Structured Covariance Estimation for State Prediction

Weichang Li and Thomas A. Badgwell

Abstract—In this paper we approach model covariance estimation as the generalization of the classical problem of covariance estimation/selection from a sequence of i.i.d samples [12], [13] to the dynamic setting. Given samples drawn from a multivariate Gaussian distribution, a number of algorithms have been recently proposed to estimate the covariance [14], [15] or its inverse, the precision matrix [16], [17], [18], [19], [20], [21], [22], with desired structure properties such as low-cardinality and/or low-rank. One general approach is to replace the rank or cardinality constraints by their convex surrogates, e.g. the trace norm and matrix $l_1$ norm respectively, and then solve via efficient optimization algorithms such as proximal gradient, coordinate descent and smoothing algorithms. A review of these methods can be found, for instance, in [23]. The rank of the covariances (or their inverses) quantifies the number of marginally (conditionally) independent components. The cardinality characterizes the corresponding pairwise dependence pattern that dominates the innovation sequence with its sample statistics. The sample statistics are typically computed from Kalman filter with a stable gain and the steady state innovation statistics are parametrized by the noise covariances. These methods are fairly robust to initial guesses of the covariances. However, a major disadvantage of these methods is the large estimation variance, mainly due to the difficulty in obtaining an accurate estimation of the sample statistics because the innovation sequences will be strongly correlated when the Kalman gain is not necessarily optimal. Another limitation associated the ALS algorithm is its high memory requirement which grows quadratically with problem size.

Except for trivial cases, ML parameter estimation in dynamic models is generally intractable. The EM algorithm is an iterative technique for ML parameter estimation [11]. Its application for parameter estimation in linear state-space or ARMA models consists of an expectation (E) step involving Kalman state smoothing and a maximization (M) step where the parameters are estimated from a set of sufficient statistics in terms of the smoothed state estimates, the residuals and the associated error covariances [4], [5], [6], [7]. Iteratively, the EM algorithm generates a sequence of parameter estimates with nondecreasing likelihood values. While the basic concept is well understood, applications of the EM algorithm often face nontrivial challenges including, for instance, ensuring parameter convergence and remaining robust to the choice of initial guesses.

In this paper we approach model covariance estimation as the generalization of the classical problem of covariance estimation/selection from a sequence of i.i.d samples [12], [13] to the dynamic setting. Given samples drawn from a multivariate Gaussian distribution, a number of algorithms have been recently proposed to estimate the covariance [14], [15] or its inverse, the precision matrix [16], [17], [18], [19], [20], [21], [22], with desired structure properties such as low-cardinality and/or low-rank. One general approach is to replace the rank or cardinality constraints by their convex surrogates, e.g. the trace norm and matrix $l_1$ norm respectively, and then solve via efficient optimization algorithms such as proximal gradient, coordinate descent and smoothing algorithms. A review of these methods can be found, for instance, in [23]. The rank of the covariances (or their inverses) quantifies the number of marginally (conditionally) independent components. The cardinality characterizes the corresponding pairwise dependence pattern that dominates the innovation sequence with its sample statistics. The sample statistics are typically computed from Kalman filter with a stable gain and the steady state innovation statistics are parametrized by the noise covariances. These methods are fairly robust to initial guesses of the covariances. However, a major disadvantage of these methods is the large estimation variance, mainly due to the difficulty in obtaining an accurate estimation of the sample statistics because the innovation sequences will be strongly correlated when the Kalman gain is not necessarily optimal. Another limitation associated the ALS algorithm is its high memory requirement which grows quadratically with problem size.

Except for trivial cases, ML parameter estimation in dynamic models is generally intractable. The EM algorithm is an iterative technique for ML parameter estimation [11]. Its application for parameter estimation in linear state-space or ARMA models consists of an expectation (E) step involving Kalman state smoothing and a maximization (M) step where the parameters are estimated from a set of sufficient statistics in terms of the smoothed state estimates, the residuals and the associated error covariances [4], [5], [6], [7]. Iteratively, the EM algorithm generates a sequence of parameter estimates with nondecreasing likelihood values. While the basic concept is well understood, applications of the EM algorithm often face nontrivial challenges including, for instance, ensuring parameter convergence and remaining robust to the choice of initial guesses.

In this paper we approach model covariance estimation as the generalization of the classical problem of covariance estimation/selection from a sequence of i.i.d samples [12], [13] to the dynamic setting. Given samples drawn from a multivariate Gaussian distribution, a number of algorithms have been recently proposed to estimate the covariance [14], [15] or its inverse, the precision matrix [16], [17], [18], [19], [20], [21], [22], with desired structure properties such as low-cardinality and/or low-rank. One general approach is to replace the rank or cardinality constraints by their convex surrogates, e.g. the trace norm and matrix $l_1$ norm respectively, and then solve via efficient optimization algorithms such as proximal gradient, coordinate descent and smoothing algorithms. A review of these methods can be found, for instance, in [23]. The rank of the covariances (or their inverses) quantifies the number of marginally (conditionally) independent components. The cardinality characterizes the corresponding pairwise dependence pattern that dominates the innovation sequence with its sample statistics. The sample statistics are typically computed from Kalman filter with a stable gain and the steady state innovation statistics are parametrized by the noise covariances. These methods are fairly robust to initial guesses of the covariances. However, a major disadvantage of these methods is the large estimation variance, mainly due to the difficulty in obtaining an accurate estimation of the sample statistics because the innovation sequences will be strongly correlated when the Kalman gain is not necessarily optimal. Another limitation associated the ALS algorithm is its high memory requirement which grows quadratically with problem size.

Except for trivial cases, ML parameter estimation in dynamic models is generally intractable. The EM algorithm is an iterative technique for ML parameter estimation [11]. Its application for parameter estimation in linear state-space or ARMA models consists of an expectation (E) step involving Kalman state smoothing and a maximization (M) step where the parameters are estimated from a set of sufficient statistics in terms of the smoothed state estimates, the residuals and the associated error covariances [4], [5], [6], [7]. Iteratively, the EM algorithm generates a sequence of parameter estimates with nondecreasing likelihood values. While the basic concept is well understood, applications of the EM algorithm often face nontrivial challenges including, for instance, ensuring parameter convergence and remaining robust to the choice of initial guesses.

In this paper we approach model covariance estimation as the generalization of the classical problem of covariance estimation/selection from a sequence of i.i.d samples [12], [13] to the dynamic setting. Given samples drawn from a multivariate Gaussian distribution, a number of algorithms have been recently proposed to estimate the covariance [14], [15] or its inverse, the precision matrix [16], [17], [18], [19], [20], [21], [22], with desired structure properties such as low-cardinality and/or low-rank. One general approach is to replace the rank or cardinality constraints by their convex surrogates, e.g. the trace norm and matrix $l_1$ norm respectively, and then solve via efficient optimization algorithms such as proximal gradient, coordinate descent and smoothing algorithms. A review of these methods can be found, for instance, in [23]. The rank of the covariances (or their inverses) quantifies the number of marginally (conditionally) independent components. The cardinality characterizes the corresponding pairwise dependence pattern that dominates the innovation sequence with its sample statistics. The sample statistics are typically computed from Kalman filter with a stable gain and the steady state innovation statistics are parametrized by the noise covariances. These methods are fairly robust to initial guesses of the covariances. However, a major disadvantage of these methods is the large estimation variance, mainly due to the difficulty in obtaining an accurate estimation of the sample statistics because the innovation sequences will be strongly correlated when the Kalman gain is not necessarily optimal. Another limitation associated the ALS algorithm is its high memory requirement which grows quadratically with problem size.

Except for trivial cases, ML parameter estimation in dynamic models is generally intractable. The EM algorithm is an iterative technique for ML parameter estimation [11]. Its application for parameter estimation in linear state-space or ARMA models consists of an expectation (E) step involving Kalman state smoothing and a maximization (M) step where the parameters are estimated from a set of sufficient statistics in terms of the smoothed state estimates, the residuals and the associated error covariances [4], [5], [6], [7]. Iteratively, the EM algorithm generates a sequence of parameter estimates with nondecreasing likelihood values. While the basic concept is well understood, applications of the EM algorithm often face nontrivial challenges including, for instance, ensuring parameter convergence and remaining robust to the choice of initial guesses.

In this paper we approach model covariance estimation as the generalization of the classical problem of covariance estimation/selection from a sequence of i.i.d samples [12], [13] to the dynamic setting. Given samples drawn from a multivariate Gaussian distribution, a number of algorithms have been recently proposed to estimate the covariance [14], [15] or its inverse, the precision matrix [16], [17], [18], [19], [20], [21], [22], with desired structure properties such as low-cardinality and/or low-rank. One general approach is to replace the rank or cardinality constraints by their convex surrogates, e.g. the trace norm and matrix $l_1$ norm respectively, and then solve via efficient optimization algorithms such as proximal gradient, coordinate descent and smoothing algorithms. A review of these methods can be found, for instance, in [23]. The rank of the covariances (or their inverses) quantifies the number of marginally (conditionally) independent components. The cardinality characterizes the corresponding pairwise dependence pattern that dominates the innovation sequence with its sample statistics. The sample statistics are typically computed from Kalman filter with a stable gain and the steady state innovation statistics are parametrized by the noise covariances. These methods are fairly robust to initial guesses of the covariances. However, a major disadvantage of these methods is the large estimation variance, mainly due to the difficulty in obtaining an accurate estimation of the sample statistics because the innovation sequences will be strongly correlated when the Kalman gain is not necessarily optimal. Another limitation associated the ALS algorithm is its high memory requirement which grows quadratically with problem size.

Except for trivial cases, ML parameter estimation in dynamic models is generally intractable. The EM algorithm is an iterative technique for ML parameter estimation [11]. Its application for parameter estimation in linear state-space or ARMA models consists of an expectation (E) step involving Kalman state smoothing and a maximization (M) step where the parameters are estimated from a set of sufficient statistics in terms of the smoothed state estimates, the residuals and the associated error covariances [4], [5], [6], [7]. Iteratively, the EM algorithm generates a sequence of parameter estimates with nondecreasing likelihood values. While the basic concept is well understood, applications of the EM algorithm often face nontrivial challenges including, for instance, ensuring parameter convergence and remaining robust to the choice of initial guesses.
the structure.

In the dynamic model setting considered here, the samples (i.e. state and observation noises) are not directly available, instead observed only through the measurement data, or equivalently, drawn implicitly, from the conditional distribution governed by the dynamic model. By applying the expectation maximization (EM) algorithm to the innovation model representation, we show that the resulting ML covariance estimates are essentially the conditional sample covariances. Subsequently we augment the negative log-likelihood function with matrix norm penalty terms that enforce low-rank and low cardinality structure in the resulting covariance estimates or their inverses. These constraints serve to reflect realistic problem structure expected from model knowledge yet are still general and flexible enough to be broadly applicable. For instance, in industrial process control applications, the process disturbances may only have a few independent components and the impact of each component on the states might not be fully correlated across different state elements. In addition, we provide a new derivation of the EM algorithm based on the innovation model which represents the estimates of both the process and observation noise covariances in terms of a common sufficient statistic. This illustrates the coupling between the two covariance estimates, and in simulated cases, enables the calculation of an upper performance bound against which the EM estimates can be compared. The use of the innovation representation also provides a tractable connection to existing techniques such as the Autocovariance Least Squares (ALS) algorithm. Numerical results comparing the constrained EM and the ALS algorithms are also provided, showing favourable performance for the EM covariance estimates.

We also would like to remark that given the connection between Gaussian graphical models and inverse covariance matrices, the extension of that connection to the dynamic system setting provides a potentially new approach to address graph based inference in dynamic networks as discussed, for instance, in [24] and [25].

The remainder of this paper is organized as follows: we formulate the problem in section II and then derive the ML estimation based on the innovation model representation in section III. The EM covariance estimation algorithm is derived in section IV. The structurally constrained EM algorithms are presented in section V. We present three numerical examples in section VI where we compare the performance of the EM and ALS covariance estimates. Finally we conclude in section VII.

As for notation, vec(·) represents the vec operation that stacks the columns of a matrix, \( \text{tr}(·) \) denotes the matrix trace, \( \text{Diag}(A_1, A_2) \) denotes a block diagonal matrix with diagonal blocks given by \( A_1 \) and \( A_2 \), and \( \mathcal{N}(\mathbf{m}, \mathbf{Q}) \) denotes Gaussian distribution with mean \( \mathbf{m} \) and covariance \( \mathbf{Q} \). The notations \( ||·||_F \), \( ||·||_p \), and \( ||·||_a \) stand, respectively, for the Frobenius norm, entry-wise \( \ell_p \) norm, and trace-norm (or nuclear norm, given by the sum of the singular values). The product \( \mathbf{A} \odot \mathbf{B} \) between two matrices with matching dimensions stands for the Hadamard or entry-wise product between \( \mathbf{A} \) and \( \mathbf{B} \). The product \( \mathbf{A} \otimes \mathbf{B} \) stands for the kronecker product. The matrix \( |\mathbf{A}| \) contains the absolute values of entries of \( \mathbf{A} \). The matrix \( (\mathbf{M})_+ \) is the componentwise positive part of the matrix \( \mathbf{M} \), and \( \text{sign}(\mathbf{M}) \) is the sign matrix associated to \( \mathbf{M} \) with the convention \( \text{sign}(0) = 0 \).

II. THE PROBLEM FORMULATION

Consider the following discrete-time linear state-space model:

\[
\begin{align*}
\mathbf{x}_{k+1} &= \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k + \mathbf{G}_k \mathbf{w}_k \\
\mathbf{y}_k &= \mathbf{C}_k \mathbf{x}_k + \mathbf{v}_k
\end{align*}
\]

where \( \mathbf{x}_k \in \mathbb{R}^N \) and \( \mathbf{y}_k \in \mathbb{R}^M \) are the state and the observed output at time \( k \), respectively. \( \mathbf{u}_k \in \mathbb{R}^L \) denotes the system input. Both the process and measurement noises \( \mathbf{w}_k \in \mathbb{R}^P \) and \( \mathbf{v}_k \in \mathbb{R}^M \) are zero-mean Gaussian processes, with covariance matrices \( \mathbf{Q}_w \) and \( \mathbf{R}_v \), respectively, and are mutually and temporally independent, and independent from the initial state \( \mathbf{x}_0 \). The initial state is assumed Gaussian, i.e. \( \mathbf{x}_0 \sim \mathcal{N}(\mu_0, \mathbf{Q}_0) \). The model is thus completely specified by the parameter set \( \mathbf{A}_k, \mathbf{B}_k, \mathbf{C}_k, \mathbf{G}_k, \mathbf{Q}_w, \mathbf{R}_v, \mu_0 \) and \( \mathbf{Q}_0 \), which we denote as

\[
\Theta_g \triangleq \{ \mathbf{a}_k; \mathbf{b}_k; \mathbf{c}_k; \mathbf{g}_k; \mathbf{q}_w; \mathbf{r}_v; \mu_0; \mathbf{q}_0 \},
\]

for \( k = 0, \cdots, K \), where \( \mathbf{a}_k = \text{vec}(\mathbf{A}_k) \), similarly for \( \mathbf{b}_k, \mathbf{c}_k, \mathbf{g}_k, \mathbf{q}_w, \mathbf{r}_v \) and \( \mathbf{q}_0 \). The linear state-space model (1) may be obtained by linearizing a more general nonlinear model. Thus the algorithms developed in this paper can be applied to nonlinear models as well, with proper linearization.

Given the set of observed input \( \mathcal{U}_K \triangleq \{ \mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_K \} \) and output \( \mathcal{Y}_K \triangleq \{ \mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_K \} \), the general problem of system identification is to estimate the full or subset of the above system parameters, for the purpose of state filtering, prediction, or control. In this paper our focus is on identifying the noise covariances \( \mathbf{Q}_w, \mathbf{R}_v \), assuming the other parameters are either given or known. The case where \( \mathbf{G}_k \) is also unknown but time-invariant may be included by absorbing \( \mathbf{G}_k = \mathbf{G} \) into \( \mathbf{Q}_w \). Therefore we rewrite the parameter set as

\[
\Theta \triangleq \{ \mathbf{q}_w; \mathbf{r}_v \}
\]

where \( \mathbf{q}_w = \text{vec}(\mathbf{Q}_w) \) and \( \mathbf{r}_v = \text{vec}(\mathbf{R}_v) \).

Several different criteria may be used in identifying the parameters above, such as minimizing the residual prediction error (RPE), maximizing the likelihood (ML), or maximizing the \textit{a posteriori} (MAP) density, leading to various different forms of parameter estimates that are interconnected with each other [1]. We focus on ML estimation in this paper.

III. ML ESTIMATION BASED ON THE L-INNOVATION REPRESENTATION

Assume that we obtain the state estimates via a sequence of time-varying stable filter gain \( \mathbf{L}_k \), according to the following update equations:

\[
\begin{align*}
\hat{\mathbf{x}}_{k+1|k} &= \mathbf{A}_k \hat{\mathbf{x}}_{k|k} + \mathbf{B}_k \mathbf{u}_k \\
\hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{L}_k \mathbf{e}_k \\
\hat{\mathbf{y}}_{k|k-1} &= \mathbf{C}_k \hat{\mathbf{x}}_{k|k-1},
\end{align*}
\]
where \( \mathbf{w}_k = \mathbf{w}_{k|k-1} \) denotes the one-step prediction residual error, or the L-innovation to differentiate from the true innovation since \( \mathbf{L}_k \) is not necessarily optimal. Defining the one-step state prediction error as

\[
\mathbf{e}_k = \mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1},
\]

then it can be shown that

\[
\begin{align*}
\mathbf{e}_{k+1} &= \mathbf{A}_k \mathbf{e}_k + \mathbf{G}_k \mathbf{w}_k \\
\mathbf{e}_k &= \mathbf{C}_k \mathbf{e}_k + \mathbf{v}_k
\end{align*}
\]

(8a), (8b)

where

\[
\begin{align*}
\mathbf{A}_k &= \mathbf{A}_k - \mathbf{A}_k \mathbf{L}_k \mathbf{C}_k \\
\mathbf{G}_k &= [\mathbf{G}_k - \mathbf{A}_k \mathbf{L}_k],
\end{align*}
\]

(9), (10)

and \( \mathbf{w}_k = [\mathbf{w}_k; \mathbf{v}_k] \sim \mathcal{N}(0, \mathbf{Q}_w) \) with \( \mathbf{Q}_w = \text{diag}(\mathbf{Q}_w, \mathbf{R}_v) \). Although the noises \( \mathbf{w}_k \) and \( \mathbf{v}_k \) are now cross-correlated, they are not correlated sequentially over time. Denoting

\[
\mathbf{z}_k = \left[ \mathbf{e}_{k+1} - \hat{\mathbf{A}}_k \mathbf{e}_k; \mathbf{e}_k - \mathbf{C}_k \mathbf{e}_k \right],
\]

(11)

it follows from (8) that

\[
\mathbf{z}_k = \begin{bmatrix} \mathbf{G}_k & -\mathbf{A}_k \mathbf{L}_k \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{w}_k \\ \mathbf{v}_k \end{bmatrix} \overset{\Delta}{=} \mathbf{H}_k \bar{\mathbf{w}}_k.
\]

(12)

As a result, the likelihood function of observing both \( \mathbf{e}_k \) and \( \mathbf{e}_k \), for \( k = 0, \cdots, K \), can be written as

\[
f(\{\mathbf{e}_k, \mathbf{e}_k\}_{k=0,\cdots,K}; \mathbf{Q}_w, \mathbf{R}_v) = p(\mathbf{e}_0) \prod_{k=0}^{K} p(\mathbf{z}_k),
\]

where \( p(\mathbf{e}_0) \) is the probability distribution of the initial state estimation error and

\[
p(\mathbf{z}_k) = \mathcal{N}(0, \mathbf{H}_k \hat{\mathbf{Q}}_w \mathbf{H}_k^\top).
\]

(14)

Taking the log of both sides of (13) gives the following log-likelihood function:

\[
LL(\{\mathbf{e}_k, \mathbf{e}_k\}_{k=0,\cdots,K}; \mathbf{Q}_w, \mathbf{R}_v) = \log p(\mathbf{e}_0) + \sum_{k=0}^{K} \log p(\mathbf{z}_k).
\]

(15)

If both \( \mathbf{e}_k \) and \( \mathbf{e}_k \) are observed, then the ML parameter estimate can be solved by maximizing either the likelihood function \( f(\{\mathbf{e}_k, \mathbf{e}_k\}_{k=0,\cdots,K}; \mathbf{Q}_w, \mathbf{R}_v) \), or equivalently, the log-likelihood function:

\[
\max_{\mathbf{Q}_w, \mathbf{R}_v} LL(\{\mathbf{e}_k, \mathbf{e}_k\}_{k=0,\cdots,K}; \mathbf{Q}_w, \mathbf{R}_v).
\]

(16)

However, we only observe \( \mathbf{y}_k \) in (1) or \( \mathbf{e}_k \) in (8) after applying the filter. The state prediction error \( \mathbf{e}_k \) is hidden. A popular approach to this type of problem is the EM algorithm which we apply to model (8) in the next section.

**IV. The EM Algorithm**

**A. The expected log-likelihood function**

The EM algorithm essentially maximizes the expected log-likelihood function, that is

\[
(\hat{\mathbf{Q}}_w, \hat{\mathbf{R}}_v) = \arg \max_{\mathbf{Q}_w, \mathbf{R}_v} J(\mathbf{Q}_w, \mathbf{R}_v).
\]

(17)

where

\[
J(\mathbf{Q}_w, \mathbf{R}_v) = \mathbb{E}\left\{ LL(\{\mathbf{e}_k, \mathbf{e}_k\}_{k=0,\cdots,K}; \mathbf{Q}_w, \mathbf{R}_v)|\mathbf{Y}_K \right\},
\]

(18)

where the expectation is taken with respect to the set of observation sequence \( \mathbf{Y}_K \). The maximization steps. Specifically, the expectation step involves computing \( E\{\mathbf{z}_k \mathbf{z}_k^\top|\mathbf{Y}_K\} \). The first term \( E\{\mathbf{e}_0 \mathbf{e}_0^\top|\mathbf{Y}_K\} \) may be dropped as it is not a function of the unknown parameters. This is one simplification of the EM algorithms in exchange for iterations. In Sections IV-B and IV-C we derive the maximization and the expectation steps respectively.

**B. The maximization step**

We first maximize \( J(\mathbf{Q}_w, \mathbf{R}_v) \) over \( \mathbf{Q}_w, \mathbf{R}_v \) by setting the corresponding partial derivatives to zero, which produces estimates of \( \mathbf{Q}_w, \mathbf{R}_v \) as functions of \( E\{\mathbf{z}_k \mathbf{z}_k^\top|\mathbf{Y}_K\} \).

Recall \( \mathbf{Q}_{g_k} \overset{\Delta}{=} \mathbf{G}_k \mathbf{Q}_w \mathbf{G}_k^\top \) taking the partial derivative of \( J(\mathbf{Q}_w, \mathbf{R}_v) \) with respect to \( \mathbf{Q}_{g_k} \) gives

\[
\frac{\partial J(\mathbf{Q}_w, \mathbf{R}_v)}{\partial \mathbf{Q}_{g_k}} = \mathbf{Q}_{g_k} - \mathbf{L}_{A_k} E\{\mathbf{z}_k \mathbf{z}_k^\top|\mathbf{Y}_K\} \mathbf{L}_{A_k}^\top
\]

\[
= \mathbf{G}_k \mathbf{Q}_w \mathbf{G}_k^\top - \mathbf{L}_{A_k} E\{\mathbf{z}_k \mathbf{z}_k^\top|\mathbf{Y}_K\} \mathbf{L}_{A_k}^\top
\]

(21)
Setting (21) to zero yields, for $k = 0, \cdots, K$,

$$(G_k \otimes G_k)q_w = vec\left(L_{A_k} E\{z_k z_k'|Y_K\} L_{A_k}'\right)$$

equals

$$vec\left(E\{z_k z_k'|Y_K\} L_{A_k}'\right).$$

(22)

In the case where $G_k = G$ is time invariant, it follows

$$Q_g \triangleq GQ_w G^t = \frac{1}{K+1} \sum_{k=0}^{K} L_{A_k} E\{z_k z_k'|Y_K\} L_{A_k}'$$

Similarly setting the partial derivative with respect to $R_v^{-1}$ to zero yields

$$\frac{\partial J}{\partial R_v^{-1}} = \sum_{k=0}^{K} \left[0 \ I\right] E\{z_k z_k'|Y_K\} \left[0 \ I\right]'$$

(24)

to zero yields

$$R_v = \frac{1}{K+1} \sum_{k=0}^{K} \left[0 \ I\right] E\{z_k z_k'|Y_K\} \left[0 \ I\right]'$$

(25)

Or in vector form

$$r_v = \left[0 \ I\right] \otimes \left[0 \ I\right] \frac{1}{K+1} \sum_{k=0}^{K} vec\left(E\{z_k z_k'|Y_K\}\right)$$

(26)

Throughout (24)-(26), we used short-hand notation $0 = 0_{M \times N}$ and $I = I_{M \times M}$. Combining (22) and (26) gives

$$\begin{bmatrix} G_K & I_{M^2} \\ I_{M^2} & r_v \end{bmatrix} q_w = \begin{bmatrix} L_K & T \\ T & vec\left(E\{z_k z_k'|Y_K\}\right) \end{bmatrix}$$

(27)

where

$$G_K \triangleq \begin{bmatrix} G_0 \otimes G_0; & G_1 \otimes G_1; & \cdots; & G_K \otimes G_K \end{bmatrix}$$

$$L_k \triangleq Diag\{L_{A_0} \otimes L_{A_0}; L_{A_1} \otimes L_{A_1}; \cdots; L_{A_K} \otimes L_{A_K}\}$$

$$T \triangleq \frac{1}{(K+1)} \begin{bmatrix} 0 \ I \end{bmatrix} \otimes \begin{bmatrix} 0 \ I \end{bmatrix}$$

$$\begin{bmatrix} I_{(N+M)^2 \times (N+M)^2}; & \cdots; & I_{(N+M)^2 \times (N+M)^2} \end{bmatrix}.$$ (28)

$T$ is of size $M^2 \times (N + M)^2$. Note that (27) only serves to provide a unified expression, the estimates of $q_w$ and $r_v$ are computed in a more efficient form instead of directly solving (27), as shown in section IV-C.

To summarize, the maximization step solve for $Q_w$ and $R_v$ in terms of $E\{z_k z_k'|Y_K\}$, as in (22) and (25) respectively, or in a unified form in (27), all arising from maximizing the log-likelihood function without any additional constraint. In section V, we will impose structural constraints associated with the desired rank and cardinality of $Q_w$ and $R_v$, or their inverses.

C. The expectation step

The expectation step essentially evaluates $E\{z_k z_k'|Y_K\}$ which is needed in (27) for the maximization step. Recall

$$z_k = [\varepsilon_{k+1} - \hat{A}_k \varepsilon_k; \varepsilon_k - C_k \varepsilon_k]$$

therefore,

$$E\{z_k z_k'|Y_K\} = \begin{bmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{12}' & \Phi_{22} \end{bmatrix}$$

(30)

with

$$\Phi_{11} = E\{(\varepsilon_{k+1} - \hat{A}_k \varepsilon_k)(\varepsilon_{k+1} - \hat{A}_k \varepsilon_k)'|Y_K\}$$

$$\Phi_{12} = E\{(\varepsilon_{k+1} - \hat{A}_k \varepsilon_k)(\varepsilon_k - C_k \varepsilon_k)'|Y_K\}$$

$$\Phi_{22} = E\{(\varepsilon_k - C_k \varepsilon_k)(\varepsilon_k - C_k \varepsilon_k)'|Y_K\},$$

which are derived and given by (55)-(58) in Appendix VII-B.

Substituting (55)-(58) into (30), then left and right multiplying (30) by $L_{A_k}$ and its transpose gives, for $k = 0, \cdots, K$,

$$\tilde{Q}_{g_k} = L_{A_k} E\{z_k z_k'|Y_K\} L_{A_k}'$$

$$= P_{k+1|K} + A_k P_{k|K} A_k - A_k P_{k,k+1|K}$$

$$- P_{k,k+1|K} A_k' + \delta x_k, k+1 | K \delta x_k, k+1 | K';$$

(31)

where $\delta x_k, k+1 | K$ is given in (56). If $G_k = G$ is time invariant, then

$$\tilde{Q}_{g_k} = L_{A_k} E\{z_k z_k'|Y_K\} L_{A_k}'$$

$$\begin{bmatrix} P_{k+1|K} + A_k P_{k|K} A_k - A_k P_{k,k+1|K} \\ - P_{k,k+1|K} A_k' + \delta x_k, k+1 | K \delta x_k, k+1 | K' \end{bmatrix}.$$ (32)

Substituting (58) into (25) gives

$$\tilde{R}_v = \frac{1}{K+1} \sum_{k=0}^{K} \left\{ C_k P_{k|K} C_k' + e_k | e_k | K \right\}$$

(33)

with $e_k | K \triangleq y_k - C_k \hat{x}_k | K$ is the smoothed residual error.

Remarks:

1. The innovation model based EM algorithm provides a common sufficient statistic, namely, $E\{z_k z_k'|Y_K\}$ for both $Q_w$ and $R_v$. This explains the coupling between the two covariance estimates, as well as providing a way to calculate an upper performance bound for EM estimates in the simulated cases.

2. Another simplification due to the use of the innovation representation is the absence of the control input term in the resulting estimates.

V. STRUCTURE CONSTRAINED EM ALGORITHM

A. Structured covariance estimation from sample sequences

Given a sequence of i.i.d samples $x_k$, for $k = 0, \cdots, K$, drawn from a multivariate Gaussian distribution $\mathcal{N}(\mathbf{0}, \Sigma)$, seeking an estimate of either the underlying covariance $\Sigma$ or its inverse, the precision matrix $P = \Sigma^{-1}$, with desired structural properties such as low-rank and/or sparsity patterns has recently been an active topic in statistics and machine learning [14], [15], [16], [17], [18], [19], [20], [21], [22]. One popular formulation is to replace the rank and the cardinality constraints by their convex surrogates, the nuclear norm and $l_1$ norm, respectively. For instance, in [15] the following problem was solved using the incremental proximal descent algorithm to estimate a simultaneously low-rank and sparse covariance matrix:

$$\min_{\Sigma} L(\mathbf{S}, \Sigma) + \alpha \|\Sigma\|_+ + \gamma \|\Sigma\|_1$$

(34)

where $\mathbf{S} \triangleq \frac{1}{K+1} x_k x_k'$ is the sample covariance, $\alpha$ and $\gamma$ are nonnegative regularization coefficients, and the loss function
\(L(S, \Sigma) = \|S - \Sigma\|_2^2\). In [14], the authors used the following negative log-likelihood loss function
\[
L(S, \Sigma) = \log \det \Sigma + tr(\Sigma^{-1}S).
\] (38)

The resulting optimization problem becomes nonconvex in \(\Sigma\) and was solved via a majorization-minimization approach. Similar formulations have been developed for estimating \(P = \Sigma^{-1}\), for instance, in [16], [18], [19], [21].

### B. Structured covariance estimation in dynamic models

To simplify the remaining discussion in this section, we focus on the case with \(G_k = G\). In addition, we consider estimating \(Q_g\) and \(R_v\) instead, noting that the Kalman filter depends directly on \(Q_g\), not \(Q_{ew}\): also this is more general without assuming \(G\) is known. However, the constrained approach developed here can be easily extended to the general case. As a result, we rewrite \(J(Q_{ew}, R_v)\) in (20) by \(J(Q_g, R_v)\).

From (35) and (36) we observe that \(\tilde{Q}_g\) and \(\tilde{R}_v\) take the form of sample covariances for the sequences \(\delta x_{k,k+1|K}\) and \(\delta e_{k|K}\) respectively, meantime accounting for the corresponding uncertainty in these sequences via the error covariance terms. In Appendix VII-C it is shown that \(\tilde{Q}_g\) and \(\tilde{R}_v\) are indeed the conditioned sample covariances of \(w_{g,k} \triangleq Gw_k\) and \(v_k\) respectively, that is
\[
\hat{Q}_g = \frac{1}{K + 1} \sum_{k=0}^{K} E \{w_{g,k}w_{g,k}'|Y_K\},
\] (39)
\[
\hat{R}_v = \frac{1}{K + 1} \sum_{k=0}^{K} E \{v_kv_k'|Y_K\}.
\] (40)

We now extend the framework in section V-A to the dynamic model setting we are considering in this paper. It becomes evident that this can be done by replacing the sample covariance in (37) by the conditional sample covariances. Specifically, for estimating \(\hat{Q}_g\) with both low-rank and sparsity constraints, it leads to
\[
\hat{Q}_{g,c} = \arg \min_{Q} L(\hat{Q}_g, Q) + \alpha \|Q\|_* + \gamma \|Q\|_1.
\] (41)

With the loss function taking the form of (38), i.e.
\[
L(\hat{Q}_g, Q) \triangleq \log \det Q + tr(Q^{-1}\hat{Q}_g),
\] (42)

it can be shown that (41) is equivalent to minimizing
\[
J_s(Q_{g,c}, R_v) = -J(Q_{g,c}, R_v) + \alpha \|Q_{g,c}\|_* + \gamma \|Q_{g,c}\|_1,
\] (43)
to which applying the majorization-minimization algorithm [14] amounts to solving the following minimization problem at each iteration
\[
\hat{Q}_{g,c}^{(l)} \triangleq \arg \min_{Q} L(\hat{Q}_{g,c}^{(l-1)}, Q) + \alpha \|Q\|_* + \gamma \|Q\|_1, \quad l \geq 1.
\] (44)

where \(\hat{Q}_{g,c}^{(l)}\) is the estimate of \(Q_{g,c}\) at the \(l\)th iteration, and
\[
L(\hat{Q}_{g,c}^{(l-1)}, Q) = tr((\hat{Q}_{g,c}^{(l-1)})^{-1}Q) + tr(Q^{-1}\hat{Q}_g).
\] (45)

The problem (44) is convex and can be solved, for instance, using the proximal gradient methods as shown in Algorithm 1 with
\[
\nabla Q \tilde{L}(\hat{Q}_{g,c}^{(l-1)}, Q) = (\hat{Q}_{g,c}^{(l-1)})^{-1} - Q^{-1}\hat{Q}_gQ^{-1}.
\] (46)

Alternatively, as in [15] we let
\[
L(\hat{Q}_g, Q) \triangleq \|\hat{Q}_g - Q\|_F^2
\] (47)
in (41) which leads to a convex problem that can be solved via proximal gradient methods such as the generalized forward-backward splitting or incremental proximal descent algorithm. Given that the only change made going from (37) to (41) and (43) is the replacement of the sample covariance by the conditional sample covariance, the strategy of choosing the regularization coefficients and step size remains the same as those in [14], [15]. However, when embedded within the EM iterations, the overall convergence becomes a challenging issue and is out of the scope of this paper. Although the above discussion has been focused on estimating \(Q_g\), the same approach is applicable to estimating \(R_v\) if similar types of structures are desirable for \(R_v\).

In Algorithm 1, we use the incremental proximal descent algorithm as an example to illustrate the overall constrained EM algorithm. \(\beta\) is the step size. The proximal operators in Algorithm 1 are given as follows [15]:

\[
\prox_{\gamma\|\cdot\|_1}(Z) = U \text{diag}((\sigma_i - \tau)_+)_i V^T.
\] (48)
\[
\prox_{\gamma\|\cdot\|_1}(Z) = \text{sgn}(Z) \circ (|Z| - \gamma)_+.
\] (49)

where \(U, V\) and \(\sigma_i\) are the singular vectors and values of the matrix \(Z\). The gradient term \(\nabla Q_g L(\hat{Q}_g, Q_{g,c})\) is given by (46) if the loss function in (42) is used, or simply the gradient of the loss function (47). It is understood that other algorithms for solving the low-rank and sparsity constrained problem may be used as well along with the EM iterations.

**Algorithm 1 Constrained EM covariance estimation with Incremental Proximal Descent Algorithm**

**Initialize** \(Q_g, R_v\),

**repeat**

**E-step:** Kalman smoothing to obtain \(P_{k|K}, P_{k+1|k}, \delta x_{k,k+1|K}\) and \(\delta e_{k|K}\) for \(k = 0, \ldots, K\).

**M-step:** compute \(\hat{Q}_g\) and \(\hat{R}_v\) via (35) and (36).

Set \(Q_{g,c} = Q_g\).

**repeat**

set \(Q_{g,c} = Q_{g,c} - \beta \nabla Q_{g,c} L(\hat{Q}_g, Q_{g,c})\).

Set \(Q_{g,c} = \prox_{\beta\|\cdot\|_1}(Q_{g,c})\).

Set \(Q_{g,c} = \prox_{\beta\|\cdot\|_1}(Q_{g,c})\).

Set \(Q_{g,c} = P_S(Q_{g,c})\).

until convergence

Set \(Q_g = Q_{g,c}\).

until convergence

**return** \(Q_g, R_v\).

Note that the rank and sparsity constraints proposed above serve to reflect realistic problem structures expected from
model knowledge, yet are still general and flexible enough to be broadly applicable. In industrial process control applications, \( Q_w \) is the covariance for process disturbances and may only have a few independent components that are not necessarily densely correlated. In addition, the matrix \( G \), mapping these disturbance components onto the state space, may be known priori to have very sparse number of significant entries. For example, in a simple first-order delayed system only a single state need to be perturbed by \( Q_w \) since clearly the states that propagate the delay will have no disturbance. In general the rank of the covariances (or inverses) quantifies the number of independent components in the sense of the marginal (conditional) dependence. The cardinality characterizes the pairwise dependence pattern in each of these independent components that dominate the structure. We have focused this section on estimating covariances with rank and sparsity structures. The same framework can be readily applied to the inverse covariances if that is preferred, such as when the information form of the Kalman filter is used, or the structural knowledge is provided in terms of the conditional instead of the marginal dependence.

VI. NUMERICAL EXAMPLES

In this section we use three numerical examples to demonstrate the performance of the constrained EM algorithm. As a baseline, we also compute the ML estimates from (22) and (25) by replacing \( E \{ z_k z_k^T | y_k \} \) with \( z_k z_k^T \), which is known in the simulation. In the first two cases we also compute the ALS estimates with optimal weighting, but not for the third case which has a dimensionality appearing too big for ALS to handle. We adopt the loss function (47) for the constrained EM algorithm in generating the results presented in this section. We also illustrate the effect of structural constraints on the algorithm convergence, using case B as an example.

A. Single Input, Single Output Example

This is the same example used in [8]. The known model parameters are

\[
A = \begin{bmatrix} 0.1 & 0 & 0.1 \\ 0 & 0.2 & 0 \\ 0 & 0 & 0.3 \end{bmatrix}, \quad G = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0.1 & 0.2 & 0 \end{bmatrix}
\]

and the data are generated with true noise covariances \( Q_w = 0.5 \) and \( R_v = 0.1 \). The filter gain is the steady-state Kalman filter based on an initial guess \( Q_w = 0.2 \) and \( R_v = 0.4 \), which is fixed in the ALS algorithm and in the EM algorithm updated with new estimates obtained at each iteration. Total 1000 data samples are generated and the simulation is repeated 100 times to obtain the mean and variance of the resulting estimates. The optimally weighted ALS has a time window length \( N = 15 \), the same as in [8].

In Figure 1, we give the scatter plot of the estimated \( Q_w \) vs \( R_v \), obtained using ML, EM and ALS methods, as well as the true value (the center of the crossties). As shown the ML estimates have very small bias, centering around the true value. The EM estimates are slightly biased but with a significantly smaller variance than the ALS estimates which have an extended amount of variations, especially along the diagonal direction.

B. 5x5 Example

In this example, the state dimension is \( N = 10 \), and the observation and the disturbances have the same dimension \( M = P = 5 \). The data are generated with the following model parameters:

\[
A = \text{diag}([0.1 \ 0.2 \ 0.3 \ 0.4 \ 0.5 \ 0.6 \ 0.7 \ 0.8 \ 0.9 \ 1.0]);
\]

and \( A(1,10) = 0.1 \);

\[
Q_w = \text{diag}([0.5 \ 0.4 \ 0.3 \ 0.2 \ 0.1]);
\]

\[
R_v = \text{diag}([0.1 \ 0.2 \ 0.3 \ 0.4 \ 0.5]);
\]

The elements of both \( G \in \mathbb{R}^{N \times P} \) and \( C \in \mathbb{R}^{M \times N} \) are randomly generated from the uniform distribution over \([0,1]\), while \( G \) is further made sparse by setting all its elements whose values are less than 0.4 to zero. All algorithms use the identity matrix as the initial guess for both \( R_v \) and \( Q_w \). The resulting estimates of \( R_v \) obtained via both the constrained EM and the optimally weighted ALS algorithms have comparable performance in terms of the bias and the variance, hence are not shown. The estimates of \( Q_w \) are given in Figure 2, with bias (top row) and variance (bottom row) obtained by averaging over 100 realizations. It is evident that the constrained EM estimates of \( Q_w \) have significantly smaller bias and variance than those from the ALS algorithm.

We have provided in Figure 4 the log-likelihood curves for the unconstrained and constrained EM algorithms, to illustrate the effect of constraints. As shown, the constrained EM algorithm takes several additional number of iterations than the unconstrained one (with the same stopping criterion). The log-likelihood value at convergence is also slightly

\[2 \text{Note that the ML estimate is computed here only for benchmarking purpose, given that we know the simulated data. It does not imply that the same can be done in practice.} \]
lower in the constrained case, which may be attributed to the additional penalty terms in the objective in (43).

C. 15x20 Example

The final example comes from a model describing the dynamic response of a hydrocracker, a type of chemical reactor found in most petroleum refineries [26]. A hydrocracker consists of several reactor beds in series that use hydrogen to crack large molecules into more smaller, more valuable molecules. The model developed here has 15 inputs and 20 outputs and 20 states following model reduction.

We compute both the ML and constrained EM estimates, starting from the identity matrix as the initial guesses for both \( \mathbf{R}_v \) and \( \mathbf{Q}_g \). The ALS algorithm fails to run due to memory limitation. In Figure 3, we plot the estimates of \( \mathbf{Q}_g \) obtained via ML and the constrained EM. As shown the constrained EM estimates has larger bias than the ML estimates, but the variance are comparable in the order of magnitude.

VII. CONCLUSION

We propose a structurally constrained EM algorithm for covariance estimation in state-space models, maximizing the log-likelihood function while incorporating rank and sparsity constraints. It is shown that the unconstrained EM covariance estimates are effectively conditional sample covariances based on the measurement data sets. Therefore we extend the optimization algorithms for low-rank and sparse covariance estimation given a sample covariance to the dynamic case, by replacing the sample covariance by the conditional sample covariance. Effectively this corresponds to augmenting the negative log-likelihood function by a combination of matrix \( \ell_1 \) norm and nuclear norm penalty terms, to enforce the desired structure. The resulting algorithms have been applied to three numerical cases, demonstrating favorable performance over the optimally weighted ALS algorithm. Finally the authors would like to thank the anonymous reviewers for their helpful comments.

APPENDIX

A. Express \( \mathbf{H}_k \mathbf{Q}_w \mathbf{H}_k^T \)^{-1} in terms of \( \mathbf{Q}_{g_k} \) and \( \mathbf{R}_v \)

Note that

\[
\mathbf{H}_k \mathbf{Q}_w \mathbf{H}_k^T = \begin{bmatrix} \mathbf{G}_k & -\mathbf{A}_k \mathbf{L}_k \end{bmatrix} \begin{bmatrix} \mathbf{Q}_w & \mathbf{R}_v \end{bmatrix} \begin{bmatrix} \mathbf{G}_k^T \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_{g_k} + (\mathbf{A}_k \mathbf{L}_k) \mathbf{R}_v (\mathbf{A}_k \mathbf{L}_k)^T & \mathbf{R}_v \\ -\mathbf{R}_v (\mathbf{A}_k \mathbf{L}_k)^T & \mathbf{R}_v \end{bmatrix},
\]

and using the following block matrix inverse identify:

\[
\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{D}^{-1} \mathbf{C} \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{D}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}
\]

(51)

where \( \mathbf{D} \triangleq \mathbf{A} - \mathbf{B} \mathbf{D}^{-1} \mathbf{C} \) is the Shur complement of \( \mathbf{D} \), it follows that

\[
(\mathbf{H}_k \mathbf{Q}_w \mathbf{H}_k^T)^{-1} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ (\mathbf{A}_k \mathbf{L}_k)^T & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_{g_k}^{-1} & \mathbf{R}_v^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{A}_k \mathbf{L}_k \\ \mathbf{0} & \mathbf{I} \end{bmatrix}
\]

\[
-\mathbf{L}_k^T \mathbf{Q}_{g_k}^{-1} \mathbf{L}_k + \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{R}_v^{-1} \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix}
\]

(52)
where $L_{Ak} \triangleq [I \ A_k L_k]$. Since

$$\det\left( \begin{bmatrix} I & 0 \\ (A_k L_k)^t & I \end{bmatrix} \right) = 1,$$  \hspace{1cm} (53)

hence it follows

$$\det(H_k Q_{\theta} H_k^t)^{-1} = \det(Q_{\theta^{-1}})^{-1}.$$  \hspace{1cm} (54)

**B. Deriving $\Phi_{ij}$**

$$\Phi_{11} = E\left\{ (\varepsilon_{k+1} - A_k \varepsilon_k) (\varepsilon_{k+1} - A_k \varepsilon_k)^t | \mathcal{Y}_k \right\} = E\left\{ (\varepsilon_{k+1} - A_k \varepsilon_k)^t A_k^t - (\varepsilon_{k+1} - A_k \varepsilon_k)^t A_k^t | \mathcal{Y}_k \right\}$$

since

$$E\left\{ (\varepsilon_{k+1} - A_k \varepsilon_k)^t | \mathcal{Y}_k \right\} = P_{k+1|K} + \hat{x}_{k|K} - \tilde{x}_{k|k} | \mathcal{Y}_k,$$

$$E\left\{ (\varepsilon_{k+1} - A_k \varepsilon_k)^t | \mathcal{Y}_k \right\} = P_{k+1|K} + (\hat{x}_{k|K} - \tilde{x}_{k|k}) | \mathcal{Y}_k.$$

hence

$$\Phi_{11} = P_{k+1|K} + A_k P_{k|K} A_k^t - A_k P_{k,k+1|K}$$

$$- P_{k,k+1|K} A_k^t + \delta x_{k+1|K} A_k^t.$$  \hspace{1cm} (55)

where

$$\delta x_{k+1|K} = (\tilde{x}_{k+1|K} - \hat{x}_{k+1|K}) - A_k (\tilde{x}_{k|K} - \hat{x}_{k|K}).$$  \hspace{1cm} (56)

Similarly

$$\Phi_{12} = - (P_{k,k+1|K} - A_k P_{k|K}) C_k^t + \epsilon_{k|K} e_{k|K}^t.$$  \hspace{1cm} (57)

$$\Phi_{22} = C_k P_{k|K} C_k^t + \epsilon_{k|K} e_{k|K}^t.$$  \hspace{1cm} (58)

where $e_{k|K} \triangleq e_k - C_k (\hat{x}_{k|K} - \tilde{x}_{k|k})$ is the smoothed residual error.

**C. Proof for $\tilde{Q}_g$ and $\tilde{R}_v$ as conditioned sample covariances**

Substituting the time-update equation for the state estimate

$$\hat{x}_{k+1|K} = A_k \hat{x}_{k|K} + B_k u_k$$

into (56) and taking expectation on both sides of the state equation w.r.t. $\mathcal{Y}_K$ gives

$$\delta x_{k+1|K} = \tilde{x}_{k+1|K} - A_k \hat{x}_{k|K} - B_k u_k$$

$$= G_k \tilde{w}_{k|K} \triangleq \tilde{w}_{g,k|K}.$$  \hspace{1cm} (60)

Assuming $u_k$ are known, the covariance of $\delta x_{k+1|K}$ is the same as that of $\tilde{x}_{k+1|K} - A_k \hat{x}_{k|K}$ which is given by the first four terms in (35). Therefore

$$\tilde{Q}_g = \frac{1}{K+1} \sum_{k=0}^K E\left\{ \tilde{w}_{g,k} \tilde{w}_{g,k}^t | \mathcal{Y}_K \right\}.$$  \hspace{1cm} (61)

Similarly taking expectation on both sides of the measurement equation w.r.t. $\mathcal{Y}_K$ gives

$$e_{k|K} = y_k - C_k \hat{x}_{k|K} = \tilde{v}_{k|K}$$

whose covariance is $C_k P_{k|K} C_k^t$. Therefore

$$\tilde{R}_v = \frac{1}{K+1} \sum_{k=0}^K E\left\{ v_{k} v_{k}^t | \mathcal{Y}_K \right\}.$$  \hspace{1cm} (63)

**REFERENCES**


