Optimal LQG Control Across Packet-Dropping Links

Vijay Gupta*, Babak Hassibi, Richard M. Murray

Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA-91125, USA.

Abstract

We examine two special cases of the problem of optimal Linear Quadratic Gaussian control of a system whose state is being measured by sensors that communicate with the controller over packet-dropping links. We pose the problem as an information transmission problem. Using a separation principle, we decompose the problem into a standard LQR state-feedback controller design, along with an optimal encoder-decoder design for propagating and using the information across the unreliable link. Our design is optimal among all causal algorithms for any arbitrary packet drop pattern. Further, the solution is appealing from a practical point of view because it can be implemented as a small modification of an existing LQG control design.

Key words: LQG control, Networked control systems, Packet-dropping links, Separation principle, Sensor fusion

1 Introduction

Recently, much attention has been directed toward systems which are controlled over a communication link (see, e.g., [1,4] and the references therein). In such systems, the control performance can be severely affected by the properties of the communication channel. Understanding and counteracting the effects of the channels will become increasingly important as emerging applications of decentralized control mature.

* Corresponding author.

Email addresses: vijay@cds.caltech.edu (Vijay Gupta), hassibi@caltech.edu (Babak Hassibi), murray@caltech.edu (Richard M. Murray).

Preprint submitted to Elsevier Science 16 November 2006
In this note, we are specifically interested in systems communicating over links that randomly drop packets. The nominal system is shown in Figure 1, where the $n$ channels represent communication links or networks that randomly erase packets being communicated from the sensors to the controller. In particular, we discuss two special cases of the problem.

(1) Case $C_1$: There is only one sensor (and one channel) present.

(2) Case $C_2$: There are 2 sensors present. However, while channel 1 drops packets randomly, channel 2 transmits all packets.

While the case $C_1$ is important in its own right, it is also the basic system we need to understand for more general systems with multiple plants, sensors and controllers. Preliminary work using this model studied stability of systems utilizing lossy packet-based communication, as in [18,22]. Performance of such systems was analyzed by Seiler in [18] and by Ling and Lemmon in [14] assuming certain dropout models. Approaches to compensate for the data loss have also been proposed. Nilsson [15] proposed keeping the old control or generating a new control by estimating the lost data. Hadjicostis and Touri [9] analyzed the performance when lost data is replaced by zeros. Ling and Lemmon [13,14] posed the problem of optimal compensator design for the case when data loss is independent and identically distributed (i.i.d.) as a nonlinear optimization. Azimi-Sadjadi [2] proposed a sub-optimal estimator and regulator to minimize a quadratic cost. Schenato et al. [17] and Imer et al. [11] extended this approach to obtain optimal controllers when the packet drops are i.i.d. The problem of optimal estimation across a packet-dropping link was considered by Sinopoli et al in [19] and extended by Gupta et al in [8].

However, most of the solutions proposed in these references aim at designing a packet-loss compensator as shown in Figure 2. Most works assume a communication link to be present only between the controller and the actuator.
The compensator accepts those packets that the link successfully transmits and comes up with an estimate for the time steps when data is lost. This estimate is then used by the controller. Our work takes a more general approach by seeking the LQG optimal control for this packet-based problem. In particular, for the case $C_1$, our architecture is as shown in Figure 3. Recognizing that the problem is of making sure that the controller has access to the maximal possible information set (hence an information transmission problem), we introduce an encoder at the sensor end. The compensator then effectively becomes a decoder for the information being transmitted over the link. We jointly design the controller, the encoder and the decoder to solve the optimal LQG problem. Even though sensors equipped with wireless or network communication capabilities will likely have some computational power available, we look for encoding and decoding algorithms that are recursive in structure. Recursive algorithms require a constant amount of memory, processing and transmission and hence can be used even in the face of limited resources.
There does not appear to be existing work dealing with the case $C_2$ specifically. We encounter this case in our work on the multi-vehicle wireless testbed [7] where each vehicle is equipped with an on-board gyro and also obtains measurements from an overhead camera. While the gyro-controller link is hard-wired and hence does not drop packets, the camera communicates to the controller over a wireless link that randomly drops packets. Thus this situation is identical to the case $C_2$. Our solution to this problem again adopts the philosophy of using some computation at the sensor end to combat the effects of the channels. Our architecture is as shown in Figure 4. We again provide recursive yet optimal designs of the encoders, the decoder and the controller.

Since the focus of the paper is on presenting the idea of information pre-processing to counter channel effects in networked control, to simplify the presentation, we will assume a channel between the sensor and the controller only. We will, however, revisit the problem for a channel being present between the controller and the actuator in Section 3.

The main contribution of the paper is posing and solving the problem of LQG control across a communication channel as an information transmission problem. Because of the real-time constraint of the control problem, block coding type operations cannot be used. For the specific cases $C_1$ and $C_2$, we obtain the optimal encoding and decoding strategies for the purpose of LQG control. The strategies are optimal in the sense that no other causal strategy can lead to a better performance even though our strategies only require bounded memory, processing and transmission.

As an intermediate step, we also solve the following problem, referred to as case $C_3$. Suppose, as shown in Figure 5, two sensors are estimating a process jointly while communicating over links that drop packets stochastically. What information should the sensors exchange? Work related to case $C_3$ has dealt
Fig. 5. Structure of the joint estimation problem (Case C₃).

with fusion of data from multiple sensors and track-to-track fusion. A usual starting point for such works is an attempt to decentralize the Kalman filter as, e.g., in [21]. However this approach requires that data about the global estimate be sent from the fusion node to the local sensors. This difficulty was overcome in [5,20] and further in [10] where both the measurement and time update steps of the Kalman filter were decentralized. Alternative approaches for data fusion from many nodes include using Bayesian methods [6], a scattering framework [12], algorithms based on decomposition of the information form of the Kalman filter [16] and so on. However these approaches assume a fixed communication topology among the nodes with a link, if present, being perfect. In our case, packets of information are dropped randomly by the communication channels. This random loss of information reintroduces the problem of correlation between the estimation errors of various nodes [3] and renders the approaches proposed in the literature as sub-optimal. We propose a recursive yet optimal strategy to find the optimal global estimate for each node.

This paper is organized as follows. We begin in the next section by posing the LQG problem in a packet-based setting. We then discuss a separation between control and estimation costs, and present an optimal solution to the estimation problem. We discuss some extensions to the algorithm. Finally, we analyze the stability and performance of our system.

2 Problem Formulation

Consider a discrete-time linear system evolving according to

$$x_{k+1} = Ax_k + Bu_k + w_k,$$

where $x_k \in \mathbb{R}^n$ is the process state, $u_k \in \mathbb{R}^m$ is the control input and $w_k$ is process noise assumed to be white, Gaussian, and zero mean with covariance
matrix $Q_w$. The initial condition $x_0$ is assumed to be independent of $w_k$ and to have mean zero and covariance matrix $Q_0$. The state of the plant is measured by two sensors according to the equations

$$y_k^i = C^i x_k + v_k^i, \quad i = 1, 2. \tag{2}$$

The measurement noises $v_k^i$’s are assumed white, zero-mean, Gaussian (with covariance matrix $Q_v^i$) and independent of the plant noise $w_k$ and of each other. Note that substituting $C^2 = 0$ and $Q_v^2 = 0$ would reduce the case $C_1$ to be a special case of $C_2$. Hence, from now on, we will carry out the derivation for case $C_2$ only and adapt the results for the case of one sensor. Each sensor communicates its own measurements (or some function of the measurements) to the controller. We impose the constraint that the function communicated should be a finite vector, whose size does not increase with time. Sensor 1 communicates over channel 1 that randomly drops packets while sensor 2 utilizes channel 2 that is perfect. For the moment we ignore delays and packet reordering; it will be shown that these effects can be accounted for with time-stamping and a slight modification to our design. At each time step $k$,

- A packet with a function of the measurements is created at both the sensors.
- The packets are sent across the link.
- The packet over channel 1 is either received instantaneously, or dropped, probabilistically.

The packet dropping in channel 1 is a random process. We refer to individual realizations of this random process as packet drop sequences. The packet drop sequence $P$ is a binary sequence $\{\eta_k\}_{k=0}^\infty$ in which $\eta_k$ takes the value “received” if the link delivers the packet at time step $k$, and “dropped” otherwise. We assume sufficient bits per packet and a high enough data rate so that quantization error is negligible. We also assume that enough error-correction coding is done within the packets so that the packets are either dropped or received without error. Denote by $s_k^i$ the finite vector transmitted from the sensor $i$ to the controller at time step $k$. By causality, $s_k^i$ can depend (possible in a time-varying manner) on $y_k^0, y_k^1, \ldots, y_k^i$, i.e., $s_k^i = f_k^i(y_k^0, y_k^1, \ldots, y_k^i)$. The information set, $I_k$ available to the controller at time $k$ is the union of two sets $I_k^1$ and $I_k^2$ defined by

$$I_k^1 = \{s_j^1 | \forall j \text{ s.t. } \eta_j = \text{“received”} \} \quad I_k^2 = \{s_j^2 | \forall j = 0 \ldots k \}$$

1 The results continue to hold for time-varying systems, but we consider the time-invariant case to simplify the presentation.

2 This assumption merely means that a sufficient number of bits is available so that the effect of quantization error is swamped by the effect of the process and the measurement noises. We do not assume an infinite number of bits, so that interleaving of bits to transmit an infinite amount of data is not admissible.
Also denote by \( t_l(k) \leq k \) the last time-step at which a packet was delivered over link \( L_1 \). That is \( t_l(k) = \max\{j \leq k \mid \eta_j = \text{“received”}\} \). The maximal information set, \( I_k^{\max} \) at time-step \( k \) is then the union of \( I^2_k \) and the set \( I_k^{\max} = \{y_j^1 \mid 0 \leq j \leq t_l(k)\} \). The maximal information set is the largest set of output measurements on which the control at time-step \( k \) can possibly depend. In general, the set of output measurements on which the control depends will be less than this set, since earlier packets, and hence measurements, may have been dropped. As stated earlier, the only restriction we impose is that the vectors \( s_k \) not increase in size as \( k \) increases. We will call the set of \( f_i^1 \)'s which fulfill this requirement as \( F \). The control \( u_k \) is a function of the information sets \( I^1_k \) and \( I^2_k \). We shall assume perfect knowledge of the system parameters \( A, B, C, Q_w \) and \( Q_v \)'s at the controller. Moreover we assume that the controller (and the decoder) have access to the previous control signals \( u_0, u_1, \cdots, u_{k-1} \). We can thus pose the packetized LQG problem as:

\[
\min_{u,f} J_K(u, f, P_1, P_2) = E \left[ \sum_{k=0}^{K} (u_k^T Q_c u_k + x_k^T R_c x_k) + x_{K+1}^T P_{K+1} x_{K+1} \right].
\]

(3)

Here \( K \) is the horizon on which the plant is operated and the expectation is taken over the uncorrelated variables \( x_0, \{w_k\} \) and \( \{v_k\} \). Note that the cost functional \( J \) above depends on the random packet-drop sequence \( P \). However, we do not average across packet-drop processes; the solution we will present is optimal for arbitrary realizations of the packet dropping process.

3 Optimal Encoder and Decoder Design

Recall that we wish to construct the optimal control input based on the information set \( I_k^{\max} \), but we have not yet specified how to design \( f^1_i \)'s that will allow the controller to compute that. If channel 1 does not drop packets, sending the current measurement \( y_k^1 \) in the current packets is sufficient. When channel 1 randomly drops packets, a naive solution would be to send the entire history of the output variables at each time step. This would certainly be an optimal solution; however, it is not allowed since it requires increasing data transmission as time evolves. Surprisingly, we can achieve performance equivalent to the naive solution using a constant amount of transmission, and memory. To this end, we first state the following separation principle.

**Theorem 1 (Separation)** For the packet-based optimal control problem defined in section 2, suppose that both the sensors transmit all the previous measurements at every time step, so that the decoder has access to the maximal information set \( I_k^{\max} \) at every time step. Then, for an optimizing choice of the control, the control and estimation costs decouple. Further, the optimal control input at time \( k \) is given by the mmse estimate of the LQ optimal control input.
\( \bar{u}_k \) given \( I^\text{max}_k \) and the previous control inputs \( u_0, \ldots, u_{k-1} \).

**PROOF.** The proof is along the lines of the standard separation principle and is omitted for space constraints. \( \square \)

Thus, using Theorem 1, the controller design part of the problem is solved. The optimal controller is the solution to the LQ control problem. Moreover, note that the optimal controller does not need to have access to the information set \( I^\text{max}_k \) at every time step \( k \). The encoders and the decoder only need to ensure that the controller receives the mmse estimate of the LQ optimal control input, or equivalently, the mmse estimate of the state \( x_k \) given \( I^\text{max}_k \) and the previous control inputs \( u_0, \ldots, u_{k-1} \).

**Optimal Transmission and Estimation Algorithm:** Let \( \hat{x}^i_{k|l} \) denote the mmse estimate of \( x_k \) based on all the measurements of sensor \( i \) up to time \( l \) and all previous control inputs. Denote the corresponding error covariance by \( P^i_{k|l} \). Also denote by \( \bar{x}^i_{k|l} \) the estimate of \( x_k \) based on all the measurements of sensor \( i \) up to time \( l \) while assuming that no control input was applied as \( x_k \) evolved according to (1). \( \hat{x}^i_{k|l} \) can be evaluated through a recursive filter identical to a Kalman filter except that no control input is applied during the time update step. We will call such a filter a modified Kalman filter.

1. **Encoder for sensor 1:** At each time step \( k \),
   - Use \( y^1_k \) to obtain \( \bar{x}^1_{k|k} \) and \( P^1_{k|k} \) through a modified Kalman filter.
   - Calculate \( \lambda^1_k = (P^1_{k|k})^{-1} \bar{x}^1_{k|k} - (P^1_{k|k-1})^{-1} \bar{x}^1_{k|k-1} \).
   - Calculate global error covariance matrices \( P_{k|k} \) and \( P_{k|k-1} \) using
     \[
     (P_{k|k})^{-1} = (P_{k|k-1})^{-1} + (C^1)^T (Q^1_1)^{-1} (C^1) + (C^2)^T (Q^2_2)^{-1} (C^2)
     \]
     \[
     P_{k|k-1} = A P_{k-1|k-1} A^T + Q_w.
     \]
   - Obtain \( \gamma^1_k = (P^1_{k|k-1})^{-1} A_{k-1} P_{k-1|k-1} \).
   - Finally calculate \( i^1_k = \lambda^1_k + \gamma^1_k i^1_{k-1} \) with \( i^1_{k-1} = 0 \) and transmit it.
2. **Encoder for sensor 2:** At each time step \( k \), transmit the measurement \( y^2_k \).
3. **Decoder:** At each time step \( k \),
   - Calculate \( i^2_k \) using \( y^2_k \) with the same algorithm for sensor 2 as the one followed by the encoder for sensor 1.
   - Calculate its estimate \( \hat{x}^{dec}_k \) as follows:
     a. If \( \eta_k = \text{"received"} \), both the links have successfully transmitted packets. In that case, calculate \( \psi_k = (P_{k|k-1})^{-1} B u_{k-1} + \gamma^1_k \psi_{k-1} \) with \( \psi_0 = 0 \) and then \( \hat{x}^{dec}_k \) using \( (P_{k|k})^{-1} \hat{x}^{dec}_k = i^1_k + i^2_k + \psi_k \).
If $\eta_k = \text{"dropped"}$, the link has dropped the current packet. In this case, propagate the estimate $\hat{x}_{k-1}^{\text{dec}}$ using the measurement $y_k^2$ and the control $u_{k-1}$ through a Kalman filter.

**Theorem 2 (Optimal Estimation)** In the above algorithm, $\hat{x}_k^{\text{dec}} = \hat{x}_{k|I_k}^{\text{max}}$.

**PROOF.** Consider a centralized filter that has access to measurements from a sensor of the form

$$
y_k = Cx_k + v_k = \begin{bmatrix} C^1 \\ C^2 \end{bmatrix} x_k + \begin{bmatrix} v^1_k \\ v^2_k \end{bmatrix}.
$$

Let $R$ be the covariance matrix of the noise $v_k$. Since $R$ is block-diagonal, the measurement update equations of the Kalman filter are

$$
\left( P_{k|k} \right)^{-1} = \left( P_{k|k-1} \right)^{-1} + C^T R^{-1} C
= \left( P_{k|k-1} \right)^{-1} + \sum_i \left[ \left( P_{i|k} \right)^{-1} - \left( P_{i|k-1} \right)^{-1} \right]
$$

$$
\left( P_{k|k} \right)^{-1} \hat{x}_{k|k} = \left( P_{k|k-1} \right)^{-1} \hat{x}_{k|k-1} + C^T R^{-1} y_k
= \left( P_{k|k-1} \right)^{-1} \hat{x}_{k|k-1} + \sum_i \left[ \left( P_{i|k} \right)^{-1} \hat{x}_{i|k} - \left( P_{i|k-1} \right)^{-1} \hat{x}_{i|k-1} \right].
$$

Recognizing that the time update equations are

$$
P_{k|k-1} = AP_{k-1|k-1}^T + Q_w, \quad \hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu_{k-1},
$$

we can write

$$
\left( P_{k|k} \right)^{-1} \hat{x}_{k|k} = \sum_i I^{i}_k + \Psi_k.
$$

The term $I^{i}_k$ is the contribution of the measurements of the $i$-th sensor and is given by

$$
I^{i}_k = \Lambda^{i}_k + \Gamma_k \Lambda^{i}_{k-1} + \Gamma_k \Gamma_{k-1} \Lambda^{i}_{k-2} + \cdots + (\Gamma_k \Gamma_{k-1} \cdots \Gamma_1) \Lambda^{i}_0,
$$

where

$$
\Lambda^{i}_k = \left( P_{i|k} \right)^{-1} \bar{x}^{i}_{k|k} - \left( P_{i|k-1} \right)^{-1} \bar{x}^{i}_{k|k-1}, \quad \Gamma_k = \left( P_{k|k-1} \right)^{-1} A P_{k-1|k-1}.
$$

The term $\Psi_k$ is the contribution of the control input and can be calculated recursively through

$$
\Psi_k = \left( P_{k|k-1} \right)^{-1} Bu_{k-1} + \Gamma_k \Psi_{k-1}.
$$
Proposition 2 presents the solution to the estimation problem in the case $C_3$ mentioned in Section 1 since we can use an encoder and a decoder described in the algorithm above at each sensor. Moreover, taken together, Propositions 1 and 2 solve the packet-based LQG control problem posed in Section 2.

**Theorem 3 (Optimal Packet-Based LQG Control)** For the packet-based optimal control problem stated in section 2, an LQR state feedback design together with the optimal transmission-estimation algorithm described above achieves the minimum of $J(u, f, P)$ for any $P$.

The information vector $I_k^i$ ‘washes away’ the effect of any previous packet losses. If $\eta_k = \text{“received”}$, $\hat{x}_{k|i}$ is calculated as if all the previous measurements from both sensors were available. Also, we have made no assumption about the packet dropping behavior. The algorithm provides the optimal estimate based on $I_k^\text{max}$ for an arbitrary packet drop sequence, irrespective of whether the packet drop can be modeled as an i.i.d. process (or a more sophisticated model like a Markov chain) or whether its statistics are known to the plant and the controller. Finally, we do not assume knowledge of the cost matrices $Q^c$ and $R^c$ at the sensor end. Thus the cost function (and hence the optimal controller) can be changed without affecting the sensor/encoder operation. This is important, e.g., in our MVWT work, where the matrices $Q^c$ and $R^c$ are user-specified while the encoder code is much harder to change.

**The single sensor case**: For case $C_1$, the algorithm reduces to the following.

- The encoder (at the sensor end) receives the measurement $y_k$. It runs the modified Kalman filter and transmits the output $\bar{x}_{k|k}$ across the link.
- The decoder (at the controller end) maintains two variables: $\psi_k$ that takes into account the effect of the control inputs, and the estimate $\hat{x}_{k|\text{dec}}$ that is updated as follows:
  - If $\eta_k = \text{“received”}$, the decoder receives $i_k$, and sets $\hat{x}_{k|\text{dec}} = \bar{x}_{k|k} + \psi_k$.
  - If $\eta_k = \text{“dropped”}$, then the decoder implements the linear predictor:
    $\hat{x}_{k|\text{dec}} = A\hat{x}_{k-1|\text{dec}} + \psi_k$.  \hspace{1cm} (5)

**Presence of delays**: The solution can readily be extended to the case when the channel applies a random delay to the packet so that packets might arrive at
the decoder delayed or even out-of-order, if we assume that there is a provi-
sion for time-stamping the packets sent by the encoder. For ease of notation,
we present the solution for optimal asynchronous estimation for the case \( C_1 \).
The case \( C_2 \) is similar. At each time step, the decoder will face one of four
possibilities, and will update its estimate as described below:

- It receives \( \tilde{x}_{k|k} \). It calculates the estimate according to
  \( \hat{x}_{dec}^k = \tilde{x}_{k|k} + \psi_k \).
- It does not receive anything. It uses the predictor equation (5) on
  \( \hat{x}_{dec}^{k-1} \).
- It receives \( \tilde{x}_{m|m} \) while at a previous time step, it has already received \( \tilde{x}_{n|n} \),
  where \( n > m \). It discards \( \tilde{x}_{m|m} \) and uses (5) on \( \hat{x}_{dec}^{k-1} \).
- It receives \( \tilde{x}_{m|m} \) and at no previous time step has it received \( \tilde{x}_{n|n} \), where
  \( n > m \). It uses \( \tilde{x}_{m|m} \) to calculate \( \hat{x}_{m}^{dec} \) and obtains \( \hat{x}_{dec}^k \) through (5).

Further, as long as the delays are finite (i.e., the packet drop probability does
not change), the stability conditions derived in Section 4 will not change.

\textbf{Channel between the controller and the actuator:} As pointed out in [17,11] if
we have a channel between the controller and the plant, the separation prin-
ciple would still hold, provided there is a provision for acknowledgment from the
receiver to the transmitter for any packet successfully received over the chan-
nel. Since the decoder has access to the control input applied at every time
step, our algorithm can easily be generalized to this case. We can also ask
the question of the optimal encoder-decoder design for the controller-actuator
channel. However, this will depend on the information that is assumed to be
known to the actuator (e.g. the cost matrices \( Q^c \) and \( R^c \) and the measurements
from the sensor) and is beyond the scope of this paper.

\textbf{Multiple sensors:} The assumption of only one sensor transmitting over a per-
fected channel can be removed. If multiple sensors transmit information over
perfect channels, the algorithm can be extended and remains optimal even
though it does not require the sensors to communicate with each other. How-
ever, the algorithm does not extend to multiple packet dropping channels since
the encoder for sensor 1 uses the fact that sensor 2 will transmit its informa-
tion at every time step. For multiple channels, this assumption will not be
satisfied. Extension of the algorithm to such cases remains an open problem.

4 Analysis of the Proposed Algorithm

In this section, we model the channel erasures as occurring according to a
Markov chain and analyze the stability and performance of our design. The
channel exists in either of two states, state 1 corresponding to a packet drop
and state 2 corresponding to no packet drop and it transitions probabilistically
between these states according to the transition probability matrix \( Q = [q_{ij}] \).
Note that i.i.d. drops can be handled by a special choice of $Q$. We assume strict causality in the Kalman filter used by the encoder. Thus to calculate the estimate of $x_k$, only the measurements till time step $k-1$ are used. The analysis for the causal case is similar. Finally we assume that the pairs $(A, B)$ and $(A, Q_{w}^{1/2})$ are stabilizable and the pairs $(A, C)$ and $(A, (R^c)^{1/2})$ are detectable, where $C$ is defined in (4).

Denote by $y_k$ the vector formed by stacking $y_1^k$ and $y_2^k$. We have three dynamical systems. The plant state $x_k$ evolves as in (1). The state $\hat{x}_k$ of a Kalman filter with access to measurements from both sensors at every time step evolves as

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + K^c_k (y_k - C\hat{x}_k).$$

Finally the state $\hat{x}^{dec}_k$ of the estimator at the decoder evolves according to

$$\hat{x}^{dec}_{k+1} = \begin{cases} A\hat{x}^{dec}_k + Bu_k + K^d_k \left(y_2^k - C^2\hat{x}^{dec}_k\right) & \text{channel in state 1} \\ \hat{x}_{k+1} & \text{otherwise.} \end{cases}$$

Denote $e_k = x_k - \hat{x}_k$ and $t_k = \hat{x}_k - \hat{x}^{dec}_k$. Since $u_k = F_k \hat{x}^{dec}_k$, (1) implies

$$x_{k+1} = (A + BF_k)x_k + w_k - BF_k(t_k + e_k).$$

Since $(A, B)$ is stabilizable and $F_k$ is the optimum control law, the system would be stable in the bounded covariance sense as long as the disturbances $w_k$, $t_k$ and $e_k$ have bounded covariances. We assume the noise $w_k$ has bounded covariance matrix. Also $e_k$ has bounded covariance matrix by our detectability assumption. Finally $t_k$ evolves according to

$$t_{k+1} = \begin{cases} (A - K^d_k C^2) t_k + L^1(e_k) + L^2(v_1^k) + L^3(v_3^k) & \text{channel in state 1} \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where $L^1(\beta)$ denotes a term linear in $\beta$. For $t_k$ to be of bounded covariance, the Markov jump system of (6) needs to be stable. Finally, since our controller and encoder/decoder design is optimal, if the closed loop is unstable with our design, it is not stabilizable by any other design. We can thus say the following.

**Theorem 4 (Stability Condition)** Consider the control problem defined in Section 2 in which the packet erasure channel is modeled as a Markov chain with transition probability matrix $Q$. The system is stabilizable, in the sense that the covariance of the state is bounded, if and only if $q_{22} |\lambda_{\max} \left( \bar{A} \right) |^2 < 1$, where $\lambda_{\max} \left( \bar{A} \right)$ is the maximum magnitude eigenvalue of the unobservable part of matrix $A$ when $(A, C^2)$ is put in the observer canonical form. Further, if the system is stabilizable, one controller and encoder/decoder design that stabilizes the system is given in Theorem 3.
Using the results of [15], we can also calculate the total quadratic cost incurred by the system for the infinite-horizon case if we make the additional assumption that the Markov chain is stationary and regular. We state the result for the case $C_1$. We consider the cost $J_\infty \triangleq \lim_{K \to \infty} J_K$ and obtain

$$J_\infty = \lim_{K \to \infty} E \left[ x_K^T R^c x_K + u_K^T Q^c u_K \right] = \text{trace} \left( P_\infty^c R^c \right) + \text{trace} \left( P_\infty^u Q^c \right), \quad (7)$$

where $P_\infty^c = \lim_{K \to \infty} E \left[ x_K^T x_K \right]$ and $P_\infty^u = \lim_{K \to \infty} E \left[ u_K^T u_K \right]$. We see that

$$P_\infty^c = \begin{bmatrix} I & 0 \\ 0 & P_\infty \end{bmatrix}$$

and

$$P_\infty^u = F \begin{bmatrix} I - I - I \\ -I \end{bmatrix} P_\infty F^T,$$

where $P_\infty = \tilde{P}_1 + \tilde{P}_2$ and $\tilde{P} = \begin{bmatrix} \text{vec}(\tilde{P}_1)^T & \text{vec}(\tilde{P}_2)^T \end{bmatrix}^T$. Then, it can be shown that $\tilde{P}$ is the unique solution to the linear equation

$$\tilde{P} = \left( Q^T \otimes I \right) \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \tilde{P} + \left( Q^T \otimes I \right) \begin{bmatrix} \pi_1 & 0 \\ 0 & \pi_2 \end{bmatrix} \otimes I \right) G.$$

In the above equation, $A_i = A_i \otimes A_i$, and $G = \begin{bmatrix} \text{vec}(G_1)^T & \text{vec}(G_2)^T \end{bmatrix}^T$, where

$$A_1 = \begin{bmatrix} A + BF & -BF & -BF \\ 0 & A - KC & 0 \\ 0 & -KC & A \end{bmatrix} \quad A_2 = \begin{bmatrix} A + BF & -BF & -BF \\ 0 & A - KC & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} I \\ I - K \\ 0 - K \end{bmatrix} \quad B_2 = \begin{bmatrix} I \\ I - K \\ 0 - K \end{bmatrix} \quad G_i = B_i \begin{bmatrix} Q_w & 0 \\ 0 & Q_v \end{bmatrix} B_i^T,$$

and $F$ and $K$ are optimal steady state controller and Kalman filter gains.

Example: We now consider some examples to illustrate the performance of our algorithm. We consider the example system considered by Ling and Lemmon in [13]. The system evolves as

$$x_{k+1} = \begin{bmatrix} 0 & -2 \\ 1 & -1 \end{bmatrix} x_k + \begin{bmatrix} 2 \\ 1 \end{bmatrix} u_k + \begin{bmatrix} 2 \\ 1 \end{bmatrix} w_k.$$

The (2,1) component of the state $x_k$ is observed without any measurement noise at every time step. The process noise $w_k$ is zero mean with unit vari-
Our algorithm: unity feedback
Our algorithm: LQ optimal control
Ling and Lemmon algorithm

Fig. 6. Comparison of performance for various algorithms.

The packet drop process is i.i.d. The cost considered is the steady state output error $\lim_{K \to \infty} y^2_K$. [13] assumes unity feedback when packets are delivered and gives an optimal compensator design when packets are lost. On analyzing the system, we observe that our algorithm allows the system to be stable up to a packet drop probability of 0.5 while the optimal compensator in [13] is stable only if the probability is less than 0.25. Figure 6 compares the performance obtained using the two algorithms. The performance is much better throughout the range of operation for our algorithm, even if we assume unity feedback in our algorithm. This shows that the difference in performance is mainly due to the novel encoding-decoding algorithm proposed.

Now we consider the same system being observed by two sensors of the form

$$y^1_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v^1_k \quad y^2_k = \begin{bmatrix} 0 & 1 \end{bmatrix} x_k + v^2_k.$$ 

The sensor noises are zero mean with variance 10 and 1 respectively. We consider the cost function $\lim_{K \to \infty} (y^2_K)^2$. Figure 7 shows the simulated performance of our algorithm as a function of the packet loss probability. We also plot the performance for a hypothetical sensor that receives information from both sensors without any packet drop and for a scheme in which sensors exchange measurements. Even in this very simple case, our algorithm can lead to a performance gain of up to 40% over the strategy of using no encoder.

5 Conclusions and Future Work

In this paper, we considered the problem of optimal LQG control when one sensor and the controller are communicating across a packet erasure channel. We identified the information that the sensor should provide to the controller to
obtain the optimal LQG performance. This can be viewed as constructing an encoder for the channel. We also designed the decoder that uses the information it receives across the link to construct an estimate of the state of the plant. The proposed algorithm is recursive yet optimal irrespective of the packet drop pattern. For the case of packet drops occurring according to a Markov chain, we carried out stability and performance analysis of our algorithm.

References


