RECURSIVE IMPLEMENTATIONS OF THE CONSIDER FILTER

Renato Zanetti* and Chris 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 work 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 UDÚ consider filter is proposed. The non-optimality of the recursive consider filter is also analyzed, and a modified algorithm is proposed to overcome this limitation.

INTRODUCTION

Consider analysis was introduced by S. F. Schmidt of NASA Ames in the mid 1960s as a means to account for 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 consider approach is especially useful when parameters have low observability or when the extra computational power to estimate them is not deemed necessary [2].

Schmidt approach is based on minimum variance estimation, Jazwinski [3] details the derivation of the optimal consider Kalman filter. In Section 8.2 of his book, Bierman [4] disputes the optimality of the Schmidt-Kalman filter, at least in its sequential implementation. Nevertheless the Schmidt-Kalman filter has received considerable attention in recent years. Tapley et al. [5] give an ample description of the consider Kalman filter, they provide a different methodology then Jazwinski and arrive to a different formulation. Woodbury et al. provide new insight into considering parameters in the measurement model [6].

While the Schmidt-Kalman filter is very well known, not much attention has been given to actual implementations of it in a real-time recursive estimation algorithm. On-board estimators commonly utilize the UDU formulation, which guarantees symmetry and positive definiteness of the covariance. To date there is no direct way of including a consider update into the UDU formulation, this paper provides a simple algorithm to obtain this capability. Another common practice is to process measurements one at a time (which is inevitable

---

*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
†GN&C Autonomous Flight Systems Engineer, Aeroscience and Flight Mechanics Division, NASA Johnson Space Center EG6, 2101 NASA Parkway, Houston, Texas, 77058. chris.dsouza@nasa.gov
when utilizing the UDU algorithm). When processing measurements one at the time in the Schmidt-Kalman filter the order in which they are processed affects the final result. This work analyzes this phenomenon and outlines a strategy to best utilize the measurements. From this strategy a globally optimal consider filter is proposed, which addresses the objections raised by Bierman.

THE KALMAN FILTER UPDATE

This work assumes linear measurements and dynamics, the extension to the nonlinear case can be readily obtained using standard extended Kalman filter techniques. Let $y$ be a set of measurements of a state vector $x$ corrupted by zero mean noise $\eta$ with covariance $R$ as

$$ y = Hx + \eta, $$

(1)

where $H$ is the measurement mapping (or sensitivity) matrix. Let $\hat{x}^-$ be an unbiased estimate of $x$ with corresponding estimation error covariance given by $P^-$. The a priori estimation error is given by

$$ e^- = x - \hat{x}^-. $$

(2)

The unbiased linear update based upon $\hat{x}^-$ and $y$ produces the a posteriori estimate given by

$$ \hat{x}^+ = \hat{x}^- + K (y - H\hat{x}^-), $$

(3)

where $K$ is some determinist matrix of appropriate dimensions to be determined. The a posteriori estimation error expressed as

$$ e^+ = x - \hat{x}^+ = (I - KH)e^- - K\eta. $$

(4)

Assuming that the measurement error $\eta$ and the a priori estimation error, $e^-$ are uncorrelated and each are zero mean, we find that the a posteriori estimation error covariance is given by the so-called Joseph update:

$$ P^+ = E \left\{ (x - \hat{x}^+) (x - \hat{x}^+)^T \right\} = (I - KH)P^-(I - KH)^T + KRK^T, $$

(5)

where $I$ is the identity matrix of appropriate dimension and $R$ is the measurement error covariance matrix.

Defining the covariance of the measurement residuals, $W$, as

$$ W \triangleq HP^-H^T + R, $$

(6)

the updated (a posteriori) covariance is equivalently written as

$$ P^+ = P^- - KHP^- - P^-H^TK^T + KWK^T. $$

(7)
Notice that up to now no assumptions have been made as to the choice of \( K \) and the Joseph update equation is valid for all \( K \), as is Eq. (7). The standard Kalman gain \( K_{opt} \) minimizes the trace of the updated covariance matrix, \( P^+ \)

\[
K_{opt} = P^- H^T W^{-1}.
\]  

Substituting this into Eq. (7) it follows that

\[
P^+ = P^- - K_{opt} HP^-
\]

\[
= P^- - K_{opt} WK_{opt}^T
\]

\[
= P^- - P^- H^T W^{-1} HP^-
\]

These equations are only valid for the optimal gain, \( K_{opt} \).

**THE SCHMIDT-KALMAN FILTER UPDATE**

Suppose now that we partition \( x \) into \( n_s \) “estimated” states, \( s \), and \( n_p \) “consider” parameters, \( p \) as

\[
x \Delta \equiv \begin{bmatrix} s \\ p \end{bmatrix},
\]

preserving the fact that \( n_x = n_s + n_p \), i.e. the total number of states is unchanged. Let us now partition the relevant matrices accordingly. Thus,

\[
P = \begin{bmatrix} P_{ss} & P_{sp} \\ P_{ps} & P_{pp} \end{bmatrix}
\]

\[
H = \begin{bmatrix} H_s \\ H_p \end{bmatrix}
\]

\[
K_{opt} = \begin{bmatrix} K_s \\ K_p \end{bmatrix} = \begin{bmatrix} P_{ss} H_s^T + P_{sp} H_p^T \\ P_{ps} H_s^T + P_{pp} H_p^T \end{bmatrix} W^{-1}.
\]

As in the previous section

\[
W = HP^- H^T + R
\]

\[
= H_s P_{ss} H_s^T + H_s P_{sp} H_p^T + H_p P_{ps} H_s^T + H_p P_{pp} H_p^T + R
\]

The updated portions of the covariance are

\[P^+_{ss} = P_{ss} - K_s H \begin{bmatrix} P_{ss}^- \\ P_{ps}^- \end{bmatrix} - \begin{bmatrix} P_{ss}^- \\ P_{ps}^- \end{bmatrix} H^T K_s^T + K_s W K_s^T\]

\[P^+_{sp} = P_{sp} - K_s H \begin{bmatrix} P_{sp}^- \\ P_{pp}^- \end{bmatrix} - \begin{bmatrix} P_{sp}^- \\ P_{pp}^- \end{bmatrix} H^T K_p^T + K_p W K_p^T\]

\[P^+_{ps} = P_{ps} - K_p H \begin{bmatrix} P_{ss}^- \\ P_{ps}^- \end{bmatrix} - \begin{bmatrix} P_{ss}^- \\ P_{ps}^- \end{bmatrix} H^T K_s^T + K_s W K_s^T\]

\[P^+_{pp} = P_{pp} - K_p H \begin{bmatrix} P_{sp}^- \\ P_{pp}^- \end{bmatrix} - \begin{bmatrix} P_{sp}^- \\ P_{pp}^- \end{bmatrix} H^T K_p^T + K_p W K_p^T\]
These equations are derived from Eq. (7) and are therefore valid for any choice \( K_s \) and \( K_p \).

We now choose the \( K_s \) and \( K_p \) carefully such that the gain for the states is the optimal value \( K_{s,\text{opt}} \) from Eq. (15) and allowing \( K_p \) to be (as yet) unspecified. Thus, substituting for the value of \( K_{s,\text{opt}} \) for the three components in Eqs. (18)–(20), it becomes

\[
P^+ = \begin{bmatrix}
P_{ss} - K_{s,\text{opt}} W K_{s,\text{opt}}^T & P_{sp} - K_{s,\text{opt}} H \left[ \begin{array}{c} P_{sp} \\ P_{pp} \end{array} \right] \\
P_{ps} - \left[ \begin{array}{c} P_{sp} \\ P_{pp} \end{array} \right]^T H^T K_{s,\text{opt}}^T & P_{pp} - K_p H \left[ \begin{array}{c} P_{sp} \\ P_{pp} \end{array} \right] - \left[ \begin{array}{c} P_{sp} \\ P_{pp} \end{array} \right]^T H^T K_p^T + K_p W K_p^T
\end{bmatrix}
\]

This equation is valid for any value of \( K_p \). Notice that there is no \( K_p \) in the cross-covariance between \( s \) and \( p \). Therefore, what is remarkable about this equation is that once the optimal \( K_{s,\text{opt}} \) is chosen, the cross-covariance between \( s \) and \( p \) is independent of the choice of \( K_p \). We will take advantage of this property in due course.

We note that the updated (\textit{a posteriori}) state is given by

\[
\begin{bmatrix} s^+ \\ p^+ \end{bmatrix} = \begin{bmatrix} s^- \\ p^- \end{bmatrix} + \begin{bmatrix} K_s \\ K_p \end{bmatrix} \left( y - H s^- - H p^- \right),
\]

(23)
a consider filter is one in which the parameters, \( p \), are not updated. From Eq. (23) it is observed that this can be achieved by taking \( K_p = 0 \). With this in mind, we start again from the most general covariance update given in Eq. (7) and we obtain that the \textit{a posteriori} consider covariance matrix is

\[
P_{\text{con}}^+ = \begin{bmatrix}
P_{ss} - K_s H \left[ \begin{array}{c} P_{ss} \\ P_{ps} \end{array} \right] - \left[ \begin{array}{c} P_{ss} \\ P_{ps} \end{array} \right]^T H^T K_s^T + K_s W K_s^T & P_{sp} - K_s H \left[ \begin{array}{c} P_{sp} \\ P_{pp} \end{array} \right] \\
P_{ps} - \left[ \begin{array}{c} P_{sp} \\ P_{pp} \end{array} \right]^T H^T K_s^T & P_{pp} - \left[ \begin{array}{c} P_{sp} \\ P_{pp} \end{array} \right]^T H^T K_p^T + K_p W K_p^T
\end{bmatrix}
\]

(24)

To find the optimal consider \( K_s \) we take the derivative of the trace of the covariance matrix with respect to \( K_s \) and set it to zero

\[
\frac{\partial}{\partial K_s} \text{(trace} [P_{\text{con}}^+]) = - \left[ \begin{array}{c} P_{ss}^- \\ P_{ps}^- \end{array} \right]^T H^T - \left[ \begin{array}{c} P_{ss}^- \\ P_{ps}^- \end{array} \right]^T H^T + K_{s,\text{opt}} W^T + K_{s,\text{opt}} W = 0.
\]

(25)
The result is the same as for the globally optimal Kalman filter, so that the optimal consider Kalman filter, also called the Schmidt-Kalman filter can be conveniently calculated from the optimal filter by zeroing out the rows corresponding to the consider states.

\[
K_{s,\text{opt}} = \left( P_{ss}^- H_s^T + P_{sp}^- H_p^T \right) W^{-1}
\]

(26)

So that the consider gain is

\[
K_{\text{con}} = \begin{bmatrix} P_{ss}^- H_s^T + P_{sp}^- H_p^T \\ O \end{bmatrix} W^{-1} = \begin{bmatrix} K_{s,\text{opt}} \\ O \end{bmatrix}
\]

(27)
The state update is

\[
s^+ = s^- + K_s (y - H_s \dot{s}^- - H_p \dot{p}^-)
\]

(28)

\[
p^+ = p^-
\]

(29)

It must be stated emphatically that the full updated covariance matrix (the *a posteriori* matrix) must be calculated by means of the Joseph update. We cannot use Eq. (11), for that equation was valid only for the optimal gain; we have a chosen a sub-optimal gain or a non-globally optimal gain (\(i.e.,\) the optimal gain corresponding to \(K_p = 0\)). Substituting \(K_{s, opt}\) into Eq. (24) we obtain

\[
P^+_{\text{con}} = \begin{bmatrix}
P_{ss}^- - K_s W K_s^T & P_{sp}^- - K_s H [P_{sp}^-; P_{pp}^-] \\
P_{ps}^- & [P_{sp}^-; P_{pp}^-]^T H^T K_s^T P_{pp}^- 
\end{bmatrix}
\]

(30)

Several comments are in order:

1. When using the Schmidt-Kalman filter, the *a priori* and *a posteriori* covariance of the parameters (\(P_{pp}\)) are the same.

2. The *a posteriori* covariance matrix of the states and the correlation between the states and the parameters are the same regardless of whether one uses the Schmidt-Kalman filter or the optimal Kalman update.

It follows that the *a posteriori* consider covariance can be obtained from the *a posteriori* optimal covariance by replacing \(P_{pp}^+\) with \(P_{pp}^-\).

Finally, item (2) above does not mean that \(P_{ss}\) from the optimal Kalman filter will be the same for all time as for the Schmidt-Kalman filter. Rather, it means that given the same *a priori* covariance the two algorithms will produce the same *a posteriori* covariance for \(P_{ss}\) and \(P_{sp}\). However, since \(P_{pp}\) is different between the two algorithms, after the very first update, the *a priori* covariance will be different henceforth, producing different results – something that is comforting. We will expand to this point in due time.

**THE UDU^T UPDATE**

A common strategy to implement covariances in real-time flight software is to use the UDU^T formulation[4]. Thus, the covariance matrix is decomposed as

\[
P = UDU^T
\]

(31)

where \(U\) is an upper triangular matrix with ones on the main diagonal, and \(D\) is a diagonal matrix. In order to efficiently update \(U\) and \(D\) with measurements, the so-called ‘rank-one’ update proposed by Agee and Turner is used [7]. The rank-one (covariance) update requires that measurements are processed one at a time and has the form

\[
P^+ = P^- + c aa^T
\]

(32)
where \( c \) is a scalar and \( a \) is a vector. If we let \( P^+ = U^+ D^+ (U^+)^T \) and \( P^- = U^- D^- (U^-)^T \), then

\[
U^+ D^+ (U^+)^T = U^- D^- (U^-)^T + c a a^T \tag{33}
\]

The algorithm for the rank-one update with \( c > 0 \) is backwards-recursive for \( U^+ \) and \( D^+ \), starting with the \((n, n)\) element \((D_{nn})\) and ending with the \((1, 1)\) element \((D_{11})\) and it is seen in Table 1.

\[
\begin{align*}
\text{For } j &= n, n-1, \ldots, 3, 2 \text{ set } c_n = c \text{ and recursively calculate:} \\
D_{jj}^+ &= D_{jj}^- + c_j a_j^2 \\
a_k &= a_k - a_j U_{kj}^- \quad k = 1, 2, \ldots, j - 1 \\
U_{kj}^+ &= U_{kj}^- + c_k a_j a_k / D_{jj}^+ \quad k = 1, 2, \ldots, j - 1 \\
c_{j-1} &= c_j D_{jj}^+ / D_{jj}^- \\
\text{and finally compute} \\
D_{11}^+ &= D_{11}^- + c_1 a_1^2
\end{align*}
\]

Table 1. Backwards-Recursive Rank-One Update

For scalar measurements \( H \) and \( K \) become vectors and \( W \) is a scalar. We recall that from Eq. (11) that the optimal update of the covariance matrix is

\[
P^+ = P^- - P^- H^T W^{-1} H P^-
\]

Thus,

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

Defining

\[
a \triangleq D^- (U^-)^T H^T \\
c \triangleq -\frac{1}{W} \tag{34}
\]

\( U^+ D^+ (U^+)^T \) becomes

\[
U^+ D^+ (U^+)^T = U^- \left[I - \frac{1}{W} a a^T\right] (U^-)^T
\]

We now proceed with the rank-one update of the term in the square bracket, defining our way to a solution, as

\[
\hat{U} \hat{D} \hat{U}^T \triangleq I - \frac{1}{W} a a^T = I - I + c a a^T
\]
so we find that

\[ U^+ = U^- \tilde{U} \quad \text{and} \quad D^+ = \tilde{D} \]  \hspace{1cm} (36) 

Recall that because the measurement update involves the potential loss of precision due to subtraction of two positive numbers which are close to one another, a modification of the rank-one update is implemented due to Carlson \[8\], resulting in a forward-recursive algorithm, one which is numerically stable. This modified algorithm is only valid for the optimal Kalman update, \( K_{opt} \). The updated \( U \) and \( D \) (\( U^+ \) and \( D^+ \), respectively) and the optimal Kalman gain matrix, \( K_{opt} \), are produced by this algorithm, which is detailed in Table 2.

<table>
<thead>
<tr>
<th>Given ( H, \tilde{U}, \tilde{D}, R = R )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f = \tilde{U}^T H^T )</td>
</tr>
<tr>
<td>( v = \tilde{D} f )</td>
</tr>
<tr>
<td>( \tilde{K}_1 = \begin{bmatrix} v_1 &amp; 0 &amp; \cdots &amp; 0 \end{bmatrix} )</td>
</tr>
<tr>
<td>( \alpha_1 = R + v_1 f_1 )</td>
</tr>
<tr>
<td>( d_1 = d_{11} = \left( \frac{R}{\alpha_1} \right) \tilde{d}_{11} )</td>
</tr>
</tbody>
</table>

For \( j = 2, 3, \ldots, n \), recursively calculate:

- \( \alpha_j = \alpha_{j-1} + v_j f_j \)
- \( d_j = d_{jj} = \left( \frac{\alpha_{j-1}}{\alpha_j} \right) \tilde{d}_{jj} \)
- \( \lambda_j = -\left( f_j / \alpha_{j-1} \right) \)
- \( U_j = \tilde{U}_j + \lambda_j \tilde{K}_{j-1} \)
- \( \tilde{K}_j = \tilde{K}_{j-1} + v_j \tilde{U}_j \)
- and finally compute \( K_{opt} = \tilde{K}_n / \alpha \)

---

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

Finally, the updated state is computed as before, \( i.e. \):

\[ x^+ = x^- + K_{opt} (y - Hx^-) \]  \hspace{1cm} (37) 

**THE SCHMIDT-KALMAN UDU^T UPDATE**

Recall that the Schmidt-Kalman update to the covariance matrix due to the measurements was stated in Eq. (24) as

\[
P_{con}^+ = \begin{bmatrix}
P_{ss}^--K_s W K_{ss}^T & P_{sp}^- - K_s H \begin{bmatrix} P_{sp}^- \\ P_{pp}^- \end{bmatrix} \\
\end{bmatrix}
\]
Unfortunately, the structure of this matrix does not allow for a rank-one update as we developed earlier. We stated earlier that to obtain the updated consider covariance, we begin with the updated optimal covariance and replace \( P_{pp}^+ \) with \( P_{pp}^- \). Now, from Eqs. (10) and (11), we have

\[
P_{opt}^+ = P^- - KWK^T
\]  

(38)

From these relations, we see that the optimal \( a \ posteriori \) covariance is equal to the \( a \ priori \) covariance minus a delta covariance. We can, therefore, harness this idea to help us along. If we were to choose the optimal value for \( K_p \), we would get the following

\[
P_{opt}^+ = \begin{bmatrix}
P_{ss}^- - K_s W K_s^T & P_{sp}^- - K_s H \left[ \begin{array}{c} P_{sp}^- \\ P_{pp}^-
\end{array} \right] \\
\left[ \begin{array}{c} P_{sp}^- \\ P_{pp}^-
\end{array} \right]^T & P_{pp}^-
\end{bmatrix}
\]  

(39)

This, of course can be written as

\[
P_{opt}^+ = P_{con}^+ - \left[ \begin{array}{c}
O_{n_a \times n_a} \\
O_{n_p \times n_p}
\end{array} \right] K_p W K_p^T
\]  

(40)

Thus, we can rewrite Eq. (40) as

\[
P_{opt}^+ = P_{con}^+ - W \left( S K_{opt} \right) \left( S K_{opt} \right)^T
\]  

(41)

Recalling from Eq.(15) that \( K_{opt} \) was defined as

\[
K_{opt} \Delta \left[ \begin{array}{c}
K_s \\
K_p
\end{array} \right]
\]

and defining \( S \) as

\[
S \Delta \left[ \begin{array}{cc}
O_{n_s \times n_s} & O_{n_a \times n_p} \\
O_{n_s \times n_p} & I_{n_p \times n_p}
\end{array} \right]
\]  

(42)

Eq. (41) becomes

\[
P_{opt}^+ = P_{con}^+ - W \left( S K_{opt} \right) \left( S K_{opt} \right)^T
\]  

(43)

We note that \( S \) is an \( n_x \times n_x \) matrix (recalling that \( n_x = n_s + n_p \)) and \( K_{opt} \) is an \( n_x \times 1 \) vector, because we are processing the measurements as scalars. Therefore \( S K_{opt} \) is an \( n_x \times 1 \) vector. Therefore,

\[
P_{con}^+ = P_{opt}^+ + W \left( S K_{opt} \right) \left( S K_{opt} \right)^T
\]  

(44)
This has the same form as the rank-one update as found in Eq. (32), i.e. $P^+ = P^- + c_{aa}^T$. We noted earlier that the scalar was negative for the optimal update creating potential numerical issues, but now

\[ a = SK_{opt} \]  \hspace{1cm} (45)
\[ c = W \]  \hspace{1cm} (46)

With this in mind, we can use the (un-modified) rank-one update which is a backward-recursive update. If, for example, all the consider parameters are in the top part of the state-space, we can effectively reduce the computations because the second update will not modify the columns of $U$ and $D$ corresponding to the estimated states.

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

\[ s^+ = s^- + K_s (y - H_s \hat{s}^- - H_p \hat{p}^-) \]  \hspace{1cm} (47)
\[ p^+ = p^- \]  \hspace{1cm} (48)

Of course, this is identical to what was obtained in Section 3.

Therefore, the procedure is as follows: first perform a complete rank-one measurement update with the optimal Kalman Gain ($K_{opt}$) according to the modified rank-one update (as in Table 2) – on the full covariance matrix. Second, perform another rank-one update with $a = SK_{opt}$ and $c = W$, according to the (un-modified) rank-one update (as in Table 1). If the consider parameters are always the same across flight we place them conveniently to reduce computations.

**OPTIMAL CONSIDER FILTER**

Through a logic construction, Bierman [4] states that the recursive Schmidt-Kalman filter is non-optimal because processing data as multiple batches provides a less accurate result that processing it all at one. We now demonstrate that statement and we deduce an optimal utilization strategy for measurements processed one at the time. Finally we propose a solution to overcome the nonoptimality of the Schmidt-Kalman filter.

Suppose we divide the measurement vector into two uncorrelated batches $y^T = [y_a^T \ y_b^T]$. It is well-known that processing the two batches together or sequentially using the Kalman filter equations produces the same result [9], i.e. defining

\[ P_{opt}^+ = (I - K_{opt} H) P^- , \quad K_{opt} = P^- H^T (H P^- H^T + R)^{-1} \]  \hspace{1cm} (49)
\[ P_{opt}^a = (I - K_{opt}^a H_a) P^- , \quad K_{opt}^a = P^- H_a^T (H_a^a P^- H_a^T + R_a)^{-1} \]  \hspace{1cm} (50)
\[ P_{opt}^b = (I - K_{opt}^b H_b) P_{opt}^a , \quad K_{opt}^b = P_{opt}^a H_b^T (H_b P_{opt}^a H_b^T + R_b)^{-1} \]  \hspace{1cm} (51)

then $P_{opt}^+ = P_{opt}^b$. Processing all the measurements at once using the consider gain

\[ K_{con} = \begin{bmatrix} P_{ss}^s H_s^T + P_{sp}^s H_p^T \\ P_{ss}^p H_p^T + P_{sp}^p H_s^T \end{bmatrix} \begin{bmatrix} H_s P_{ss}^s H_s^T + H_s P_{sp}^s H_p^T + H_p P_{ss}^s H_s^T + H_p P_{sp}^s H_p^T + R \end{bmatrix}^{-1} \]
we have that $s^+, P_{ss}, P_{ps}^+$ have the same values for both the Schmidt-Kalman filter and the regular Kalman filter. If we process the first batch of measurements with the consider gain we have that $P_{ss, opt}^a = P_{ss, con}^a$, $P_{ps, opt}^a = P_{ps, opt}^a$, but $P_{pp, opt}^a \neq P_{pp, con}^a = P_{pp}^a$. Therefore

$$H_s P_{ss, con}^a H_s^T + H_s P_{sp, con}^a H_p^T + H_p P_{ps, con}^a H_s^T + H_p P_{pp, con}^a H_p^T + R_b >$$

$$H_s P_{ss, opt}^a H_s^T + H_s P_{sp, opt}^a H_p^T + H_p P_{ps, opt}^a H_s^T + H_p P_{pp, opt}^a H_p^T + R_b$$

(52)

hence the rows of $K_{con}^b$ corresponding to the estimated states will be “smaller” than the corresponding rows of $K_{opt}^b$, resulting in a smaller update. If we have two measurements of the same kind, but one more accurate than the other, and we need to process them sequentially with the consider gain, which one should we process first? After the first consider update we loose the improvement to $P_{pp}$ that we would have using the optimal gain, the covariance of the parameters is then used in calculating the second gain.

Previously we derived the “optimal” consider filter by minimizing the trace of the posterior covariance. The optimality holds only if all measurements are processed at once. Processing all measurements at once is more accurate than processing them in two batches, which in turn is more accurate that processing them in three batches, and so on. While this result might not seem of great consequences (usually only a few measurements are available at each given time), it has one larger implication: processing all of the past and current measurements at once is more accurate than the usual recursive formulation in which only current measurements are incorporated and followed by a propagation phase to the next measurement epoch. In standard aerospace applications, measurements can be incorporated in hundreds of batches, one for each time the filter update phase is invoked.

Notice that we can incorporate the measurements in batches using the regular Kalman filter update updating the full covariance but only the estimate of the states; only after the last batch is processed we substitute the optimal $P_{pp}^+$ with the prior $P_{pp}^-$. Adopting this strategy we obtain exactly the same result independently if we process the measurements individually or together.

It follows that obtaining an optimal recursive Schmidt-Kalman filter that gives identical results to a batch estimator is rather simple. We need to implement a regular Kalman filter in which the entire covariance is updated and propagated, but only the estimated states are updated using the measurements, while the consider parameters remain unchanged across measurements. As discussed earlier the optimal consider filter and the Kalman filter share the same estimate of the states, their covariance, and their correlation to the consider parameters. The covariance of the parameters of this optimal consider filter does not represent their actual uncertainty, instead it represents the uncertainty the parameters would have if they were optimally estimated. If it is desired to know what the uncertainty of the parameters actually is, another, smaller, covariance of the parameters only needs to be carried in the filter. This “true” covariance is never used nor changes in the measurement update phase.
NUMERICAL RESULTS

In this section we present a very simple numerical example to demonstrate the theory developed in this paper. A single state $s$ and parameter $p$ for the state vector $x^T = [s \ p]$. The state is measured directly, while the parameter is a systematic measurement error

$$y_k = s_k + p_k + \eta_k = \begin{bmatrix} 1 & 1 \end{bmatrix} x_k + \eta_k = Hx_k + \eta_k$$

(53)

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

$$x_{k+1} = \Phi_k x_k + \nu_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{-\Delta t_k/\tau} \end{bmatrix} x_k + \begin{bmatrix} \nu_k \\ \mu_k \end{bmatrix}$$

(54)

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 $P_{pp,ss} = 1$. An initial unbiased estimate $\hat{x}_0^T = [0 \ 0]$ of the true state vector is available and has associated estimation error covariance given by

$$P_0 = \begin{bmatrix} 10 & 3 \\ 3 & 1 \end{bmatrix}.$$ 

(55)

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 time constant $\tau$ is such that $e^{-200/\tau} = 0.5$. A standard Kalman filter implementation performs a first update and produces

$$P_{KF}^+(t_0) = \begin{bmatrix} 0.6111 & 0.1111 \\ 0.1111 & 0.1111 \end{bmatrix}.$$ 

(56)

The next step is a time update after which

$$P_{KF}^- (t_1) = \begin{bmatrix} 1.6111 & 0.0786 \\ 0.0786 & 0.5556 \end{bmatrix}.$$ 

(57)

Finally a second update is performed that produces

$$P_{KF}^+ (t_1) = \begin{bmatrix} 0.7522 & -0.2438 \\ -0.2438 & 0.4346 \end{bmatrix}.$$ 

(58)

From the discussion above and since the initial uncertainty of the parameter coincides with its steady-state value we know that an optimal consider filter provides an estimate with error covariance

$$P_{OCKF}^+ (t_1) = \begin{bmatrix} 0.7522 & -0.2438 \\ -0.2438 & 1 \end{bmatrix},$$

(59)

however applying the Schmidt-Kalman filter equations recursively we obtain after the first update

$$P_{C KF}^- (t_0) = \begin{bmatrix} 0.6111 & 0.1111 \\ 0.1111 & 1 \end{bmatrix},$$

(60)
after the propagation

\[ \mathbf{P}_{CKF}^-(t_1) = \begin{bmatrix} 1.6111 & 0.0786 \\ 0.0786 & 1 \end{bmatrix}, \quad (61) \]

and finally after the second update is

\[ \mathbf{P}_{CKF}^+(t_1) = \begin{bmatrix} 0.8535 & -0.4051 \\ -0.4051 & 1 \end{bmatrix}. \quad (62) \]

The optimal consider 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 \( \hat{x}_0 \) 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

\[ \hat{x}_{k+1} = \Phi_k \hat{x}_k. \quad (63) \]

Figures 1 to 3 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 4 shows a comparison of the three algorithms. All the covariances shown in the plots are 3\( \sigma \) values.

**CONCLUSIONS**

The paper analyzes the recursive implementations of the Schmidt-Kalman filter. A numerically stable, recursive implementation of the UDU Schmidt-Kalman filter is introduced. The non-optimality of the recursive Schmidt-Kalman filter is discussed, and a simple modification is proposed to overcome this deficiency. Monte Carlo simulations are used to confirm the validity of this approach.

**REFERENCES**


Figure 1. Kalman Filter Monte Carlo Results
Figure 2. Schmidt-Kalman Filter Monte Carlo Results
Figure 3. Optimal Consider Filter Monte Carlo Results
Figure 4. Comparison of Filters Estimation Errors