Computationally Efficient Algorithms for Real-time Attitude Estimation

N 9 3-24709

Steven R. Pringle

McDonnell Douglas Space Systems Company

/54735

P- 9

ABSTRACT

For many practical spacecraft applications, algorithms for determining spacecraft attitude must combine inputs from diverse sensors and provide redundancy in the event of sensor failure. A Kalman filter is suitable for this task, however, it may impose a computational burden which may be avoided by sub optimal methods. A sub optimal estimater is presented which was implemented successfully on the Delta Star spacecraft which performed a 9 month SDI flight experiment in 1989. This design sought to minimize algorithm complexity to accommodate the limitations of an 8K guidance computer. The algorithm used is interpreted in the framework of Kalman filtering and a derivation is given for the computation.

INTRODUCTION

Historically, satellite attitude determination has relied on simple deterministic calculations for batch processing of telemetry data because real-time recursive algorithms such as Kalman filters imposed an impractical computational burden. This burden has become less daunting with advances in flight-qualified microprocessors, however, simple algorithms remain important for maintaining the reliability and controlling the development cost of real-time software.

This paper examines the algorithm used to estimate attitude for Delta Star. This algorithm applies deterministic gains to measurement data. Nonetheless, it is desirable to perform an statistical error analysis. The attitude estimation problem is cast as a Kalman filtering problem such that the performance of the sub optimal deterministic gains can be quantified. As a convenient byproduct, the Kalman gains implicit in this setup provide an alternative estimation procedure with only a modest increase in computations.

DELTA STAR BACKGROUND

The SDIO sponsored Delta Star spacecraft operated on-orbit for nine months during 1989. Its objectives included multi-spectral observation of low earth orbital phenomena against various earth and space backgrounds. Numerous pointing and tracking guidance modes required modest but reliable, knowledge of spacecraft attitude.

The spacecraft consisted of two sections: a guidance and propulsion section and a sensor section. Each was controlled by separate processors designated guidance computer (GC)

and flight processor (FP) respectively. All primary GN&C functions resided in the GC. Because the guidance and propulsion section was based on the Delta launch vehicle second stage, the GC was a Delco Magic 352 guidance computer, featuring 8K of random access memory to accommodate data, program instructions and a resident operating system. In addition to GN&C functions, the GC flight program sequenced discretes to control avionics subsystems, processed telemetry and uplinks from the ground, and provided a protocol for communication with the FP. The limited memory budgeted for attitude determination made a simple design imperative.
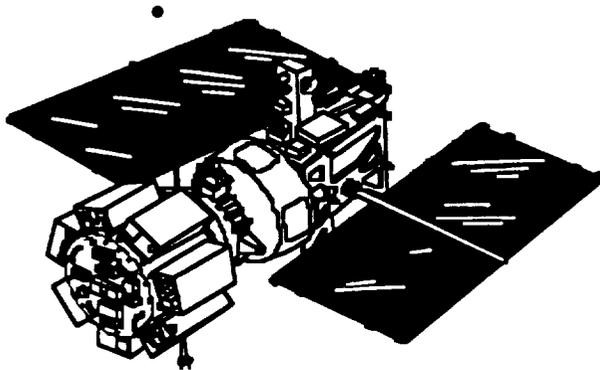
Figure 1. Delta Star Spacecraft

QUATERNION CONVENTIONS

In this section notation and conventions are developed for the quaternion q. The two primary coordinate frames of interest in this report are an inertial reference frame I, and a spacecraft body-fixed frame B. A coordinate frame is given by a triad of orthonormal basis vectors which obey the right-hand rule. A change of basis is specified by a rotation or direction cosine matrix T_I^B defined by

$$y_{\bullet} = T_I^B y_I \qquad (1.)$$

where
$$y_{\bullet} \in \Re^3$$
$$y_I \in \Re^3$$

are the same vector expressed in I and B coordinates.

The rotation matrix T_I^B can be represented by a quaternion q. the quaternion is a globally nonsingular mapping of the rotation matrix. The set of attitude quaternions is defined as

$$Q = \left\{ (q_{\bullet}, q_v) \in \Re \times \Re^3 : q_{\bullet}^2 + \|q_v\|^2 = 1; q_{\bullet} \geq 0 \right\}$$
)

where the first condition is the unit quaternion normality constraint and the second is a convention to eliminate the ambiguity of sign which arises because $(q_{\bullet}, q_v)$ and $(-q_{\bullet}, -q_v)$ represent the same attitude. With these conventions, T_I^B can be computed from the quaternion q by the formula

$$T_I^B = I + 2\Omega_{q_v}^2 - 2q_{\bullet}\Omega_{q_v} \qquad (3.)$$

where for $a \in \Re^3$, Ω_a is defined as

$$\Omega_a : \Re^3 \times \Re$$
$$c = \Omega_a b \qquad (4.)$$
$$c = a \times b$$

for $a, b \in \Re^3$. Quaternion multiplication is defined as follows:

218
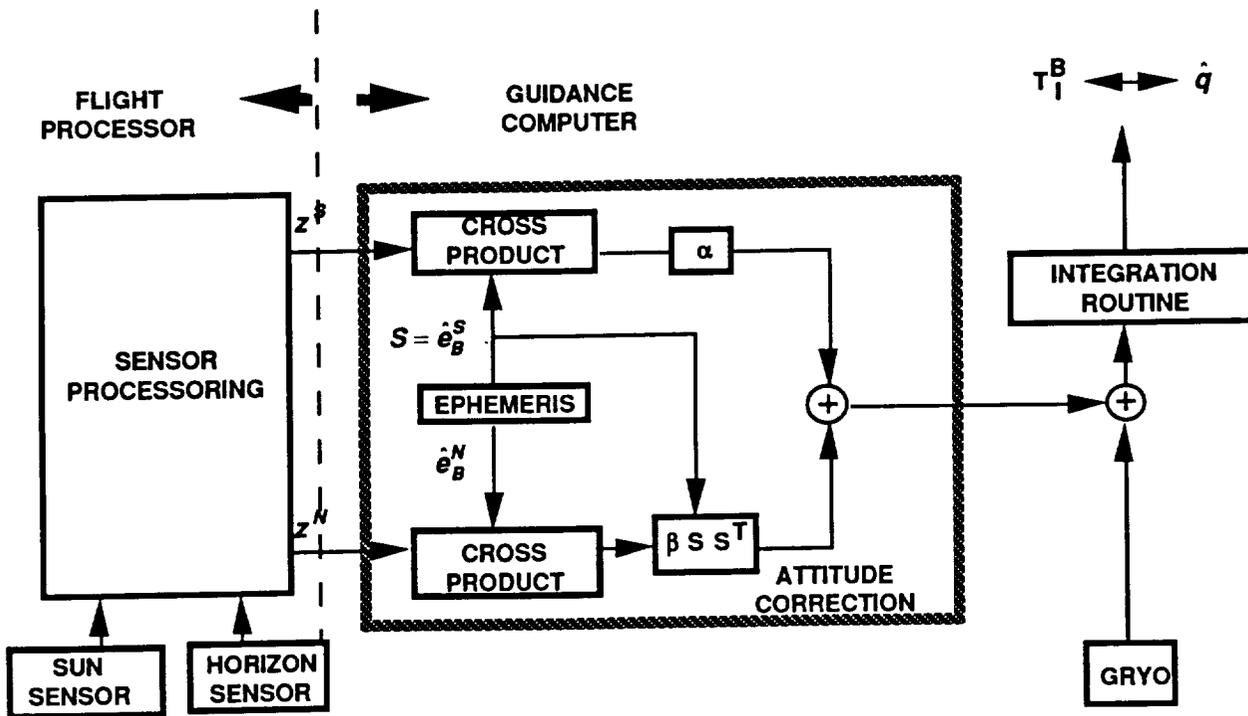
FIGURE 2. Delta Star Attitude Filter

$$c = ab; \quad a,b \in Q$$
$$c_s = a_s b_s - [a_v, b_v]$$
$$c_v = a_s b_v + b_s a_v + a_v \times b_v \qquad (5.)$$
$$c = (c_s, c_v) \in Q$$

Quaternion multiplication is important because it corresponds to compositions of quaternion rotations. That is, for coordinate frames **A,B,C**, if q_1 rotates **A** into **B** and q_2 rotates **B** into **C**, then $q_3 = q_1 q_2$ rotates **A** into **C**.

The quaternion q. representing T_I^B, evolves in time according to the equation

$$\dot{q} = \frac{1}{2} q(0, \omega) \qquad (6.)$$

where ω is the instantaneous angular velocity of the B-frame with respect to the I-frame specified in B coordinates.

The inverse of q is denoted q^* which is also called the conjugate of q [3] and is defined by

$$q^* = (q_s, -q), \text{ and } qq^* = q^* q = (1, 0)$$
)

Finally, a vector $v_I \in \Re^3$ in I coordinates is transformed into $v_B \in \Re^3$ in B coordinates by

$$(0, v_B) = q^*(0, v_I)q = (0, T_I^B v_I) \qquad (8.)$$

DELTA STAR ATTITUDE DETERMINATION

The attitude sensors on Delta Star spacecraft consisted of five sun sensors and a dual conical scan horizon sensor. The five sun sensors were configured to provide omni-directional sun coverage. The horizon sensor had a 26°x26° field of view. These sensors provided attitude measurements for comparison against on-line

ephemeris. The FP edited sensor data for wild points and compensated horizon sensor data for earth oblateness before passing unit vectors for sun and nadir across a communication interface. Modern earth sensors are equipped to provide such compensation using embedded microprocessors. The Delta Star Attitude Filter (DSAF)design is shown in *Figure 2*.

Traditional deterministic methods which compute a direction cosine matrix from a pair of independent measured vectors, such as the TRIAD algorithm [2], offer extreme simplicity but suffer from several deficiencies,

> 1. Only the current vector pair factor into the attitude estimate (i.e. noisy measurements are not averaged).
>
> 2. Nearly collinear measured vectors produce dubious solutions
>
> 3. The two vectors of a pair must be synchronous for a solution and cause complications if they arrive asynchronously.
>
> 4. Measurements from different sensors cannot be weighted to reflect relative noise levels.

A Kalman filter will eliminate these deficiencies. However, the computations required by such a filter were considered prohibitive for the Delta Star application. The design shown in *Figure 2*. also eliminates these deficiencies, but without the matrix computations required by the Kalman filter to propagate a covariance matrix and compute a gain as a function of the covariance.

The constants α and β are design parameters used to control noise rejection and to weight measurements from sun and horizon sensors with respect to each other. A method for preforming a statistical error analysis of this design is presented below. Sub optimal gains are derived in terms of α and β. A statistical interpretation of these gains is given which provides considerations for selecting α and β.

In *Figure 2.*, a running estimate of attitude is maintained by integrating angular rates from gyros according to (6.). This running estimate denoted by $\hat{q}$ differs from q as a result of gyro drift and initial condition errors.

A sun sensor produces two measurements from which the sun vector in B-coordinates can be derived. An earth nadir vector is similarly derived from the outputs of an horizon sensor. Specification of this processing will not be given here. These computations were performed by the FP for Delta Star are not formally considered a part of the DSAF design.

We will distinguish between observation vectors and measurements of these vectors. An observation vector will be denoted by $e^A_{C_k} \in \Re^3$ for time t_k where A is a tag denoting the type of observation {S:sun,N:nadir}.and C denotes the coordinate frame in which the vector is expressed.

A measurement $z^A_k \in \Re^3$ of $e^A_{B_k}$ is derived from sun and horizon sensors. For our

220

purposes, the former is a "noisy" version of the latter.

The vectors $e^N_{l\,k}$ and $e^S_{l\,k}$ are available from an on board ephemeris calculator. From these reference vectors estimates of e^A_{Bk} denoted by $\hat{e}^A_{Bk}$ are computed by

$$(0,\hat{e}^A_{Bk}) = \hat{q}^*(0,e^A_{l\,k})\hat{q} \qquad (9.)$$

The discrepancy between q and $\hat{q}$ is then estimated from the discrepancy between $\hat{e}^A_{Bk}$ and z^A_k. In *Figure 2.*, we note that the discrepancy between $\hat{e}^A_{Bk}$ and z^A_k is captured in the form of the cross product of these two vectors which is used to compute a corrective rate by which to improve the estimate $\hat{q}$.

The discrepancy between q and $\hat{q}$ will be defined by

$$q = \delta q\,\hat{q} \qquad (10.)$$

Because δq_s can be computed from δq_v using the normality constraint in (2.), δq_v will be used to define the attitude error. In the statistical error analysis below, we investigate the behavior of $P = Cov(\delta q_v)$ for the DSAF given specified statistical assumptions.

In the following, we derive a measurement sequence $\{v^A_k\}^-_{k=1}$ and matrix sequences $\{\Phi_k, H_k, K_k\}^-_{k=1}$ such that under specified conditions the following error propagation and update equations apply for the DSAF:

$$\delta q^-_{v_{k+1}} = \Phi_k \delta q^+_{v_k}$$
$$v^A_k = H^A_k \delta q^-_{v_k} + \eta^A_k \; ; Cov(\eta^A_k) = R^A_k \qquad (11.)$$
$$\delta \hat{q}^+_{v_k} = \delta \hat{q}^-_{v_k} + K^A_k (v^A_k - H^A_k \delta q_{v_k})$$

To begin, v^A_k is defined by

$$v^A_k = \hat{e}^A_{Bk} \times z^A_k \qquad (12.)$$

where η^A_k is considered gaussian white noise with covariance matrix R^A_k. Given $\hat{q}$, we can define $h^A_k : \Re^3 \to \Re^3$ from (11.) by

$$v^A_k = h^A_c(\delta q_{vk}) + \eta_k \qquad (13.)$$

The Jacobian matrix of $h^A_k : \Re^3 \to \Re^3$ is given by

$$H^A_k = Dh_k(\delta q_{v_k}) = -2\hat{T}^B_l \Omega^2_e \qquad (14.)$$

where D is the derivative operator, $\hat{T}^l_B$ is computed from (2.) and $e = \hat{e}^A_B$

Then

$$v^A_k - h_k(0) = H^A_k \delta q_{v_k} + \eta^A_k + o(\|\delta q_{v_k}\|)$$
$$)$$

so that to first order

$$v^A_k = H^A_k \delta q_{v_k} + \eta^A_k \qquad (16.)$$

This is the linearized observation equation.

By (5.) and (9.) if $\varepsilon = 0$ over the interval $[t_k, t_{k+1})$

$$\delta q_{v_{k+1}} = \delta q_{v_k}$$
$$= \Phi_k \delta q_{v_k} \; ; \Phi_k = I \qquad (17.)$$

221

This is the linearized state provides Φ_k and gives the linearized state equation Finally,

$$\dot{\hat{q}} = \frac{1}{2}\hat{q}(0,\omega+\varepsilon); \ \hat{q}(t_k) = \hat{q}_k; \ \hat{q}_{k+1} = \hat{q}(t_k)$$

(18.)

To first order this is equivalent to the Extended Kalman Filter (EKF) update scheme defined by the procedure

$$\bar{q}_k^- = \hat{q}_k$$

$$\bar{q}_k^+ = (*,\frac{\tau}{2}T_B^I\varepsilon)\bar{q}_k^-$$

$$\dot{\bar{q}} = \frac{1}{2}\bar{q}(0,\omega+\varepsilon); \ \bar{q}(t_k) = \bar{q}_k^+; \ \bar{q}_{k+1}^- = \bar{q}(t_k)$$

)

where

$$\bar{q}_{k+1}^- = \hat{q}_{k+1} + o(\tau); \quad \tau = t_{k+1} - t_k$$

(20.)

Then to summarize , (16.),(17.) and (19.) provide an a set of propagation and update equations in the same form as (11.) with

$$K_k^S = \alpha\tau T_{Bk}^I$$

$$K_k^N = \beta\tau T_{Bk}^I SS^T; S = \hat{e}_{Bk}^S$$

(22.)

This signal generation model enables us to analyze the behavior of the covariance

$$P_k = Cov(\delta q_{vk}) \cdot$$

(23.)

The covariance for this setup propagates according to

$$P_{k+1} = F_k^A P_k (F_k^A)^T + G_k^A$$

)

where

$$F_k^A = (I - K_k^A H_k^A); G_k^A = K_k^A R_k^A (K_k^A)^T \cdot$$

By (17.) we need not distinguish between pre-update and post-update covarances (i.e..

$P_{k+1}^- = P_k^+)$. If R_k^A is constant, which we shall assume, then it is easy to see than G_k^A is also constant.
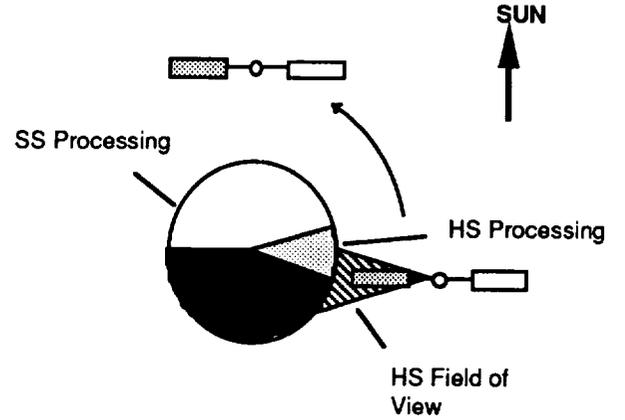
FIGURE 3. Solar Inertial Geometry

To understand the significance of α, and β in (22.), consider the simple geometry shown in *Figure 3.*. The sun vector lies in the orbit plane and intervals of sun sensor and horizon sensor usage are as shown. Define the set of basis vectors S,B_1,B_2 where S is the sun vector and B_1,B_2 are chosen to form a right handed orthonormal triad or coordinate frame. We will call this coordinate from the I'-frame.

The vectors S,B_1,B_2 are all eigenvectors of F_k^S, with eigenvalues $\lambda_S,\lambda_{B_1},\lambda_{B_2}$ such that

$$\lambda_S = 1$$
$$\lambda_{B_1} = \lambda_{B_2} = 1-\alpha\tau$$

(25.)

The matrix F_k^S modifies the covariance according to (24.) when sun sensor data is processed. If δq_v is expressed in I' coordinates, then by (24.)

$$\sigma_{Sk+1}^2 = \lambda_S^2 \sigma_{Sk+1}^2 + \gamma_S$$
$$\sigma_{B_1k+1}^2 = \lambda_{B_1}^2 \sigma_{B_1k+1}^2 + \gamma_{B_1}$$
$$\sigma_{B_2k+1}^2 = \lambda_{B_2}^2 \sigma_{B_2k+1}^2 + \gamma_{B_2}$$

(26.)

where

$$\sigma^2_{S_k} = P_{1,1_k}; \quad \gamma_S = G_{1,1_k}$$
$$\sigma^2_{B_{1_k}} = P_{2,2_k}; \quad \gamma_{B_1} = G_{2,2_k}$$
$$\sigma^2_{B_{2_k}} = P_{3,3_k}; \quad \gamma_{B_2} = G_{3,3_k}$$

and it is assumed that G^A_k is constant. The behavior of the variances defined in (26.) is simple to understand in terms of the difference equations. The error around the sun vector increases at a constant rate (in terms of variance) at a rate determined by the sensor noise and the gain α. The orthogonal components decay to a steady-state value as the corresponding eigenvalues are less than unity. The steady-state residual can be computed using the Final Value Theorem for Z-transforms. The decay rate is exponential and easily determined from $\lambda_{B_1}, \lambda_{B_2}$. A design value for α is achieved by establishing acceptable values for error growth around the sun vector, and steady-state residual and decay rate for error about the orthogonal vectors. and trading off one for the other for an "optimal" compromise The horizon sensor gain β, can be selected similarly.

ERROR ANALYSIS

The covariance propagation above is limited because only the effects of sensor noise are considered. To investigate the effects of other errors such as a constant gyro drift, the method described in [1] is used. The basic idea is shown in *Figure 4*.
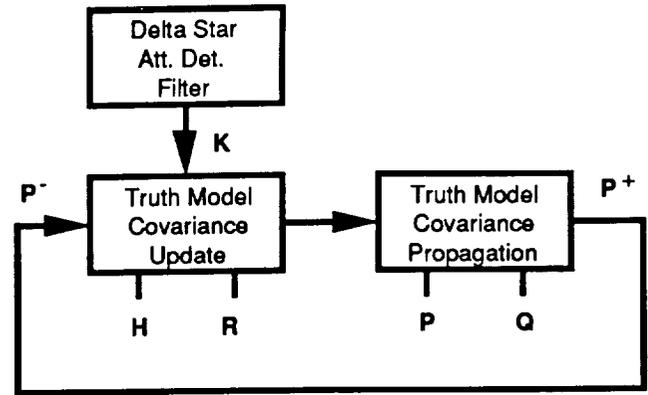
FIGURE 4. Error Analysis Method

For the geometry shown in *Figure 3* this method was used to generate a covariance history assuming that,

$$R^S_k = Diag[3.05\times10^{-6}, 3.05\times10^{-6}, 3.05\times10^{-6}]$$
$$R^N_k = Diag[8.46\times10^{-8}, 8.46\times10^{-8}, 8.46\times10^{-8}]$$
$$P^S_k = Diag[7.61\times10^{-7}, 7.61\times10^{-7}, 7.61\times10^{-7}]$$
$$Q_k = Diag[6.53\times10^{-13}, 6.53\times10^{-13}, 6.53\times10^{-13}]$$
$$\alpha = 0.01$$
$$\beta = 0.06$$

where Q_k is the covariance of the constant gyro bias error in radians per second, quaternion error is dimensionless but, approximately half of angular error in radians and sensor error is similarly approximately half of the angle error produced by sensor noise given in radians The factor of two comes from the definition of quaternion in terms of rotation angle and rotation vector[3]. The result is shown in *Figure 5*.
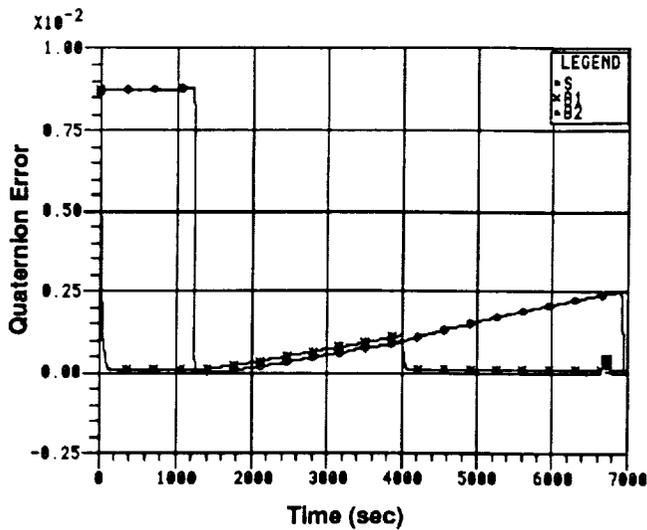
FIGURE 5. DSAF Covariance Analysis
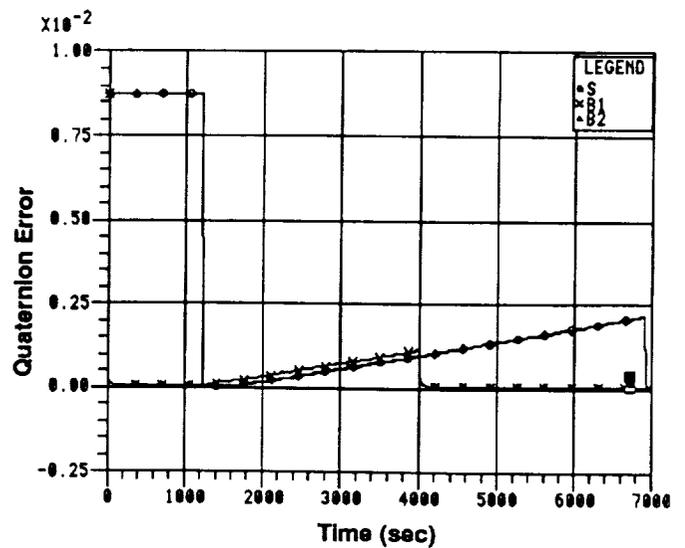
FIGURE 6. Kalman Filter Covariance Analysis

For comparison, the gains in (22.) are replaced by Kalman gains computed using

$$K_k^A = P_k H_k^{A^T} \left(H_k^A P_k H_k^{A^T} + R_k^A \right)^{-1}$$

(27.)

The result is shown in *Figure 6*

In each of these cases, the spacecraft begins by processing sun sensor data. The error around the sun line slowly increases, and the orthogonal components are reduced. At approximately 1100 seconds, an horizon sensor update occurs. On the time interval (1200-4000), no sensor data is processed and pure gyro drift is observed. At 4000 seconds sun sensor data is processed again and a new cycle begins. We observe that the convergence rates are faster for the Kalman filter, that the Kalman filter variances converge to smaller values and that the

Kalman filter performs better with respect to orthogonal components of error during horizon sensor updating. Note that no attempt is made to estimate gyro drift from the sensor data.

HS Field of
View
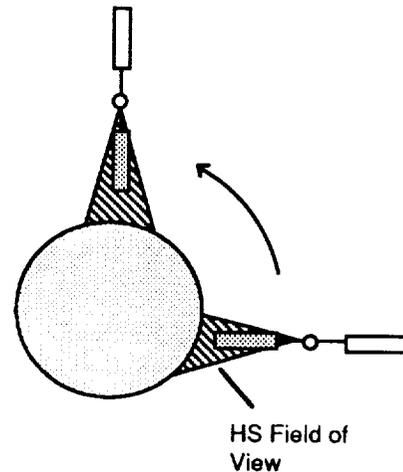
FIGURE 7. Nadir Pointing Geometry

We see that for the scenario described by *Figure 4*. the DSAF compares favorably with the Kalman filter without having to propagate a covariance or compute a Kalman gain. The Kalman filter does, however, afford an advantage which is not evident in the above analysis. The DSAF will not work if only horizon

224

sensor data is available. Clearly, such a capability is desirable for attitude determination reliability in the event of a sensor failure. The Kalman filter does have this capability. For the same statistical assumptions as above, but using only horizon sensor data for the geometry shown in *Figure 7.* we obtain the covariance history shown in *Figure 8.*
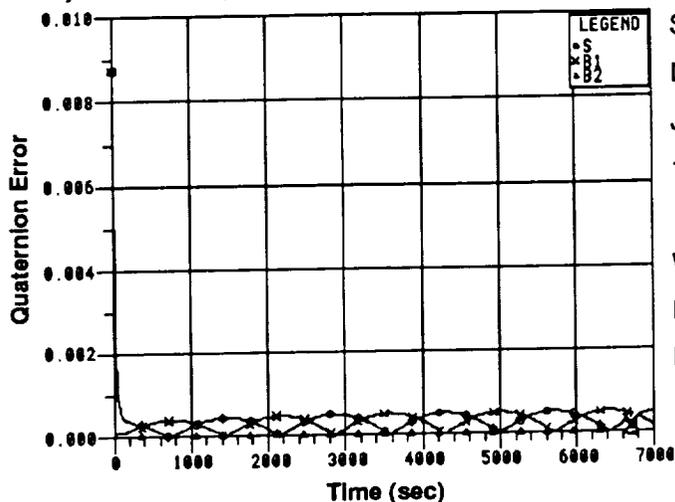
FIGURE 8. DSAF Covariance Analysis
Horizon Sensor Only

presented, and performance is compared against a Kalman filter.

REFERENCES

Gelb, A., Ed., *Applied Optimal Estimation*, MIT Press, Cambridge Mass. 1974.

Shuster, M.D. and Oh, S.D. "Three Axis Attitude Determination from Vector Observations", Journal of Guidance and Control, Vol., Jan.-Feb. 1981, pp. 70-77.

Wertz, J.R., Ed., Spacecraft Attitude Determination and Control, Kluwer Academic Publishers, The Netherlands, 1978.

CONCLUSION

This paper has presented a simple filtering algorithm DSAF for determining spacecraft attitude from vector observations. This algorithm was used successfully on-orbit for the Delta Star SDIO flight experiment in 1989. It offers several advantages over simple deterministic methods such as TRIAD, but does not require as much computation as a Kalman filter mechanization. If a Kalman filter is required or desired for an application, the DSAF is easily extendible to a Kalman filter by means of a more elaborate gain computation. The design parameters of the DSAF are motivated, an error analysis is

225