# RECURSIVE IMPLEMENTATIONS OF THE SCHMIDT-KALMAN 'CONSIDER' FILTER

**Renato Zanetti*[*] and Christopher D'Souza[†]**

One method to account for parameters errors in the Kalman filter is to 'consider' their effect in the so-called Schmidt-Kalman filter. This paper addresses issues that arise when implementing a consider Kalman filter as a real-time, recursive algorithm. A favorite implementation of the Kalman filter as an onboard navigation subsystem is the UDU formulation. A new way to implement a UDU Schmidt-Kalman filter is proposed. The non-optimality of the recursive Schmidt-Kalman filter is also analyzed, and a modified algorithm is proposed to overcome this limitation.

## INTRODUCTION

In the mid-1960s, S.F. Schmidt introduced a variant of the Kalman Filter, labeled the Schmidt-Kalman filter as means to account for – to *consider* the effect of – errors in both the dynamic and measurement models due to uncertain parameters [1]. The *consider* Kalman filter, also called the Schmidt-Kalman filter resulted from this body of work. The Schmidt-Kalman filter is especially useful when parameters have low observability or when the extra computational power to estimate them is not deemed necessary [2].

Schmidt's approach was based on minimum variance estimation, Jazwinski [3] details the derivation of the optimal filter when some elements of the state vector are not estimated (i.e. they are *considered*). Jazwinski's conclusion is that the Schmidt-Kalman filter is the optimal solution. In Section 8.2 of his book, Bierman [4] disputes the optimality of the Schmidt-Kalman filter, at least in its sequential implementation form. Nevertheless the Schmidt-Kalman filter has received considerable attention in recent years. Tapley *et al.* [5] give an ample description of how to include the contributions of non-estimated states into the Kalman filter algorithm; they provide a different methodology than Jazwinski and arrive to a different formulation. Woodbury *et al.* provide new insight into consider parameters in the measurement model [6]. Equivalent formulations to the Schmidt-Kalman filter were also studied [7, 8] and applied to Mars entry navigation [9] and orbit determination [10].

While the Schmidt-Kalman filter is fairly well known and has been used in covariance analysis, not much attention has been given to actual implementations of it in a real-time recursive estimation algorithm. Onboard estimators commonly utilize the UDU formulation, which guarantees symmetry and positive definiteness of the covariance matrix. Yet, to date

[*] Senior Member of the Technical Staff, Vehicle Dynamics and Controls, The Charles Stark Draper Laboratory, 17629 El Camino Real, Suite 470, Houston, Texas, 77058. rzanetti@draper.com

[†] Deputy Chief, GN&C Autonomous Flight Systems Branch, Aeroscience and Flight Mechanics Division, NASA Johnson Space Center EG6, 2101 NASA Parkway, Houston, Texas, 77058. chris.dsouza@nasa.gov

there has been no direct way of including a Schmidt-Kalman update into the UDU formulation. This paper provides a simple algorithm to obtain this capability. Another common real-time practice is to process measurements one at a time (which is inevitable when utilizing the UDU algorithm). In applying this technique in Schmidt-Kalman filter paradigm and in the presence of nonlinearities, the order in which they are processed does affect the final result. This work analyzes this phenomenon and proposes a novel *globally optimal recursive* Schmidt-Kalman filter, which addresses the objections raised by Bierman.

The paper is organized as follows: in the next section the Kalman filter update equations are reviewed in order to set the stage for the Schmidt-Kalman filter update in the following section. The so-called UDU covariance parameterization is then presented followed by a discussion of it's implementation in the Schmidt-Kalman filter. Bierman's objections to the optimality of the recursive Schmidt-Kalman update are then discussed and a new *globally optimal recursive* Schmidt-Kalman filter is proposed. A numerical example is then introduced before a few concluding remarks.

## THE KALMAN FILTER UPDATE

To begin, the assumption of linear measurements and linear dynamics is invoked; the extension to the nonlinear case can be readily obtained using standard extended Kalman filter (EKF) techniques. Let $\mathbf{y}$ be a set of measurements of a state vector $\mathbf{x}$ corrupted by zero mean noise $\boldsymbol{\eta}$ with covariance $\mathbf{R}$

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\eta}, \tag{1}$$

where $\mathbf{H}$ is the measurement mapping (or sensitivity) matrix. Let $\hat{\mathbf{x}}^-$ be an unbiased *a priori* estimate of $\mathbf{x}$ with corresponding estimation error covariance matrix given by $\mathbf{P}^-$. The *a priori* estimation error is defined as $\mathbf{e}^- = \mathbf{x} - \hat{\mathbf{x}}^-$. The unbiased linear update based upon $\hat{\mathbf{x}}^-$ and $\mathbf{y}$ produces the *a posteriori* estimate given by

$$\hat{\mathbf{x}}^+ = \hat{\mathbf{x}}^- + \mathbf{K}\left(\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}^-\right), \tag{2}$$

where $\mathbf{K}$ is some deterministic matrix of appropriate dimensions yet to be determined. The *a posteriori* estimation error is expressed as

$$\mathbf{e}^+ = \mathbf{x} - \hat{\mathbf{x}}^+ = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{e}^- - \mathbf{K}\boldsymbol{\eta}. \tag{3}$$

Assuming that the measurement error $\boldsymbol{\eta}$ and the *a priori* estimation error, $\mathbf{e}^-$, are uncorrelated and each are zero mean, the *a posteriori* estimation error covariance is given by the so-called Joseph formula [11]:

$$\mathbf{P}^+ = \mathrm{E}\left\{\left(\mathbf{x} - \hat{\mathbf{x}}^+\right)\left(\mathbf{x} - \hat{\mathbf{x}}^+\right)^{\mathrm{T}}\right\} = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}^-(\mathbf{I} - \mathbf{K}\mathbf{H})^{\mathrm{T}} + \mathbf{K}\mathbf{R}\mathbf{K}^{\mathrm{T}}. \tag{4}$$

Defining the covariance of the measurement residual, $\mathbf{W}$, as $\mathbf{W} \triangleq \mathbf{H}\mathbf{P}^-\mathbf{H}^{\mathrm{T}} + \mathbf{R}$, the updated (*a posteriori*) covariance is equivalently written as

$$\mathbf{P}^+ = \mathbf{P}^- - \mathbf{K}\mathbf{H}\mathbf{P}^- - \mathbf{P}^-\mathbf{H}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}} + \mathbf{K}\mathbf{W}\mathbf{K}^{\mathrm{T}}. \tag{5}$$

Notice that up to this point no assumptions have been made regarding the choice of $\mathbf{K}$; the Joseph update equation is valid for all $\mathbf{K}$, as is Eq. (5). The standard Kalman gain $\mathbf{K}_{opt}$ minimizes the trace of the updated covariance matrix, $\mathbf{P}^{+}$, is found to be

$$\mathbf{K}_{opt} = \mathbf{P}^{-}\mathbf{H}^{\mathrm{T}}\mathbf{W}^{-1}. \tag{6}$$

Substituting this into Eq. (5) it follows that

$$\mathbf{P}^{+}_{opt} = \mathbf{P}^{-} - \mathbf{K}_{opt}\mathbf{H}\mathbf{P}^{-} = \mathbf{P}^{-} - \mathbf{K}_{opt}\mathbf{W}\mathbf{K}^{\mathrm{T}}_{opt} = \mathbf{P}^{-} - \mathbf{P}^{-}\mathbf{H}^{\mathrm{T}}\mathbf{W}^{-1}\mathbf{H}\mathbf{P}^{-} \tag{7}$$

These equations are only valid for the optimal gain, $\mathbf{K}_{opt}$. Thus, the Kalman filter update equations are Eq. (2), Eq. (6), and Eq. (7). These will serve as a foundation for what follows. One strategy for Kalman filtering is to implement these update equations directly. This is a rather brute-force method and makes no distinctions between dynamic states (such as position, velocity and attitude) and slowly varying states (sensor biases, etc). Hence it tends to be rather inefficient numerically and computationally.

## THE SCHMIDT-KALMAN FILTER UPDATE

Suppose that the system under consideration contains dynamic states and sensor states, and suppose it is desired to only estimate the dynamic states, yet *consider* the effect of the sensor states on the dynamic states. In such a situation, $\mathbf{x}$ is now partitioned into $n_{\mathbf{s}}$ “estimated” states, $\mathbf{s}$, and $n_{\mathbf{p}}$ “consider” parameters, $\mathbf{p}$, as

$$\mathbf{x}^{\mathrm{T}} \triangleq \begin{bmatrix}\mathbf{s}^{\mathrm{T}} & \mathbf{p}^{\mathrm{T}}\end{bmatrix}, \tag{8}$$

Thus,

$$\mathbf{P} = \begin{bmatrix}\mathbf{P}_{\mathrm{ss}} & \mathbf{P}_{\mathrm{sp}} \\ \mathbf{P}_{\mathbf{ps}} & \mathbf{P}_{\mathbf{pp}}\end{bmatrix}, \qquad \mathbf{H} = \begin{bmatrix}\mathbf{H}_{\mathrm{s}} & \mathbf{H}_{\mathbf{p}}\end{bmatrix} \tag{9}$$

$$\mathbf{K}_{opt} = \begin{bmatrix}\mathbf{K}_{\mathbf{s},opt} \\ \mathbf{K}_{\mathbf{p},opt}\end{bmatrix} = \begin{bmatrix}\mathbf{P}^{-}_{\mathbf{ss}}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{P}^{-}_{\mathbf{sp}}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} \\ \mathbf{P}^{-}_{\mathbf{ps}}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{P}^{-}_{\mathbf{pp}}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}}\end{bmatrix}\mathbf{W}^{-1}. \tag{10}$$

As in the previous section

$$\mathbf{W} = \mathbf{H}\mathbf{P}^{-}\mathbf{H}^{\mathrm{T}} + \mathbf{R} = \mathbf{H}_{\mathbf{s}}\mathbf{P}^{-}_{\mathbf{ss}}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{s}}\mathbf{P}^{-}_{\mathbf{sp}}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{-}_{\mathbf{ps}}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{-}_{\mathbf{pp}}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{R}$$

The updated portions of the covariance are

$$\mathbf{P}^{+}_{\mathbf{ss}} = \mathbf{P}^{-}_{\mathbf{ss}} - \mathbf{K}_{\mathbf{s}}\mathbf{H}\begin{bmatrix}\mathbf{P}^{-}_{\mathbf{ss}} \\ \mathbf{P}^{-}_{\mathbf{ps}}\end{bmatrix} - \begin{bmatrix}\mathbf{P}^{-}_{\mathbf{ss}} \\ \mathbf{P}^{-}_{\mathbf{ps}}\end{bmatrix}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{K}_{\mathbf{s}}\mathbf{W}\mathbf{K}^{\mathrm{T}}_{\mathbf{s}} \tag{11}$$

$$\mathbf{P}^{+}_{\mathbf{ps}} = (\mathbf{P}^{+}_{\mathbf{sp}})^{\mathrm{T}} = \mathbf{P}^{-}_{\mathbf{ps}} - \mathbf{K}_{\mathbf{p}}\mathbf{H}\begin{bmatrix}\mathbf{P}^{-}_{\mathbf{ss}} \\ \mathbf{P}^{-}_{\mathbf{ps}}\end{bmatrix} - \begin{bmatrix}\mathbf{P}^{-}_{\mathbf{sp}} \\ \mathbf{P}^{-}_{\mathbf{pp}}\end{bmatrix}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{K}_{\mathbf{p}}\mathbf{W}\mathbf{K}^{\mathrm{T}}_{\mathbf{s}} \tag{12}$$

$$\mathbf{P}^{+}_{\mathbf{pp}} = \mathbf{P}^{-}_{\mathbf{pp}} - \mathbf{K}_{\mathbf{p}}\mathbf{H}\begin{bmatrix}\mathbf{P}^{-}_{\mathbf{sp}} \\ \mathbf{P}^{-}_{\mathbf{pp}}\end{bmatrix} - \begin{bmatrix}\mathbf{P}^{-}_{\mathbf{sp}} \\ \mathbf{P}^{-}_{\mathbf{pp}}\end{bmatrix}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{K}_{\mathbf{p}}\mathbf{W}\mathbf{K}^{\mathrm{T}}_{\mathbf{p}} \tag{13}$$

These equations are directly derived from Eq. (5) and are therefore valid for any choice of $\mathbf{K}_\mathrm{s}$ and $\mathbf{K}_\mathbf{p}$.

Substituting for the value of $\mathbf{K}_{\mathbf{s},opt}$ for the three components in Eqs. (11)–(12) and allowing $\mathbf{K}_\mathbf{p}$ to be (as yet) unspecified, $\mathbf{P}^+$ becomes

$$\mathbf{P}^+ = \begin{bmatrix} \mathbf{P}^-_\mathbf{ss} - \mathbf{K}_{\mathbf{s},opt}\mathbf{W}\mathbf{K}^\mathrm{T}_{\mathbf{s},opt} & \mathbf{P}^-_\mathbf{sp} - \mathbf{K}_{\mathbf{s},opt}\mathbf{H}\begin{bmatrix}\mathbf{P}^-_\mathbf{sp}\\ \mathbf{P}^-_\mathbf{pp}\end{bmatrix} \\ \mathbf{P}^-_\mathbf{ps} - \begin{bmatrix}\mathbf{P}^-_\mathbf{sp}\\ \mathbf{P}^-_\mathbf{pp}\end{bmatrix}^\mathrm{T}\mathbf{H}^\mathrm{T}\mathbf{K}^\mathrm{T}_{\mathbf{s},opt} & \mathbf{P}^-_\mathbf{pp} - \mathbf{K}_\mathbf{p}\mathbf{H}\begin{bmatrix}\mathbf{P}^-_\mathbf{sp}\\ \mathbf{P}^-_\mathbf{pp}\end{bmatrix} - \begin{bmatrix}\mathbf{P}^-_\mathbf{sp}\\ \mathbf{P}^-_\mathbf{pp}\end{bmatrix}^\mathrm{T}\mathbf{H}^\mathrm{T}\mathbf{K}^\mathrm{T}_\mathbf{p} + \mathbf{K}_\mathbf{p}\mathbf{W}\mathbf{K}^\mathrm{T}_\mathbf{p} \end{bmatrix}.$$

Once again, it is emphasized that this equation is valid for *any* value of $\mathbf{K}_\mathbf{p}$. Notice that there is no $\mathbf{K}_\mathbf{p}$ in the cross-covariance between $\mathbf{s}$ and $\mathbf{p}$. Therefore, what is remarkable about this equation is that *once the optimal* $\mathbf{K}_{\mathbf{s},opt}$ *is chosen, the cross-covariance between* $\mathbf{s}$ *and* $\mathbf{p}$ *is independent of the choice of* $\mathbf{K}_\mathbf{p}$. We will take advantage of this property in due course.

The updated (*a posteriori*) state is given by

$$\begin{bmatrix}\mathbf{s}^+\\ \mathbf{p}^+\end{bmatrix} = \begin{bmatrix}\mathbf{s}^-\\ \mathbf{p}^-\end{bmatrix} + \begin{bmatrix}\mathbf{K}_\mathbf{s}\\ \mathbf{K}_\mathbf{p}\end{bmatrix}\left(\mathbf{y} - \mathbf{H}_\mathbf{s}\hat{\mathbf{s}}^- - \mathbf{H}_\mathbf{p}\hat{\mathbf{p}}^-\right). \tag{14}$$

The Schmidt-Kalman filter is one in which the parameters, $\mathbf{p}$, are not updated. From Eq. (14) it is observed that this can be achieved by taking $\mathbf{K}_\mathbf{p} = \mathbf{O}$. From the most general covariance update given in Eq. (5), setting $\mathbf{K}_\mathbf{p} = \mathbf{O}$ yields

$$\mathbf{P}^+_{SKF} = \begin{bmatrix} \mathbf{P}^-_\mathbf{ss} - \mathbf{K}_\mathbf{s}\mathbf{H}\begin{bmatrix}\mathbf{P}^-_\mathbf{ss}\\ \mathbf{P}^-_\mathbf{ps}\end{bmatrix} - \begin{bmatrix}\mathbf{P}^-_\mathbf{ss}\\ \mathbf{P}^-_\mathbf{ps}\end{bmatrix}^\mathrm{T}\mathbf{H}^\mathrm{T}\mathbf{K}^\mathrm{T}_\mathbf{s} + \mathbf{K}_\mathbf{s}\mathbf{W}\mathbf{K}^\mathrm{T}_\mathbf{s} & \mathbf{P}^-_\mathbf{sp} - \mathbf{K}_\mathbf{s}\mathbf{H}\begin{bmatrix}\mathbf{P}^-_\mathbf{sp}\\ \mathbf{P}^-_\mathbf{pp}\end{bmatrix} \\ \mathbf{P}^-_\mathbf{ps} - \begin{bmatrix}\mathbf{P}^-_\mathbf{sp}\\ \mathbf{P}^-_\mathbf{pp}\end{bmatrix}^\mathrm{T}\mathbf{H}^\mathrm{T}\mathbf{K}^\mathrm{T}_\mathbf{s} & \mathbf{P}^-_\mathbf{pp} \end{bmatrix} \tag{15}$$

The necessary condition for an optimal Schmidt-Kalman gain, $\mathbf{K}_{\mathbf{s}_{SKF}}$, is given by

$$\frac{\partial}{\partial \mathbf{K}_\mathbf{s}}\left(\mathrm{trace}\left[\mathbf{P}^+_{SKF}\right]\right) = -\begin{bmatrix}\mathbf{P}^-_\mathbf{ss}\\ \mathbf{P}^-_\mathbf{ps}\end{bmatrix}^\mathrm{T}\mathbf{H}^\mathrm{T} - \begin{bmatrix}\mathbf{P}^-_\mathbf{ss}\\ \mathbf{P}^-_\mathbf{ps}\end{bmatrix}^\mathrm{T}\mathbf{H}^\mathrm{T} + \mathbf{K}_\mathbf{s}\mathbf{W}^\mathrm{T} + \mathbf{K}_\mathbf{s}\mathbf{W} = \mathbf{O}. \tag{16}$$

The result obtained from this condition is the same as for the globally optimal Kalman filter:

$$\mathbf{K}_{\mathbf{s}_{SKF}} = \left(\mathbf{P}^-_\mathbf{ss}\mathbf{H}^\mathrm{T}_\mathbf{s} + \mathbf{P}^-_\mathbf{sp}\mathbf{H}^\mathrm{T}_\mathbf{p}\right)\mathbf{W}^{-1}. \tag{17}$$

Therefore the Schmidt-Kalman filter gain can be conveniently calculated from the optimal Kalman filter gain by zeroing out the rows corresponding to the ’consider’ parameters $\mathbf{p}$. With this in hand, the Schmidt-Kalman filter gain is

$$\mathbf{K}_{SKF} = \begin{bmatrix}\mathbf{P}^-_\mathbf{ss}\mathbf{H}^\mathrm{T}_\mathbf{s} + \mathbf{P}^-_\mathbf{sp}\mathbf{H}^\mathrm{T}_\mathbf{p}\\ \mathbf{O}\end{bmatrix}\mathbf{W}^{-1} = \begin{bmatrix}\mathbf{K}_{\mathbf{s}_{SKF}}\\ \mathbf{O}\end{bmatrix} \tag{18}$$

The resulting estimated state update and 'consider' parameters update is

$$\mathbf{s}^{+} = \mathbf{s}^{-} + \mathbf{K}_{\mathbf{s}_{SKF}} \left(\mathbf{y} - \mathbf{H}_{\mathbf{s}}\hat{\mathbf{s}}^{-} - \mathbf{H}_{\mathbf{p}}\hat{\mathbf{p}}^{-}\right) \qquad \mathbf{p}^{+} = \mathbf{p}^{-}. \tag{19}$$

It must be stated emphatically that the full updated covariance matrix (the *a posteriori* matrix) must be calculated by means of the Joseph update. Eq. (7) is not applicable, because that equation is valid only for the optimal Kalman gain; recall that the Schmidt-Kalman gain is a not the globally optimal gain because it sets $\mathbf{K}_{\mathbf{p}} = \mathbf{O}$. Substituting $\mathbf{K}_{\mathbf{s}_{SKF}}$ into Eq. (15) yields

$$\mathbf{P}^{+}_{SKF} = \begin{bmatrix} \mathbf{P}^{-}_{\mathbf{ss}} - \mathbf{K}_{\mathbf{s}_{SKF}} \mathbf{W} \mathbf{K}^{\mathrm{T}}_{\mathbf{s}_{SKF}} & \mathbf{P}^{-}_{\mathbf{sp}} - \mathbf{K}_{\mathbf{s}_{SKF}} \mathbf{H} \begin{bmatrix} \mathbf{P}^{-}_{\mathbf{sp}} \\ \mathbf{P}^{-}_{\mathbf{pp}} \end{bmatrix} \\ \mathbf{P}^{-}_{\mathbf{ps}} - \begin{bmatrix} \mathbf{P}^{-}_{\mathbf{sp}} \\ \mathbf{P}^{-}_{\mathbf{pp}} \end{bmatrix}^{\mathrm{T}} \mathbf{H}^{\mathrm{T}} \mathbf{K}^{\mathrm{T}}_{\mathbf{s}_{SKF}} & \mathbf{P}^{-}_{\mathbf{pp}} \end{bmatrix}. \tag{20}$$

In the Schmidt-Kalman filter, the *a priori* and *a posteriori* covariance of the parameters ($\mathbf{P}_{\mathbf{pp}}$) are the same. At the same time, the *a posteriori* covariance matrix of the states and the cross-covariance between the states and the parameters are the same regardless of whether the Schmidt-Kalman filter or the optimal Kalman filter equations are used.

Therefore, it follows that the *a posteriori* Schmidt-Kalman covariance can be easily obtained from the *a posteriori* optimal covariance by simply replacing $\mathbf{P}^{+}_{\mathbf{pp}}$ with $\mathbf{P}^{-}_{\mathbf{pp}}$. Notice that applying these equations recursively $\mathbf{P}_{\mathbf{ss}}$ from the optimal Kalman filter and the Schmidt-Kalman filter are not the same for all time. Only when given the same *a priori* covariance the two algorithms will produce the same *a posteriori* covariance for $\mathbf{P}_{\mathbf{ss}}$ and $\mathbf{P}_{\mathbf{sp}}$. But since $\mathbf{P}_{\mathbf{pp}}$ is different between the two algorithms, after the very first update, the subsequent *a priori* and *a posteriori* covariances will be different from that time forward, thereby producing different results.

## THE UDU UPDATE

When the dimension of the state-space is small, nothing is lost by implementing the Kalman filter as described above (both the 'classic' Kalman filter and the Schmidt-Kalman filter). As the number of states increases, computational throughput and stability considerations become a cause of concern. Much work has gone into reducing the computational throughput and increasing computational stability of Kalman filter implementations. Two such methods are the so-called Square Root formulations and matrix factorization formulations. Real-time flight software covariance matrices are usually stored/represented using the UDU matrix factorization paradigm[12] decomposed as $\mathbf{P} = \mathbf{U}\mathbf{D}\mathbf{U}^{\mathrm{T}}$, where $\mathbf{U}$ is an upper triangular matrix with ones on the main diagonal, and $\mathbf{D}$ is a diagonal matrix. This section describes one such instantiation of an efficient UDU implementation.

In order to efficiently update $\mathbf{U}$ and $\mathbf{D}$ with measurements, the so-called 'rank-one' update proposed by Agee and Turner is sometimes used [13]. The rank-one (covariance) update requires that measurements be processed one at a time and has the form

$$\mathbf{P}^{+} = \mathbf{P}^{-} + c\mathbf{a}\mathbf{a}^{\mathrm{T}} \tag{21}$$

where $c$ is a scalar and $\mathbf{a}$ is a vector. With $\mathbf{P}^+ = \mathbf{U}^+\mathbf{D}^+(\mathbf{U}^+)^{\mathrm{T}}$ and $\mathbf{P}^- = \mathbf{U}^-\mathbf{D}^-(\mathbf{U}^-)^{\mathrm{T}}$ the algorithm for the rank-one update is backwards-recursive for $\mathbf{U}^+$ and $\mathbf{D}^+$, starting with the $(n, n)$ element ($D_{nn}$) and ending with the $(1, 1)$ element ($D_{11}$) and it is seen in Table 1.

For $j = n, n-1, \ldots, 3, 2$ set $c_n = c$ and recursively calculate:

$$D^+_{jj} = D^-_{jj} + c_j a_j^2$$

$$a_k := a_k - a_j U^-_{kj} \quad k = 1, 2, \ldots j-1$$

$$U^+_{kj} = U^-_{kj} + c_k a_j a_k / D^+_{jj} \quad k = 1, 2, \ldots j-1$$

$$c_{j-1} = c_j D^+_{jj} / D^-_{jj}$$

and finally compute

$$D^+_{11} = D^-_{11} + c_1 a_1^2$$

**Table 1. Backwards-Recursive Rank-One Update**

For scalar measurements $\mathbf{H}$ and $\mathbf{K}$ become vectors and $\mathbf{W}$ is a scalar. From Eq. (7) the *optimal update* of the covariance matrix is

$$\mathbf{U}^+\mathbf{D}^+(\mathbf{U}^+)^{\mathrm{T}} \;\; = \;\; \mathbf{U}^- \left[\mathbf{D}^- - \frac{1}{W}\mathbf{D}^-(\mathbf{U}^-)^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{H}\mathbf{U}^-\mathbf{D}^-\right](\mathbf{U}^-)^{\mathrm{T}}.$$

Defining $\mathbf{a} = \mathbf{D}^-(\mathbf{U}^-)^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}$, $c = -1/W$, $\mathbf{U}^+\mathbf{D}^+(\mathbf{U}^+)^{\mathrm{T}}$ becomes

$$\mathbf{U}^+\mathbf{D}^+(\mathbf{U}^+)^{\mathrm{T}} \;\; = \;\; \mathbf{U}^- \left[\mathbf{I}\mathbf{D}^-\mathbf{I} + c\mathbf{a}\mathbf{a}^{\mathrm{T}}\right](\mathbf{U}^-)^{\mathrm{T}},$$

where $\mathbf{I}$ is the identity matrix (which is upper triangular with ones on the diagonal). A rank-one update of the term in the square bracket is first performed, defining an intermediate step to the solution as

$$\widetilde{\mathbf{U}}\widetilde{\mathbf{D}}\widetilde{\mathbf{U}}^{\mathrm{T}} = \mathbf{D}^- + c\,\mathbf{a}\mathbf{a}^{\mathrm{T}}$$

The updated matrices are given by $\mathbf{U}^+ = \mathbf{U}^-\widetilde{\mathbf{U}}$ and $\mathbf{D}^+ = \widetilde{\mathbf{D}}$. Since $c$ is negative, this measurement update involves the potential loss of precision due to subtraction of two positive numbers which are close to one another. A modified rank-one update due to Carlson [14] results in a numerically stable, forward-recursive algorithm. The caveat is that this modified algorithm is only valid for the optimal Kalman update, $\mathbf{K}_{opt}$. The updated $\mathbf{U}$ and $\mathbf{D}$ ($\mathbf{U}^+$ and $\mathbf{D}^+$, respectively) and the optimal Kalman gain matrix, $\mathbf{K}_{opt}$, are produced by this algorithm, which is detailed in Table 2.

The state update is computed as before, *i.e.* $\mathbf{x}^+ = \mathbf{x}^- + \mathbf{K}_{opt}\left(\mathbf{y} - \mathbf{H}\mathbf{x}^-\right)$.

## THE SCHMIDT-KALMAN FILTER UDU UPDATE

The major goals of real-time filter implementation are efficiency and computational stability; as well, the formulation ought to flexible enough to handle (slight) modifications to

| |
|---|
| Given $\mathbf{H}$, $\mathbf{U}^{-}$, $\mathbf{D}^{-}$, $\mathbf{R} = R$<br>$\mathbf{f} = (\mathbf{U}^{-})^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}, \qquad \mathbf{v} = \mathbf{D}^{-}\,\mathbf{f}$<br>$\bar{\mathbf{K}}_1 = [\ v_1 \quad 0 \quad \cdots \quad 0\ ]^{\mathrm{T}}$<br>$\alpha_1 = R + v_1 f_1, \qquad d_1 = d_{11} = (R/\alpha_1)\,\bar{d}_{11}$ |
| For $j = 2, 3, \ldots, n$, recursively calculate:<br>$\alpha_j = \alpha_{j-1} + v_j f_j, \qquad \lambda_j = -(f_j/\alpha_{j-1})$<br>$d_j = d_{jj} = (\alpha_{j-1}/\alpha_j)\,\bar{d}_{jj}$<br>$\mathbf{U}_j = \mathbf{U}_j^{-} + \lambda_j \bar{\mathbf{K}}_{j-1}, \qquad \bar{\mathbf{K}}_j = \bar{\mathbf{K}}_{j-1} + v_j \mathbf{U}_j^{-}$ |
| $\mathbf{K}_{opt} = \bar{\mathbf{K}}_n/\alpha_n$ |

**Table 2. The Forward-Recursive *Modified* Rank-One Update**

the structure of the problem. Despite the efficiencies of the UDU formulation, it is rather brittle and does not readily meet the desire of flexibility. In point of fact, the structure of the updated consider covariance matrix in Eq. (15) does not allow for a rank-one update as developed earlier. However, a method has been developed to do exactly what we desire – a Schmidt-Kalman UDU Update – at the expense of a modest increase in computation.

Begin with the optimal covariance matrix update (from the original Kalman update, partitioned into 'states' and 'parameters') which is given by

$$\mathbf{P}_{opt}^{+} = \begin{bmatrix} \mathbf{P}_{\mathbf{ss}}^{-} - \mathbf{K}_{\mathbf{s},opt}\mathbf{W}\mathbf{K}_{\mathbf{s},opt}^{\mathrm{T}} & \mathbf{P}_{\mathbf{sp}}^{-} - \mathbf{K}_{\mathbf{s},opt}\mathbf{H}\begin{bmatrix}\mathbf{P}_{\mathbf{sp}}^{-} \\ \mathbf{P}_{\mathbf{pp}}^{-}\end{bmatrix} \\ \mathbf{P}_{\mathbf{ps}}^{-} - \begin{bmatrix}\mathbf{P}_{\mathbf{ps}}^{-} & \mathbf{P}_{\mathbf{pp}}^{-}\end{bmatrix}\mathbf{H}^{\mathrm{T}}\mathbf{K}_{\mathbf{s},opt}^{\mathrm{T}} & \mathbf{P}_{\mathbf{pp}}^{-} - \mathbf{K}_{\mathbf{p},opt}\mathbf{W}\mathbf{K}_{\mathbf{p},opt}^{\mathrm{T}} \end{bmatrix} \tag{22}$$

which can be rearranged to be expressed as

$$\mathbf{P}_{opt}^{+} = \begin{bmatrix} \mathbf{P}_{\mathbf{ss}}^{-} - \mathbf{K}_{\mathbf{s},opt}\mathbf{W}\mathbf{K}_{\mathbf{s},opt}^{\mathrm{T}} & \mathbf{P}_{\mathbf{sp}}^{-} - \mathbf{K}_{\mathbf{s},opt}\mathbf{H}\begin{bmatrix}\mathbf{P}_{\mathbf{sp}}^{-} \\ \mathbf{P}_{\mathbf{pp}}^{-}\end{bmatrix} \\ \mathbf{P}_{\mathbf{ps}}^{-} - \begin{bmatrix}\mathbf{P}_{\mathbf{ps}}^{-} & \mathbf{P}_{\mathbf{pp}}^{-}\end{bmatrix}\mathbf{H}^{\mathrm{T}}\mathbf{K}_{\mathbf{s},opt}^{\mathrm{T}} & \mathbf{P}_{\mathbf{pp}}^{-} \end{bmatrix} - \begin{bmatrix} \mathbf{O}_{n_{\mathbf{s}}\times n_{\mathbf{s}}} & \mathbf{O}_{n_{\mathbf{s}}\times n_{\mathbf{p}}} \\ \mathbf{O}_{n_{\mathbf{p}}\times n_{\mathbf{s}}} & \mathbf{K}_{\mathbf{p},opt}\mathbf{W}\mathbf{K}_{\mathbf{p},opt}^{\mathrm{T}} \end{bmatrix}$$

or equivalently

$$\mathbf{P}_{opt}^{+} = \mathbf{P}_{SKF}^{+} - (\mathbf{S}\mathbf{K}_{opt})\,\mathbf{W}\,(\mathbf{S}\mathbf{K}_{opt})^{\mathrm{T}}, \qquad \mathbf{S} = \begin{bmatrix} \mathbf{O}_{n_{\mathbf{s}}\times n_{\mathbf{s}}} & \mathbf{O}_{n_{\mathbf{s}}\times n_{\mathbf{p}}} \\ \mathbf{O}_{n_{\mathbf{p}}\times n_{\mathbf{s}}} & \mathbf{I}_{n_{\mathbf{p}}\times n_{\mathbf{p}}} \end{bmatrix}. \tag{23}$$

$\mathbf{K}_{opt}$ is an $n_{\mathbf{x}} \times 1$ vector, because measurements are processed as scalars. For the same reason, $\mathbf{W} = W$, which results in

$$\mathbf{P}_{SKF}^{+} = \mathbf{P}_{opt}^{+} + W\,(\mathbf{S}\mathbf{K}_{opt})\,(\mathbf{S}\mathbf{K}_{opt})^{\mathrm{T}} = \mathbf{P}_{opt}^{+} + c\,\mathbf{a}\mathbf{a}^{\mathrm{T}} \tag{24}$$

This has the same form as the rank-one update as found in Eq. (21). As was noted in the previous section, the optimal measurement update using the rank-one algorithm had a negative scalar involved in the product, thereby creating potential numerical issues. This drawback is not present in Eq. (24). With this in mind, the original backward-recursive ('un-modified') rank-one update is now used to obtain $\mathbf{P}_{con}^{+}$ from $\mathbf{P}_{opt}^{+}$. If, for example, all the 'consider' parameters are in the top part of the state-space (i.e. the 'estimated states'), we can effectively reduce the computations because the second update will not modify the columns of $\mathbf{U}$ and $\mathbf{D}$ corresponding to the estimated states.

Finally, the states (and the consider parameters) are updated as

$$\hat{\mathbf{s}}^{+} = \hat{\mathbf{s}}^{-} + \mathbf{K}_{\mathbf{s}}\left(\mathbf{y} - \mathbf{H}_{\mathbf{s}}\hat{\mathbf{s}}^{-} - \mathbf{H}_{\mathbf{p}}\hat{\mathbf{p}}^{-}\right) \qquad \hat{\mathbf{p}}^{+} = \hat{\mathbf{p}}^{-} \tag{25}$$

Therefore, the procedure to incorporate a consider update into the UDU factorization is as follows: first perform a complete rank-one measurement update with the optimal Kalman Gain ($\mathbf{K}_{opt}$) according to the *modified* rank-one update (as in Table 2) on the full covariance matrix. Second, perform another rank-one update with $\mathbf{a} = \mathbf{S}\mathbf{K}_{opt}$ and $c = W$, according to the (un-modified) rank-one update (as in Table 1). While this involves a modest amount of additional computations, it still fits within the framework of the UDU matrix factorization, retaining its numerical stability. Additionally, if the consider parameters are always the same across flight they can be placed conveniently to reduce computations.

## THE GLOBALLY OPTIMAL SCHMIDT-KALMAN RECURSIVE FILTER

The Schmidt-Kalman filter has been derived as the optimal update under the condition that the rows of the gain corresponding to the 'consider' parameters are zero. However, as pointed out by Bierman, this optimality holds only for a single update. When the Schmidt-Kalman filter is made into a recursive algorithm the optimality is lost after the very first measurement update. The burning question is this: *is there a globally optimal recursive Schmidt-Kalman filter formulation in which the 'consider' parameters are not updated but the 'estimated' states are?*

The answer is: yes, there is. But to get at this, one needs to expend a bit of effort to find it.

First, a bit of background is in order. Through a logic construction, Bierman [12] stated that the recursive Schmidt-Kalman filter is non-optimal because processing data in multiple batches provides a less accurate estimate than processing it all at once. In this section a new algorithm is proposed to overcome the non-optimality of the recursive Schmidt-Kalman filter.

Begin as follows: suppose the measurement vector is divided into two batches $\mathbf{y} = \begin{bmatrix}\mathbf{y}_a^{\mathrm{T}} & \mathbf{y}_b^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} = \begin{bmatrix}\mathbf{H}_a & \mathbf{H}_b\end{bmatrix}\mathbf{x} + \begin{bmatrix}\boldsymbol{\eta}_a^{\mathrm{T}} & \boldsymbol{\eta}_b^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$, the measurement noises $\boldsymbol{\eta}_a$ and $\boldsymbol{\eta}_b$ are zero mean, uncorrelated from each other, and have covariance matrices given by $\mathbf{R}_a$ and $\mathbf{R}_b$, respectively. It is well-known that processing the two batches together or sequentially using the standard Kalman filter equations produces the same result [15], i.e. defining

$$\mathbf{P}_{opt}^{+} = \left(\mathbf{I} - \mathbf{K}_{opt}\mathbf{H}\right)\mathbf{P}^{-}, \qquad \mathbf{K}_{opt} = \mathbf{P}^{-}\mathbf{H}^{\mathrm{T}}\left(\mathbf{H}\mathbf{P}^{-}\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right)^{-1} \tag{26}$$

$$\mathbf{P}_{opt}^{a} = \left(\mathbf{I} - \mathbf{K}_{opt}^{a}\mathbf{H}_a\right)\mathbf{P}^{-}, \qquad \mathbf{K}_{opt}^{a} = \mathbf{P}^{-}\mathbf{H}_a^{\mathrm{T}}\left(\mathbf{H}_a\mathbf{P}^{-}\mathbf{H}_a^{\mathrm{T}} + \mathbf{R}_a\right)^{-1} \tag{27}$$

$$\mathbf{P}_{opt}^{b} = \left(\mathbf{I} - \mathbf{K}_{opt}^{b}\mathbf{H}_b\right)\mathbf{P}_{opt}^{a}, \qquad \mathbf{K}_{opt}^{b} = \mathbf{P}_{opt}^{a}\mathbf{H}_b^{\mathrm{T}}\left(\mathbf{H}_b\mathbf{P}_{opt}^{a}\mathbf{H}_b^{\mathrm{T}} + \mathbf{R}_b\right)^{-1} \tag{28}$$

then $\mathbf{P}^{+}_{opt} = \mathbf{P}^{b}_{opt}$. Processing all the measurements together (i.e. in one batch) using the Schmidt-Kalman gain given by

$$\mathbf{K}_{SKF} = \begin{bmatrix} \mathbf{P}^{-}_{\mathbf{ss}}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{P}^{-}_{\mathbf{sp}}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} \\ \mathbf{O} \end{bmatrix} \left(\mathbf{H}_{\mathbf{s}}\mathbf{P}^{-}_{\mathbf{ss}}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{s}}\mathbf{P}^{-}_{\mathbf{sp}}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{-}_{\mathbf{ps}}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{-}_{\mathbf{pp}}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{R}\right)^{-1}$$

it is easy to observe that $\hat{\mathrm{s}}^{+}$, $\mathbf{P}^{+}_{\mathbf{ss}}$, $\mathbf{P}^{+}_{\mathbf{ps}}$ have the same values for both the Schmidt-Kalman filter and the classical Kalman filter.

Processing the first batch of measurements with the Schmidt-Kalman gain results in $\mathbf{P}^{a}_{\mathbf{ss},opt} = \mathbf{P}^{a}_{\mathbf{ss},SKF}$, $\mathbf{P}^{a}_{\mathbf{ps},opt} = \mathbf{P}^{a}_{\mathbf{ps},SKF}$; however, $\mathbf{P}^{a}_{\mathbf{pp},opt} \neq \mathbf{P}^{a}_{\mathbf{pp},SKF} = \mathbf{P}^{-}_{\mathbf{pp}}$. The covariance of the 'consider' parameters remains unchanged across a Schmidt-Kalman update, therefore we have that $\mathbf{P}^{a}_{\mathbf{pp},SKF} \geq \mathbf{P}^{a}_{\mathbf{pp},opt}$. It follows that

$$\begin{aligned} &\mathbf{H}_{\mathbf{s}}\mathbf{P}^{a}_{\mathbf{ss},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{s}}\mathbf{P}^{a}_{\mathbf{sp},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{a}_{\mathbf{ps},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{a}_{\mathbf{pp},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{R}_{b} \geq \\ &\qquad \mathbf{H}_{\mathbf{s}}\mathbf{P}^{a}_{\mathbf{ss},opt}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{s}}\mathbf{P}^{a}_{\mathbf{sp},opt}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{a}_{\mathbf{ps},opt}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{a}_{\mathbf{pp},opt}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{R}_{b} \end{aligned}$$

Since

$$\begin{aligned} \mathbf{K}^{b}_{SKF} = &\begin{bmatrix} \mathbf{P}^{a}_{\mathbf{ss},SKF}(\mathbf{H}^{b}_{\mathbf{s}})^{\mathrm{T}} + \mathbf{P}^{a}_{\mathbf{sp},SKF}(\mathbf{H}^{b}_{\mathbf{p}})^{\mathrm{T}} \\ \mathbf{O} \end{bmatrix} \\ &\left(\mathbf{H}_{\mathbf{s}}\mathbf{P}^{a}_{\mathbf{ss},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{s}}\mathbf{P}^{a}_{\mathbf{sp},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{a}_{\mathbf{ps},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{s}} + \mathbf{H}_{\mathbf{p}}\mathbf{P}^{a}_{\mathbf{pp},SKF}\mathbf{H}^{\mathrm{T}}_{\mathbf{p}} + \mathbf{R}_{b}\right)^{-1} \end{aligned} \tag{29}$$

it follows that the rows of $\mathbf{K}^{b}_{SKF}$ corresponding to the estimated states will be "smaller" than the corresponding rows of $\mathbf{K}^{b}_{opt}$, resulting in a smaller update, hence $\mathbf{P}^{b}_{\mathbf{ss},SKF} \geq \mathbf{P}^{a}_{\mathbf{ss},opt}$. This is why Bierman came to the conclusion he did.

However it is still possible to recover the Schmidt-Kalman covariance of the single update even using a recursive algorithm. Recall that the "optimal" Schmidt-Kalman filter was derived by minimizing the trace of the posterior covariance. The optimality holds only when all the measurements at a given time are processed together, given a prior covariance. Processing all the measurements together (in one batch) is more accurate than processing them in two batches, which is more accurate than processing them in three batches, and so on. Whereas this may not seem of great consequence (because usually only a few measurements are available at any given time), it does have a significant implication: while processing measurements together or in batches yields the same result in the Kalman filter and is a key to its optimality, this is *not* the case for the recursive Schmidt-Kalman filter. In the 'classic' Kalman filter only current measurements are incorporated and this is followed by a propagation cycle to the next measurement epoch, this process continuing so long as measurements are available. Of course, this is equivalent to processing all measurements (past and present) in one batch because of the property previously discussed.

Recall, as well, that in a earlier section, the Schmidt-Kalman filter gain was obtained from the 'classic' (optimal) Kalman filter measurement update equations, zeroing out the rows of the Kalman gain corresponding to the parameters

$$\mathbf{K}_{SKF} = \begin{bmatrix} \mathbf{I}_{n_{\mathbf{s}}\times n_{\mathbf{s}}} & \mathbf{O}_{n_{\mathbf{s}}\times n_{\mathbf{p}}} \\ \mathbf{O}_{n_{\mathbf{p}}\times n_{\mathbf{s}}} & \mathbf{O}_{n_{\mathbf{p}}\times n_{\mathbf{p}}} \end{bmatrix} \mathbf{K}_{opt} = \begin{bmatrix} \mathbf{K}_{\mathbf{s}_{opt}} \\ \mathbf{O}_{n_{\mathbf{p}}\times 1} \end{bmatrix} \tag{30}$$

Likewise, the updated covariance can be computed by first using the (optimal) Kalman gain and then by replacing $\mathbf{P}^{+}_{\mathbf{pp}}$ by $\mathbf{P}^{-}_{\mathbf{pp}}$ which is equivalent to

$$\mathbf{P}^{+}_{SKF} = \mathbf{P}^{+}_{opt} + W \left(\mathbf{S}\mathbf{K}_{opt}\right)\left(\mathbf{S}\mathbf{K}_{opt}\right)^{\mathrm{T}} = \mathbf{P}^{+}_{opt} + c\,\mathbf{a}\mathbf{a}^{\mathrm{T}} \tag{31}$$

*This strategy yields the* **same** *result as the Schmidt-Kalman filter* and this is the key to the *optimal, recursive Schmidt-Kalman filter*. Significantly, the state and covariance can be updated in many batches (at the same epoch) with the optimal Kalman gain and the replacement for $\mathbf{P}_{\mathbf{pp}}$ is performed only after the last batch has been processed.

The *optimal* recursive Schmidt-Kalman filter which gives identical results to a batch Schmidt-Kalman estimator is obtained by implementing an optimal Kalman filter in which the entire covariance is updated and propagated through all the measurements but only the estimated states are updated while the consider parameters are not. As was discussed earlier, given the same *a priori* covariance and state, the optimal Kalman filter and the Schmidt-Kalman filter yield the same values of the states, the covariance of the states, and the cross-covariance of the states and the consider parameters. This optimal recursive Schmidt-Kalman filter produces exactly the same estimate for the states $\mathbf{s}$ as the globally optimal Kalman filter, hence it produces the best possible estimate of $\mathbf{s}$. Because the parameters are not updated but their covariance $\mathbf{P}_{\mathbf{pp}}$ is, the covariance of the 'consider' parameters in the proposed optimal recursive Schmidt-Kalman filter does not reflect the actual uncertainty; rather, it represents the uncertainty the parameters would have had they been optimally estimated.

The propagation phase of the proposed optimal recursive Schmidt-Kalman filter is identical to that of the standard Schmidt-Kalman filter which in turn is identical to that of the classic Kalman filter. The recursive formulation of the classic Kalman filter provides an estimate at any time $t_k$ that is identical to processing all measurements (past and present) together in a single batch. The proposed optimal recursive Schmidt-Kalman filter provides an estimate identical to processing all measurements (past and present) together in a single batch using the standard Schmidt-Kalman filter algorithm. Processing all measurements together in a single batch using the two algorithms produces identical values for the estimate of $\mathbf{s}$, and the covariance matrices $\mathbf{P}_{\mathbf{ss}}$ and $\mathbf{P}_{\mathbf{sp}}$. The "true" covariance of the parameters $\mathbf{P}_{\mathbf{pp}}$ is not needed in the proposed algorithm nor it is computed. If desired, the "true" covariance $\mathbf{P}_{\mathbf{pp}}$ can be computed as an additional quantity outside of the main filter equations. Without these not needed computations the optimal recursive Schmidt-Kalman filter is slightly less computationally intensive than the classic Kalman filter. While the covariance computations are identical, the proposed algorithm does not need to calculate the parameters' gain $\mathbf{K}_{\mathbf{p}}$ nor to update the parameters' estimate.

Across several measurement cycles, the proposed optimal recursive Schmidt-Kalman filter will have a smaller covariance than the traditional Schmidt-Kalman filter. Hence the uncertainty of the classic Schmidt-Kalman filter is always larger than that of the optimal recursive Schmidt-Kalman filter proposed here. This is seen in the example in the following section.

## NUMERICAL RESULTS

This section introduces a simple numerical example to demonstrate the theory developed in this paper. A single state $s$ and parameter $p$ form the state vector $\mathbf{x}^{\mathrm{T}} = \begin{bmatrix} s & p \end{bmatrix}$. The state is measured directly, while the parameter is a systematic measurement error

$$y_k = s_k + p_k + \eta_k = \begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{x}_k + \eta_k = \mathbf{H}\mathbf{x}_k + \eta_k \tag{32}$$

where $\eta_k$ is a zero mean, white sequence with variance $R_k = 1$ and uncorrelated from any other error source. The true state is a random walk while the parameter is a first order Markov process

$$\mathbf{x}_{k+1} = \mathbf{\Phi}_k \mathbf{x}_k + \boldsymbol{\nu}_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{-\Delta t_k/\tau} \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \nu_k \\ \mu_k \end{bmatrix} \tag{33}$$

where $\nu_k$ and $\mu_k$ are zero mean, uncorrelated white sequences with variances given by $Q_k = 1$ and $(1 - e^{-2\Delta t_k/\tau}) P_{pp,ss}$, respectively. The steady-state value of the Markov process variance is chosen as $P_{pp,ss} = 1$ and the time constant $\tau$ is such that $e^{-200/\tau} = 0.5$. An initial unbiased estimate is given by $\hat{\mathbf{x}}_0^{\mathrm{T}} = \begin{bmatrix} 0 & 0 \end{bmatrix}$. A first measurement $y_0$ is assumed to be available at time $t_0 = 0$ and a second measurement $y_1$ becomes available at time $t_1 = 100$. The initial estimation error covariance and its updated value once the first measurement is incorporated with a standard Kalman filter are given by

$$\mathbf{P}_0 = \begin{bmatrix} 10 & 3 \\ 3 & 1 \end{bmatrix}, \qquad \mathbf{P}_{KF}^{+}(t_0) = \begin{bmatrix} 0.6111 & 0.1111 \\ 0.1111 & 0.1111 \end{bmatrix}. \tag{34}$$

After the time propagation and the second update:

$$\mathbf{P}_{KF}^{-}(t_1) = \begin{bmatrix} 1.6111 & 0.0786 \\ 0.0786 & 0.5556 \end{bmatrix}, \qquad \mathbf{P}_{KF}^{+}(t_1) = \begin{bmatrix} 0.7522 & -0.2438 \\ -0.2438 & 0.4346 \end{bmatrix}. \tag{35}$$

From the discussion above and since the parameter is a Markov process at its steady-state value, it follows that an optimal recursive Schmidt-Kalman filter provides an estimate with error covariance

$$\mathbf{P}_{OSKF}^{+}(t_1) = \begin{bmatrix} 0.7522 & -0.2438 \\ -0.2438 & 1 \end{bmatrix}, \tag{36}$$

however applying the usual Schmidt-Kalman filter equations recursively the first update is given by

$$\mathbf{P}_{SKF}^{+}(t_0) = \begin{bmatrix} 0.6111 & 0.1111 \\ 0.1111 & 1 \end{bmatrix}, \tag{37}$$

and the propagation and second update produce

$$\mathbf{P}_{SKF}^{-}(t_1) = \begin{bmatrix} 1.6111 & 0.0786 \\ 0.0786 & 1 \end{bmatrix}, \qquad \mathbf{P}_{SKF}^{+}(t_1) = \begin{bmatrix} 0.8535 & -0.4051 \\ -0.4051 & 1 \end{bmatrix}. \tag{38}$$

It is apparent that the classic Schmidt-Kalman filter does not extract all the possible information because it produces an estimate of the state $s$ with variance $0.8535$ which is greater than the optimal estimate with variance $0.7522$.

The optimal recursive Schmidt-Kalman filter algorithm proposed in this paper carries the covariance of the Kalman filter with the understanding the the portion corresponding to the covariance of the consider parameters is fictitious. At any time it is desired to know the actual covariance it can be built from the Kalman filter covariance and an externally carried covariance of the parameters. For the simple example in this section the parameter's covariance stays constant at its steady state value. To demonstrate the validity of this method a Monte Carlo simulation is performed. True initial states are obtained sampling from a Gaussian distribution with mean $\hat{\mathbf{x}}_0$ and covariance $\mathbf{P}_0$. Each Monte Carlo run also samples different values for the process noise and measurement noise, with zero mean and covariance as specified above. The propagation phase is given by

$$\hat{\mathbf{x}}_{k+1} = \mathbf{\Phi}_k \hat{\mathbf{x}}_k \tag{39}$$

$$\mathbf{P}_{k+1} = \mathbf{\Phi}_k \mathbf{P}_k \mathbf{\Phi}_k^{\mathrm{T}} + \begin{bmatrix} Q_k & 0 \\ 0 & (1 - e^{-2\Delta t_k/\tau}) P_{pp,ss} \end{bmatrix} \tag{40}$$

Figures 1(a) to 1(c) show the results of 100 Monte Carlo runs for each algorithm. The figures convey that all the algorithms perform correctly since their predicted estimation error is consistent with its actual value. Figure 1(d) shows a comparison of the three algorithms. All the covariances shown in the plots are $3\sigma$ values.

## CONCLUSIONS

This paper analyzes recursive implementations of the Schmidt-Kalman filter. A recursive implementation is essential for onboard estimation systems. A common strategy for aerospace navigation is to carry the filter's estimation error covariance utilizing the UDU formulation. This paper introduces a recursive formulation of the UDU Schmidt-Kalman filter. A second algorithm is proposed in this paper that addresses the non-optimality of the recursive Schmidt-Kalman filter. A numerical example is shown to demonstrate the validity of the proposed approach.

## REFERENCES

[1] S. F. Schmidt, "Application of State-Space Methods to Navigation Problems," *Advances in Control Systems*, Vol. 3, 1966, pp. 293–340.

[2] D. Woodbury and J. Junkins, "On the Consider Kalman Filter," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, August 2010.

[3] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, Vol. 64 of *Mathematics in Sciences and Engineering*. New York, New York 10003: Academic Press, 1970.

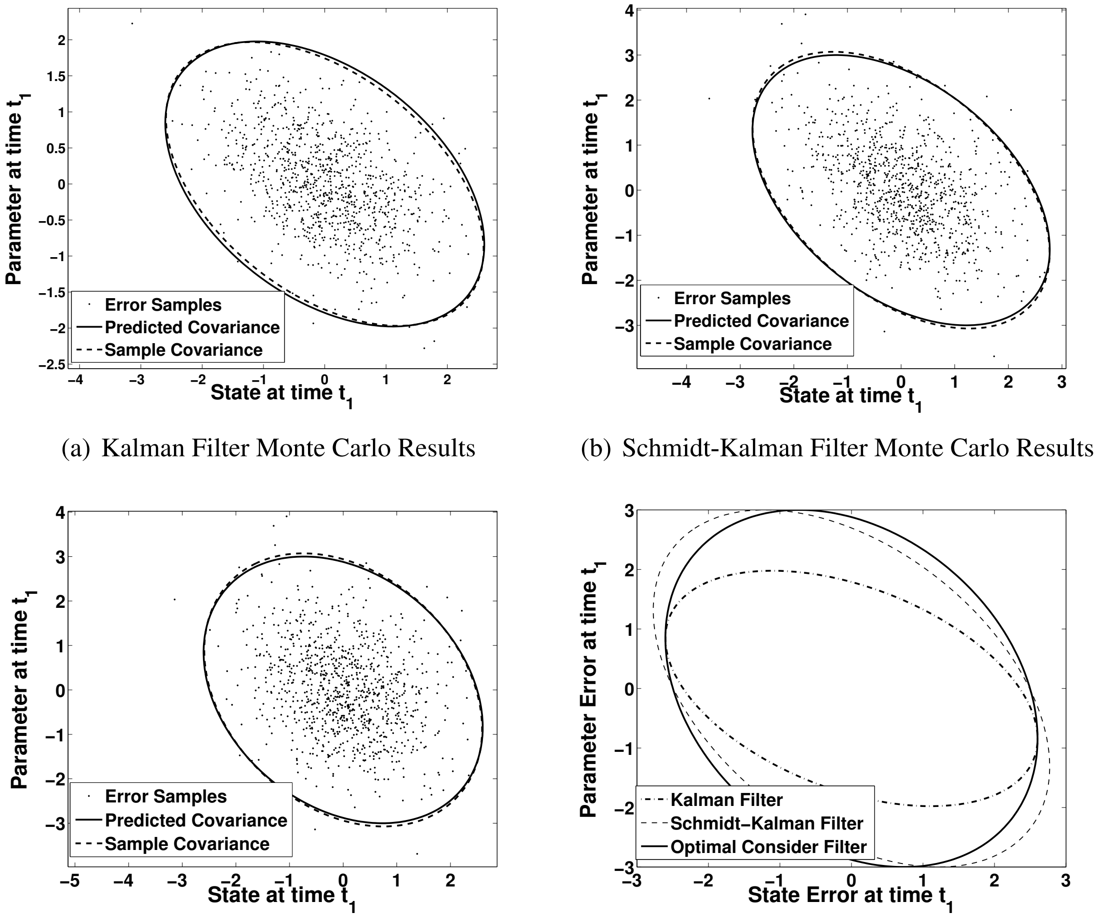
[4] G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation*, Vol. 128 of *Mathematics in Sciences and Engineering*. Academic Press, 1978.

[5] B. D. Tapley, B. E. Schutz, and G. H. Born, *Statistical Orbit Determination*. Elsevier Academic Press, 2004.

[6] D. Woodbury, M. Majji, and J. Junkins, "Considering Measurement Model Parameter Errors in Static and Dynamic Systems," *Proceedings of the George H. Born Symposiu*, May 2010.

[7] R. Zanetti, *Advanced Navigation Algorithms for Precision Landing*. PhD thesis, The University of Texas at Austin, Austin,Texas, December 2007.

[8] R. Zanetti and R. H. Bishop, "Kalman Filters with Uncompensated Biases," *Journal of Guidance, Control, and Dynamics*, Vol. 35, January–February 2012, pp. 327–330.



(a) Kalman Filter Monte Carlo Results

(b) Schmidt-Kalman Filter Monte Carlo Results

(c) Optimal Schmidt-Kalman Filter Monte Carlo Results

(d) Comparison of Filters Estimation Errors

**Figure 1. Algorithms performance and comparison**

[9] R. Zanetti and R. H. Bishop, "Entry Navigation Dead-Reckoning Error Analysis: Theoretical Foundations of the Discrete-Time Case," *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference held August 19–23, 2007, Mackinac Island, Michigan*, Vol. 129 of *Advances in the Astronautical Sciences*, 2007, pp. 979–994. AAS 07-311.

[10] M. E. Hough, "Orbit Determination with Improved Covariance Fidelity, Including Sensor Measurement Biases," *Journal of Guidance, Control, and Dynamics*, Vol. 34, May–June 2011, pp. 903–911.

[11] R. S. Bucy and P. D. Joseph, *Filtering for Stochastic Processing with Applications to Guidance*. Providence, RI: AMS Chelsea Publishing, 2nd ed., 2005.

[12] G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation*. New York: Dover Publications, 2006.

[13] W. Agee and R. Turner, "Triangular Decomposition of a Positive Definite Matrix Plus a Symmetric Dyad with Application to Kalman Filtering," *White Sands Missile Range Tech. Rep. No. 38*, 1972.

[14] N. Carlson, "Fast Triangular Factorization of the Square Root Filter," *AIAA Journal*, Vol. 11, No. 9, 1973, pp. 1259–1265.

[15] R. G. Brown and P. Y. Hwang, *Introduction To Random Signals And Applied Kalman Filtering*. John Wiley and Sons, third ed., 1997.