Research Article

Iterative and Algebraic Algorithms for the Computation of the Steady State Kalman Filter Gain

Nicholas Assimakis\textsuperscript{1,2} and Maria Adam\textsuperscript{2}

\textsuperscript{1} Department of Electronic Engineering, Technological Educational Institute of Central Greece, 3rd km Old National Road Lamia-Athens, 35100 Lamia, Greece
\textsuperscript{2} Department of Computer Science and Biomedical Informatics, University of Thessaly, 2-4 Papasiopoulou Street, 35100 Lamia, Greece

Correspondence should be addressed to Maria Adam; madam@dib.uth.gr

Received 24 February 2014; Accepted 31 March 2014; Published 4 May 2014

Academic Editors: F. Ding, L. Guo, and H. C. So

Copyright © 2014 N. Assimakis and M. Adam. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Kalman filter gain arises in linear estimation and is associated with linear systems. The gain is a matrix through which the estimation and the prediction of the state as well as the corresponding estimation and prediction error covariance matrices are computed. For time invariant and asymptotically stable systems, there exists a steady state value of the Kalman filter gain. The steady state Kalman filter gain is usually derived via the steady state prediction error covariance by first solving the corresponding Riccati equation. In this paper, we present iterative per-step and doubling algorithms as well as an algebraic algorithm for the steady state Kalman filter gain computation. These algorithms hold under conditions concerning the system parameters. The advantage of these algorithms is the autonomous computation of the steady state Kalman filter gain.

1. Introduction

The Kalman filter gain arises in Kalman filter equations in linear estimation and is associated with linear systems. State space systems have been widely used in estimation theory to describe discrete time systems [1–5]. It is known [1] for time invariant systems that if the signal process model is asymptotically stable, then there exists a steady state value of the Kalman filter gain. Thus, the steady state gain is associated with time invariant systems described by the following state space equations:

\[
\begin{align*}
x_{k+1} &= Fx_k + w_k, \\
z_k &= Hx_k + v_k,
\end{align*}
\]

for \( k \geq 0 \), where \( x_k \) is the \( n \)-dimensional state vector at time \( k \), \( z_k \) is the \( m \)-dimensional measurement vector at time \( k \), \( F \) is the \( n \times n \) system transition matrix, \( H \) is the \( m \times n \) output matrix, \( w_k \) is the plant noise at time \( k \), and \( v_k \) is the measurement noise at time \( k \). Also, \( \{w_k\} \) and \( \{v_k\} \) are Gaussian zero-mean white random processes with covariance matrices \( Q \) and \( R \), respectively.

The discrete time Kalman filter [1, 6] is the most well-known algorithm that solves the filtering problem. In fact, Kalman filter faces simultaneously two problems as follows.

(i) **Estimation**: the aim is to recover at time \( k \) information about the state vector at time \( k \) using measurements up till time \( k \).

(ii) **Prediction**: the aim is to obtain at time \( k \) information about the state vector at time \( k+1 \) using measurements up till time \( k \); it is clear that prediction is related to the forecasting side of information processing.

Kalman filter uses the measurements up till time \( k \) in order to produce the (one step) prediction of the state vector and the corresponding prediction error covariance matrix \( P_{k+1/k} \), as well as producing the estimation of the state vector and the corresponding estimation error covariance matrix \( P_{k/k} \). The Kalman filter equations needed for the computation of
the prediction and estimation error covariance matrices are as follows:

\[ K_k = P_{k/\nu - 1} H^T [HP_{k/\nu - 1} H^T + R]^{-1}, \]

(2)

\[ P_{k/\nu} = (I - K_k H) P_{k/\nu - 1}, \]

(3)

\[ P_{k + 1/\nu} = Q + F P_{k/\nu} F^T, \]

(4)

for \( k \geq 0 \), with initial condition \( P_{0/\nu - 1} = P_0 \) for the time instant, where there are no measurements given. Note that \( K_k \) is the Kalman filter gain.

From (2) to (4), we are able to derive the Riccati equation, which is an iterative equation with respect to the prediction error covariance:

\[ P_{k + 1/\nu} = Q + F P_{k/\nu} F^T - F P_{k/\nu} H^T [HP_{k/\nu - 1} H^T + R]^{-1} HP_{k/\nu - 1} F^T. \]

(5)

In the general case, where \( R \) and \( P_0 \) are positive definite matrices, using in (5) the matrix inversion lemma:

\[ [A + BCD]^{-1} = A^{-1} - A^{-1} B [C^{-1} + DA^{-1} B]^{-1} DA^{-1}, \]

(6)

the Riccati equation is formulated as

\[ P_{k + 1/\nu} = Q + F P_{k/\nu}^{-1} + H^T R^{-1} H]^{-1} F^T. \]

(7)

The Riccati equation is a nonlinear iterative equation with respect to the prediction error covariance. For time invariant systems, it is well known [1] that if the signal process model is asymptotically stable, then there exists a steady state value \( \bar{P}_p \) of the prediction error covariance matrix. In fact, the prediction error covariance tends to the steady state prediction error covariance.

The steady state prediction error covariance satisfies the steady state Riccati equation

\[ \bar{P}_p = Q + F \bar{P}_p^{-1} + H^T R^{-1} H]^{-1} F^T. \]

(8)

Then, from (2), it is clear that there also exists a steady state value \( \bar{K} \) of the Kalman filter gain [7]. The steady state gain can be calculated by

\[ \bar{K} = \bar{P}_p H^T [HP_p H^T + R]^{-1}. \]

(9)

Also, from (3), it is clear that there also exists a steady state value \( \bar{P}_p \) of the estimation error covariance matrix [7], which can be calculated by

\[ \bar{P}_p = \bar{P}_p - \bar{P}_p H^T [HP_p H^T + R]^{-1} H \bar{P}_p \bar{P}_p = \left( \bar{P}_p^{-1} + H^T R^{-1} H \right)^{-1}. \]

(10)

It is obvious from (9) that the steady state Kalman filter gain can be derived via the steady state prediction error covariance. The covariance matrix in Kalman filter plays an important role in many applications [1, 4, 6, 8–10]. The steady state prediction error covariance can be derived by solving the Riccati equation emanating from Kalman filter. The discrete time Riccati equation has attracted recent attention. In view of the importance of the Riccati equation, there exists considerable literature on its algebraic solutions; for example, in [1, 7, 11, 12], the authors have derived an eigenvector solution, while the author of [13] has included solving scalar polynomials. Other methods are based on the iterative solutions [1, 13–18] concerning per-step or doubling algorithms. The iterative algorithms that provide the steady state Kalman filter gain together with the prediction error covariance are the Chandrasekhar algorithms [1], as well as the iterative algorithm that calculate the Kalman gain only once for a period of the stationary channel, as opposed to each data sample in the conventional filter [19]. A geometric illustration of the Kalman filter gain is given in [20].

In this paper, we present algorithms for the steady state Kalman filter gain autonomous computation. These algorithms hold under conditions concerning the system parameters. The paper is organized as follows: two new per-step iterative algorithms, a new doubling iterative algorithm, and an algebraic algorithm for the computation of the steady state Kalman filter gain are presented in Section 2. In Section 3, two examples verify the results. Finally, Section 4 summarizes the conclusions.

2. New Algorithms for the Steady State Kalman Filter Computation

2.1. Assumptions. We assume the general case, where \( R \) and \( P_0 \) are positive definite matrices.

The Kalman filter gain \( K_k \) is a matrix of dimension \( n \times m \). We define the matrix

\[ G_k = K_k H. \]

(11)

It is clear that \( G_k \) is a nonsymmetric matrix of dimension \( n \times n \). It is also clear that there exists a steady state value

\[ \bar{G} = \bar{K} H. \]

(12)

Also, we define the matrix

\[ S = H^T R^{-1} H. \]

(13)

Note that \( S \) is an \( n \times n \) symmetric positive semidefinite matrix and \( S \) is a positive definite if \( \text{rank}(H) = n \); this means that \( S \) is a nonsingular matrix in the case, \( \text{rank}(H) = n \) with \( m \geq n \), [21].

2.2. Indirect Steady State Kalman Filter Gain Computation. In this section, we present algorithms for \( \bar{G} \) computation. Then, we show how to compute the steady state Kalman filter \( \bar{K} \) through \( \bar{G} = \bar{K} H \).

2.2.1. Iterative Algorithms for \( \bar{G} \) Computation. In this section, we present two iterative per-step algorithms and an iterative doubling algorithm for \( \bar{G} \) computation.
Per-Step Iterative Algorithm 1. Using (2) and (11), it is derived that

\[ G_k = K_k H \]

\[ = P_{k/k-1} H^T [HP_{k/k-1} H^T + R]^{-1} H \]

\[ = P_{k/k-1} H^T [HP_{k/k-1} H^T + R]^{-1} RR^{-1} H \]

\[ = P_{k/k-1} H^T \left( [HP_{k/k-1} H^T + R] \times \left( [HP_{k/k-1} H^T + R] - [HP_{k/k-1} H^T + R]^{-1} HH_{k/k-1} H^T \right) R^{-1} H \right) \]

\[ = P_{k/k-1} H^T \left( I - \left( HP_{k/k-1} H^T + R \right)^{-1} HP_{k/k-1} H^T \right) R^{-1} H \]

\[ = P_{k/k-1} H^T R^{-1} H \]

\[ = P_{k/k-1} H^T \left( HP_{k/k-1} H^T + R \right)^{-1} HP_{k/k-1} H^T R^{-1} H \]

\[ = \left( P_{k/k-1} - P_{k/k-1} H^T \left( HP_{k/k-1} H^T + R \right)^{-1} \right) H^T R^{-1} H \]

\[ = \left[ P_{k/k-1}^{-1} + H^T R^{-1} H \right]^{-1} H^T R^{-1} H \]

\[ = \left[ P_{k/k-1}^{-1} + S \right]^{-1} S. \] (14)

Thus, arises

\[ K_k H = \left[ P_{k/k-1}^{-1} + S \right]^{-1} S. \] (15)

Using the Riccati equation (7), (15), the nonsingularity of S, and some algebra we have

\[ P_{k+1/k} = Q + F \left( P_{k/k-1}^{-1} + H^T R^{-1} H \right)^{-1} F^T \]

\[ = Q + F \left( P_{k/k-1}^{-1} + S \right)^{-1} F^T = Q + FK_k HS^{-1} F^T \]

\[ \Rightarrow P_{k+1/k} S = QS + FK_k HS^{-1} F^T S \]

\[ = QF^{-T} S \left( S^{-1} F^{-T} S \right)^{-1} + FK_k H \left( S^{-1} F^{-T} S \right)^{-1} \]

\[ \Rightarrow P_{k+1/k} S \left( S^{-1} F^{-T} S \right)^{-1} = QF^{-T} S + FK_k H \]

\[ \Rightarrow S^{-1} F^{-T} S = S^{-1} P_{k+1/k} \left( QF^{-T} S + FK_k H \right) \]

\[ \Rightarrow K_{k+1} H \left( S^{-1} F^{-T} S \right) = K_{k+1} H \left( S^{-1} F^{-T} S + QF^{-T} S + FK_k H \right) \]

\[ \Rightarrow K_{k+1} H = \left( QF^{-T} S + FK_k H \right) \left( S^{-1} F^{-T} S + QF^{-T} S + FK_k H \right)^{-1}. \] (16)

Thus, the above equation can be written as

\[ K_{k+1} H = \left( QF^{-T} S + FK_k H \right) \left( S^{-1} F^{-T} S + QF^{-T} S + FK_k H \right)^{-1}. \] (21)

Combining (21) with (11), the following nonlinear iterative equation with respect to \( G_k \) is derived:

\[ G_{k+1} = \left( \left( QF^{-T} S + FK_k H \right) \times \left( S^{-1} F^{-T} S + QF^{-T} S + FK_k H \right)^{-1} \right) \]

\[ = \left( C + DG_k \right) \left( A + BG_k \right)^{-1}. \] (22)
where
\[ A = (Q + S^{-1}) F^{-T} S, \]
\[ B = F, \]
\[ C = QF^{-T} S, \]
\[ D = F. \]  \hfill (23)\]

The algorithm uses the initial condition \( G_0 = K_0 H = P_0 H^T [H P_0 H^T + R]^{-1} H \). It is known [1] that the prediction error covariance tends to the steady state prediction error covariance and that the convergence is independent of the initial uncertainty, that is, independent of the value of the initial condition \( P_0 \). Thus, we are able to assume zero initial condition \( P_0 = 0 \) and so we are to use the initial condition \( G_0 = 0 \).

It is clear that \( G_k \) tends to a steady state value \( \bar{G} \) and by (22) \( \bar{G} \) satisfies
\[ \bar{G} = (C + D \bar{G}) \left[ A + B \bar{G} \right]^{-1}. \]  \hfill (24)\]

**Per-Step Iterative Algorithm 2.** We rewrite (22) as
\[ G_{k+1} = (C + D G_k) \left[ A + B G_k \right]^{-1} \]
\[ = (CA^{-1} A + (D + CA^{-1} B - CA^{-1} B) G_k) \left[ A + B G_k \right]^{-1} \]
\[ = (CA^{-1} A + B G_k) + (D - CA^{-1} B) G_k \left[ A + B G_k \right]^{-1} \]
\[ = CA^{-1} + (D - CA^{-1} B) \left[ G_k A + B G_k \right]^{-1} \]
\[ = CA^{-1} + (D - CA^{-1} B) \left[ c^{-1} + A^{-1} B \right]^{-1} A^{-1}. \]  \hfill (25)\]

Thus, the following nonlinear iterative equation with respect to \( G_k \) is derived:
\[ G_{k+1} = c + a \left[ G_k^{-1} + b \right]^{-1} d, \]  \hfill (26)\]

where
\[ a = D - CA^{-1} B = F - Q \left[ Q + S^{-1} \right]^{-1} F, \]
\[ b = A^{-1} B = S^{-1} F^T \left[ Q + S^{-1} \right]^{-1} F, \]
\[ c = CA^{-1} = Q \left[ Q + S^{-1} \right]^{-1}, \]
\[ d = A^{-1} = S^{-1} F^T \left[ Q + S^{-1} \right]^{-1}. \]  \hfill (27)\]

The algorithm uses the initial condition \( G_0 = K_0 H = P_0 H^T [H P_0 H^T + R]^{-1} H \). It is known [1] that the prediction error covariance tends to the steady state prediction error covariance and that the convergence is independent of the initial uncertainty, that is, independent of the value of the initial condition \( P_0 \). Thus, we are able to assume zero initial condition \( P_0 = 0 \) and so we are to use the initial condition \( G_0 = c \).

It is clear that \( G_k \) tends to a steady state value \( \bar{G} \) and by (26) \( \bar{G} \) satisfies
\[ \bar{G} = c + a \left[ \bar{G}^{-1} + b \right]^{-1} d. \]  \hfill (28)\]

**Doubling Iterative Algorithm.** In (22), setting
\[ G_k = K_k H = Y_k X_k^{-1}, \]  \hfill (29)\]
we take
\[ Y_{k+1} X_{k+1}^{-1} = (C + D Y_k X_k^{-1}) \left[ A + B Y_k X_k^{-1} \right]^{-1} \]
\[ = (C X_k + D Y_k) \left[ A X_k + B Y_k \right]^{-1} \]
or
\[ \begin{bmatrix} X_{k+1} \\ Y_{k+1} \end{bmatrix} = \Phi \begin{bmatrix} X_k \\ Y_k \end{bmatrix}, \]  \hfill (31)\]

where
\[ \Phi = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} (Q + S^{-1}) F^{-T} S & F \\ QF^{-T} S & F \end{bmatrix} \]
\[ = \begin{bmatrix} d^{-1} & d^{-1} b \\ cd^{-1} & cd^{-1} b + a \end{bmatrix} \]  \hfill (32)\]
is a matrix of dimension \( 2n \times 2n \) and \( A, B, C, D \) as in (23).

We are able to use zero initial condition \( P_0 = 0 \), so \( G_0 = K_0 H = Y_0 X_0^{-1} = 0 \); that is,
\[ \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \]  \hfill (33)\]
and hence
\[ \begin{bmatrix} X_k \\ Y_k \end{bmatrix} = \Phi^k \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \]  \hfill (34)\]

We define
\[ \Phi^x = \begin{bmatrix} d_k^{-1} & d_k^{-1} b_k \\ c_k d_k^{-1} & c_k d_k^{-1} b_k + a_k \end{bmatrix} \]  \hfill (35)\]
with initial condition
\[ \Phi^0 = \begin{bmatrix} d_0^{-1} & d_0^{-1} b_0 \\ c_0 d_0^{-1} & c_0 d_0^{-1} b_0 + a_0 \end{bmatrix} = \Phi = \begin{bmatrix} d^{-1} & d^{-1} b \\ cd^{-1} & cd^{-1} b + a \end{bmatrix}. \]  \hfill (36)\]

Then,
\[ \begin{bmatrix} X \left( 2^k \right) \\ Y \left( 2^k \right) \end{bmatrix} = \Phi^x \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \]  \hfill (37)\]
and, using the doubling principle \([1]\) \(\Phi^{2^{k+1}} = \Phi^{2^k} \Phi^{2^k}\), we have
\[
\begin{bmatrix}
d^{-1}_{k+1} & d^{-1}_{k+1} b_{k+1} \\
d^{-1}_{k} & d^{-1}_{k} b_k \\
\end{bmatrix}
\begin{bmatrix}
c_{k+1} d^{-1}_{k+1} b_{k+1} + a_{k+1} \\
c_k d^{-1}_{k} b_k + a_k \\
\end{bmatrix}
= \begin{bmatrix}
d^{-1}_{k} & d^{-1}_{k} b_k \\
c_k d^{-1}_{k} b_k + a_k \\
\end{bmatrix}
\begin{bmatrix}
d^{-1}_{k+1} & d^{-1}_{k+1} b_{k+1} \\
c_{k+1} d^{-1}_{k+1} b_{k+1} + a_{k+1} \\
\end{bmatrix}.
\tag{38}
\]

Then we are able to derive, after some algebra, the following nonlinear iterative equations:
\[
\begin{align*}
a_{k+1} &= a_k \left(I - c_k \left[I + b_k c_k \right]^{-1} b_k \right) a_k, \\
b_{k+1} &= b_k + d_k \left[I + b_k c_k \right]^{-1} b_k a_k, \\
c_{k+1} &= c_k + a_k c_k \left[I + b_k c_k \right]^{-1} d_k, \\
d_{k+1} &= d_k \left[I + b_k c_k \right]^{-1} d_k,
\end{align*}
\tag{39}
\]

with initial conditions
\[
\begin{align*}
a_0 &= D - CA^{-1} B = F - Q \left[Q + S^{-1}\right]^{-1} F, \\
b_0 &= A^{-1} B = S^{-1} F^T \left[Q + S^{-1}\right]^{-1} F, \\
c_1 &= CA^{-1} = Q \left[Q + S^{-1}\right]^{-1}, \\
d_1 &= A^{-1} = S^{-1} F^T \left[Q + S^{-1}\right]^{-1}.
\end{align*}
\tag{40}
\]

Then, since
\[
\begin{bmatrix}
X \left(2^k\right) \\
Y \left(2^k\right)
\end{bmatrix}
= \begin{bmatrix} \Phi^{2^k} & 0 \\ 0 & \end{bmatrix}
\begin{bmatrix}
d^{-1}_{k} \\
c_k d^{-1}_{k}
\end{bmatrix},
\tag{41}
\]

it is clear that \(c_k = Y_{2k} X_{2k}^{-1} = G_{2k}\) tends to a steady state value \(\overline{G}\).

2.2.2. Algebraic Algorithm for \(\overline{G}\) Computation. In this section, we present an algebraic algorithm for \(\overline{G}\) computation. As in (29), setting
\[
G_k = K_k H = Y_k X_k^{-1}
\tag{42}
\]

and using the parameters \(A, B, C, D\) by (23), we derive
\[
\Phi = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} \left(Q + S^{-1}\right) F^{-T} S & F \\ Q F^{-T} S & F \end{bmatrix},
\tag{43}
\]

which is a matrix of dimension \(2n \times 2n\). Since
\[
\det \Phi = \det \left(A \det \left(D - CA^{-1} B\right)\right)
= \det \left(\left(Q + S^{-1}\right) F^{-T} S\right)
\times \det \left(F - QF^{-T} S \left[Q + S^{-1}\right]^{-1} F\right)
= \det \left(Q + S^{-1}\right) \det \left(F^{-1}\right) \det \left(S\right)
\times \det \left(F - QF^{-T} S \left[Q + S^{-1}\right]^{-1} F\right)
= \det \left(S\right) \det \left(F - Q \left[Q + S^{-1}\right]^{-1} F\right)
\times \det \left(F^{-1}\right) \det \left(Q + S^{-1}\right)
= \det \left(S\right) \det \left(I - Q \left[Q + S^{-1}\right]^{-1}\right)
\times \det \left(F\right) \det \left(F^{-1}\right) \det \left(Q + S^{-1}\right)
= \det \left(S\right) \det \left(Q + S^{-1} - Q\right)
= \det \left(S\right) \det \left(S^{-1}\right)
= 1,
\tag{44}
\]

it is evident that \(\Phi\) is a nonsingular matrix and its eigenvalues occur in reciprocal pairs.

Thus, (43) can be written
\[
\Phi = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} \left(Q + S^{-1}\right) F^{-T} S & F \\ Q F^{-T} S & F \end{bmatrix} = W L W^{-1},
\tag{45}
\]

where
\[
L = \begin{bmatrix} \Lambda & 0 \\ 0 & \Lambda^{-1} \end{bmatrix}
\tag{46}
\]

is a diagonal matrix containing the eigenvalues of \(\Phi\), with \(\Lambda\) diagonal matrix with all the eigenvalues of \(\Phi\) lying outside the unit circle, and
\[
W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}
\tag{47}
\]

is the matrix containing the corresponding eigenvectors of \(\Phi\), with
\[
V = W^{-1} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}.
\tag{48}
\]

We are able to use zero initial condition \(P_0 = 0\), so \(G_0 = K_0 H = Y_0 X_0^{-1} = 0\); that is,
\[
\begin{bmatrix} X_0 \\ Y_0 \end{bmatrix} = \begin{bmatrix} I \\ 0 \end{bmatrix},
\tag{49}
\]

and hence
\[
\begin{bmatrix} X_k \\ Y_k \end{bmatrix} = \Phi^k \begin{bmatrix} I \\ 0 \end{bmatrix}.
\tag{50}
\]
Then, from (50) and (45)–(48), we are able to write
\[
\begin{bmatrix} X_k \\ Y_k \end{bmatrix} = \Phi X_{k-1} + \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \Lambda^{-k} \begin{bmatrix} V_{11} \\ V_{12} \\ V_{21} \\ V_{22} \end{bmatrix} \begin{bmatrix} 0 \\ I \end{bmatrix},
\]
that is,
\[
\begin{bmatrix} X_k \\ Y_k \end{bmatrix} = \begin{bmatrix} W_{11} \Lambda^{-k} V_{11} + W_{12} \Lambda^{-k} V_{21} \\ W_{21} \Lambda^{-k} V_{11} + W_{22} \Lambda^{-k} V_{21} \end{bmatrix}.
\]

(51)

Substituting in (42) the matrices \(X_k, Y_k\) from (52), we have that
\[
G_k = Y_k X_k^{-1} = \begin{bmatrix} W_{21} \Lambda^{-k} V_{11} + W_{22} \Lambda^{-k} V_{21} \end{bmatrix} \begin{bmatrix} W_{11} \Lambda^{-k} V_{11} + W_{12} \Lambda^{-k} V_{21} \end{bmatrix}^{-1}.
\]

(53)

Furthermore, the diagonal matrix \(\Lambda^{-k}\) contains all the eigenvalues of \(\Phi\) lying inside the unit circle, which follows that \(\lim_{k \to \infty} \Lambda^{-k} = 0\). Then, \(G_k\) tends to a steady state value \(\bar{G}\) with \(\bar{G} = \lim_{k \to \infty} G_k\), and from (53) arises
\[
\bar{G} = W_{21} W_{11}^{-1}.
\]

(54)

2.2.3. Steady State Kalman Filter Gain Computation. All algorithms presented in Sections 2.2.1 and 2.2.2 compute the steady state value \(\bar{G}\). Taking into account the assumptions of Section 2.1, we are able to conclude that, under the condition \(\text{rank}(H) = n\), the steady state gain is
\[
\bar{K} = \bar{G} [H^T H]^{-1} H^T.
\]

(55)

2.3. Direct Steady State Kalman Filter Gain Computation. In this section, we present algorithms for the direct computation of the steady state Kalman filter \(\bar{K}\). The proposed algorithms compute directly the steady state Kalman filter gain, that is, without using \(\bar{G} = KH\). All these algorithms hold under the assumption that \(n = m\). Note that, since \(\text{rank}(H) = n\), \(H\) and \(S\) are nonsingular matrices.

2.3.1. Iterative Algorithms for \(\bar{K}\) Computation. In this section, we present two iterative per-step algorithms and an iterative doubling algorithm for \(\bar{K}\) computation.

Per-Step Iterative Algorithm 1. Using (11), (22), and (13), we are able to derive the following nonlinear iterative equation with respect to the Kalman filter gain \(K_k\):
\[
K_{k+1} = G_{k+1} H^{-1}
\]
\[
= (QF^{-T} S + FG_k) \left[ S^{-1} F^{-T} S + QF^{-T} S + FG_k \right]^{-1} H^{-1}
\]
\[
= (QF^{-T} H^T R^{-1} H + FG_k)
\]
\[
\times \left[ H^{-1} RH^{-T} F^{-T} H^T R^{-1} H + HQF^{-T} H^T R^{-1} H + HFK_k H \right]^{-1} H^{-1}
\]
\[
= (QF^{-T} H^T R^{-1} + FK_k)
\]
\[
\times \left[ RH^{-T} F^{-T} H^T R^{-1} H + HQF^{-T} H^T R^{-1} H + HFK_k H \right]^{-1}
\]
\[
= (QF^{-T} H^T R^{-1} + FK_k)
\]
\[
\times \left[ RH^{-T} F^{-T} H^T R^{-1} + HQF^{-T} H^T R^{-1} + HFK_k H \right]^{-1}
\]

(56)

The nonsingularity of \(S\) and (13) allow us to write the equality in (56) as
\[
K_{k+1} = (QF^{-T} H^T R^{-1} + FK_k)
\]
\[
\times \left[ H {QF^{-T} H^T R^{-1} + HS^{-1} F^{-T} H^T R^{-1} + HFK_k} \right]^{-1}
\]
\[
= (C + DK_k) [A + BK_k]^{-1},
\]

(57)

where
\[
A = HQF^{-T} H^T R^{-1} + HS^{-1} F^{-T} H^T R^{-1}
\]
\[
= H \left( Q + S^{-1} \right) F^{-T} H^T R^{-1},
\]
\[
B = HF,
\]
\[
C = QF^{-T} H^T R^{-1},
\]
\[
D = F.
\]

(58)
The initial condition is $K_0 = P_0 H_T [H P_0 H_T + R]^{-1}$. It is known [1] that the prediction error covariance tends to the steady state prediction error covariance and that the convergence is independent of the initial uncertainty, that is, independent of the value of the initial condition $P_0$. Thus, we are able to assume zero initial condition $P_0 = 0$ and so we are to use the initial condition $K_0 = 0$.

It is clear that $K_k$ tends to a steady state value $\bar{K}$ satisfying

$$\bar{K} = (C + D\bar{K}) [A + B\bar{K}]^{-1}. \quad (59)$$

**Per-Step Iterative Algorithm 2.** Using (57), we are able to derive the following nonlinear iterative equation with respect to the Kalman filter gain $K_k$:

$$K_{k+1} = (C + D K_k) [A + B K_k]^{-1}$$

$$= (C A^{-1} A + (D - CA^{-1} B - CA^{-1} B) K_k) [A + B K_k]^{-1}$$

$$= CA^{-1} + (D - CA^{-1} B) K_k [A + B K_k]^{-1}$$

$$= CA^{-1} + (D - CA^{-1} B) \left[AK_k^{-1} + B\right]^{-1}$$

$$= CA^{-1} + (D - CA^{-1} B) \left[K_k^{-1} + A^{-1} B\right]^{-1} A^{-1}$$

$$= c + a[K_k^{-1} + b]^{-1}d, \quad (60)$$

where $A, B, C, D$ are given by (58) and

$$a = D - CA^{-1} B,$$

$$b = A^{-1} B,$$

$$c = CA^{-1},$$

$$d = A^{-1}. \quad (61)$$

The algorithm uses the initial condition $K_0 = P_0 H_T [H P_0 H_T + R]^{-1}$. It is known [1] that the prediction error covariance tends to the steady state prediction error covariance and that the convergence is independent of the initial uncertainty, that is, independent of the value of the initial condition $P_0$. Thus, we are able to assume zero initial condition $P_0 = 0$. In this case, in order to avoid $K_0^{-1}$, we are to use the initial condition $K_1 = c$.

It is clear that $K_k$ tends to a steady state value $\bar{K}$ satisfying

$$\bar{K} = c + a[\bar{K}^{-1} + b]^{-1}d. \quad (62)$$

**Doubling Iterative Algorithm.** In (57), setting

$$K_k = Y_k X_k^{-1}, \quad (63)$$

we take

$$Y_{k+1} X_{k+1}^{-1} = (C + D Y_k X_k^{-1}) \left[A + B Y_k X_k^{-1}\right]^{-1}$$

$$= (C K + D Y_k) [A X_k + B Y_k]^{-1} \quad (64)$$

or

$$\begin{bmatrix} X_{k+1} \\ Y_{k+1} \end{bmatrix} = \Phi \begin{bmatrix} X_k \\ Y_k \end{bmatrix}, \quad (65)$$

where

$$\Phi = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} d^{-1} & d^{-1} b \\ cd^{-1} & cd^{-1} b + a \end{bmatrix} \quad (66)$$

is a matrix of dimension $2n \times 2n$ and $A, B, C, D$ as in (58).

Working as in the doubling iterative algorithm of Section 2.2.1 and using zero initial condition $P_0 = 0$, we are able to derive the following nonlinear iterative equations:

$$a_{k+1} = a_k \left(1 - c_k [I + b_k c_k]^{-1} b_k\right) a_k,$$

$$b_{k+1} = b_k + d_k [I + b_k c_k]^{-1} b_k a_k,$$

$$c_{k+1} = c_k + a_k c_k [I + b_k c_k]^{-1} d_k,$$

$$d_{k+1} = d_k [I + b_k c_k]^{-1} d_k, \quad (67)$$

with initial conditions

$$a_1 = a,$$

$$b_1 = b,$$

$$c_1 = c,$$

$$d_1 = d. \quad (68)$$

It is clear that $c_k = Y_{2k} X_{2k}^{-1} = K_{2k}$ tends to a steady state value $\bar{K}$.

### 2.3.2. Algebraic Algorithm for $\bar{K}$ Computation.

In this section, we present an algebraic algorithm for $\bar{K}$ computation. Working as in the algebraic algorithm of Section 2.2.2 and using the parameters $A, B, C, D$ by (58), we derive

$$\Phi = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \begin{bmatrix} \Lambda & 0 \\ 0 & \Lambda^{-1} \end{bmatrix} \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}^{-1}, \quad (69)$$

which is a matrix of dimension $2n \times 2n$.

Then, the steady state Kalman filter is

$$\bar{K} = W_{22} W_{11}^{-1}. \quad (70)$$

### 2.4. Advantages of the Proposed Algorithms.

All algorithms for the computation of the steady state Kalman filter gain $\bar{K}$, presented in Section 2, are summarized in Table 1. It is clear that the direct computation of the Kalman filter gain
is feasible only if the following restriction holds: \( n = m \). The advantage of the presented algorithms is the autonomous computation of the steady state Kalman filter gain. Especially, the steady state Kalman filter gain is important, when we want to compute the parameters of the steady state Kalman filter. 

\[
x_{k+1|k+1} = (I - \tilde{K}H)Fx_{k|k} + \tilde{K}z_{k+1}
\]

(71)

It is obvious from (71) that the parameters of the steady state Kalman filter are related to the steady state Kalman filter gain. In particular, the steady state prediction error covariance can be computed via the steady state gain and is given by

\[
\hat{P}_p = (I - \tilde{K}H)^{-1}\tilde{K}RH\left[H^TH\right]^{-1}.
\]

(72)

Indeed, from (2), arises \( \tilde{K} = \tilde{P}_pH^T[H\tilde{P}_pH^T + R]^{-1} \), which leads to

\[
\tilde{K} \left( H\tilde{P}_pH^T + R \right) = \tilde{P}_pH^T \implies \tilde{K}\tilde{P}_pH^T + \tilde{K}R = \tilde{P}_pH^T
\]

\[
\implies \tilde{P}_pH^T - \tilde{K}\tilde{P}_pH^T = \tilde{K}R
\]
\[
(1 - \bar{K}H)\bar{P}_pH^T = \bar{K}R
gives \quad \bar{P}_pH^T = [I - \bar{K}H]^{-1}\bar{K}R
\Rightarrow \quad \bar{P}_pH^TH = [I - \bar{K}H]^{-1}\bar{K}RH.
\]

(73)

Since \(\text{rank}(H) = n\), the matrix \(H^TH\) is nonsingular \([21]\); thus from the last equation arises immediately the formula of the steady state prediction error covariance in (72).

Also, by (3), the steady state estimation error covariance can be computed via the steady state prediction error covariance

\[
\bar{P}_e = [I - \bar{K}H] \bar{P}_p. 
\]

(74)

3. Examples

In this section, two examples verify the results of Section 2.

Example 1. A model of dimensions \(n = 1\) and \(m = 2\) is assumed with parameters:

\[
F = 0.8,
\]

\[
H = \begin{bmatrix} 1 \\ 2 \end{bmatrix},
\]

\[
Q = 5,
\]

\[
R = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.4 \end{bmatrix}.
\]

(75)

In this example, we have \(\text{rank}(H) = n = 1\) with \(m > n\).

Using all algorithms presented in Section 2.2, we computed

\[
\bar{G} = 0.9902.
\]

(76)

Then, using (55), we computed the steady state gain

\[
\bar{K} = \begin{bmatrix} -0.4033 & 0.6207 \\ 0.4597 & -0.1975 \end{bmatrix}.
\]

(80)

We also computed the same steady state gain, using all algorithms presented in Section 2.3, since \(m = n = 2\).

4. Conclusions

The Kalman filter gain arises in Kalman filter equations in linear estimation and is associated with linear systems. The gain is a matrix through which the estimation and the prediction of the state as well as the corresponding estimation and prediction error covariance matrices are computed. For time invariant and asymptotically stable systems, there exist steady state values of the estimation and prediction error covariance matrices. There exists also a steady state value of the Kalman filter gain.

The steady state Kalman filter gain is usually derived via the steady state prediction error covariance by first solving the corresponding Riccati equation. In view of the importance of the Riccati equation, there exists considerable literature on its algebraic or iterative solutions, including the Chandrasekhar algorithms, which are the only iterative algorithms that provide the steady state Kalman filter gain together with the prediction error covariance.

Iterative per-step and doubling algorithms as well as an algebraic algorithm for the steady state Kalman filter computation were presented. These algorithms hold under conditions concerning the system parameters. The advantage of these algorithms is the autonomous computation of the steady state Kalman filter gain. This is important if we want to compute only the steady state Kalman filter gain or to compute the parameters of the steady state Kalman filter, which are related to the steady state Kalman filter gain.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References


Submit your manuscripts at http://www.hindawi.com