Information Formulation of the UDU Kalman Filter

Christopher D'Souza and Renato Zanetti

Abstract

A new information formulation of the Kalman filter is presented where the information matrix is parameterized as the product of an upper triangular matrix, a diagonal matrix, and the transpose of the triangular matrix (UDU factorization). The UDU factorization of the Kalman filter is known for its numerical stability, this work extends the technique to the information filter. A distinct characteristic of the new algorithm is that measurements can be processed as vectors, while the classic UDU factorization requires scalar measurement processing, i.e. a diagonal measurement noise covariance matrix.

I. Introduction

The UDU formulation of the Kalman Filter has been used in aerospace engineering applications for several decades. Thornton [1], Bierman and Thornton [2] and Bierman [3] introduced an elegant formulation where the covariance matrix P is replaced by two factors: a diagonal matrix D and an upper triangular matrix U with ones on the main diagonal, such that $P = UDU^T$. Whereas the UDU factorization improves the computational stability and efficiency of large navigation filters, it was originally used in a batch formulation [4]. However, this formulation lent itself to sequential implementations, well-suited for platforms where both computational stability and numerical efficiency are at a premium. It serves as the backbone of the Orion Navigation System [5].

Factorization of the covariance matrix in a Kalman filter [6] is almost as old as the filter itself. In 1963 James Potter developed a square-root formulation of the Kalman filter to implement on the Apollo onboard computer [7]. The main driver at the time was numerical precision, as computer words were only 8 bits long. Replacing the covariance by a square root matrix S, such as $P = SS^T$, reduces the spread of the elements of P bringing them closer to 1, doubling the numerical precision of the stored variable. Potter's algorithm requires the computation of scalar square roots (one per measurement). At the time, the Apollo Kalman filter was designed without any process noise, because computations required for inclusion of the process noise required too many computations [8]. A very desirable by-product of this factorization is that the symmetry and semi-positive definiteness of the covariance are insured by construction, and does not need to be checked or enforced to correct for numerical and round-off errors. It should be noted that this Apollo factorization was not a triangular square root matrix.

An alternative square root covariance factorization is the Cholesky factorization [9], [10]. The Cholesky method is very similar to Potter's but computes the square root of the covariance matrix with a Cholesky decomposition (S is a triangular matrix) [11]. Another relevant covariance factorization work is that proposed by Oshman and Bar-Itzhack [12], which utilizes the spectral decomposition of the covariance matrix.

The UDU factorization is not a square root filter; the numerical precision of the stored variable does not increase due to the factorization. For example, if P is diagonal, $U = I$ and $D = P$; therefore the full range of values in P are preserved in this factorization. However, the UDU formulation of the Kalman filter has great numerical stability properties [3]; it insures symmetry of the covariance by construction, and it requires a trivial check and correction to ensure semi-positive definiteness (it suffices to enforce that the diagonal elements of D remain non-negative). The UDU formulation is free from square root operations, making it computationally cheaper than the Cholesky approach. For these reasons the UDU has endured as one of the preferred practical implementation of Kalman filters in aerospace applications.

While the UDU factorization is well known, it has never been applied to the information formulation of the Kalman filter [13], [14], [15]. In this formulation the inverse of the covariance matrix, known as the information matrix, is carried in the recursive algorithm rather than the covariance matrix itself. The information formulation is a popular approach in several situations. In particular, the Square Root Information Filter (SRIF) [16], [17], [18], [3], [19], [20] is a go-to Kalman filter factorization method used in orbit determination packages such as Monte because of its great stability and accuracy. In this work we introduce the UDU Information Filter, a never developed before algorithm with two key properties that make it a very desirable implementation of a recursive estimator: i. unlike the regular UDU filter, measurements do not need to be processed as scalars, i.e. the measurement noise covariance matrix R does not need to be diagonal or diagonalized, and ii. unlike the regular information formulation the state estimation error covariance matrix does not actually need to be inverted.

C. D'Souza is with the Aeroscience and Flight Mechanics Division, EG6, 2101 NASA Parkway, NASA Johnson Space Center, Houston, Texas 77058.
R. Zanetti is with the Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, Texas 78712

II. Background

The well-known Kalman filter measurement update equations are given by

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + K_k(\mathbf{y}_k - H_k\bar{\mathbf{x}}_k) \tag{1}$$

$$\begin{aligned} P_k &= \bar{P}_k - \bar{P}_k H_k^T (H_k \bar{P}_k H_k^T + R_k)^{-1} H_k \bar{P}_k \\ &= (I - K_k H_k)\,\bar{P}_k \end{aligned} \tag{2}$$

$$K_k = \bar{P}_k H_k^T (H_k \bar{P}_k H_k^T + R_k)^{-1} \ = \ P_k H_k^T R_k^{-1} \tag{3}$$

where the bar represents the *a priori* value, K_k is the $n\times m$ Kalman gain, $\mathbf{x} \in \Re^n$ is the state vector, P_k is the $n\times n$ estimation error covariance matrix, $\mathbf{y} \in \Re^m$ is the measurement vector defined as

$$\mathbf{y}_k = H_k \mathbf{x}_k + \boldsymbol{\eta}_k \tag{4}$$

where $\boldsymbol{\eta}_k$ is a zero mean, white sequence with covariance matrix R_k. The propagation equations are

$$\bar{\mathbf{x}}_{k+1} = \Phi(t_{k+1}, t_k)\hat{\mathbf{x}}_k \tag{5}$$

$$\begin{aligned} \bar{P}_{k+1} &= \Phi(t_{k+1}, t_k) P_k \Phi(t_{k+1}, t_k)^T + G_k Q_k G_k^T \\ &= \Phi_k P_k \Phi_k^T + G_k Q_k G_k^T \end{aligned} \tag{6}$$

where $\Phi(t_{k+1}, t_k)$ (which we will denote as Φ_k) is the state $n\times n$ transition matrix from t_k to t_{k+1}, Q_k is the $p\times p$ process noise covariance matrix, and G_k is the $n \times p$ process noise shaping matrix.

The UDU factorization implements the above equations by replacing the covariance matrix P_k with an upper triangular matrix with ones on the diagonal (U_k) and a diagonal matrix D_k, such that

$$P_k = U_k D_k U_k^T \tag{7}$$

The UDU approach to propagate U_k and D_k forward in time makes use of the Modified Weighted Gram-Schmidt (MWGS) orthogonalization algorithm that avoids loss of orthogonality due to round-off errors [21]. Measurements are processed one at the time as scalars by noting that when R_k is diagonal the update in Eq. (2) is obtained by recursively processing one element of $\mathbf{y}_k$ at a time, using the corresponding row of H_k and diagonal element of R_k. The measurement residual covariance matrix $W_k = H_k \bar{P}_k H_k^T + R_k$ thus becomes a scalar, and the quantity $\bar{P}_k H_k^T = \mathbf{w}_k$ becomes a vector; thus each of the scalar updates takes the form

$$P_k = \bar{P}_k - \frac{1}{W_k}\mathbf{w}_k \mathbf{w}_k^T \tag{8}$$

since matrix $\bar{P}_k$ is updated with a rank one matrix ($\frac{1}{W_k}\mathbf{w}_k\mathbf{w}_k^T$), we call this a rank one update, Agee and Turner [22] detailed how to directly update the U_k and D_k factors due to a rank one update. The subtraction in Eq. (8) could cause some numerical instabilities in Agee-Turner's algorithm. Carlson [23] introduced an alternative rank-one update algorithm that, while less generic, is more stable for the measurement update. Carlson's rank-one update is not valid for generic values of $\mathbf{w}_k$ and W_k, but only when the update is done with the optimal Kalman gain.

An alternative formulation of the Kalman filter is the information formulation, where the covariance matrix P is replaced by its inverse. The covariance update and Kalman gain are calculated as [13]

$$P_k^{-1} = \bar{P}_k^{-1} + H_k^T R_k^{-1} H_k \tag{9}$$

$$K_k = P_k H_k^T R_k^{-1} \tag{10}$$

The information formulation is particularly useful when there is no prior information, i.e. $P_0 = \infty$, in this case the covariance formulation of the KF is not defined, while the information formulation is, and starts from $P_0^{-1} = O$. In the covariance formulation, the $m\times m$ measurement residual covariance matrix $W_k = H_k\bar{P}_k H_k^T + R_k$ is inverted to process the measurement, while in the information formulation the $n \times n$ covariance matrix is inverted in the time propagation step. In situations when $m > n$, therefore, the information formulation could be computationally cheaper, although measurements are often processed one at the time as scalars. Processing measurements as scalars is only possible when R_k is diagonal, otherwise the additional steps of a change of variables to diagonalize R_k is required.

III. The UDU Information Filter

A. The Measurement Update

Begin with factorizing the covariance P into an LDL form; that is, rather than using an upper triangular matrix we will use a lower triangular matrix. We denote the diagonal matrix with Δ

$$P_k \ = \ L_k\,\Delta_k\,L_k^T \quad \text{and} \quad \overline{P}_k \ = \ \overline{L}_k\,\overline{\Delta}_k\,\overline{L}_k^T \tag{11}$$

and we define U and D as the inverses of L^T and Δ, respectively

$$P_k^{-1} = L_k^{-T}\,\Delta_k^{-1}\,L_k^{-1} = U_k\,D_k\,U_k^T \tag{12}$$

$$\overline{P}_k^{-1} = \overline{L}_k^{-T}\,\overline{\Delta}_k^{-1}\,\overline{L}_k^{-1} = \overline{U}_k\,\overline{D}_k\,\overline{U}_k^T \tag{13}$$

so that the measurement update (Eq. (9)) becomes

$$U_k D_k U_k^T = \overline{U}_k \overline{D}_k \overline{U}_k^T + H_k^T R_k^{-1} H_k \tag{14}$$

We now factorize the $m \times m$ matrix R_k into LDL form as

$$R_k = L_{R_k} \Delta_{R_k} L_{R_k}^T \tag{15}$$

and

$$R_k^{-1} = L_{R_k}^{-T} \Delta_{R_k}^{-1} L_{R_k}^{-1} = U_{R_k} D_{R_k} U_{R_k}^T \tag{16}$$

so that Eq. (14) becomes

$$U_k D_k U_k^T = \overline{U}_k \overline{D}_k \overline{U}_k^T + H_k^T U_{R_k} D_{R_k} U_{R_k}^T H_k \tag{17}$$

We now work on the term $U_{R_k} H$ where we note that it is of dimension $m \times n$ so that it can be expressed as

$$U_{R_k}^T H_R = \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_m^T \end{bmatrix} \tag{18}$$

where each $\mathbf{v}_i$ is an $n \times 1$ vector.

The factor $H_k^T R_k^{-1} H_k$ can be expressed as

$$\begin{aligned} H_k^T R_k^{-1} H_k &= H_k^T U_{R_k} D_{R_k} U_{R_k}^T H_k \\ &= \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_m^T \end{bmatrix}^T \begin{bmatrix} 1/d_{1_R} & 0 & \cdots & 0 \\ 0 & 1/d_{2_R} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/d_{m_R} \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_m^T \end{bmatrix} \\ &= \sum_{i=1}^{m} \frac{1}{d_{i_R}} \mathbf{v}_i \mathbf{v}_i^T \end{aligned} \tag{19}$$

so that the measurement update equation is now

$$U_k D_k U_k^T = \overline{U}_k \overline{D}_k \overline{U}_k^T + \sum_{i=1}^{m} \frac{1}{d_{i_R}} \mathbf{v}_i \mathbf{v}_i^T \tag{20}$$

Thus it reduces to a series of m rank-one updates.

Notice that Eq. (20) is the update equation due to a vector measurement so that the need to process scalar measurements, as in the covariance UDU formulation, is avoided in the proposed information UDU formulation. Rather than performing an eigenvalue decomposition of R_k and a corresponding change of variables for $\mathbf{y}_k$, we simply perform the UDU factorization of R_k^{-1}.

As stated earlier, one of the benefits of using an information formulation is that if P_0 is singular, this allows for an estimate to be obtained [14]. When P_0 is singular, $\mathbf{x}_0$ is not completely defined. To this end, $\hat{\mathbf{z}}_k$ and $\bar{\mathbf{z}}_k$, which are directly related to $\hat{\mathbf{x}}_k$ and $\bar{\mathbf{x}}_k$, are defined as

$$\hat{\mathbf{z}}_k \triangleq P_k^{-1} \hat{\mathbf{x}}_k\,, \qquad \bar{\mathbf{z}}_k \triangleq \bar{P}_k^{-1} \bar{\mathbf{x}}_k \tag{21}$$

Premultiplying Eq. (1) by P_k^{-1}, we get

$$\hat{\mathbf{z}}_k = P_k^{-1} (I - K_k H_k)\,\bar{\mathbf{x}}_k + P_k^{-1} K_k \mathbf{y}_k \tag{22}$$

$$= \bar{\mathbf{z}}_k + H_k^T R_k^{-1} \mathbf{y}_k \tag{23}$$

B. The Time Update

Prior to propagation, the standard information formulation of the Kalman filter inverts the information matrix to obtain the covariance, it then propagates the covariance with Eq. (6), and finally inverted the propagated covariance matrix to prepare for the measurement update phase. We propose an algorithm that propagates the factors of the information matrix directly. Starting from the covariance propagation Eq. (6), we factorize Q_k via a UDU parameterization so that

$$Q_k = U_{Q_k}\,\Delta_{Q_k} U_{Q_k}^T \tag{24}$$

where Δ_{Q_k} is a diagonal $p \times p$ matrix and define G_{Q_k} as

$$G_{Q_k} \triangleq G_k\, U_{Q_k} \tag{25}$$

so that $\overline{P}_k$ becomes

$$\overline{P}_{k+1} = \Phi_k\, P_k\, \Phi_k^T + G_{Q_k}\,\Delta_{Q_k}\, G_{Q_k}^T \tag{26}$$

Invoking the matrix inversion lemma

$$(Z + XAY)^{-1} = Z^{-1} - Z^{-1}X\left(A^{-1} + Y\,Z^{-1}X\right)^{-1} Y Z^{-1} \tag{27}$$

and letting

$$Z = \Phi_k\, P_k\, \Phi_k^T; \qquad A = \Delta_{Q_k}; \qquad X = G_{Q_k}; \qquad Y = G_{Q_k}^T \tag{28}$$

and

$$Z^{-1} = M_k \triangleq \Phi_k^{-T}\, P_k^{-1}\, \Phi_k^{-1} \tag{29}$$

The inverse of the propagated covariance is

$$\overline{P}_{k+1}^{-1} = M_k - M_k\, G_{Q_k}\left[G_k^T\, M_k\, G_{Q_k} + \Delta_{Q_k}^{-1}\right]^{-1} G_{Q_k}^T M_k \tag{30}$$

Defining

$$\overline{G}_k \triangleq \Phi_k^{-1}\, G_{Q_k} \qquad D_{Q_k} = \Delta_{Q_k}^{-1} \tag{31}$$

P_{k+1}^{-1} becomes

$$\begin{aligned}\overline{P}_{k+1}^{-1} = \Phi_k^{-T}\Big\{ & P_k^{-1} - \\ & P_k^{-1}\,\overline{G}_k\left[\overline{G}_k^T\, P_k^{-1}\,\overline{G}_k + D_{Q_k}\right]^{-1}\overline{G}_k^T\, P_k^{-1}\Big\}\Phi_k^{-1}\end{aligned} \tag{32}$$

and defining $\mathcal{K}_k$ as

$$\mathcal{K}_k \triangleq P_k^{-1}\,\overline{G}_k\left[\overline{G}_k^T\, P_k^{-1}\,\overline{G}_k + D_{Q_k}\right]^{-1} \tag{33}$$

Defining the quantity inside the brackets in Eq. (32) as $\mathcal{P}_k^{-1}$

$$\begin{aligned}\mathcal{P}_k^{-1} &\triangleq P_k^{-1} - P_k^{-1}\,\overline{G}_k\left[\overline{G}_k^T\, P_k^{-1}\,\overline{G}_k + D_{Q_k}\right]^{-1}\overline{G}_k^T P_k^{-1} \\ &= \left[I - \mathcal{K}_k\overline{G}_k^T\right] P_k^{-1}\end{aligned} \tag{34}$$

We notice Eq. (34) is an analog to Eq. (2) with

$$\begin{array}{lll}\mathcal{P}_k^{-1} \to P_k & \overline{G}_k \to H_k^T & D_{Q_k} \to R_k \\ P_k^{-1} \to \bar{P}_k & \mathcal{K}_k \to K_k & \end{array}$$

and since D_{Q_k} is a diagonal $p \times p$ matrix, we can solve for the UDU factorization of $\mathcal{P}_k^{-1}$ directly by using a Carlson Rank-One Update [23] performed p times on

$$\begin{aligned}\mathcal{U}_k\mathcal{D}_k\mathcal{U}_k^T = {} & U_k D_k U_k^T - \\ & U_k D_k U_k^T\,\overline{G}_k\left[\overline{G}_k^T\, U_k D_k U_k^T\,\overline{G}_k + \Delta_{Q_k}^{-1}\right]^{-1}\overline{G}_k^T U_k D_k U_k\end{aligned} \tag{35}$$

so that we can find the time-propagated UDU factors of $\overline{P}_k^{-1}$ as

$$\overline{P}_{k+1}^{-1} = \overline{U}_{k+1}\overline{D}_{k+1}\overline{U}_{k+1}^T = \Phi_k^{-T}\,\mathcal{U}_k\mathcal{D}_k\mathcal{U}_k^T\,\Phi_k^{-1} \tag{36}$$

Since this equation is equivalent to a covariance propagation without process noise, the MWGS orthogonalization algorithm can be used to obtain the factors $\overline{U}_{k+1}$ and $\overline{D}_{k+1}$.

Notice that Φ_k^{-1} does not necessarily need to be computed by a direct matrix inversion. Usually Φ_k is computed via integration of a matrix differential equation or by series approximation. Similarly, Φ_k^{-1} can be obtained directly by backwards integration or by series approximation.

Beginning with Eq. (5) the time update for $\bar{\mathbf{z}}_{k+1}$ is obtained as follows

$$\bar{P}_{k+1}^{-1}\bar{\mathbf{x}}_{k+1} = \bar{P}_{k+1}^{-1}\Phi_k\, P_k\, P_k^{-1}\hat{\mathbf{x}}_k \tag{37}$$

which becomes

$$\bar{\mathbf{z}}_{k+1} = \bar{P}_{k+1}^{-1}\Phi_k\, P_k\,\hat{\mathbf{z}}_k \tag{38}$$

substituting from Eqs. (32) and (34)

$$\bar{\mathbf{z}}_{k+1} = \Phi_k^{-T}\left[I - \mathcal{K}_k\overline{G}_k^T\right]\hat{\mathbf{z}}_k \tag{39}$$

or

$$\bar{\mathbf{z}}_{k+1} = \Phi^{-T}(t_{k+1}, t_k)\left[I - \mathcal{K}_k\overline{G}_k^T\right]\hat{\mathbf{z}}_k \tag{40}$$

The following table summarizes the UDU Information Filter. While Table I contains a compact notation for the covariance time propagation and measurement update; in the actual algorithm the covariance factors U_k and D_k are individually propagated and updated using the Rank-1 Update and the Modified Weighted Gram-Schmidt orthogonalization algorithms.

Initialization	
State	$\mathbf{z}_0 = P^{-1}(t_0)\mathbf{x}(t_0)$
Covariance	$U_0 D_0 U_0^T = P^{-1}(t_0)$
Time Propagation	
Truth	$\mathbf{x}_{k+1} = \Phi_k\mathbf{x}_k + G_k\boldsymbol{\nu}_k, \quad \boldsymbol{\nu}_k \sim n(\mathbf{0}, Q_k)$
Process Noise	$U_{Q_k}\Delta_{Q_k}U_{Q_k}^T = Q_k, \quad \overline{G}_k = \Phi_k^{-1}G_k\, U_{Q_k}$
Gain	$\mathcal{K}_k = U_k D_k U_k^T\overline{G}_k\left[\overline{G}_k^T U_k D_k U_k^T\overline{G}_k + \Delta_{Q_k}\right]^{-1}$
Covariance	$\mathcal{U}_k\mathcal{D}_k\mathcal{U}_k^T = \left[I - \mathcal{K}_k\overline{G}_k^T\right]U_k D_k U_k^T$ (Rank-1 Update) $\overline{U}_{k+1}\overline{D}_{k+1}\overline{U}_{k+1}^T = \Phi_k^{-T}\mathcal{U}_k\mathcal{D}_k\mathcal{U}_k^T\Phi_k^{-1}$ (MWGS)
State	$\bar{\mathbf{z}}_{k+1} = \Phi_k^{-T}\left[I - \mathcal{K}_k\overline{G}_k^T\right]\hat{\mathbf{z}}_k$
Measurement Update	
Truth	$\mathbf{y}_{k+1} = H_{k+1}\mathbf{x}_k + \boldsymbol{\eta}_{k+1}, \quad \boldsymbol{\eta}_{k+1} \sim n(\mathbf{0}, R_{k+1})$
Meas. Noise	$U_{R_{k+1}}D_{R_{k+1}}U_{R_{k+1}}^T = R_{k+1}^{-1}$ $\sum_{i=1}^m \frac{1}{d_{i_R}}\mathbf{v}_i\mathbf{v}_i^T = H_{k+1}^T U_{R_{k+1}}D_{R_{k+1}}U_{R_{k+1}}^T H_{k+1}$
Covariance	$U_{k+1}D_{k+1}U_{k+1}^T = \overline{U}_{k+1}\overline{D}_{k+1}\overline{U}_{k+1}^T +$ $+\sum_{i=1}^m (1/d_{i_R})\,\mathbf{v}_i\mathbf{v}_i^T$ (Rank-1 Update)
State	$\hat{\mathbf{z}}_{k+1} = \bar{\mathbf{z}}_{k+1} + H_{k+1}R_{k+1}^{-1}\mathbf{y}_{k+1}$

TABLE I
SUMMARY OF UDU INFORMATION FILTER EQUATIONS

C. An Efficient Algorithm to compute U^{-1}

The algorithm proposed does not necessitate to invert the covariance matrix nor its U or L factor. However, in case the initial covariance was provided, it might be convenient to factorize it first, and to efficiently invert its factors rather than inverting the full covariance. In this section we compute the inverse in an efficient manner, taking advantage of the '1's' and '0's'. It is as follows: Given an $n \times n$ upper triangular 'unit' matrix U expressed as

$$U = \begin{bmatrix} 1 & U_{1,2} & U_{1,3} & \cdots & U_{1,n-1} & U_{1,n} \\ 0 & 1 & U_{2,3} & \cdots & U_{2,n-1} & U_{2,n} \\ 0 & 0 & 1 & \cdots & U_{3,n-1} & U_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & U_{n-1,n} \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{41}$$

the inverse is also an $n \times n$ upper triangular 'unit' matrix V (so that $\det(V) = 1$) which is

$$V = U^{-1} = \begin{bmatrix} 1 & V_{1,2} & V_{1,3} & \cdots & V_{1,n-1} & V_{1,n} \\ 0 & 1 & V_{2,3} & \cdots & V_{2,n-1} & V_{2,n} \\ 0 & 0 & 1 & \cdots & V_{3,n-1} & V_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & V_{n-1,n} \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{42}$$

since

$$U\,V \quad = \quad I \tag{43}$$

and the ij-th element of UV is given by

$$\sum_{k=i}^{j} U_{i,k}V_{k,j} = U_{i,i}V_{i,j} + U_{i,j}V_{j,j} + \sum_{k=i+1}^{j-1} U_{i,k}V_{k,j} = V_{i,j} + U_{i,j} + \sum_{k=i+1}^{j-1} U_{i,k}V_{k,j} \tag{44}$$

we can solve for the elements of V as

$$j = n, \cdots, 2, \qquad i = j-1, \cdots, 1$$

$$V_{i,j} = -\left[U_{i,j} + \sum_{k=i+1}^{j-1} U_{i,k}V_{k,j} \right] \tag{45}$$

IV. A Numerical Example

In this section we show the performance of the algorithm in linear, time-varying example with correlated measurement noise covariance matrix R. The system is given by

$$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \boldsymbol{\nu}_k \tag{46}$$

$$\mathbf{y}_k = H_k \mathbf{x}_k + \boldsymbol{\eta}_k \tag{47}$$

$$\Phi_k = \begin{bmatrix} I & A_k \\ B_k & I \end{bmatrix} \tag{48}$$

$$A_k = \begin{bmatrix} t_k - t_{k-1} & 0 \\ 0 & t_k - t_{k-1} \end{bmatrix} \tag{49}$$

$$B_k = 0.1 \begin{bmatrix} \sin(t_k) - \sin(t_{k-1}) & -\big(\cos(t_k) - \cos(t_{k-1})\big) \\ 0 & \sin(t_k) - \sin(t_{k-1}) \end{bmatrix} \tag{50}$$

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{51}$$

where I is the identity matrix $t_k - t_{k-1} = 1$ second, $\boldsymbol{\nu}_k$ is a zero mean, Gaussian white sequence with covariance matrix $Q_k = 0.01\ I$, and $\boldsymbol{\eta}_k$ is a zero mean, Gaussian white sequence with covariance matrix R_k

$$R_k = \begin{bmatrix} 2.96 & 2.8 \\ 2.8 & 2.96 \end{bmatrix} \tag{52}$$

The initial estimate is unbiased, and the initial estimation error is Gaussian with covariance $P_0 = I$. Fig. 1 shows the result of a single run and the 3σ predicted standard deviations from a straight formulation of a Kalman filter. In order to show the equivalence between the Kalman filter (KF) and the UDU information approach (UDUI), Fig. 2 shows the norm of the difference between the two state estimates

$$\epsilon_{\mathbf{x}} = \|\hat{\mathbf{x}}_{KF} - \hat{\mathbf{x}}_{UDUI}\|$$

while Fig. 3 compares the Kalman filter covariance P_{KF} with the UDU factorization of the Information matrix $P_{UDUI}^{-1} = UDU^T$ by plotting the following quantity:

$$\epsilon_P = \|P_{KF}\ P_{UDUI}^{-1} - I\|$$

The figures show that the proposed algorithm results closely match the Kalman filter hence validating the proposed algorithm as its UDU information formulation. The growth of the error in Figs. 2 and 3 is due to the accumulation of round-off errors in the algorithms. It is known [3] that numerical errors accumulate faster in the full covariance formulation of the Kalman filter than in the UDU's.

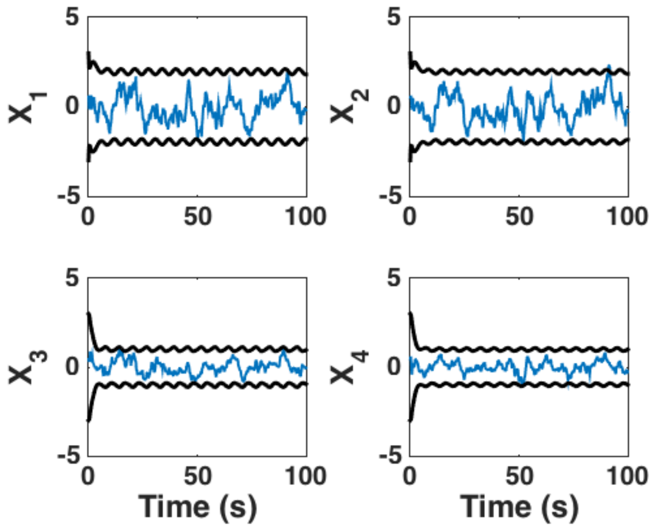

Fig. 1. Estimation Error and 3σ predicted standard deviations

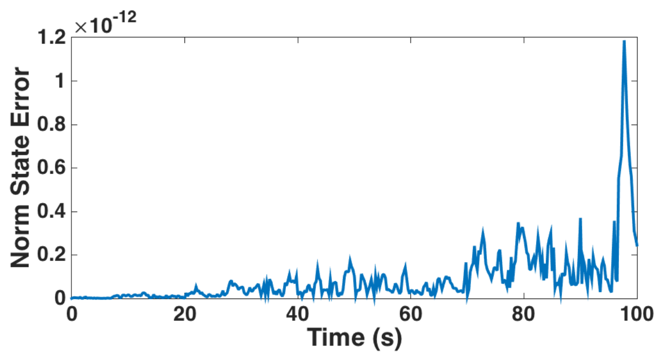

Fig. 2. Norm of the State Error ($\|\hat{\mathbf{x}}_{KF} - \hat{\mathbf{x}}_{UDUI}\|$)

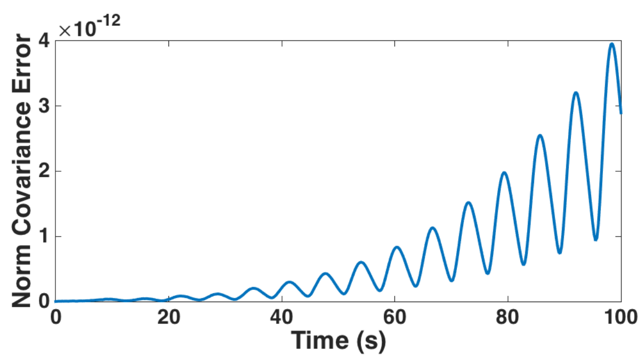

Fig. 3. Norm of the Covariance error ($\|P_{KF}\ P_{UDUI}^{-1} - I\|$)

V. Conclusions

A new algorithmic mechanization of the classic Kalman filter is presented, the new algorithm combines the information formulation with the UDU factorization. While the covariance formulation of the Kalman filter is usually employed, the information formulation has distinct advantages in some applications, for example when no initial condition is available. The UDU factorization is a widely adopted technique to produce a numerically stable and accurate algorithm to keep the covariance matrix symmetric and positive definite. A numerical example confirms the equivalency between the Kalman filter and the proposed algorithm.

Acknowledgments

This work was supported in part by NASA JSC contract NNX17AI35A.

REFERENCES

[1] C. Thornton, "Triangular Covariance Factorizations for Kalman Filtering," Ph.D. dissertation, University of California at Los Angeles, October 1976.

[2] G. Bierman and C. Thornton, "Numerical comparison of kalman filter algorithms: Orbit determination case study," *Automatica*, vol. 13, pp. 23–25, 1977.

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

[4] S. Evans, W. Taber, T. Drain, J. Smith, H.-C. Wu, M. Guevara, R. Sunseri, and J. Evans, "Monte: The next generation of mission design and navigation software," *6th International Conference on Astrodynamics Tools and Techniques (ICATT)*, March 2016.

[5] J. Sud, R. Gay, G. Holt, and R. Zanetti, "Orion Exploration Flight Test 1 (EFT1) Absolute Navigation Design," in *Proceedings of the AAS Guidance and Control Conference*, ser. Advances in the Astronautical Sciences, vol. 151, Breckenridge, CO, January 31–February 5, 2014 2014, pp. 499–509, aAS 14-092.

[6] R. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME (J. Basic Eng.)*, vol. 82D, pp. 34–45, March 1960.

[7] J. Potter and R. Stern, "Statistical filtering of space navigation measurements," *Proceedings of the AIAA Guidance and Control Conference*, 1963.

[8] R. Battin, *An Introduction to the Mathematics and Methods of Astrodynamics*. Reston, VA: AIAA, 1999.

[9] G. H. Golub and C. F. V. Loan, *Matrix Computations*, 2nd ed. Baltimore, MD: John Hopkins University Press, 1989, pp. 141–142.

[10] M. Verhaegen and P. Van Dooren, "Numerical aspects of different kalman filter implementations," *IEEE Transactions on Automatic Control*, vol. 31, no. 10, pp. 907–917, October 1986.

[11] P. Kaminski, A. Bryson, and S. Schmidt, "Discrete square root filtering: A survey of current techniques," *IEEE Transactions on Automatic Control*, vol. 16, no. 6, pp. 727–736, December 1971.

[12] Y. Oshman and I. Y. Bar-Itzhack, "Square root filtering via covariance and information eigenfactors," *Automatica*, vol. 22, no. 5, pp. 599–604, 1986.

[13] R. G. Brown and P. Y. Hwang, *Introduction To Random Signals And Applied Kalman Filtering*, 3rd ed. John Wiley and Sons, 1997, pp. 246–250.

[14] P. Maybeck, *Stochastic Models, Estimation and Control, Vol. 1*. New York, NY: Academic Press, 1979.

[15] ——, *Stochastic Models, Estimation and Control, Vol. 2*. New York, NY: Academic Press, 1979.

[16] P. Dyer and S. McReynolds, "Extension of square-root filtering to include process noise," *Journal of Optimization Theory and Applications*, vol. 3, no. 6, pp. 444–458, December 1969.

[17] G. Golub, "Numerical methods for solving linear least squares problems," *Numerische Mathematik*, vol. 7, no. 3, pp. 206–216, Jun 1965, doi: 10.1007/BF01436075.

[18] R. J. Hanson and C. L. Lawson, "Extensions and applications of the householder algorithm for solving linear least squares problems," *Mathematics of Computation*, vol. 23, no. 108, pp. 787–812, 1969.

[19] G. Bierman, "Sequential square root filtering and smoothing of discrete linear systems," *Automatica*, vol. 10, pp. 147–158, 1974.

[20] ——, "The treatment of bias in the square root information filter/smoother," *Journal of Optimization Theory and Applications*, vol. 16, no. 1/2, pp. 165–178, July 1975.

[21] C. Thornton and G. Bierman, "Gram-schmidt algorithms for covariance propagation," *IEEE Conference on Decision and Control*, pp. 489–498, 1975.

[22] 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.

[23] N. Carlson, "Fast triangular factorization of the square root filter," *AIAA Journal*, vol. 11, no. 9, pp. 1259–1265, September 1973.