
Average	$10^{-12} km^2/s^4$
Poor	$10^{-10} km^2/s^4$

Table 6. Poor, average, and good accelerometers specifications

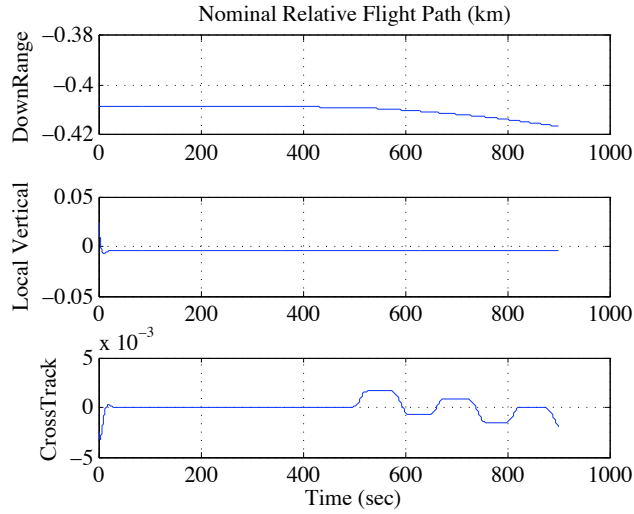


Figure 7. Flight path for accelerometer comparison

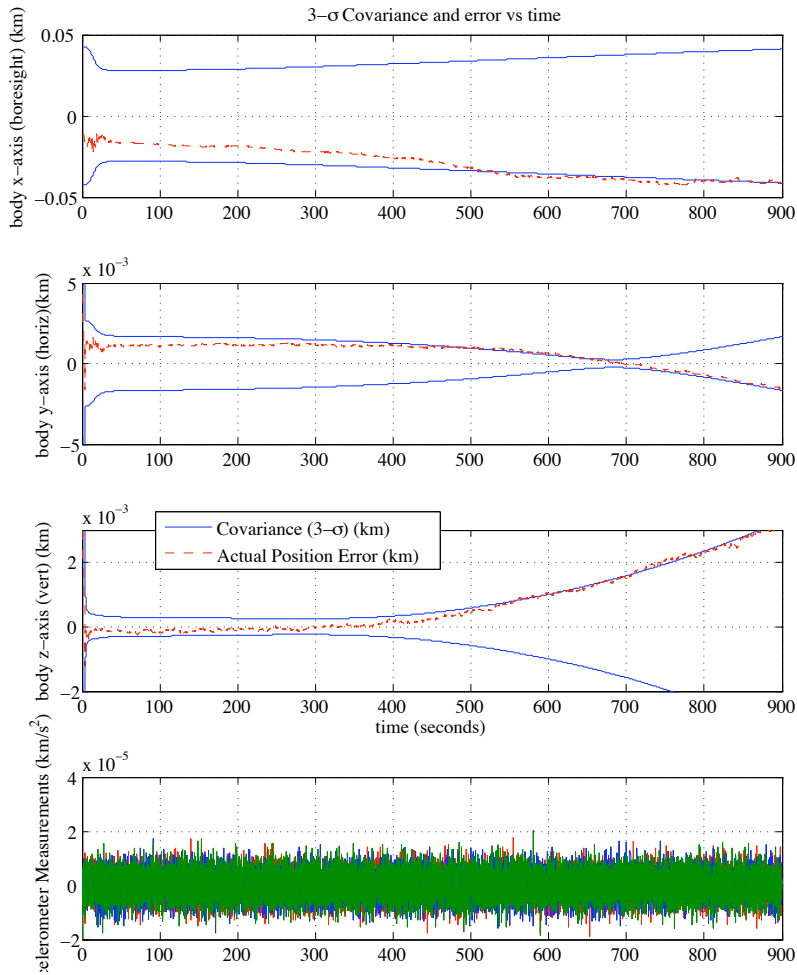


Figure 8. Relative position covariance and errors with no maneuvers

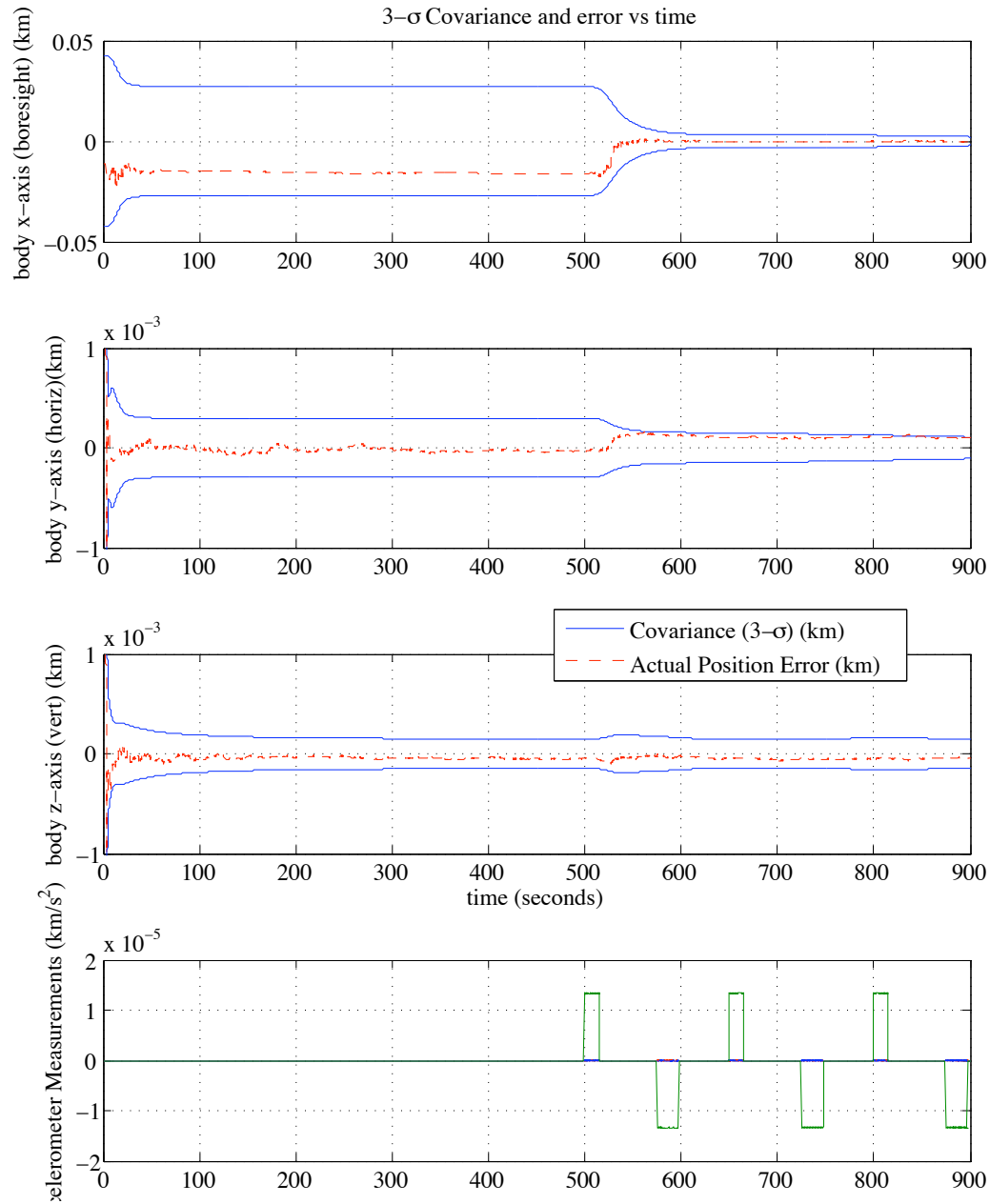


Figure 9. Relative position covariance and errors with perfect accelerometers

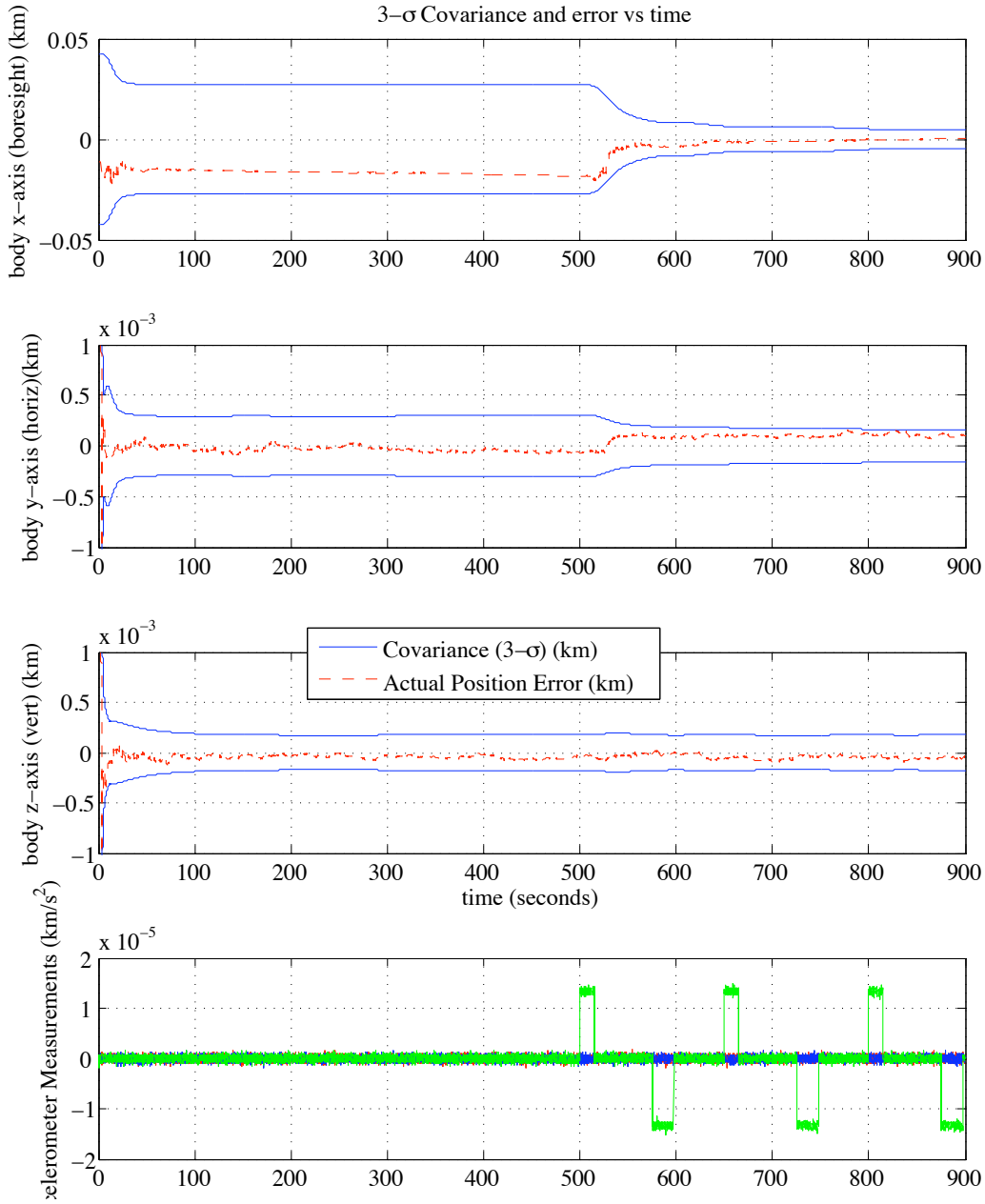


Figure 10. Relative position covariance and errors with good accelerometers

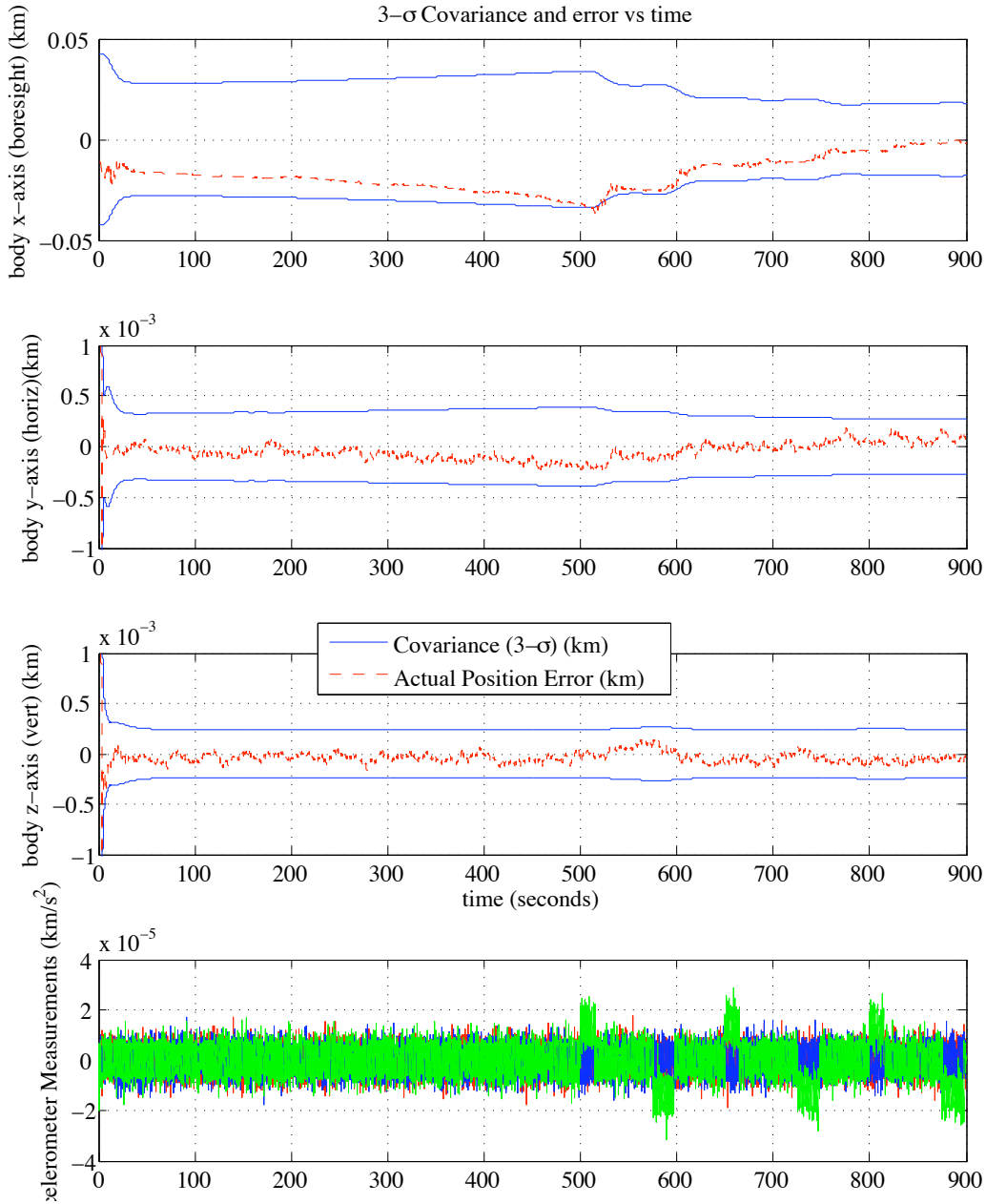


Figure 11. Relative position covariance and errors with average accelerometers

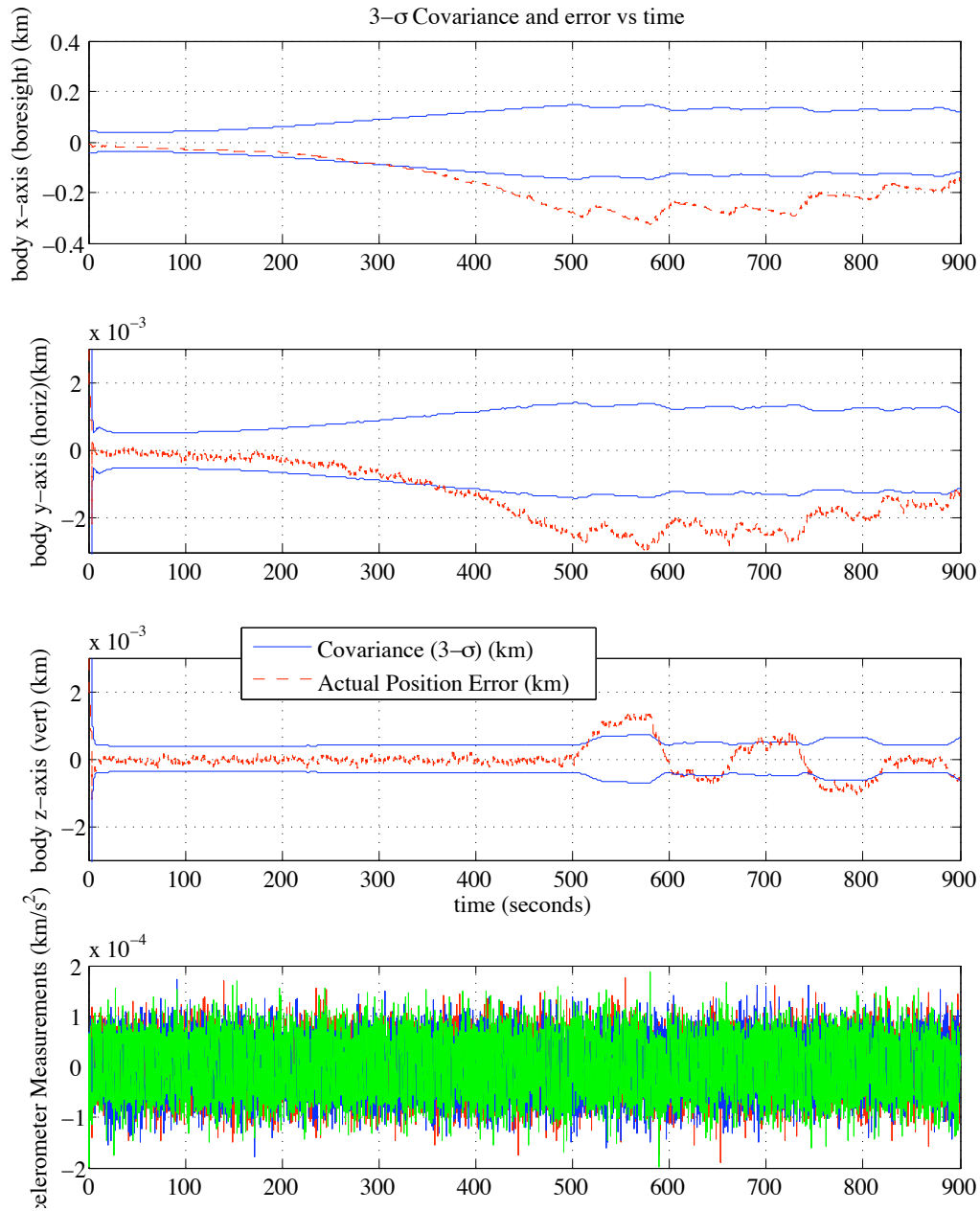


Figure 12. Relative position covariance and errors with poor accelerometers

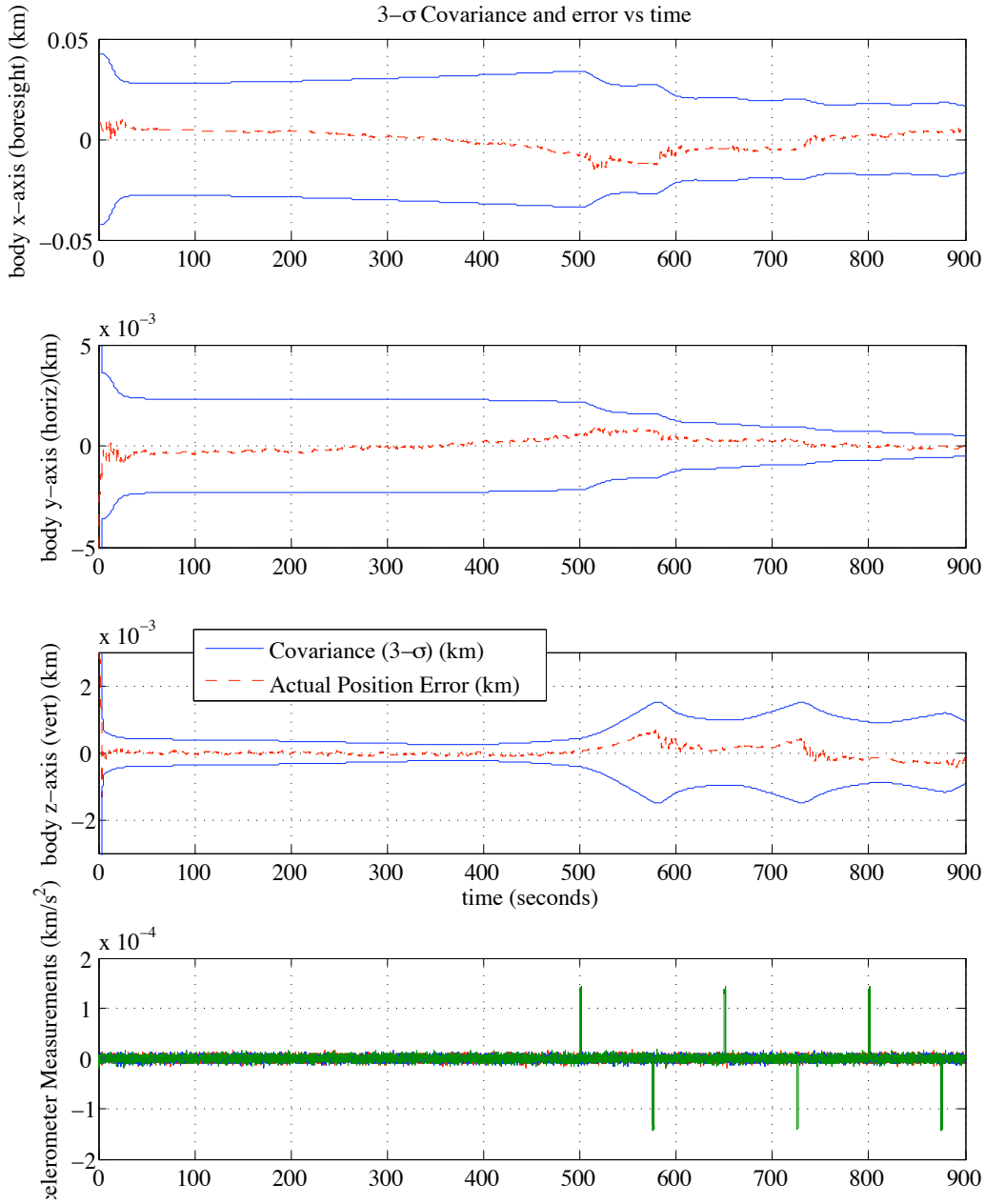


Figure 13. Relative Position Covariance and errors with higher thrust levels

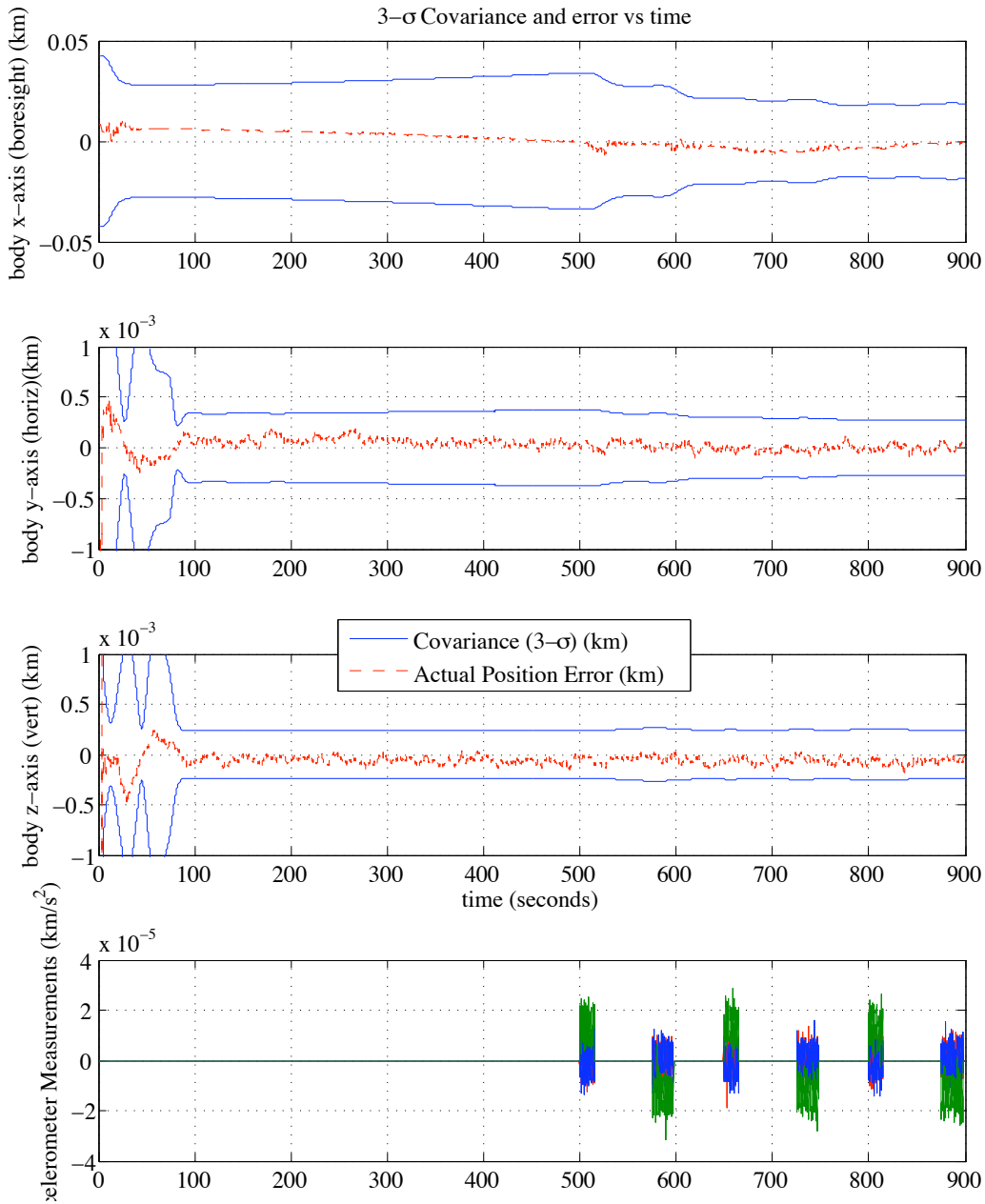


Figure 14. Relative Position Covariance and errors with active accelerometers only when thrusting

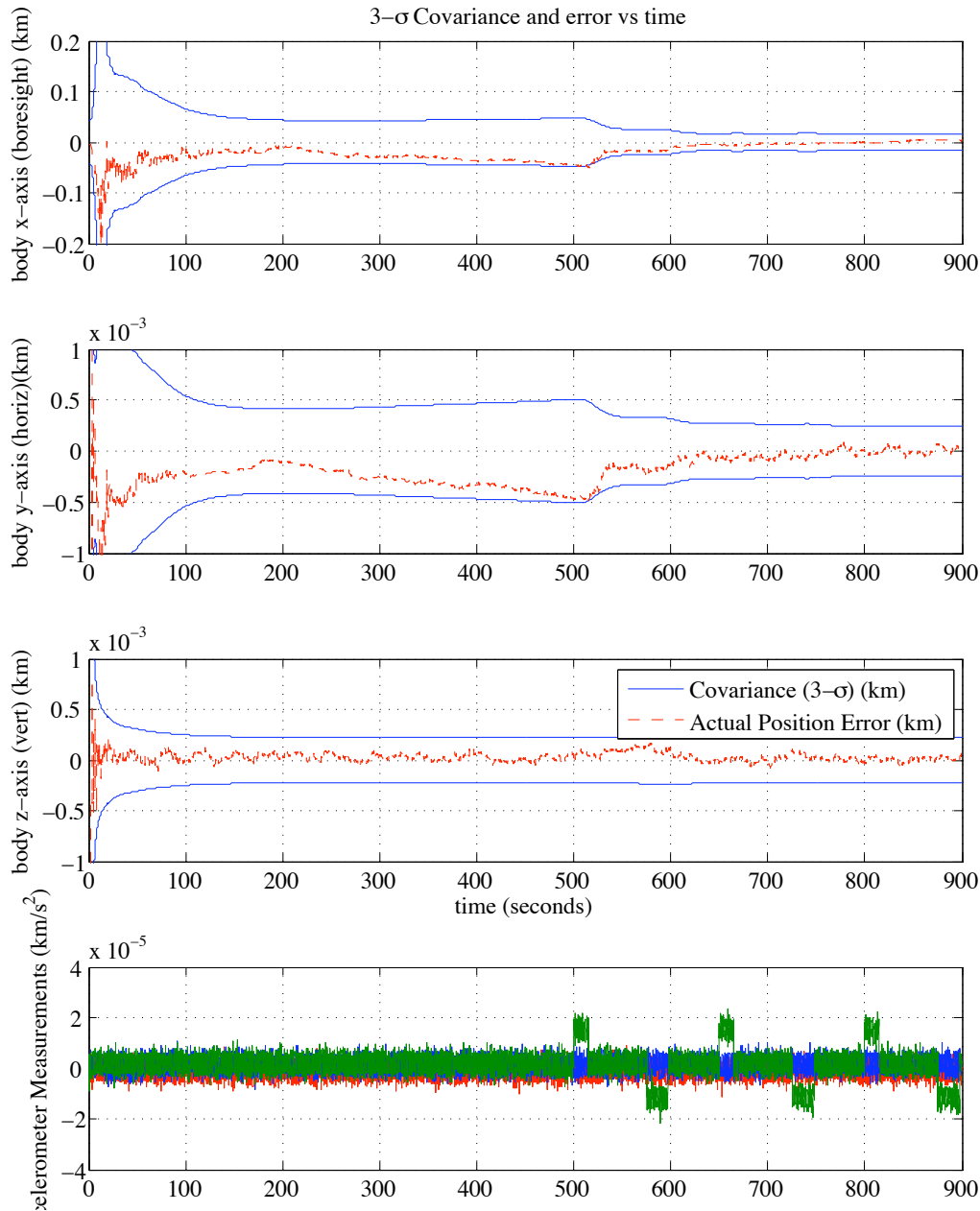


Figure 15. Relative Position Covariance and errors with LN-200 type accelerometers

VI. Conclusions

As expected, the results show a correlation between navigation performance and accelerometer accuracy. If the acceleration level of the maneuvers are not overwhelmed by the measurement noise then the filter can estimate range. If the acceleration levels are overwhelmed by the the noise, then the filter will fail to estimate the relative state accurately (See figure 12).

It is common to improve filter performance by processing the accelerometer measurements only when a burn

is being commanded. Increasing the magnitude of the thrust is also effective at improving navigation performance. Comparing figures 11, 13 and 14 reveals that processing accelerometer data only when maneuvers are being executed is slightly more effective at improving state estimation than increasing burn strength and both techniques are better than the original case (figure 11).

The run with LN-200 type accelerometers (figure 15) also converges to the correct state despite the increased bias on the accelerometers.

These results are preliminary and further testing and

investigation will be required before the effects of accelerometer error and processing on state estimation can be fully quantified. However, these results indicate that a camera combined with average commercial off-the-shelf accelerometers and occasional maneuvers may be able to estimate the relative state accurately enough for proximity operations, and deserves further investigation.

References

- ¹Chari, R. J. V., *Autonomous Orbital Rendezvous Using Angles-Only Navigation*, Master's thesis, Massachusetts Institute of Technology, June 2001.
- ²Edwin Eugene Aldrin, J., *Line-of-sight Guidance Techniques for Manned Orbital Rendezvous*, Ph.D. thesis, Massachusetts Institute of Technology, January 1963.
- ³Marc D. Rayman, Phillip Varghese, D. H. L. L. L., "Results from the Deep Space 1 Technology Validation Mission," *Acta Astronautica*, Vol. 47, July-November 2000, pp. 475–487.
- ⁴Bryan E. Bingham, D. K. G., *Preliminary Orbit Determination for Orbital Rendezvous using Gauss' Method*, Master's thesis, Utah State University, Logan, Utah, 2007.
- ⁵Aidala, V. J., "Kalman Filter Behavior in Bearings-Only Tracking Applications," *IEEE Transactions on Aerospace and Electronic Systems.*, Vol. AES-15, IEEE, January 1979, pp. 29–39.
- ⁶Woffinden, D. C., *Angles-only Navigation for Autonomous Orbital Rendezvous*, Ph.D. thesis, Logan, Utah, 2008.
- ⁷Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME – Journal of Basic Engineering*, Vol. 82, 1960, pp. 35–45.
- ⁸Maybeck, P. S., *Stochastic Models, Estimation, and Control*, Vol. 1, chap. 7, Navtech Book & Software Store, 1994.
- ⁹Byron D. Tapley, B. E. S. and Born, G. H., *Statistical Orbit Determination*, Elsevier Academic Press, 2004.
- ¹⁰Higham, N., "Analysis of the Cholesky decomposition of a semi-definite matrix," *Reliable Numerical Computation*, edited by S. J. H. M. G. Cox, Oxford University Press, 1990, pp. 161–185.
- ¹¹Systems, L. G. . C., "Product Overview: Summary of LN-200 IMU Characteristics," .