

The Journal of the Astronautical Sciences, Vol. 57, Nos. 1 & 2, January–June 2009, pp. 000–000

Generalized Linear Covariance Analysis¹

F. Landis Markley² and J. Russell Carpenter³

Abstract

This paper presents a comprehensive approach to filter modeling for generalized covariance analysis of both batch least-squares and sequential estimators. We review and extend in two directions the results of prior work that allowed for partitioning of the state space into “solve-for” and “consider” parameters, accounted for differences between the formal values and the true values of the measurement noise, process noise, and *a priori* solve-for and consider covariances, and explicitly partitioned the errors into subspaces containing only the influence of the measurement noise, process noise, and *a priori* solve-for and consider covariances. In this work, we explicitly add sensitivity analysis to this prior work, and relax an implicit assumption that the batch estimator’s epoch time occurs prior to the definitive span. We also apply the method to an integrated orbit and attitude problem, in which gyro and accelerometer errors, though not estimated, influence the orbit determination performance. We illustrate our results using two graphical presentations, which we call the “variance sandpile” and the “sensitivity mosaic,” and we compare the linear covariance results to confidence intervals associated with ensemble statistics from a Monte Carlo analysis.

Introduction

This paper gives an overview of a comprehensive approach to filter modeling for error analysis. The models presented include both batch least-squares methods, which originated with Gauss [1], and sequential estimators, which were pioneered by Kalman [2]. Both batch least-squares and Kalman filtering methods produce a covariance matrix that encapsulates the expected errors of the estimated parameters and the correlations among them. It was recognized soon after the development of the Kalman filter that the covariance matrix computation does not depend on processing actual data, and that it was very useful to generate covariance matrices corresponding to candidate dynamic models and measurement strategies [3]. Jazwinski [4] describes several early extensions of these covariance analysis techniques to assess the contributions of the various error sources to the uncertainties, and provides references to the original papers.

¹Presented at the F. Landis Astronautics Symposium, Cambridge, Maryland, June 29–July 2, 2008.

²Attitude Control Systems Engineering Branch, NASA Goddard Space Flight Center, Code 591, Greenbelt, MD 20771.

³Navigation and Mission Design Branch, NASA Goddard Space Flight Center, Code 595, Greenbelt, MD 20771.

The number of parameters describing a dynamical system can be quite large, and it is often impractical to solve for all of them, especially in real-time applications. Schmidt developed a variant of the Kalman filter that divides the parameters into two subsets: a set that the estimator solves for and a set of unknown parameters that is not solved for [4–6]. The Schmidt Kalman filter does not estimate the unknown parameters, which are assumed to be constant, but it accounts for the full covariance of the entire set of estimated and unknown parameters. Filters of this type are sometimes called “consider filters,” with the unknown parameters referred to as “consider parameters;” but this usage must be carefully distinguished from the concept of consider parameters used in consider covariance analysis, which is the subject of this paper.

This paper closely follows the approach of references [7] and [8], which extended the methods described in Maybeck [9] and Gelb [10] by partitioning the state space into “solve-for” and “consider” parameters. The consider parameters in this consider covariance analysis differ from the unknown parameters in the Schmidt Kalman filter in two significant respects. Our consider parameters are not required to be constant, and their errors are completely ignored by the estimation algorithm. Their effect on the accuracy of the estimates obtained is considered in the covariance analysis, though, accounting for their name. This treatment is very similar to that described in Bierman [11] and in Tapley, et al [12]. The method described here differs from many other treatments in explicitly partitioning the errors into subspaces containing only the influence of the measurement noise, process noise, and *a priori* solve-for and consider covariances. McReynolds [13] has also implemented such a partitioning in both batch and sequential filter/smooth error budget analysis.

In this paper, we review and extend in two directions the results of references [7] and [8], by explicitly adding sensitivity analysis, and by generalizing the earlier results, which implicitly assumed that the batch estimator’s epoch time occurs prior to the definitive span. We also apply the method to an integrated orbit and attitude problem, in which gyro and accelerometer errors, though not estimated, influence the orbit determination performance. We illustrate our results using two graphical presentations, which we call the “variance sandpile” and the “sensitivity mosaic,” and we compare the linear covariance results to confidence intervals associated with ensemble statistics from a Monte Carlo analysis.

General Dynamics Model

The state vector \mathbf{x} is an n -dimensional vector of parameters that completely characterizes a system, including both time-dependent and constant parameters. By assumption, the state vector evolves according to

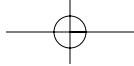
$$\frac{d}{dt} \mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t), t) + \mathbf{w}(t) \quad (1)$$

where the dynamic noise $\mathbf{w}(t)$ is a Gaussian white noise process with mean and covariance given by

$$E[\mathbf{w}(t)] = \mathbf{0}_n \quad \text{and} \quad E[\mathbf{w}(t)\mathbf{w}(\tau)^T] = \mathbf{Q}(t)\delta(t - \tau) \quad (2)$$

with $E[\dots]$ denoting the expectation operator and $\mathbf{0}_n$ denoting an n -dimensional vector of zeros. In the covariance equation, \mathbf{Q} is the $n \times n$ dynamic noise spectral density matrix and $\delta(t - \tau)$ denotes the Dirac delta, or unit impulse, function.

The true value of the state vector is never exactly known, but can only be estimated. Between update times, which are times when measurement information is



Generalized Linear Covariance Analysis

3

incorporated, the state estimate vector, $\hat{\mathbf{x}}$, evolves according to

$$\frac{d}{dt} \hat{\mathbf{x}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t), t) \quad (3)$$

The state error vector, given by

$$\mathbf{e}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t) \quad (4)$$

is assumed to always remain small. Then linear error analysis techniques give, to first order,

$$\frac{d}{dt} \mathbf{e}(t) = A(t)\mathbf{e}(t) + \mathbf{w}(t) \quad (5)$$

where

$$A(t) = \left. \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \right|_{\hat{\mathbf{x}}(t)} \quad (6)$$

Formal integration of equation (5) gives

$$\mathbf{e}(t) = \Phi(t, t_i)\mathbf{e}_i + \mathbf{w}_d(t, t_i) \quad (7)$$

where the state transition matrix, $\Phi(t, t_i)$, is the solution of

$$\dot{\Phi}(t, t_i) = A(t)\Phi(t, t_i) \quad (8)$$

with the initial condition

$$\Phi(t_i, t_i) = I_n = \text{the } n \times n \text{ identity matrix} \quad (9)$$

and the random excitation vector, $\mathbf{w}_d(t, t_i)$, is given by the formal integral

$$\mathbf{w}_d(t, t_i) = \int_{t_i}^t \Phi(t, \tau) \mathbf{w}(\tau) d\tau \quad (10)$$

The state vector error at time t can be written as the sum of three contributions: the error $\mathbf{e}_a(t)$, due to an *a priori* error at an epoch time t_* , the error $\mathbf{e}_v(t)$ due to noise in the measurements used for the estimation, and the error $\mathbf{e}_w(t)$ due to dynamic noise. Thus

$$\mathbf{e}(t) = \mathbf{e}_a(t) + \mathbf{e}_v(t) + \mathbf{e}_w(t) \quad (11)$$

and equation (7) gives

$$\mathbf{e}_a(t) = \Phi(t, t_i)\mathbf{e}_{ai} \quad (12)$$

$$\mathbf{e}_v(t) = \Phi(t, t_i)\mathbf{e}_{vi} \quad (13)$$

and

$$\mathbf{e}_w(t) = \Phi(t, t_i)\mathbf{e}_{wi} + \mathbf{w}_d(t, t_i) \quad (14)$$

General Estimation Model

A filter produces state estimates based on an *a priori* estimate at an epoch time t_* and information obtained from measurements obtained at discrete times t_i . Let \mathbf{y}_i be a vector of measurement values obtained at times t_i . Measurements are related to the state vector by the measurement model

$$\mathbf{y}_i = \mathbf{h}_i(\mathbf{x}_i, t_i) + \mathbf{v}_i \quad (15)$$

where \mathbf{v}_i is a Gaussian white noise process with mean and covariance given by

$$\mathbf{E}[\mathbf{v}_i] = 0, \quad \mathbf{E}[\mathbf{v}_i \mathbf{v}_i^T] = \mathbf{R}_i, \quad \text{and} \quad \mathbf{E}[\mathbf{v}_i \mathbf{v}_j^T] = 0, \quad \text{for } i \neq j \quad (16)$$

The functions \mathbf{h}_i are assumed to be known functions of imprecisely known arguments. Therefore, it is possible to compute predicted measurements from

$$\hat{\mathbf{y}}_i^- = \mathbf{h}_i(\hat{\mathbf{x}}_i^-, t_i) \quad (17)$$

where $\hat{\mathbf{x}}_i^-$ is the state estimate before incorporating the measurement information. Conversely, $\hat{\mathbf{x}}_i$ denotes the state estimate at time t_i including the measurement information. The sequential and batch filters differ in exactly what information is or is not included. The difference between the observed and computed measurements, or “o-minus-c,” is then, to first order in \mathbf{e} ,

$$\mathbf{r}_i^- = \mathbf{y}_i - \hat{\mathbf{y}}_i^- = \mathbf{H}_i \mathbf{e}_i^- + \mathbf{v}_i \quad (18)$$

where the measurement sensitivity matrix \mathbf{H}_i is given by

$$\mathbf{H}_i = \left. \frac{\partial \mathbf{h}(\mathbf{x}_i, t_i)}{\partial \mathbf{x}_i} \right|_{\hat{\mathbf{x}}_i^-} \quad (19)$$

The o-minus-c before the update, \mathbf{r}_i^- , is commonly referred to as the “innovation,” and the o-minus-c after the update, \mathbf{r}_i^+ , as the “residual,” though this terminology is by no means universal.

It is usually not necessary to estimate all of the state parameters. Therefore, a filter may produce estimates for a subset of n_s solve-for parameters. The filter does not estimate the remaining $n_c = n - n_s$ parameters, which are called consider parameters since they contain uncertainties that are considered in the error analysis. The solve-for parameter vector $\mathbf{s}(t)$ and the consider parameter vector $\mathbf{c}(t)$ are given by

$$\mathbf{s}(t) = \mathbf{S}(t)\mathbf{x}(t) \quad \text{and} \quad \mathbf{c}(t) = \mathbf{C}(t)\mathbf{x}(t) \quad (20)$$

where the $n_s \times n$ matrix $\mathbf{S}(t)$ and the $n_c \times n$ matrix $\mathbf{C}(t)$ are such that the matrix

$$\mathbf{M} = \begin{bmatrix} \mathbf{S} \\ \mathbf{C} \end{bmatrix} \quad (21)$$

is nonsingular.⁴ The inverse of \mathbf{M} is partitioned into an $n \times n_s$ matrix $\tilde{\mathbf{S}}$ and an $n \times n_c$ matrix $\tilde{\mathbf{C}}$ as

$$\mathbf{M}^{-1} = [\tilde{\mathbf{S}}, \tilde{\mathbf{C}}] \quad (22)$$

The properties of the matrix inverse then lead immediately to the identities

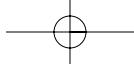
$$\mathbf{S}\tilde{\mathbf{S}} = \mathbf{I}_{n_s}, \quad \mathbf{C}\tilde{\mathbf{C}} = \mathbf{I}_{n_c}, \quad \mathbf{S}\tilde{\mathbf{C}} = \mathbf{0}_{n_s \times n_c}, \quad \mathbf{C}\tilde{\mathbf{S}} = \mathbf{0}_{n_c \times n_s} \quad (23)$$

and

$$\tilde{\mathbf{S}}\tilde{\mathbf{S}} + \tilde{\mathbf{C}}\tilde{\mathbf{C}} = \mathbf{I}_n \quad (24)$$

In the usual case that the elements of the solve-for and consider vectors are merely selected and possibly permuted components of the state vector, the matrix \mathbf{M} is an

⁴Here and subsequently we suppress the time arguments where there is no ambiguity so as to simplify the notation.



Generalized Linear Covariance Analysis

5

orthogonal permutation matrix. In this case, and in any case for which M is orthogonal, the matrices \tilde{S} and \tilde{C} are just the transposes of S and C , respectively, which makes inversion of M unnecessary and simplifies many of the following equations.

It follows from equations (20) and (24) that

$$\mathbf{x}(t) = \tilde{S}(t)\mathbf{s}(t) + \tilde{C}(t)\mathbf{c}(t) \quad (25)$$

Relations similar to equation (20) give the estimated solve-for vector $\hat{\mathbf{s}}(t)$ and the assumed consider vector $\hat{\mathbf{c}}(t)$ in terms of the estimated state $\hat{\mathbf{x}}(t)$. Thus

$$\mathbf{e}_s(t) = \mathbf{s}(t) - \hat{\mathbf{s}}(t) = S(t)\mathbf{e}(t) \quad (26)$$

$$\mathbf{e}_c(t) = \mathbf{c}(t) - \hat{\mathbf{c}}(t) = C(t)\mathbf{e}(t) \quad (27)$$

and

$$\mathbf{e}(t) = \tilde{S}(t)\mathbf{e}_s(t) + \tilde{C}(t)\mathbf{e}_c(t) \quad (28)$$

Since the estimation process does not alter the consider parameters, the estimation error after an update is related to its value before the update by

$$\mathbf{e}_i^+ = \mathbf{e}_i^- + \tilde{S}_i(\mathbf{e}_{si}^+ - \mathbf{e}_{si}^-) \quad (29)$$

where $\tilde{S}_i = \tilde{S}(t_i)$.

General Covariance Analysis

The function of a full estimation system is to determine an estimate $\hat{\mathbf{s}}(t)$ given measurements \mathbf{y}_i . Error analysis, however, does not actually compute an estimate, but determines how good an estimate would be produced in a given situation. The covariance matrix of the estimation error, defined by

$$\mathbf{P}(t) = E[\mathbf{e}(t)\mathbf{e}(t)^T] \quad (30)$$

assuming $E[\mathbf{e}] = 0$, provides a statistical measure of the quality of an estimate at time t of a given scenario.⁵ It is clear that equations (12), (13), and (14) do not introduce correlations among $\mathbf{e}_a(t)$, $\mathbf{e}_v(t)$, and $\mathbf{e}_w(t)$ at time t if no correlations were present at time t_i . If these errors are uncorrelated, the covariance matrix can be written as the sum of *a priori*, measurement noise, and dynamic noise contributions as

$$\mathbf{P}(t) = \mathbf{P}_a(t) + \mathbf{P}_v(t) + \mathbf{P}_w(t) \quad (31)$$

where

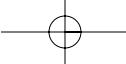
$$\mathbf{P}_a(t) = E[\mathbf{e}_a(t)\mathbf{e}_a^T(t)], \quad \mathbf{P}_v(t) = E[\mathbf{e}_v(t)\mathbf{e}_v^T(t)], \quad \text{and} \quad \mathbf{P}_w(t) = E[\mathbf{e}_w(t)\mathbf{e}_w^T(t)] \quad (32)$$

In most cases, the full covariance is not required, but only its solve-for part

$$\mathbf{P}_{ss}(t) = E[\mathbf{e}_s(t)\mathbf{e}_s^T(t)] = \mathbf{S}\mathbf{P}_a(t)\mathbf{S}^T + \mathbf{S}\mathbf{P}_v(t)\mathbf{S}^T + \mathbf{S}\mathbf{P}_w(t)\mathbf{S}^T \quad (33)$$

In addition to providing a solve-for parameter estimate, an estimation system will generally also compute an estimate $\hat{\mathbf{P}}$ of the solve-for covariance \mathbf{P}_{ss} . Since the true *a priori* error and noise covariances may not be known, the estimator may need to use assumed values for the covariances. The three basic contributions to $\hat{\mathbf{P}}$, also known as the formal error covariance, are the covariance contribution $\hat{\mathbf{P}}_a(t)$ induced by the assumed *a priori* covariance, the covariance $\hat{\mathbf{P}}_v(t)$ induced by the assumed

⁵It should be kept in mind that if systematic errors cause $E[\mathbf{e}] \neq 0$, then $\mathbf{P}(t)$ will describe the mean square error matrix, rather than the covariance.



measurement noise covariance, and the covariance $\hat{P}_w(t)$ induced by the assumed dynamic noise covariance, so that

$$\hat{P} = \hat{P}_a + \hat{P}_v + \hat{P}_w \quad (34)$$

If the assumed covariances do not reflect the actual values (the filter is mistuned) then there will be some covariance contribution due to residual *a priori*, measurement noise, and dynamic noise. Thus

$$P_{ss}(t) = \hat{P}(t) + \Delta P_a(t) + \Delta P_v(t) + \Delta P_w(t) \quad (35)$$

where the residual covariance matrices, which may not be non-negative definite, are

$$\Delta P_a = S P_a S^\top - \hat{P}_a, \quad \Delta P_v = S P_v S^\top - \hat{P}_v, \quad \text{and} \quad \Delta P_w = S P_w S^\top - \hat{P}_w \quad (36)$$

Sequential Filter Covariance Analysis

A sequential filter produces an estimate $\hat{\mathbf{s}}(t)$ based on measurements taken at discrete times t_i , where $t_* \leq t_i < t$. At each measurement time t_i , the solve-for parameters are updated based on the propagated state $\hat{\mathbf{s}}(t_i)$ and the measurements \mathbf{y}_i . Typically, this update has the form

$$\hat{\mathbf{s}}_i^+ = \hat{\mathbf{s}}_i^- + K_i \mathbf{r}_i^- \quad (37)$$

where $\hat{\mathbf{s}}_i^+$ and $\hat{\mathbf{s}}_i^-$ denote estimates of the solve-for parameters immediately after and immediately before incorporating information contained in the measurements at time t_i . The gain matrix K_i determines how much the propagated solve-for parameters are corrected, based on the measurement innovation, \mathbf{r}_i^- . The estimation error immediately after an update is

$$\mathbf{e}_{si}^+ = \mathbf{s}_i^+ - \hat{\mathbf{s}}_i^+ = \mathbf{e}_{si}^- - K_i \mathbf{r}_i^- \quad (38)$$

since the true state is continuous at t_i . Substituting equation (18) for \mathbf{r}_i^- and using equation (29) gives

$$\begin{aligned} \mathbf{e}_i^+ &= \mathbf{e}_i^- - \tilde{S}_i K_i (H_i \mathbf{e}_i^- + \mathbf{v}_i) \\ &= (I_n - \tilde{S}_i K_i H_i) \mathbf{e}_i^- - \tilde{S}_i K_i \mathbf{v}_i \end{aligned} \quad (39)$$

Substituting equation (11) into equation (39) gives update equations for each of the contributions to the state error as

$$\mathbf{e}_{ai}^+ = (I_n - \tilde{S}_i K_i H_i) \mathbf{e}_{ai}^- \quad (40)$$

$$\mathbf{e}_{vi}^+ = (\mathbb{I} - \tilde{S}_i K_i H_i) \mathbf{e}_{vi}^- - \tilde{S}_i K_i \mathbf{v}_i \quad (41)$$

$$\mathbf{e}_{wi}^+ = (I_n - \tilde{S}_i K_i H_i) \mathbf{e}_{wi}^- \quad (42)$$

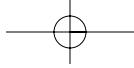
These equations do not introduce correlations between the different parts of $\mathbf{e}(t)$, so equations (40–42) give the covariance updates as

$$P_a(t_i^+) = (I_n - \tilde{S}_i K_i H_i) P_a(t_i^-) (I_n - \tilde{S}_i K_i H_i)^\top \quad (43)$$

$$P_v(t_i^+) = (I_n - \tilde{S}_i K_i H_i) P_v(t_i^-) (I_n - \tilde{S}_i K_i H_i)^\top + \tilde{S}_i K_i R_i K_i^\top \tilde{S}_i^\top \quad (44)$$

$$P_w(t_i^+) = (I_n - \tilde{S}_i K_i H_i) P_w(t_i^-) (I_n - \tilde{S}_i K_i H_i)^\top \quad (45)$$

where R_i is defined by equation (16). The *a priori*, measurement noise, and dynamic noise contributions to the covariance at epoch, before any measurements are processed, are



Generalized Linear Covariance Analysis

7

$$P_a(t_*) = E[\mathbf{e}(t_*)\mathbf{e}^\top(t_*)] = P(t_*), \quad P_v(t) = P_w(t) = 0_{n \times n} \quad (46)$$

Between measurement times t_i , the state estimate propagates according to the integral of equation (3), and the estimation errors propagate according to equations (12), (13), and (14). Since $\hat{\mathbf{x}}_i^+$ does not include any information from times greater than t_i , the correlation terms between \mathbf{e}_{wi} and $\mathbf{w}_d(t, t_i)^\top$ in $P_w(t)$ vanish. Thus, the covariance propagation equations for $t_i \leq t < t_{i+1}$ are given by

$$P_a(t) = \Phi(t, t_i)P_a(t_i^+)\Phi^\top(t, t_i) \quad (47)$$

$$P_v(t) = \Phi(t, t_i)P_v(t_i^+)\Phi^\top(t, t_i) \quad (48)$$

$$P_w(t) = \Phi(t, t_i)P_w(t_i^+)\Phi^\top(t, t_i) + Q_d(t, t_i) \quad (49)$$

where $Q_d(t, t_i)$ is the random excitation covariance matrix, also known as the process noise covariance matrix, defined by

$$Q_d(t, t_i) = E[\mathbf{w}_d(t, t_i)\mathbf{w}_d^\top(t, t_i)] \quad (50)$$

$$= \int_{t_i}^t \int_{t_i}^t \Phi(t, \tau)Q(\tau)\delta(\tau - \tau')\Phi^\top(t, \tau')d\tau d\tau' \quad (51)$$

The covariances $P_a(t_{i+1}^-)$, $P_v(t_{i+1}^-)$, and $P_w(t_{i+1}^-)$ immediately before the $(i+1)$ th update are the limits of equations (47–49) as $t \rightarrow t_{i+1}$.

A Kalman filter is a sequential filter that produces solve-for parameter estimates with minimum covariance due to noise sources known to the filter [9,10]. In addition to the solve-for parameters, a Kalman filter maintains an estimate \hat{P} of the solve-for parameter covariance, and uses this to compute an optimal gain K_i at each time t_i . The covariance estimate \hat{P} is given by algorithms similar to those for P , but without consider parameters. In the update equations, only the solve-for part of the matrix $I_n - \tilde{S}_i K_i H_i$ is used, namely

$$\begin{aligned} S_i(I_n - \tilde{S}_i K_i H_i) \tilde{S}_i &= I_{n_s} - K_i H_i \tilde{S}_i \\ &= I_{n_s} - K_i H_{si} \end{aligned} \quad (52)$$

where $\tilde{S}_i = S(t_i)$ and

$$H_{si} = H_i \tilde{S}_i \quad (53)$$

The update for the covariance estimate and its components are thus given by

$$\hat{P}(t_i^+) = (I_{n_s} - K_i H_{si}) \hat{P}(t_i^-) (I_{n_s} - K_i H_{si})^\top + K_i \hat{R}_i K_i^\top \quad (54)$$

$$\hat{P}_a(t_i^+) = (I_{n_s} - K_i H_{si}) \hat{P}_a(t_i^-) (I_{n_s} - K_i H_{si})^\top \quad (55)$$

$$\hat{P}_v(t_i^+) = (I_{n_s} - K_i H_{si}) \hat{P}_v(t_i^-) (I_{n_s} - K_i H_{si})^\top + K_i \hat{R}_i K_i^\top \quad (56)$$

$$\hat{P}_w(t_i^+) = (I_{n_s} - K_i H_{si}) \hat{P}_w(t_i^-) (I_{n_s} - K_i H_{si})^\top \quad (57)$$

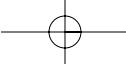
(58)

where $\hat{R}(t_i)$ is the filter's estimate of the measurement noise covariance. The initial values of the covariances are

$$\hat{P}_a(t_*) = \hat{P}(t_*), \quad \hat{P}_v(t) = \hat{P}_w(t) = 0_{n_s \times n_s} \quad (59)$$

where $\hat{P}(t_*)$ is the assumed value for the solve-for part of the *a priori* covariance.

In the propagation equations, only the solve-for part of the state transition matrix is used, namely



$$\Phi_{ss}(t_i, t_j) = \mathcal{S}(t_i)\Phi(t_i, t_j)\tilde{\mathcal{S}}(t_j) \quad (60)$$

The covariance estimate and its component contributions are then propagated in the time interval $t_i \leq t < t_{i+1}$ according to

$$\hat{P}(t) = \Phi_{ss}(t, t_i)\hat{P}(t_i^+)\Phi_{ss}^T(t, t_i) + \hat{Q}_d(t, t_i) \quad (61)$$

$$\hat{P}_a(t) = \Phi_{ss}(t, t_i)\hat{P}_a(t_i^+)\Phi_{ss}^T(t, t_i) \quad (62)$$

$$\hat{P}_v(t) = \Phi_{ss}(t, t_i)\hat{P}_v(t_i^+)\Phi_{ss}^T(t, t_i) \quad (63)$$

$$\hat{P}_w(t) = \Phi_{ss}(t, t_i)\hat{P}_w(t_i^+)\Phi_{ss}^T(t, t_i) + \hat{Q}_d(t, t_i) \quad (64)$$

The $n_s \times n_s$ matrix $\hat{Q}_d(t, t_i)$ is the filter's estimate of the random excitation covariance matrix, which is based on the $n_s \times n_s$ spectral density matrix \hat{Q} of the dynamic noise on the solve-for parameters by

$$\hat{Q}_d(t, t_i) = \int_{t_i}^t \Phi_{ss}(t, \tau)\hat{Q}(\tau)\Phi_{ss}^T(t, \tau)d\tau \quad (65)$$

The covariance estimates $\hat{P}(t_{i+1}^-)$, $\hat{P}_a(t_{i+1}^-)$, $\hat{P}_v(t_{i+1}^-)$, and $\hat{P}_w(t_{i+1}^-)$ immediately before the $(i + 1)$ th update are the limits of equations (61–64) as $t \rightarrow t_{i+1}$.

The Kalman gain is given by [9, 10]

$$K_i = \hat{P}(t_i^-)H_{si}^T[H_{si}\hat{P}(t_i^-)H_{si}^T + \hat{R}_i]^{-1} \quad (66)$$

A Kalman filter will produce an estimate with minimum covariance \hat{P} due to the assumed covariances $\hat{P}(t_*)$, \hat{R}_i , and $\hat{Q}_d(t_i, t_{i-1})$. If the filter is mistuned, the true covariance will not be minimized, and there will be residual covariance contributions given by equations (36).

One may perform covariance analysis of other sequential filters, such as the Schmidt-Kalman filter [6], in similar fashion.

Batch Filter Covariance Analysis

A batch filter produces an estimate $\hat{\mathbf{s}}_*$ at an epoch time t_* , which is not necessarily prior to all of the measurement times t_i , based on a single batch of measurements \mathbf{y} that combines all the measurements \mathbf{y}_i made at various times. Thus,

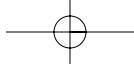
$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_m \end{bmatrix}, \quad \hat{\mathbf{y}} = \begin{bmatrix} \hat{\mathbf{y}}_1 \\ \vdots \\ \hat{\mathbf{y}}_m \end{bmatrix}, \quad \mathbf{r} = \mathbf{y} - \hat{\mathbf{y}} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_m \end{bmatrix} \quad (67)$$

where by equation (16), $E[\mathbf{v}\mathbf{v}^T]$ is block diagonal with R_i as its elements. The batch filter's update may be written in the same form as that of the sequential filter, but with a single update at the epoch time rather than individual updates at each measurement time as

$$\hat{\mathbf{s}}^+ = \hat{\mathbf{s}}_*^- + K\mathbf{r} = \hat{\mathbf{s}}_*^- + \sum_{i=1}^m K_i \mathbf{r}_i \quad (68)$$

where $\hat{\mathbf{s}}_*^-$ is an *a priori* estimate of \mathbf{s}_* and K is a gain matrix consisting of a “row” of gain matrices for each measurement, K_i . Substituting equations (7) and (18) into equation (67) gives

$$\mathbf{r} + H\mathbf{e}_*^- + \mathbf{u}_d + \mathbf{v} \quad (69)$$



Generalized Linear Covariance Analysis

9

where

$$\tilde{H} = \begin{bmatrix} \tilde{H}_1 \\ \vdots \\ \tilde{H}_m \end{bmatrix} = \begin{bmatrix} H_1 \Phi(t_1, t_*) \\ \vdots \\ H_m \Phi(t_m, t_*) \end{bmatrix}, \quad \mathbf{u}_d = \begin{bmatrix} \mathbf{u}_{d1} \\ \vdots \\ \mathbf{u}_{dm} \end{bmatrix} = \begin{bmatrix} H_1 \mathbf{w}_d(t_1, t_*) \\ \vdots \\ H_m \mathbf{w}_d(t_m, t_*) \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_m \end{bmatrix} \quad (70)$$

Using equation (29) for \mathbf{e}_* gives, with equations (68) and (69),

$$\mathbf{e}_*^+ = \left(I_n - \tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \tilde{H}_i \right) \mathbf{e}_*^- - \tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \mathbf{v}_i - \tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \mathbf{u}_{di} \quad (71)$$

which resembles the sequential filter update but with two complications arising from the difference between the measurement times and the update time: the term involving \mathbf{u}_d and the appearance of the state transition matrix in \tilde{H} . Now equation (11) gives the individual components of the state error as

$$\mathbf{e}_{a*}^+ = \left(I_n - \tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \tilde{H}_i \right) \mathbf{e}_*^- \quad (72)$$

$$\mathbf{e}_{v*}^+ = -\tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \mathbf{v}_i \quad (73)$$

$$\mathbf{e}_{w*}^+ = -\tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \mathbf{u}_{di} \quad (74)$$

These errors are uncorrelated, so equations (32) give the covariance at epoch as the sum of the terms

$$P_a(t_*^+) = \left(I_n - \tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \tilde{H}_i \right) P_*^- \left(I_n - \tilde{\mathcal{S}}_* \sum_{i=1}^m K_i \tilde{H}_i \right)^T \quad (75)$$

$$P_v(t_*^+) = \tilde{\mathcal{S}}_* \left(\sum_{i=1}^m K_i R_i K_i^T \right) \tilde{\mathcal{S}}_*^T \quad (76)$$

and

$$P_w(t_*^+) = \tilde{\mathcal{S}}_* \gamma \begin{bmatrix} Q_d(t_*; t_1, t_1) & Q_d(t_*; t_1, t_2) & \dots \\ Q_d(t_*; t_2, t_1) & Q_d(t_*; t_2, t_2) & \dots \\ \vdots & \dots & \ddots \end{bmatrix} \gamma^T \tilde{\mathcal{S}}_*^T \quad (77)$$

with $P_*^- = \hat{P}(t_*)$,

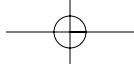
$$\gamma = [K_1 \tilde{H}_1 \quad K_2 \tilde{H}_2 \quad \cdots \quad K_m \tilde{H}_m] \quad (78)$$

and

$$Q_d(t_*; t_i, t_j) = E[\mathbf{w}_d(t_i, t_*) \mathbf{w}_d^T(t_j, t_*)] \quad (79)$$

$$= \int_{t_*}^{t_i} \int_{t_*}^{t_j} \Phi(t_i, \tau) Q(\tau) \delta(\tau - \tau') \Phi^T(t_j, \tau') d\tau d\tau' \quad (80)$$

$$= \begin{cases} \int_{t_*}^{\min(t_i, t_j)} \Phi(t_i, \tau) Q(\tau) \Phi^T(t_j, \tau) d\tau & t_* < t_i, t_* < t_j \\ \int_{\max(t_i, t_j)}^{t_*} \Phi(t_i, \tau) Q(\tau) \Phi^T(t_j, \tau) d\tau & t_i < t_*, t_j < t_* \\ 0 & \text{otherwise} \end{cases} \quad (81)$$



$$= \begin{cases} Q_d(t_i, t_*) \Phi^T(t_j, t_i) & t_* < t_i \leq t_j \\ \Phi(t_i, t_j) Q_d(t_j, t_*) & t_* < t_j \leq t_i \\ \Phi(t_i, t_j) Q_d(t_*, t_j) & t_i \leq t_j < t_* \\ Q_d(t_*, t_i) \Phi^T(t_j, t_i) & t_j \leq t_i < t_* \\ 0 & \text{otherwise} \end{cases} \quad (82)$$

Note that if $t_i = t_j > t_*$, then the definition above reduces to the usual definition of the process noise covariance, i.e. $Q_d(t_*, t_i, t_i) = Q_d(t_i, t_*)$. If $t_i = t_j > t_*$ then $Q_d(t_*, t_i, t_i) = Q_d(t_*, t_i)$, i.e. we have the usual process noise covariance, but it increases as time flows backwards. In the same fashion, the first and fourth cases in equation (82) produce the same numerical result, with the former corresponding to forward and the latter to reverse time flow. The same holds for the second and third cases in equation (82).

The state estimate propagates to any other time according to the integral of equation (3), and the estimation errors propagate according to equations (12), (13), and (14). Therefore, the various components of the error covariances at such other times are given by

$$P_a(t) = \Phi(t, t_*) P_a(t_*) \Phi^T(t, t_*) \quad (83)$$

$$P_v(t) = \Phi(t, t_*) P_v(t_*) \Phi^T(t, t_*) \quad (84)$$

$$P_w(t) = \Phi(t, t_*) P_w(t_*) \Phi^T(t, t_*) + \Phi(t, t_*) N_d(t) + N_d^T(t) \Phi^T(t, t_*) + Q_d(t_*, t, t) \quad (85)$$

where

$$N_d(t) = E[\mathbf{e}_{w^*}^+ \mathbf{w}_d^T(t, t_*)] \quad (86)$$

$$= -E[\sum_i \tilde{\mathbf{S}}_i K_i \mathbf{u}_{di} \mathbf{w}_d^T(t, t_*)] \quad (87)$$

$$= -\sum_i \tilde{\mathbf{S}}_i K_i H_i Q_d(t_*, t, t_i) \quad (88)$$

We note that when we allow for time reversal here, we are considering the *post-diction* problem; the fact that the process noise terms increase the uncertainty as time runs backwards reflects our increasing uncertainty as we propagate further into the past. One should not infer from this work that we take any position as to whether or not a physical Brownian process is diffusing backward in time.

The batch estimator gain is chosen to minimize a quadratic cost function with contributions from the measurement innovations vector \mathbf{r} and the deviation from the the *a priori* estimate weighted by non-negative definite symmetric matrices W and W_* , respectively. This gives

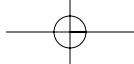
$$K = [W_* + \tilde{H}_s^T W \tilde{H}_s]^{-1} \tilde{H}_s^T W \quad (89)$$

where

$$\tilde{H}_s = \begin{bmatrix} \tilde{H}_{s1} \\ \vdots \\ \tilde{H}_{sm} \end{bmatrix} = \begin{bmatrix} H_{s1} \Phi_{ss}(t_1, t_*) \\ \vdots \\ H_{sm} \Phi_{ss}(t_m, t_*) \end{bmatrix} \quad (90)$$

Assuming that W is block-diagonal with the matrices W_i along its main diagonal, the gains for each measurement are

$$K_i = \left[W_* + \sum_{j=1}^m \tilde{H}_{sj}^T W_j H_{sj} \right]^{-1} \tilde{H}_{si}^T W_i \quad (91)$$



The gain computation uses only the solve-for parts of the measurement sensitivity matrices, \tilde{H}_{si} , from equation (53), and the state transition matrix, Φ_{ss} , from equation (60), because the estimator does not take consider parameter errors into account. The $n_s \times n_s$ matrix

$$\tilde{H}_s^\top \tilde{H}_s = \sum_{i=1}^m \tilde{H}_{si}^\top \tilde{H}_{si} \quad (92)$$

is known as the observability grammian. Singularity or ill-conditioning of the observability grammian indicates a lack of observability of the solve-for states from the measurements \mathbf{y} .

A minimum variance batch estimator produces solve-for estimates with minimum covariance due to the noise sources known to the filter, using the weights

$$W_*^{-1} = \hat{P}_*^-, \quad \text{and} \quad W^{-1} = \hat{R} \quad (93)$$

where \hat{R} is an assumed value for the measurement noise covariance and \hat{P}_*^- is the assumed value for the solve-for part of the *a priori* contribution to the covariance at epoch. In equation (75) the matrix $(I_n - \sum_i \tilde{S}_i K_i \tilde{H}_i)$ is replaced by

$$I_{n_s} - \sum_{i=1}^m K_i \tilde{H}_{si} = (W_* + \tilde{H}_s^\top W \tilde{H}_s)^{-1} W_* \quad (94)$$

which, if W is block-diagonal as above

$$I_{n_s} - \sum_{i=1}^m K_i \tilde{H}_{si} = \left(W_* + \sum_{i=1}^m \tilde{H}_{si}^\top W_i \tilde{H}_{si} \right)^{-1} W_* \quad (95)$$

Therefore, assuming a block-diagonal W , the minimum variance batch filter's assumed covariance components at the epoch time are

$$\hat{P}_{a*}^+ = (W_* + \sum_i \tilde{H}_{si}^\top W_i \tilde{H}_{si})^{-1} W_* (W_* + \sum_i \tilde{H}_{si}^\top W_i \tilde{H}_{si})^{-1} \quad (96)$$

$$\hat{P}_{v*}^+ = (W_* + \sum_i \tilde{H}_{si}^\top W_i \tilde{H}_{si})^{-1} (\sum_i \tilde{H}_{si}^\top W_i \tilde{H}_{si}) (W_* + \sum_i \tilde{H}_{si}^\top W_i \tilde{H}_{si})^{-1} \quad (97)$$

$$\hat{P}_{w*}^+ = 0 \quad (98)$$

and

$$\hat{P}_*^+ = \hat{P}_{a*}^+ + \hat{P}_{v*}^+ = (W_* + \sum_i \tilde{H}_{si}^\top W_i \tilde{H}_{si})^{-1} \quad (99)$$

The reason that $\hat{P}_{w*}^+ = 0$ is that the batch filter does not account for process noise at all. The covariance estimate is propagated to other times by using equations (83) and (84), but only involving the solve-for part of the state transition matrix defined in equation (60). Thus,

$$\hat{P}_a(t) = \Phi_{ss}(t, t_*) \hat{P}_a(t_*^+) \Phi_{ss}^\top(t, t_*) \quad (100)$$

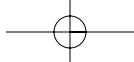
$$\hat{P}_v(t) = \Phi_{ss}(t, t_*) \hat{P}_v(t_*^+) \Phi_{ss}^\top(t, t_*) \quad (101)$$

and

$$\hat{P}(t) = \Phi_{ss}(t, t_*) \hat{P}_*^+ \Phi_{ss}^\top(t, t_*) \quad (102)$$

Sensitivity Matrix Time and Measurement Update

The sensitivity matrix shows the linear sensitivity of the solution at a specified time to mismodeling of the distributions of the *a priori* parameters (both solve-fors and considers). For the sequential filter, it is given by



$$\Sigma_a(t_i) = [I - \tilde{S}(t_i)K_iH(t_i)]\Phi(t_i, t_{i-1})\Sigma_a(t_{i-1}), \quad \Sigma_a(t_o) = [\tilde{S}(t_o), \tilde{C}(t_o)] \quad (103)$$

and for the batch estimator, it is

$$\Sigma_a(t) = S(t)\Phi(t, t_*)[I - \sum_{i=1}^k \tilde{S}(t_i)K_iH(t_i)\Phi(t_i, t_*)][\tilde{S}(t_*), \tilde{C}(t_*)] \quad (104)$$

Although it is possible to compute the sensitivities to each particular measurement and process noise sample,⁶ this does not appear to be particularly useful. Instead, suppose R and \hat{R} have the same structure, but differ only by a scalar multiplier, i.e., $R = r \cdot \bar{R}$ and $\hat{R} = \hat{r} \cdot \bar{R}$ (for example, let $\bar{R} = I$). Then, for the sequential filter,

$$\Delta P_v = SP_v S^\top - \hat{P}_v \quad (105)$$

will contain terms of the form $KK^\top(r - \hat{r})$. In this case, if one chooses $(r - \hat{r}) = 1$, then ΔP_v will represent the sensitivity to measurement noise mismodeling. Similarly, when $Q = q \cdot I$, and $\hat{Q} = \hat{q} \cdot I$, then $(q - \hat{q})$ will factor out of the process noise partition, so that if one chooses $q - \hat{q} = 1$, then ΔP_w will represent the sensitivity to process noise mismodeling. For the batch estimator

$$\Delta P_v = K\bar{R}K^\top(r - \hat{r}) \quad (106)$$

In this case, if one chooses $(r - \hat{r}) = 1$, ΔP_v will represent the sensitivity to measurement noise mismodeling across the entire definitive data span. Similarly, when $Q = q \cdot \bar{Q}$, q will factor out of the (true) process noise partition, so that if one chooses $q = 1$, ΔP_w (which equals P_w since the batch has no process noise) will represent the sensitivity to process noise mismodeling across the entire interval of interest.

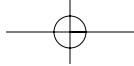
Inertial Navigation System Model

Next, we apply the generalized method to a study of the effect of inertial navigation system (INS) parameters on an entry trajectory. This section describes the model of the INS. As part of this description, we address some issues related to attitude error modeling which appear to us to have never been adequately described in other works.

Our model includes additive bias and noise terms, and multiplicative errors. One of the multiplicative errors is associated with uncertainties in the scale factors used in the devices to convert their internal units into output units. The other error considered is that caused by small misalignments in the mounting of the gyros and accelerometers, so that they are not quite orthogonal to one another. This model shares many of the characteristics of the model used in reference [14], where Space Shuttle navigation performance with strapdown systems was evaluated, as well as with the model described in reference [15], a popular short course on inertial navigation systems. Similar models for platform-type inertial systems are described in references [16] and [17]. A more complicated model that shares some similarities is that of reference [18], in which a Kalman filter for calibration and alignment of inertial navigation systems is described.

⁶It is particularly easy to show that for the batch estimator, the sensitivity to measurement and process noise inputs is

$$\Sigma_{vk}(t_*) = -K_k, \quad \Sigma_{wk}(t_*) = -K_k H(t_k)$$



Gyro Model

INS are generally of two types, “platform” and “strapdown,” which use the gyros in a somewhat different fashion. The platform type consists of a set of gimbals that interconnect the vehicle body to a rigid platform on which the gyros are mounted. This arrangement allows the axes along which the gyros are sensitive to rotate freely with respect to the vehicle body. A feedback system attempts to null the gyro outputs, which thus maintains the gyro platform approximately fixed in inertial space. The vehicle attitude with respect to inertial space is then given by reading out the gimbal angles. The strapdown type, which has largely replaced the platform type in current usage, dispenses with the gimbal system and rigidly fixes the gyros to the vehicle body. The strapdown INS computationally integrates the gyro outputs so as to keep track of the orientation of the body with respect to the inertial frame.

In each type of INS, errors in the gyro output affect the determination of the vehicle attitude, but in a somewhat different manner. In a platform INS, gyro output errors cause the platform to physically drift with respect to the true inertial frame. The strapdown INS measures at each instant an incremental change in the angular position of the body with respect to some inertial frame. Thus, errors in the gyro outputs create error in the computational accumulation of these angular velocity increments.

To illustrate the implications of this difference, we assume (without loss of generality) that the INS represents the attitude as a direction cosine matrix that maps from the INS case frame⁷ to the inertial frame, and we denote its estimate of the attitude as \hat{M}_C^I . One may view this estimate as composed of one of two successive rotations as

$$\hat{M}_C^I = M_C^I \hat{M}_{\bar{C}}^C, \quad \text{or} \quad \hat{M}_C^I = M_I^{\hat{I}} M_{\bar{C}}^C \quad (107)$$

We assume that we can represent $M_{\bar{C}}^C$ and $M_I^{\hat{I}}$ as small angle rotations as

$$M_{\bar{C}}^C = I - \boldsymbol{\theta}^\times \quad \text{and} \quad M_I^{\hat{I}} = I - \boldsymbol{\psi}^\times \quad (108)$$

where the superscript ‘ \times ’ indicates that the elements of the vector are arranged in a skew-symmetric matrix such that $\mathbf{a}^\times \mathbf{b} = \mathbf{a} \times \mathbf{b}$. Note that $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ are in general two different sets of small angle rotations—it is not necessarily the case that they are the same vector expressed in two different coordinate systems.⁸ The (coordinate-independent) derivatives of these vectors with respect to the case and inertial frames are related by

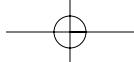
$$\frac{^I d}{dt} \boldsymbol{\theta} = \frac{^C d}{dt} \boldsymbol{\theta} + {}^I \boldsymbol{\omega}^C \times \boldsymbol{\theta} \quad \text{and} \quad \frac{^I d}{dt} \boldsymbol{\psi} = \frac{^C d}{dt} \boldsymbol{\psi} + {}^I \boldsymbol{\omega}^C \times \boldsymbol{\psi} \quad (109)$$

where we denote the angular velocity of frame C in frame I as ${}^I \boldsymbol{\omega}^C$. Similarly, the angular velocity estimate of the INS, ${}^I \hat{\boldsymbol{\omega}}^C$, will be corrupted by the gyro output error, and we may view this estimate as either

$$\widehat{{}^I \boldsymbol{\omega}^C} = {}^I \boldsymbol{\omega}^C + {}^C \hat{\boldsymbol{\omega}}^{\hat{C}} \quad \text{and} \quad \widehat{{}^I \boldsymbol{\omega}^C} = {}^I \hat{\boldsymbol{\omega}}^I + {}^I \boldsymbol{\omega}^C \quad (110)$$

⁷For either type of INS, the case frame is rigidly fixed to the vehicle body, but may not be aligned with the body axes.

⁸However, if these rotations arise from two optimal attitude estimators processing the same data from the same initial conditions, the constraint that both estimators produce the optimal estimate implies that $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ do become the same vector expressed in two different coordinate systems.



Thus, the coordinate-independent angular velocity error vector, $\delta^I \boldsymbol{\omega}^C = \widehat{^I \boldsymbol{\omega}^C} - {^I \boldsymbol{\omega}^C}$, may be viewed as either

$$\delta^I \boldsymbol{\omega}^C = {}^C \boldsymbol{\omega}^{\hat{C}} \quad \text{or} \quad \delta^I \boldsymbol{\omega}^C = \hat{^I \boldsymbol{\omega}^I} \quad (111)$$

Recalling the description of how the platform INS uses the gyro output, it is clear that for the platform system, the gyro error corresponds to a drift of the platform with respect to the inertial, and since the platform represents the INS's knowledge of the inertial frame, it is clear that the platform's drift corresponds to $\hat{^I \boldsymbol{\omega}^I}$. Thus, ψ must be related to $\hat{^I \boldsymbol{\omega}^I}$, according to

$$\frac{^I d}{dt} \psi = \hat{^I \boldsymbol{\omega}^I} \quad (112)$$

Recalling that the strapdown INS integrates the case-fixed gyro outputs, it is clear that a constant gyro drift from a strapdown gyro will appear to be constant to an observer who is also fixed in the case frame, and hence gyro drift from a strapdown INS corresponds to ${}^C \boldsymbol{\omega}^{\hat{C}}$, or

$$\frac{{}^C d}{dt} \theta = {}^C \boldsymbol{\omega}^{\hat{C}} \quad (113)$$

Thus, depending on whether the INS is of the platform type or the strapdown type, a constant gyro output bias will appear to be constant relative to the inertial frame, or constant relative to the case frame, respectively. These gyro biases will integrate into a misalignment ψ with respect to the inertial frame (platform INS), or into a *possibly different* misalignment θ with respect to the case frame (strapdown INS). Note that one is free to write the equations of motion with respect to any frame one chooses, so that for example a set of completely permissible expressions would be

$$\frac{^I d}{dt} \theta = {}^C \boldsymbol{\omega}^{\hat{C}} + {^I \boldsymbol{\omega}^C} \times \theta \quad \text{and} \quad \frac{{}^C d}{dt} \psi = \hat{^I \boldsymbol{\omega}^I} + {^I \boldsymbol{\omega}^C} \times \psi \quad (114)$$

Henceforth, we only consider strapdown INS. For many such systems, the gyro error may be modeled as

$$\delta^I \boldsymbol{\omega}_C^C = {}^C \boldsymbol{\omega}_C^{\hat{C}} = \mathbf{b}_{gC} + \mathbf{S}_g {^I \boldsymbol{\omega}_C^C} + \Gamma_g {^I \boldsymbol{\omega}_C^C} + \boldsymbol{\epsilon}_\omega \quad (115)$$

where \mathbf{b}_{gC} is an angular velocity in the case frame that biases the gyro, \mathbf{S}_g is the gyro scale factor matrix, Γ_g is the gyro nonorthogonality matrix, ${}^I \boldsymbol{\omega}_C^C$ is the representation of ${}^I \boldsymbol{\omega}^C$ in coordinates fixed to the INS case, $\boldsymbol{\epsilon}_\omega$ is a Gaussian white noise that produces angular random walk, and where

$$\mathbf{S}_g = \begin{bmatrix} S_{gx} & 0 & 0 \\ 0 & S_{gy} & 0 \\ 0 & 0 & S_{gz} \end{bmatrix} \quad \text{and} \quad \Gamma_g = \begin{bmatrix} 0 & 0 & 0 \\ -\gamma_{gyz} & 0 & 0 \\ \gamma_{gzy} & -\gamma_{gxz} & 0 \end{bmatrix} \quad (116)$$

Here, the elements of the matrix \mathbf{S}_g represent scale factor errors, and the elements of the matrix Γ_g represent nonorthogonality errors. The latter arise from the fact that (for example) the y gyro will measure not only the angular velocity about C_y as it is intended to, but also a small projection of the angular velocity about C_x . Figure 1 illustrates our convention, from which the structure of the Γ_g matrix should be clear.

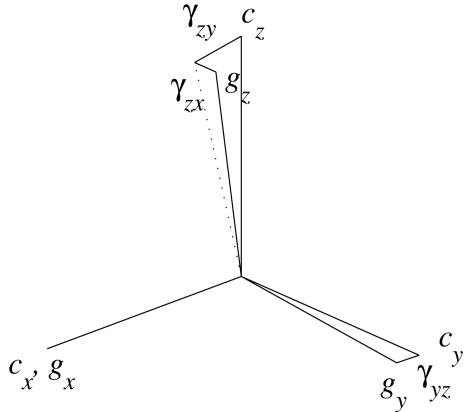
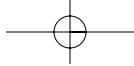


FIG. 1. Non-orthogonality of INS case frame and gyro sensitive axes frame. Here we assume that as part of the INS calibration procedure, the x gyro is defined to be colinear with the case x axis, the x - y case frame is defined to include the y gyro, and misalignments of the y and z gyros from the two axes orthogonal to this reference are measured. In practice, the case frame is defined by an alignment device such as an optical corner cube attached to the outside of the case, and the relationship between the frame defined externally by the optical cube and the internally-defined case frame is loaded into the INS's firmware.

Accelerometer Model

Since we have assumed a strapdown INS, the accelerometers measure the case coordinates of the “sensed acceleration,” \mathbf{a}_C , i.e. that part of the second time derivative of \mathbf{r} , the INS position vector, with respect to an inertial frame, which is due to nongravitational, or “contact,” forces. If the INS is located at the spacecraft center of gravity (c.g.), \mathbf{r} represents the spacecraft position vector as well. If not, the accelerometer will measure not only the external acceleration on the spacecraft, but also the linear acceleration at the c.g. to INS moment arm due to any spacecraft rotational motion. This must be subtracted from the INS measured acceleration, typically by the navigation software. This model does not explicitly include the effect INS errors have on this c.g. offset calculation.

For many strapdown accelerometers, we may assume that the error may be modeled as

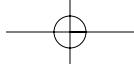
$$\delta\mathbf{a}_C = \mathbf{b}_{aC} + \mathcal{S}_a \mathbf{a}_C + \Gamma_a \mathbf{a}_C \quad (117)$$

where \mathbf{b}_{aC} is the accelerometer bias, \mathcal{S}_a is the accelerometer scale factor matrix, Γ_a is the accelerometer nonorthogonality matrix, and where

$$\mathcal{S}_a = \begin{bmatrix} S_{ax} & 0 & 0 \\ 0 & S_{ay} & 0 \\ 0 & 0 & S_{az} \end{bmatrix}, \quad \text{and} \quad \Gamma_a = \begin{bmatrix} 0 & 0 & 0 \\ -\gamma_{ayz} & 0 & 0 \\ \gamma_{azy} & -\gamma_{azx} & 0 \end{bmatrix} \quad (118)$$

Combined Model

It is convenient to recast equations (115) and (117) as follows. Define $D(\mathcal{I}\boldsymbol{\omega}_C^C) = \text{diag}(\mathcal{I}\boldsymbol{\omega}_C^C)$, and $\mathbf{s}_g = [S_{gx}, S_{gy}, S_{gz}]^\top$. Then $\mathcal{S}_g \mathcal{I}\boldsymbol{\omega}_C^C = D(\mathcal{I}\boldsymbol{\omega}_C^C) \mathbf{s}_g$. Similarly, define $D(\mathbf{a}_c) = \text{diag}(\mathbf{a}_c)$, and $\mathbf{s}_a = [S_{ax}, S_{ay}, S_{az}]^\top$. Then $\mathcal{S}_a \mathbf{a}_c = D(\mathbf{a}_c) \mathbf{s}_a$. Define $\boldsymbol{\gamma}_g$ and $F(\mathcal{I}\boldsymbol{\omega}_C^C)$ such that $\Gamma_g \mathcal{I}\boldsymbol{\omega}_C^C = F(\mathcal{I}\boldsymbol{\omega}_C^C) \boldsymbol{\gamma}_g$, i.e. $\boldsymbol{\gamma}_g = [\gamma_{gyz}, \gamma_{gzy}, \gamma_{gzx}]^\top$, and



$$\mathcal{F}({}^I\boldsymbol{\omega}_C^C) = \begin{bmatrix} 0 & 0 & 0 \\ -{}^I\omega_{Cx}^C & 0 & 0 \\ 0 & {}^I\omega_{Cx}^C & -{}^I\omega_{Cy}^C \end{bmatrix} \quad (119)$$

Similarly, define $\boldsymbol{\gamma}_a$ and $\mathcal{F}(\mathbf{a}_C)$ such that $I_a \mathbf{a}_C = \mathcal{F}(\mathbf{a}_C) \boldsymbol{\gamma}_a$. Then, equations (115) and (117) may be written as

$$\delta {}^I\boldsymbol{\omega}_C^C = \mathbf{b}_{gC} + D({}^I\boldsymbol{\omega}_C^C) \mathbf{s}_g + \mathcal{F}({}^I\boldsymbol{\omega}_C^C) \boldsymbol{\gamma}_g + \boldsymbol{\epsilon}_\omega \quad (120)$$

$$\delta \mathbf{a}_C = \mathbf{b}_{aC} + D(\mathbf{a}_C) \mathbf{s}_a + \mathcal{F}(\mathbf{a}_C) \boldsymbol{\gamma}_a \quad (121)$$

Now, let $\mathbf{x}_g = [\boldsymbol{\theta}_C^\top, \mathbf{b}_{gC}^\top, \mathbf{s}_g^\top, \boldsymbol{\gamma}_g^\top]^\top$. Then, one can rewrite equation (115) as

$$\begin{bmatrix} \frac{d}{dt} \boldsymbol{\theta}_C \\ \frac{d}{dt} \mathbf{b}_{gC} \\ \frac{d}{dt} \mathbf{s}_g \\ \frac{d}{dt} \boldsymbol{\gamma}_g \end{bmatrix} = \begin{bmatrix} \mathbf{O}_{3 \times 3} & I_3 & D({}^I\boldsymbol{\omega}_C^C) & \mathcal{F}({}^I\boldsymbol{\omega}_C^C) \\ & \mathbf{O}_{9 \times 12} & & \end{bmatrix} \mathbf{x}_g + \begin{bmatrix} I_3 \\ \mathbf{O}_{9 \times 3} \end{bmatrix} \boldsymbol{\epsilon}_\omega \quad (122)$$

$$\dot{\mathbf{x}}_g = A_g \mathbf{x}_g + B_g \boldsymbol{\epsilon}_\omega \quad (123)$$

where we have chosen to write the equations of motion for the gyro misalignment and drift with respect to the case frame, and we recall that since the scale factor and nonorthogonality coefficients are only shown as column arrays for convenience, they are not properly viewed as physical vectors that take derivatives in any particular coordinate system. In similar fashion, let $\mathbf{x}_a = [\mathbf{b}_{aC}^\top, \mathbf{s}_a^\top, \boldsymbol{\gamma}_a^\top]^\top$, with $\dot{\mathbf{x}}_a = \frac{d}{dt} \mathbf{x}_a = 0$, and cast the acceleration error $\delta \mathbf{a}_C$ as

$$\delta \mathbf{a}_C = [I_3 \quad D(\mathbf{a}_C) \quad \mathcal{F}(\mathbf{a}_C)] \mathbf{x}_a \quad (124)$$

$$= H_a \mathbf{x}_a \quad (125)$$

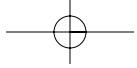
We refer to the combination of the two models above as our Inertial Measurement Unit (IMU) model. When we integrate these results to produce position and velocity errors, we have a full INS model. When an INS incorporates sensed acceleration into its computation of the total acceleration on the vehicle, its gyro misalignment will corrupt the calculation as

$$\widehat{\frac{d}{dt} \mathbf{v}_I} = -\frac{\mu}{r^3} \hat{\mathbf{r}}_I + \widehat{M_C^I} \hat{\mathbf{a}}_C \quad (126)$$

$$\approx -\frac{\mu}{r^3} \mathbf{r}_I + \mathbf{G}(\mathbf{r}_I) \delta \mathbf{r}_I + M_C^I [I - \boldsymbol{\theta}_C^\times] (\mathbf{a}_C + \delta \mathbf{a}_C) \quad (127)$$

where the gravity gradient matrix, $\mathbf{G}(\mathbf{r}_I)$, is the Jacobian of the two-body gravity term, $-\mu \mathbf{r}_I / r^3$. Thus, to first order, we may approximate the total acceleration error equations of motion as

$$\widehat{\frac{d}{dt} \delta \mathbf{v}_I} = \mathbf{G}(\mathbf{r}_I) \delta \mathbf{r}_I + M_C^I \mathbf{a}_C^\times \boldsymbol{\theta}_C + M_C^I \delta \mathbf{a}_C \quad (128)$$



These models may be combined as

$$\begin{bmatrix} \frac{^I d}{dt} \delta \mathbf{r}_I \\ \frac{^I d}{dt} \delta \mathbf{v}_I \\ \dot{\mathbf{x}}_a \\ \dot{\mathbf{x}}_g \end{bmatrix} = \begin{bmatrix} \mathbf{O}_{3 \times 3} & I_3 & \mathbf{O}_{3 \times 9} & \mathbf{O}_{3 \times 12} \\ G(\mathbf{r}_I) & \mathbf{O}_{3 \times 3} & M_C^T H_a & M_C^T X_{ag} \\ \mathbf{O}_{9 \times 3} & \mathbf{O}_{9 \times 3} & \mathbf{O}_{9 \times 9} & \mathbf{O}_{9 \times 12} \\ \mathbf{O}_{12 \times 3} & \mathbf{O}_{12 \times 3} & \mathbf{O}_{12 \times 9} & A_g \end{bmatrix} \begin{bmatrix} \delta \mathbf{r}_I \\ \delta \mathbf{v}_I \\ \mathbf{x}_a \\ \mathbf{x}_g \end{bmatrix} + \begin{bmatrix} \mathbf{O}_{15 \times 3} \\ B_g \\ \mathbf{x}_g \end{bmatrix} \boldsymbol{\epsilon}_\omega \quad (129)$$

$$\dot{\mathbf{x}} = A\mathbf{x} + B\boldsymbol{\epsilon}_\omega \quad (130)$$

where $X_{ag} = [a_C^\times, \mathbf{O}_{3 \times 9}]$ couples the gyro and accelerometer errors.

Example

Next, we describe our example problem, then use the generalized covariance method to analyze the performance and sensitivities of the inertial navigation system.

Description of Example Problem

In our example, we assume that a global navigation satellite system type of receiver provides vector measurements of position throughout an entry trajectory, with measurement noise standard deviation three times less than the estimator assumes. Figures 2, 3 and 4 show some salient characteristics of the example. The simulation begins after the de-orbit burn, and one sees in Fig. 2 that the vehicle reaches a peak deceleration of about $17 \times$ Earth's surface gravity about half-way

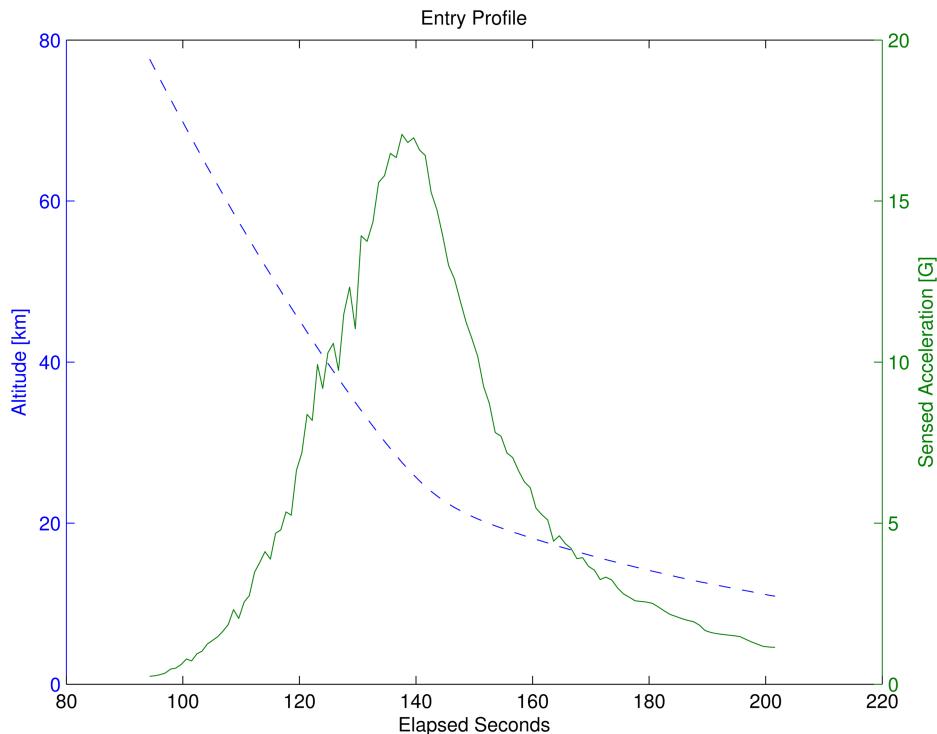
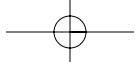


FIG. 2. Entry Profile.



IMU Model Acceleration Errors

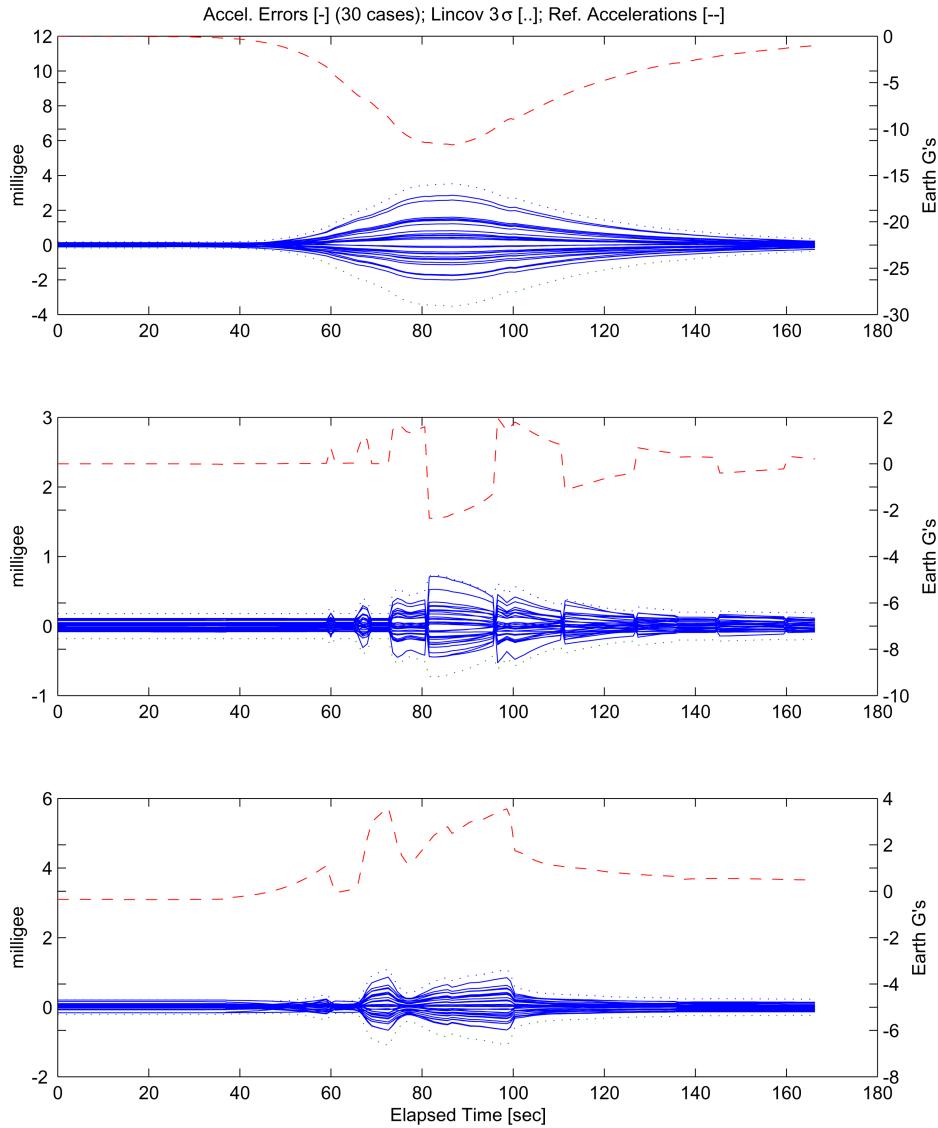
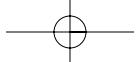


FIG. 3. Monte Carlo Simulation of Case Coordinates of Uncorrected Accelerometer Errors (Left Scale, Solid Blue Lines) and Their Formal 3σ Values (Left Scale, Dotted Blue Lines), Shown with Reference Accelerations (Right Scale, Dashed Red Lines); the x , y , and z Coordinates are Shown in the Top, Middle, and Bottom Subplots, Respectively.

through the entry, at an altitude of about 25 km and about 140 sec elapsed time. The vehicle executes 14 bank maneuvers, which are a combination of rolls and yaws, as the top and bottom plots of Fig. 4 show, so as to control its landing point. The middle and bottom plots of Fig. 3 show accelerations from side forces created by the vehicle's lift during these bank maneuvers.



IMU Model Gyro Rate Errors

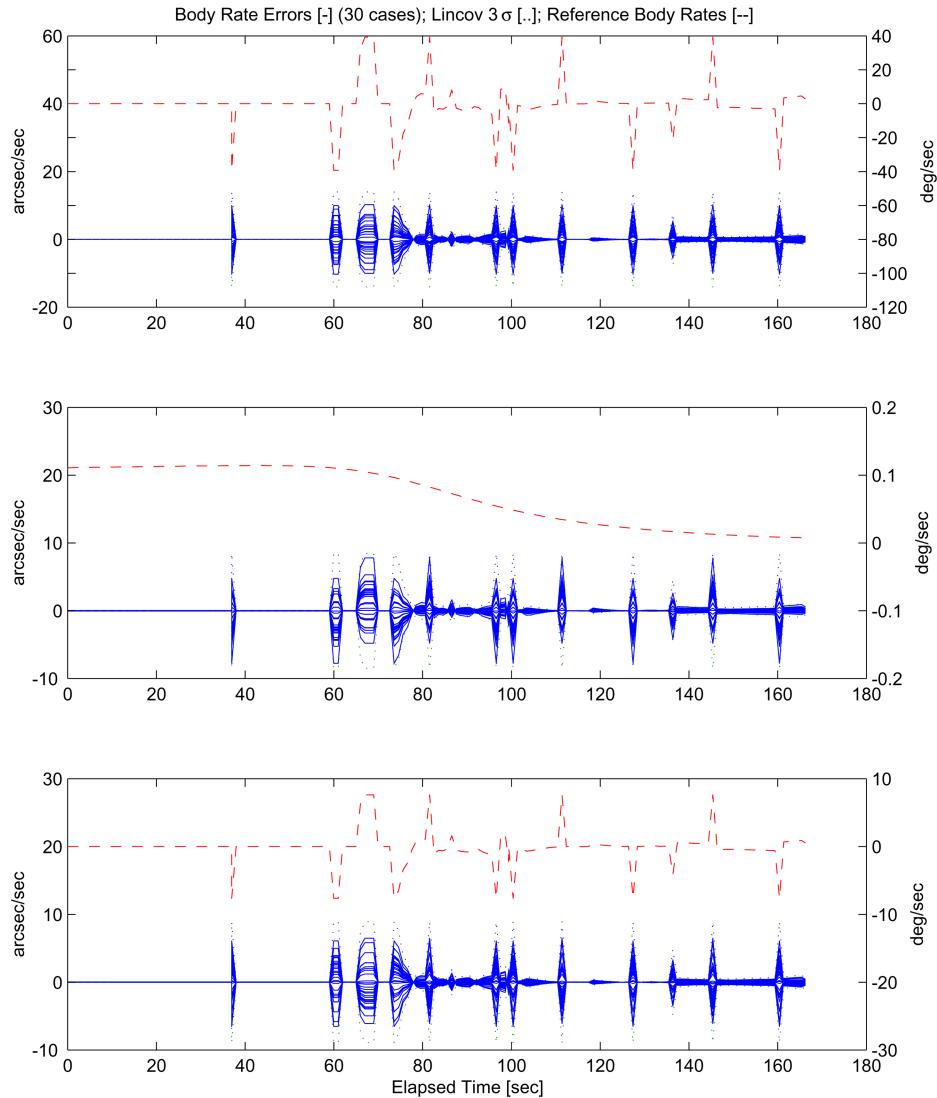
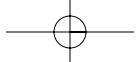


FIG. 4. Monte Carlo Simulation of Case Coordinates of Uncorrected Gyro Rate Errors (Left Scale, Solid Blue Lines) and Their Formal 3σ Values (Left Scale, Dotted Blue Lines), Shown with Reference Body Rates (Right Scale, Dashed Red Lines); the x , y , and z Coordinates are Shown in the Top, Middle, and Bottom Subplots, Respectively.

A Monte Carlo simulation produced the gyro and accelerometer error data that Figs. 3 and 4 also show. This simulation assumed that the elements of $\theta_C(t_o)$, \mathbf{b}_{gC} , \mathbf{s}_g , $\boldsymbol{\gamma}_g$, \mathbf{b}_{aC} , \mathbf{s}_a , $\boldsymbol{\gamma}_a$ and $\boldsymbol{\epsilon}_\omega$ are zero mean, timewise uncorrelated, Gaussian-distributed random processes, and are uncorrelated with one another. Table 1 lists the relevant simulation and estimation parameters that we used.

**TABLE 1.** Simulation and Estimation Parameters

Simulation Parameter	Value	Units
Gravitational Constant	4.305×10^4	km^3/sec^2
Measurement Time Interval	2	sec
Estimation Parameter	Standard Deviation	Units
True Position Measurement Noise	0.305	m
Formal Position Measurement Noise	0.914	m
Initial Position Error	30.5	meters
Initial Velocity Error	3.05	cm/sec
Accelerometer Bias	60	μg
Accelerometer Scale Factor	500	ppm
Accelerometer Nonorthogonality	10	ppm
Initial Gyro Angular Error	42	arcsec
Gyro Bias Drift	0.01	deg/hr
Gyro Scale Factor	33	ppm
Gyro Nonorthogonality	20	ppm
Gyro Random Walk Intensity	0.158	deg/hr ^{1/2}

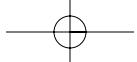
Navigation Analysis of Sample Problem

In using any linear covariance analysis method, we always recommend at least one actual simulation as a check against linearization problems. In the present work, we generate random deviations from the reference as initial conditions for each of 25 Monte Carlo cases. We integrate each deviated case, and use this as truth for measurement simulation and estimation error generation. From these, we generate the time series of estimation errors and residuals for each case.

Description of Plots. We discuss a number of plots below, and it will be helpful to describe our plot conventions in some detail before delving into the results themselves.

A typical output of Monte Carlo simulations is the ensemble of the time histories of the true estimation errors, along with their formal errors, which are the standard deviations from the diagonals of the estimator's covariance matrix. Figure 6 is an example of such a result. In this and similar figures to follow, we plot the true error time series as a sheaf of yellow lines, and the ensemble mean of the true errors as a single blue line. We plot the empirical ensemble deviation of the true errors as a thick cyan band that covers ± 1 to $\pm 3\sigma$'s. We show twice the true standard deviation from the linear covariance analysis as a series of black "+" marks. We show twice the ensemble mean of the formal standard deviations from the Monte Carlo analysis as a series of green "+" marks. We compute a 95% confidence interval based on a hypothesis test of the variance of the Monte Carlo results against the true variance from the linear covariance analysis, which we show as a translucent red band bounded by red "+" marks. These sorts of results are available from existing linear covariance and Monte Carlo simulation procedures.

We have chosen to illustrate the capability of the generalized covariance method to partition the covariance using what we call "variance sandpiles," which are stacked area charts showing the time series of each solve-for variance's contribution from *a priori* error variance, measurement noise variance, and process noise variance. Figure 9 is an example of such a result. Here, we plot the components of the true variance as a positive sandpile, and the components of the formal variance



Generalized Linear Covariance Analysis

21

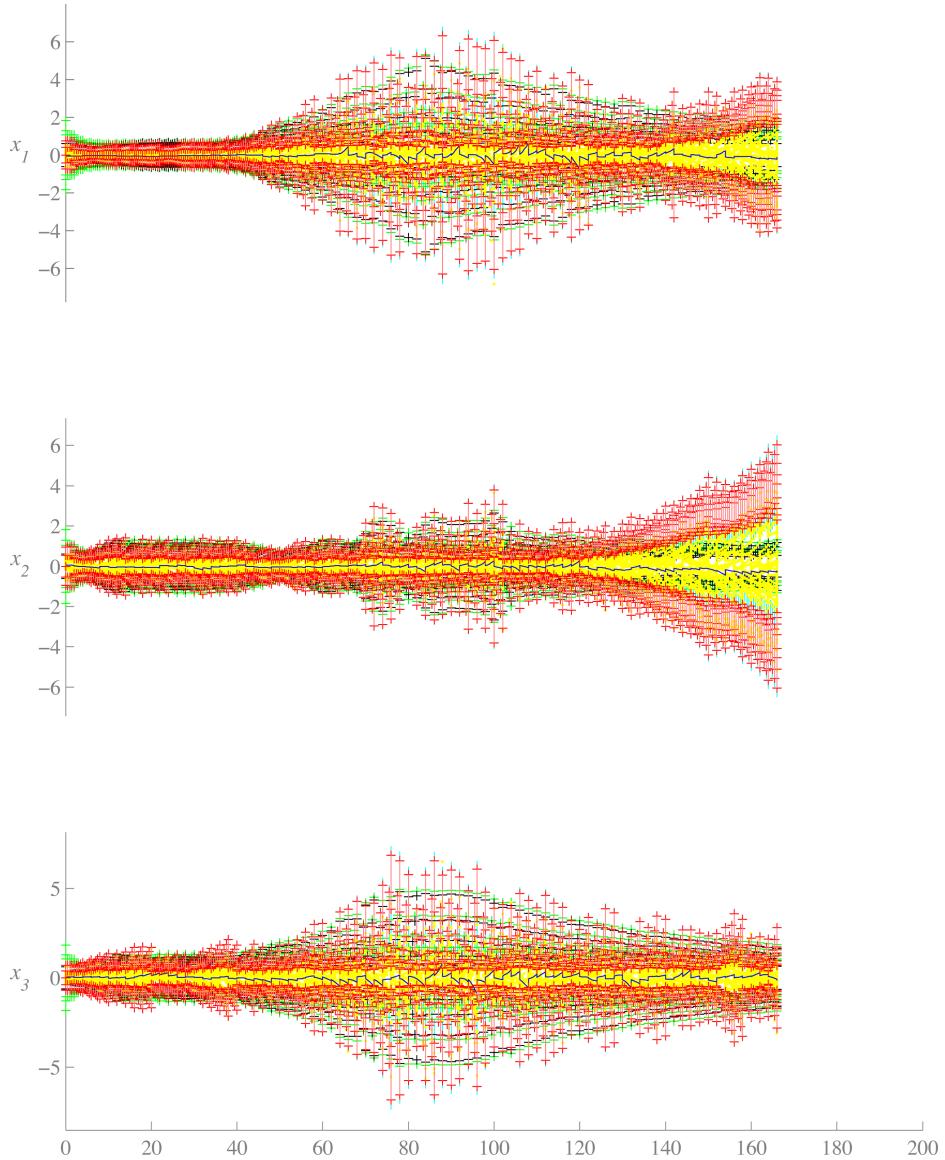


FIG. 5. Position Errors (m) vs. Elapsed Time (sec) from 25-case Monte-Carlo Simulation.

as an inverted sandpile on the negative ordinate, and relabel the negative ordinate to indicate this.⁹

One of several insights that plots such as Fig. 9 convey is a sense of the degree to which the estimator is able to use its measurements to improve its *a priori* knowledge of a given state. As Maybeck [9] describes, for states which are observable, *a priori* information is “forgotten” as more data are processed. Thus, one

⁹Depending on the relative magnitudes of these results, we could also illustrate the sandpiles in other ways, e.g. if all the delta variances were positive, the sandpile could show the formal variance, and the deltas due to each component. If all the deltas were negative, the sandpile could show the true variance, and negatives of the deltas.

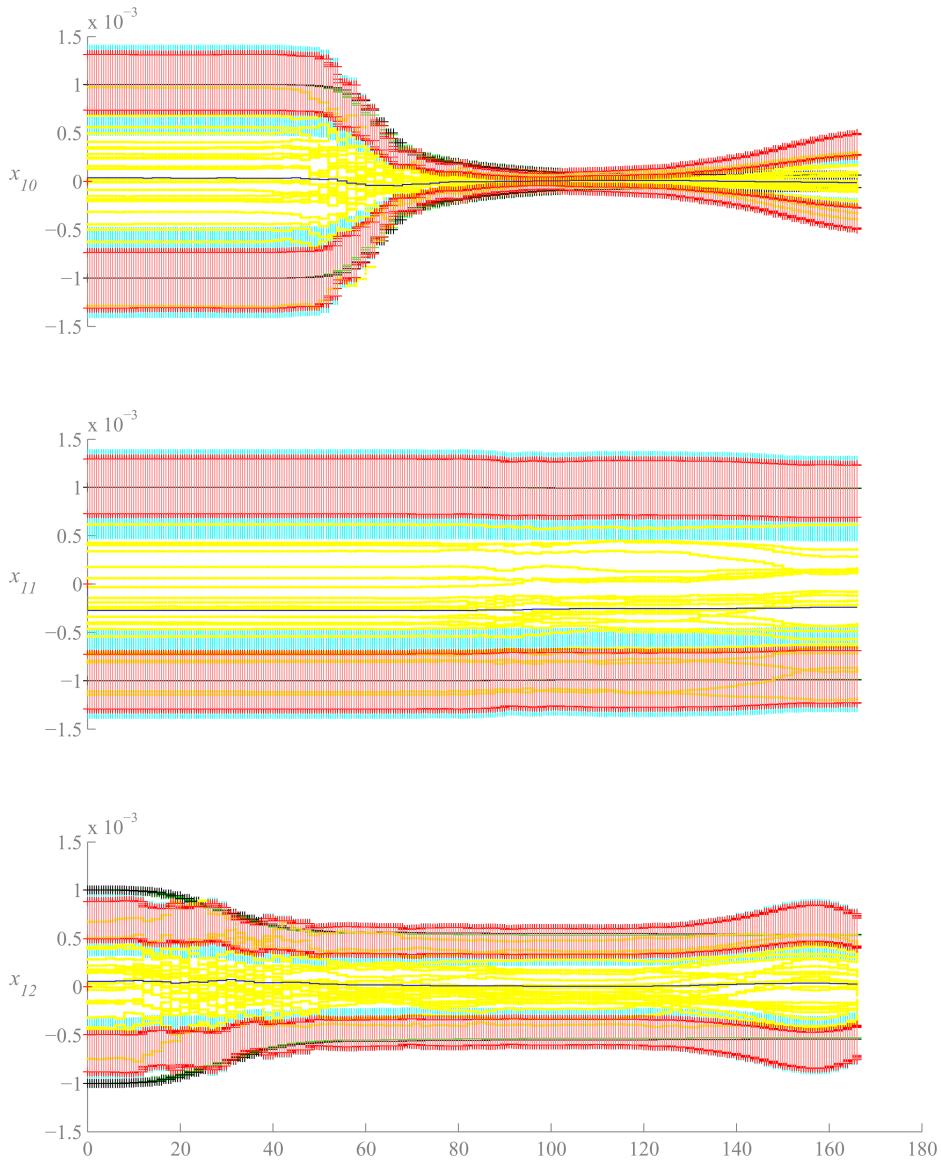
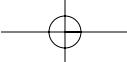
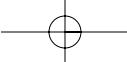


FIG. 6. Accelerometer Scale Factor Errors (dimensionless) vs. Elapsed Time (sec) from 25-case Monte-Carlo Simulation.

may infer from plots such as Fig. 9 whether or not and at what rate the filter is able to “forget” the *a priori* variance for a given state. If no decrease in the *a priori* variance occurs, one has a strong clue that the state in question may not be observable. If the *a priori* variance decreases, but does not appear to be decreasing asymptotically, one has a clue that the state in question may represent a combination of some other states, only a subset of which are observable. We may conclude that such a state is “weakly observable.” Conversely if we observe a rapid, apparently asymptotic decrease in the *a priori* variance we may conclude such a state is “strongly observable.” Although such distinctions are subjective, they are often



Generalized Linear Covariance Analysis

23

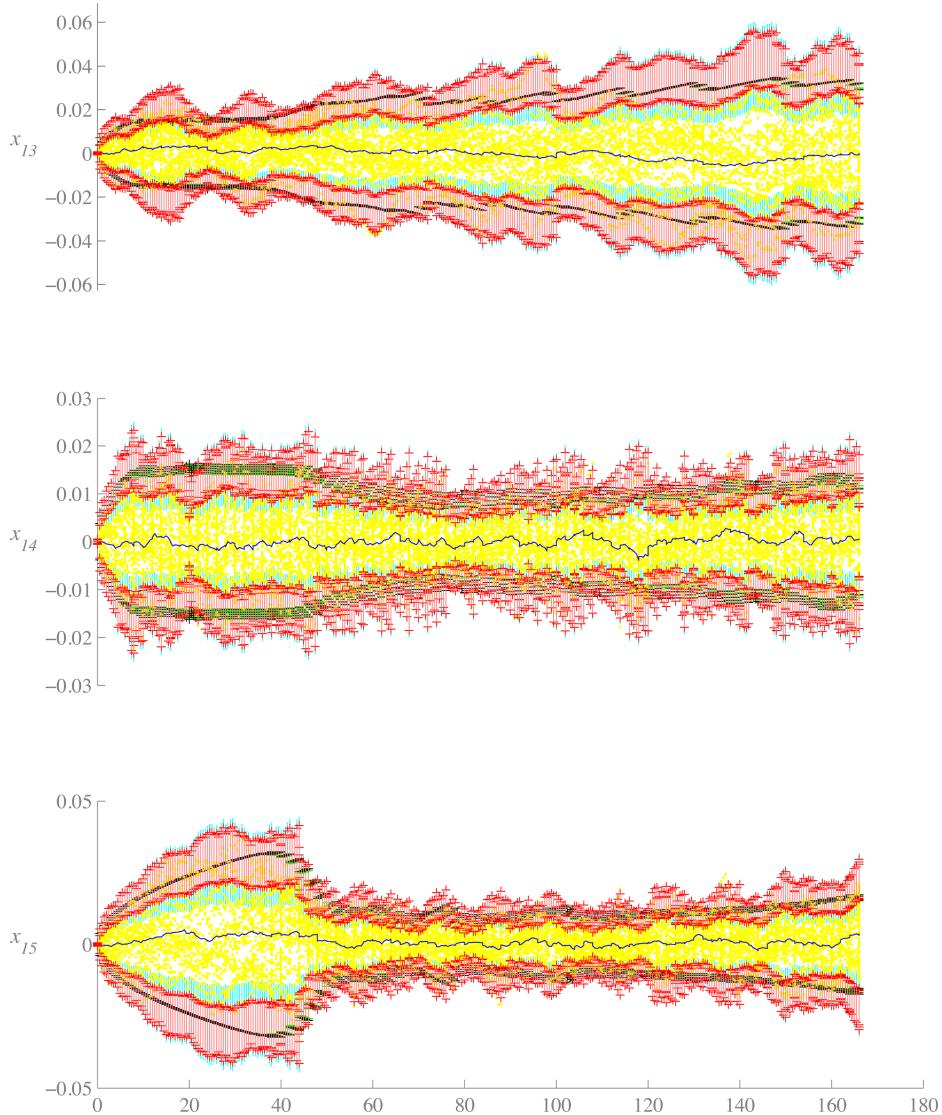
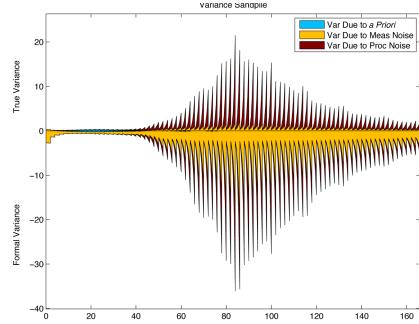
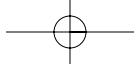
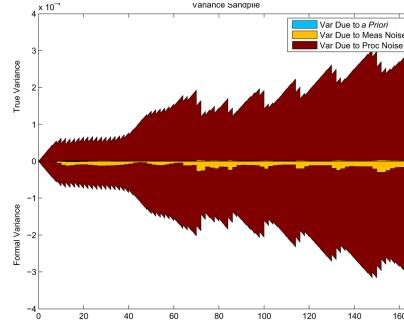
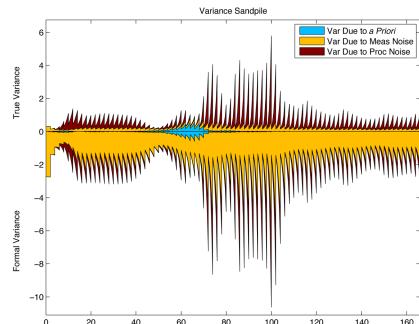
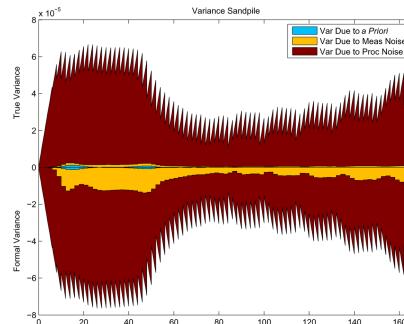
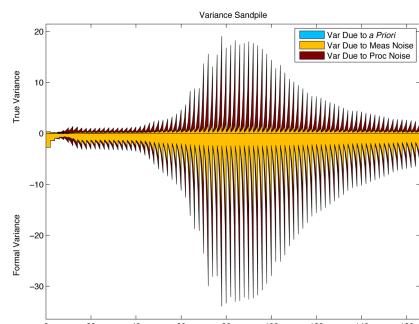
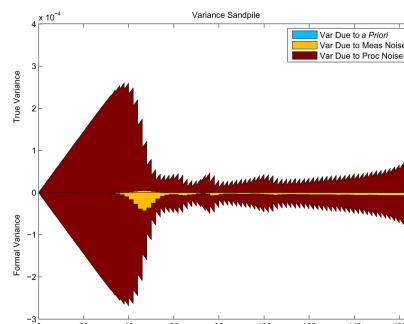
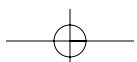


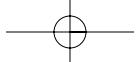
FIG. 7. Gyro Angular Errors (rad) vs. Elapsed Time (sec) from 25-case Monte-Carlo Simulation.

more useful in practice than strict observability conditions based on grammian rank, which apply to the entire solve-for vector, and which may be difficult to conclusively compute due to numerical issues.

Another useful result of many types of covariance analysis is an error budget showing the sensitivity of each solve-for parameter to errors in the consider parameters. We show such results as “sensitivity mosaics,” which are checkerboard plots of the log magnitude of the elements of the sensitivity matrices. Figure 10 is an example of such a plot.

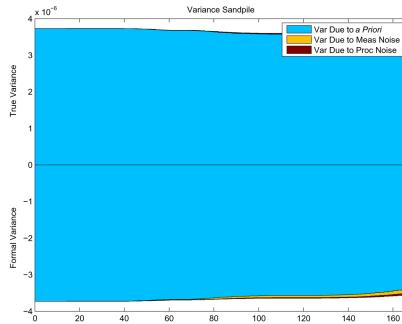
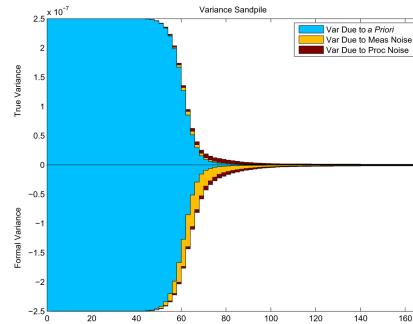
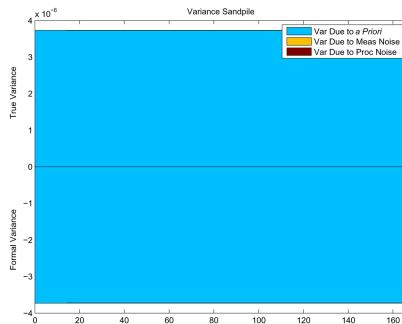
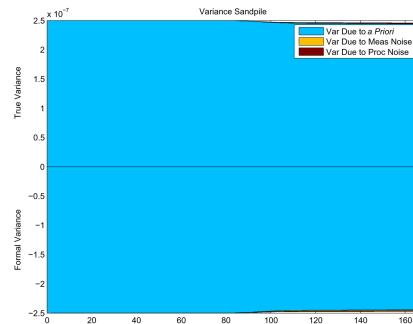
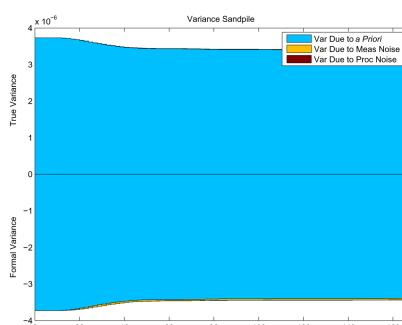
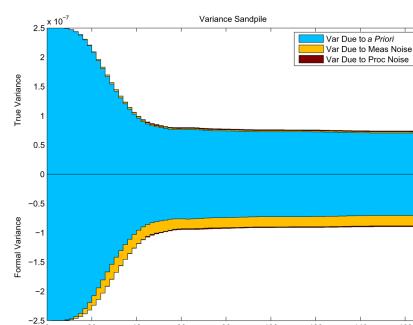
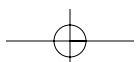
Results. We performed a number of studies of our example problem using the generalized method, which we briefly summarize before discussing our final results.

(a) X -component of Inertial Position Variance (m^2)(b) X -comp. of Case-Fixed Gyro Angular Variance (rad^2)(c) Y -component of Inertial Position Variance (m^2)(d) Y -comp. of Case-Fixed Gyro Angular Variance (rad^2)(e) Z -component of Inertial Position Variance (m^2)(f) Z -comp. of Case-Fixed Gyro Angular Variance (rad^2)**FIG. 8.** Position Error (a, c, e) and Gyro Angular Error (b, d, f) Variance Sandpiles (vs. Elapsed Seconds).



Generalized Linear Covariance Analysis

25

(a) X -comp. of Case-Fixed Accelerometer Bias (m/s^2)(b) X -comp. of Case-Fixed Accelerometer Scale Factor(c) Y -comp. of Case-Fixed Accelerometer Bias (m/s^2)(d) Y -comp. of Case-Fixed Accelerometer Scale Factor(e) Z -comp. of Case-Fixed Accelerometer Bias (m/s^2)(f) Z -comp. of Case-Fixed Accelerometer Scale Factor**FIG. 9.** Accelerometer Bias (a, c, e) and Scale Factor (b, d, f) Variance Sandpiles (vs. Elapsed Seconds).

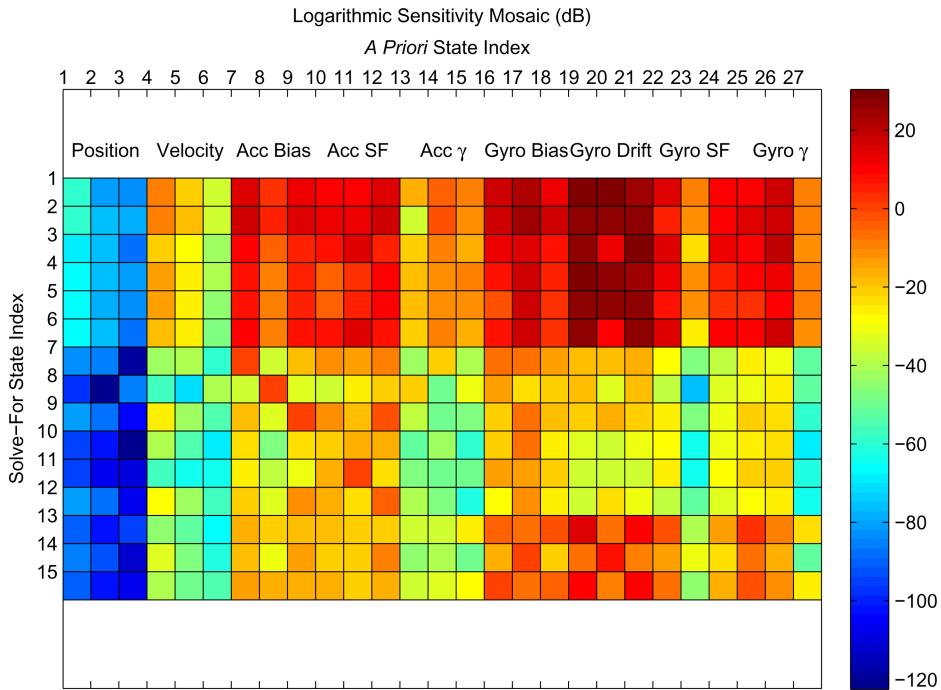
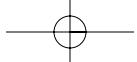
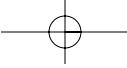


FIG. 10. Sensitivity of final solve-for errors to variations in *a priori* parameter specifications. The solve-for indices are as follows: 1–3, position; 4–6, velocity; 7–9, accelerometer bias; 10–12, accelerometer scale factor; 13–15, gyro angular error.

First, we solved for all of the states in the INS model to study the observability of the various states. From this work, we concluded that accelerometer bias was weakly observable and accelerometer scale factor was strongly observable along the two directions generally normal to the drag acceleration. Accelerometer nonorthogonality and all of the gyro rate errors were not observable. Gyro misalignment was strongly observable, along with position and velocity. Based on these results, we next examined various combinations of observable states as solve-for parameters. Among these combinations, we studied (1) solving for position, velocity, accelerometer bias and scale factor, and gyro misalignment; (2) solving for only position and velocity; and (3) solving for linear combinations of the observable accelerometer and gyro states. The latter was particularly easy with the generalized formulation, since we needed only to modify the S and C matrices. From this work, we concluded that the best performance was available from case (1). Figures 5–10 show the results from this case we obtained with the sequential form of the method. Finally, we studied this case using the batch estimator. Although this produced comparable results, we found that the large size of the i matrix, by which the batch form captures the effect of process noise, caused the batch analysis to take considerably longer to run.

Discussion. There are several notable features of Fig. 5, which shows the Monte Carlo and linear covariance results for the position states.¹⁰ The agreement between the true standard deviation from the linear covariance analysis (black + marks) and the ensemble mean of the formal standard deviations from the Monte

¹⁰Velocity results are generally similar to position, so we omit them for brevity.



Carlo analysis (green + marks) indicates that nonlinearities are not significantly affecting the results. These are bounded by the Monte Carlo confidence interval (red band), indicating that the null hypothesis, which is that the Monte Carlo cases actually are governed by the moments of the linear covariance analysis, cannot be rejected at the 95% confidence level. The large sawtooth increases in the errors between measurement updates lead us to suspect that process noise, which only comes from gyro random walk in this example, is a strong factor influencing performance.

The position variance sandpiles (left column of Fig. 8) allow us to confirm that the spikes are due to process noise. They also show that measurement noise is a significant contributor to the formal variance, but not the true variance (recall that the simulation was run with three times less measurement noise on the true measurements than the filter assumed). Referring to the sensitivity mosaic in Fig. 10, we can see that the IMU parameters, especially gyro drift, are significant contributors to position and velocity errors.

Considering now the accelerometer and gyro errors in Figs. 6–7, we see that the z -component of the accelerometer scale factor error somewhat exceeds the 95% confidence limit during the initial convergence of the filter, possibly indicating stronger nonlinear effects not captured by the linear covariance analysis (Fig. 6). Once the z -component of the gyro angular error converges after $t = 40$ (Fig. 7), the confidence interval violation for the accelerometer scale vector stops, suggesting that the cross-product between gyro angular error and sensed acceleration is playing a role. The sandpiles for the weakly observable accelerometer parameter variances (Fig. 9) are dominated by *a priori* error, while the strongly observable gyro angular error variance sandpile (right column of Fig. 8) is dominated by process and measurement noise, as with the other strongly observable states.

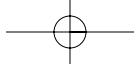
Summary and Conclusion

We have updated and extended previous work on linear covariance analysis and used our generalized method to study the navigation performance of an inertial navigation system during an entry trajectory. As part of this work, we clarified some issues concerning the equations of motion for modeling attitude errors.

We showed that sensing drag acceleration during the entry, along with position measurements from a GPS-like device, allows the INS to update its attitude error states. We found that the linear covariance analysis results agreed well with a 25-case Monte Carlo simulation, indicating the lack of any strongly nonlinear effects in our example problem. Using the unique capability of the generalized method to partition the errors, we were able to *quantify* the extent to which process and measurement noise dominate the performance in estimating the strongly observable states, and to quantify the extent to which *a priori* error strongly affects the performance in estimating the weakly observable states. We illustrated the sensitivity of the INS to various unobservable parameters, and showed that position and velocity errors are strongly sensitive to gyro bias.

Acknowledgments

Lee Bryant of the Johnson Space Center produced the reference data for the entry simulation. Chunlei Rui of Northrop Grumman initiated the discussion of attitude error modeling which caused us to clarify our thinking on this matter. We also thank the reviewers, particularly Reviewer 2, whose comments have led to a greatly improved paper.



References

- [1] GAUSS, K. F. *Theory of the Motion of the Heavenly Bodies Moving About the Sun in Conic Sections*, Dover, New York, 1963.
- [2] KALMAN, 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.
- [3] FAGIN, S. L. "Recursive Linear Regression Theory, Optimal Filter Theory, and Error Analysis of Optimal Systems," *1964 IEEE International Convention Record*, 1964, pp. 216–240.
- [4] JAZWINSKI, A. H. *Stochastic Processes and Filtering Theory*, Academic Press, 1970.
- [5] SCHMIDT, S. F. "Application of State-Space Methods to Navigation Problems," *Advances in Control Systems: Theory and Applications*, (edited by LEONDES, C. T.), Academic Press, New York, Vol. 3, 1966, pp. 293–340.
- [6] BROWN, R. G. and HWANG, P. Y. *Introduction to Random Signals and Applied Kalman Filtering*, John Wiley and Sons, Inc., New York, NY, 3rd ed., 1997.
- [7] MARKLEY, F. L., SEIDEWITZ, E., and NICHOLSON, M. "A General Model for Attitude Determination Error Analysis," *Proceedings of the 1998 Flight Mechanics/Estimimation Theory Symposium*, May 1988, pp. 3–25.
- [8] MARKLEY, F. L., SEIDEWITZ, E., and DEUTSCHMANN, J. "Attitude Determination Error Analysis: General Model and Specific Application," *Proceedings of the CNES Space Dynamics Conference*, Toulouse, France, November 1989, pp. 251–266.
- [9] MAYBECK, P. S. *Stochastic Models, Estimation, and Control*, Vol. 1, Academic Press, New York, 1979.
- [10] GELB, A. (editor) *Applied Optimal Estimation*, The M.I.T. Press, 1974.
- [11] BIERMAN, G. J. *Factorization Methods for Discrete Sequential Estimation*, Academic Press, New York, 1977.
- [12] TAPLEY, B. D., SCHUTZ, B. E., and BORN, G. R. *Statistical Orbit Determination*, Academic Press, 2004.
- [13] MCREYNOLDS, S. "Covariance Projection Methods for Estimators," presented as paper AIAA 96-3739 at the AIAA Guidance, Navigation and Control Conference, American Institute of Aeronautics and Astronautics, Inc., July 1996.
- [14] RIEGSECKER, D. "On-Orbit Attitude Errors," Memorandum E43-96-105, C. S. Draper Laboratory, July 25, 1996.
- [15] BOSE, S. C. "Integrated Navigation Systems (GPS/INS)," Short course lecture notes.
- [16] "Mathematical Models of Attitude Hardware," *Spacecraft Attitude Determination and Control*, (edited by WERTZ, J. R.), Kluwer Academic Publishers, Boston, 1978.
- [17] "Inertial Measurement Unit," *Onboard Navigation System Characteristics*, JSC-26289, NASA Johnson Space Center, June 1993.
- [18] GREWAL, M., HENDERSON, V., and MIYASAKO, R. "Application of Kalman Filtering to the Calibration and Alignment of Inertial Navigation Systems," *IEEE Transactions on Automatic Control*, Vol. 36, No. 1, Jan 1991, pp. 4–12.