

A Scheme for Joint Quantization, Error Protection and Feedback Control over Noisy Channels

Lei Bao, Mikael Skoglund and Karl Henrik Johansson
School of Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden
Email: lei.bao@ee.kth.se, skoglund@ee.kth.se, kallej@ee.kth.se

Abstract— We study a closed-loop scalar control system with feedback transmitted over a discrete noisy channel. For this problem, we propose a joint design of the state measurement quantization, protection against channel errors, and control. The goal is to minimize a linear quadratic cost function over a finite horizon. In particular we focus on a special case where we verify that certainty equivalence holds, and for this case we design joint source–channel encoder and decoder/estimator pairs. The proposed algorithm leads to a practically feasible design of time-varying non-uniform quantization and control. Numerical results demonstrate the promising performance obtained by employing the proposed iterative optimization algorithm.

I. INTRODUCTION

In recent years, the demand for sharing resources efficiently in large networked systems has been continuously increasing. However, in many situations, there is a challenging conflict between the amount of transmitted data and the response time. In particular for emerging control applications, limits imposed on available signaling bandwidth from communication channels can severely restrict the closed-loop performance and even destabilize the system. Networked control based on limited sensor and actuator information has therefore attracted much attention during the past decade. The literature particularly related to the results in this paper includes [5], [8], [9], [10], [11], [12], [13], [14]. Up till now, results on control with limited information have often been derived based on rather simple scenarios. In some early work, for instance, the initial state was the only unknown entity that the encoder needs to convey to the decoder. Generalizations to more complex situations, e.g., systems with process noise, measurement noise, and transmission errors, are challenging research topics. Some cases studied recently include systems with feedback over noisy channels, e.g., [8], [11], [12], [13], and systems corrupted by process noise, e.g., [10].

The main contribution of the present paper is a practical synthesis technique for joint optimization of the quantization and error protection for state observations over a bandlimited and noisy channel. We study a special case where we verify that certainty equivalence holds, and for this case we design joint source–channel encoder and decoder/estimator pairs¹. We assume that the probability density functions (pdfs) of the plant initialization and the process noise, and the transition

¹As in previous works on quantizer design for noisy channels, we use “encoder” as a generic term that describes combined source–channel encoding, i.e., joint quantization and error control. The “decoder” is the inverse mapping, and as we will see in Sec. V-A, decoding will specialize to state estimation based on the received symbols.

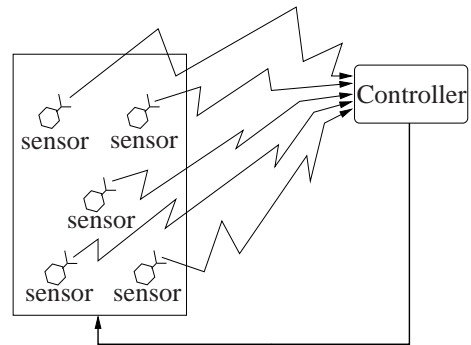


Fig. 1. Feedback control over a wireless sensor network. The approach developed in the paper can be applied to optimize the use of the wireless medium.

probability of the channel are known, or can be accurately estimated. We show that this problem is numerically solvable and leads to channel optimized encoder–decoder pairs that provide better control performance than existing solutions in the literature. The corresponding scenario for a plant without process noise was considered in our previous papers [2], [3]. The case with process noise is more realistic, but leads to a more complex optimization problem which depends on the pdf of the noise.

The paper is organized as follows. A motivating example is presented in Sec. II. Sec. III defines the control system with encoder, decoder, controller, and communication channel. The problem statement, which concerns a linear quadratic (LQ) objective over finite horizon, is presented in Sec. IV. The joint encoder–decoder design and the training procedure based on dynamic programming are described in Sec. V. In Sec. VI, a sequential Monte Carlo method is introduced to solve the nonlinear estimation problem. Sec. VII presents numerical results and discusses some implementation issues. Finally, the conclusions are given in Sec. VIII.

II. MOTIVATING EXAMPLE

Consider the wireless networked control system in Fig. 1. Suppose there is a large number of sensor nodes that can connect through a shared wireless medium to a control node. The sensors are spatially distributed over a large area and they measure the state of the control object, which is affected by rarely occurring local disturbances. The control commands for keeping the states around their equilibrium working points are executed through a common actuator, therefore all the sensors report their state measurements

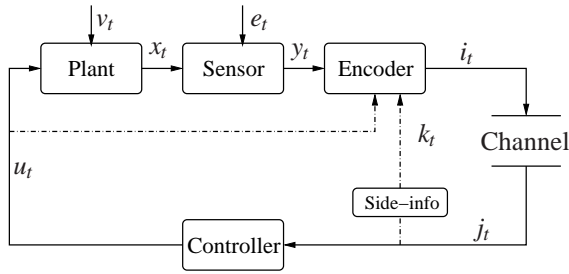


Fig. 2. System description.

to a common control node. The described system is quite representative for many emerging applications with control using wireless sensor networks, in areas such as industrial automation, environmental monitoring, surveillance etc.

It is desirable to efficiently utilize the communication resource, in order to allow for more sensors and tighter control. Therefore, consider the following decentralized control strategy. Let the control command corresponding to a specific sensor be zero as long as no disturbance is detected. When the sensor detects a disturbance, the sensor reading is encoded and transmitted to the controller node. The message is decoded at the controller node and a control command is derived and actuated to counteract the specific disturbance acting on the transmitting sensor. The controller and the encoder–decoder pair can be designed based on a finite-horizon LQ criterion. Assuming that the local disturbance is possible to detect and that an estimate of the disturbance magnitude pdf at each sensor location is known, we can optimize the encoder–decoder pair together with the control law. Such a design takes into account uncertainties due to initial state, process noise, and noisy channel. A class of time-varying non-uniform encoder–decoders are derived in the paper, which, together with a simple state feedback control, is suitable to handle the described scenario.

III. PRELIMINARIES

Consider the control system with a communication channel depicted in Fig. 2. Sensor data are encoded and transmitted over an unreliable communication channel. Control commands are then derived based on the received data. In this section, we describe this system in detail.

Let $\mathbf{x}_a^b = \{x_a, \dots, x_b\}$ denote the evolution of a discrete-time signal x_t from $t = a$ to $t = b$. The plant is given by the scalar system

$$\begin{aligned} x_{t+1} &= \theta_t(\mathbf{x}_{t-M_x}^t, u_t, v_t) \\ y_t &= h_t(x_t, e_t), \end{aligned} \quad (1)$$

where $x_t, u_t, y_t \in \mathbb{R}$ are the state, the control, and the measurement, respectively. The variable v_t is referred to as the process noise and e_t is the measurement noise. The system has a memory of order M_x .

Let \mathcal{E}_t denote the set of information available at the encoder at time t , i.e., the set of variables whose values are known to the encoder. In particular, we consider an encoder that *causally utilizes full information*, in the sense that $\mathcal{E}_t = \{\mathbf{y}_0^t, \mathbf{i}_0^{t-1}, \mathbf{k}_0^{t-1}, \mathbf{u}_0^{t-1}\}$, where i_t is the transmitted index and k_t represents the encoder’s knowledge about the

index j_t received by the decoder. (The mapping from j_t to k_t will be discussed below.) The *encoder* is a mapping from \mathcal{E}_t to a discrete set of symbols. We take each symbol to be represented by an integer index. At time t , the index is $i_t \in \mathcal{I}_L = \{0, 1, \dots, L-1\}$, where $L = 2^R$ with R denoting the *rate* of the transmission, in bits per state measurement. Hence, the encoder is described by a mapping

$$i_t = f_t(\mathbf{y}_0^t, \mathbf{i}_0^{t-1}, \mathbf{k}_0^{t-1}, \mathbf{u}_0^{t-1}). \quad (2)$$

Let the *discrete channel* have input variable i_t and output $j_t \in \mathcal{I}_L$, with one channel use defined by

$$j_t = \kappa_t(\mathbf{i}_{t-M_c}^t), \quad (3)$$

where $\kappa_t : \mathcal{I}_L^{M_c+1} \rightarrow \mathcal{I}_L$ is a random mapping, and $M_c \geq 0$ indicates (potential) channel memory.

At the receiver, we denote the information available at the controller by \mathcal{C}_t ; in particular, we consider a controller that causally utilizes full information, i.e., $\mathcal{C}_t = \{\mathbf{j}_0^t, \mathbf{u}_0^{t-1}\}$, see Fig. 2. The *controller* is then a mapping

$$u_t = \phi_t(\mathbf{j}_0^t, \mathbf{u}_0^{t-1}), \quad (4)$$

from \mathcal{C}_t to \mathbb{R} .

In some particular cases, the overall controller ϕ_t can be separated into two components, a decoder part (estimator) and a separate controller part. To model these situations, let $\overline{\mathcal{D}}_t = \{\mathbf{j}_0^t, \mathbf{u}_0^{t-1}\}$ denote the full information at the decoder part. The *decoder* is a mapping

$$d_t = g_t(\mathbf{j}_0^t, \mathbf{u}_0^{t-1}), \quad (5)$$

from $\overline{\mathcal{D}}_t$ to \mathbb{R} . Also, let $\overline{\mathcal{E}}_t = \{\mathbf{d}_0^t, \mathbf{u}_0^{t-1}\}$ denote the (full) information available at the controller part. Separate control based on the output from the decoder/estimator is then defined by the mapping

$$u_t = z_t(\mathbf{d}_0^t, \mathbf{u}_0^{t-1}), \quad (6)$$

from $\overline{\mathcal{E}}_t$ to \mathbb{R} .

IV. PROBLEM STATEMENT

In this section we describe the special case of the system defined above that is considered in the paper. The performance measure is detailed together with the encoder–decoder side information.

A. System Description and Performance Measure

Consider a stable scalar linear time-invariant plant controlled over a binary symmetric channel (BSC). The process noise v_t , which contributes to the uncertainty in system, is white noise with pdf p_{v_t} . We assume that the pdf of the initial state, p_{x_0} , is known, and that full state information is available at the encoder, i.e., $y_t = x_t$. The system is then given by

$$x_{t+1} = ax_t + u_t + v_t, \quad y_t = x_t, \quad |a| < 1. \quad (7)$$

Let $c(i_t) \in \{0, 1\}^R$ be a binary codeword of length R representing the encoder output $i_t \in \mathcal{I}_L$. The mapping between i_t and $c(i_t)$ is referred to as the *index assignment* [6]. For

example, the natural binary code assigns the natural binary representation of an integer i to the codeword c_i . At time t , the relation between the index i_t and the transmitted binary codeword $c(i_t)$ is

$$i_t = n \Leftrightarrow c(i_t) = c_n, \quad n \in \mathcal{J}_L.$$

In a similar way, $c(j_t)$ denotes the received binary codeword, where $j_t \in \mathcal{J}_L$ is the received index. The (memoryless) communication channel is a binary symmetric channel described by a channel transition probability $\varepsilon = p(0|1) = p(1|0)$. Assuming independent transmission of each binary bit, the conditional probability of a codeword $p(c_j|c_i)$ is

$$p(c_j|c_i) = (1 - \varepsilon)^{R - d_H(c_i, c_j)} \varepsilon^{d_H(c_i, c_j)}, \quad (8)$$

where $d_H(c_i, c_j)$ is the Hamming distance between c_i and c_j , i.e., number of bits which they differ.

Our goal is to solve an optimal encoder–decoder and control problem for the plant (7). The performance measure for this integrated communication and control problem is the linear quadratic cost function

$$J_T = E \left\{ x_T^2 + \sum_{t=0}^{T-1} x_t^2 + \rho u_t^2 \right\}, \quad (9)$$

where T is the horizon and $\rho \geq 0$ the control weight.

B. Encoder–Decoder Common Information

For memory based coding schemes, the system performance relies heavily on the encoder’s knowledge about the decoder memory state and vice versa. In the presence of a noisy channel, care has to be taken in specifying how to synchronize the states of the encoder and decoder.

We use the term *encoder side-information* to specify the encoder’s knowledge about the symbol j_t received by the decoder. Consequently, *no* encoder side-information is the extreme case when the encoder has no information at all about j_t , and *full* encoder side-information denotes the situation that the encoder, at each time instant t , knows exactly the previously received symbols j_0^{t-1} . This is the case when the channel is noiseless, so that $j_t = i_t$, or when there is an error-free side-information channel from the output of the forward channel to the encoder. Note that full side-information can also be obtained if the joint decoder–controller function is an invertible mapping, i.e., if j_t can be deduced from \mathbf{u}_0^t .

To model different degrees of side information, we define the *common information* shared by the encoder and decoder to be the value of the variable

$$k_t = \gamma_t(j_t) \in \mathcal{K}_K = \{0, 1, \dots, K-1\}, \quad K \leq L,$$

where $\gamma_t(j_t)$ is a mapping that specifies how k_t is obtained from the received variable j_t . In particular, $k_t = j_t$ and $K = L$ when full side-information is available, while $K = 1$ when there is no side-information at the encoder. Between the extremes, there are a variety of cases with incomplete side-information, for which $1 < K < L$. One example is the case with no side-information channel and a non-invertible decoder–controller mapping, e.g., if u_t takes on only $K < L$ distinct values.

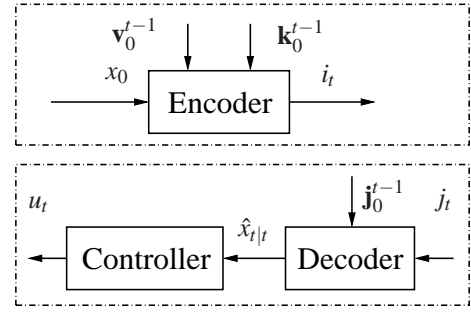


Fig. 3. The encoder and the decoder structures. The encoder is a mapping from $\{x_0, \mathbf{v}_0^{t-1}, \mathbf{k}_0^{t-1}\}$ to an integer symbol i_t , while the decoder maps $\{\mathbf{j}_0^{t-1}, \mathbf{u}_0^{t-1}\}$ to $\hat{x}_{t|t}$. Then the controller executes u_t based on $\hat{x}_{t|t}$.

We assume the general case, $1 \leq K \leq L$, in the paper. Hence we let \mathbf{k}_0^{t-1} specify the state of the encoder. Since this sequence is known also at the decoder, the encoder and the decoder are “synchronized” in this way. Note that in Fig. 2 we illustrate the mapping from j_t to k_t as an explicit side-information channel, even though this information can be obtained by other means, e.g., by inverting the decoder–controller mapping as previously discussed. Note also that the encoder has direct access to the controls u_t .

V. ENCODER–DECODER DESIGN

This section presents the main result of the paper. After discussing certainty equivalence and encoder–decoder mappings, we present the overall design algorithm. The encoder–decoder structure is shown in Fig. 3. The encoder–decoder design problem is on finding a strategy for the encoder to inform the decoder about x_0 and \mathbf{v}_0^{t-1} . Note that by knowing \mathbf{x}_0^t and \mathbf{u}_0^{t-1} , the encoder can deduce \mathbf{v}_0^{t-1} , based on (7). We can therefore assume that i_t is generated as

$$i_t = f_t(x_0, \mathbf{v}_0^{t-1}, \mathbf{k}_0^{t-1}), \quad t = 0, \dots, T, \quad (10)$$

based on x_0, \mathbf{v}_0^{t-1} and the side-information \mathbf{k}_0^{t-1} .

At the receiver, \mathbf{u}_0^t is computed from \mathbf{j}_0^t , hence in the following the assumed decoder–controller mapping will be $u_t = \phi_t(\mathbf{j}_0^t)$, and, if implemented separately, the decoder is a function $d_t = g_t(\mathbf{j}_0^t)$. The optimal state estimator based on the received indices is

$$\hat{x}_{s|t} = E\{x_s | \mathbf{j}_0^t\}, \quad s \leq t = 0, \dots, T-1. \quad (11)$$

A. Certainty Equivalence

Here we will show that under the assumptions of Sec. IV-B, the mapping ϕ_t can be split into a separate decoder g_t and controller z_t , i.e., *certainty equivalence* holds [4].

The necessary and sufficient condition for certainty equivalence presented in [4] corresponds in our case to the requirement that the average estimation error

$$E\{(x_t - \hat{x}_{t|t})^2\}, \quad t = 0, \dots, T \quad (12)$$

is not a function of \mathbf{u}_0^s , $s = 0, \dots, t-1$, for any sequence \mathbf{u}_0^t . This condition is satisfied in our scenario, a fact easily shown as follows. Note that

$$\hat{x}_{t|t} = E\{x_t | \mathbf{j}_0^t\} = q_{j_0}^t + \sum_{s=0}^{t-1} a^{t-1-s} u_s \quad (13)$$

with $q_{j_0^t} = E\{\bar{x}_t | \mathbf{j}_0^t\}$ where \bar{x}_t is defined as

$$\bar{x}_t = a^t x_0 + \sum_{s=0}^{t-1} a^{t-1-s} v_s. \quad (14)$$

Now, according to (10),

$$\begin{aligned} i_0 &= f_0(x_0) \\ i_1 &= f_1(x_0, v_0, k_0) \\ &\vdots \\ i_t &= f_t(x_0, \mathbf{v}_0^{t-1}, \mathbf{k}_0^{t-1}), \end{aligned}$$

with $k_t = \gamma_t(j_t)$, the variables \mathbf{i}_0^t , \mathbf{k}_0^t and \mathbf{j}_0^t depend only on x_0 , \mathbf{v}_0^{t-1} , γ_0^{t-1} and potential channel errors, but not on \mathbf{u}_0^s for any $s < t$. Hence, $x_t - \hat{x}_{t|t} = \bar{x}_t - q_{j_0^t}$ is not a function of \mathbf{u}_0^s . It follows that certainty equivalence control is optimal for fixed encoders $\{f_t\}_0^T$, with the given structure $f_t(x_0, \mathbf{v}_0^{t-1}, \mathbf{k}_0^{t-1})$. Consequently, the optimal control is given as

$$u_t = -\ell_t d_t, \quad d_t = \hat{x}_{t|t} \quad (15)$$

with

$$\ell_t = \frac{aP_{t+1}}{P_{t+1} + \rho}, \quad P_t = 1 + \frac{a^2 P_{t+1} \rho}{P_{t+1} + \rho} \quad (16)$$

for $t = 0, \dots, T-1$, and P_t initialized with $P_T = 1$. In the notation of Sec. III, we thus have that the optimal ϕ_t can be split into decoding (estimation) and control, according to

$$\phi_t(\mathbf{j}_0^t) = -\ell_t g_t(\mathbf{j}_0^t) = -\ell_t d_t = -\ell_t \hat{x}_{t|t}. \quad (17)$$

The proof of the above statement is completed in the appendix.

B. Design Algorithm

In this section we propose a training method to obtain the optimal encoder–decoder pair. It can be interpreted as a design suitable for short codewords, accomplishing source compression and channel protection simultaneously. The encoder–decoder mapping consists of functions with memory, whose role is to successively provide refinements of information about the initial state and the process noise. The proposed iterative approach starts with an initial set-up, and updates reconstruction points and encoding rules considering not only the past but also the predicted future state evolution. Similar to traditional quantizer design [6], the idea here is to fix the encoder and update the decoder, then fix the decoder and update the encoder etc.

For fixed encoders, the *optimal decoder* is

$$g_t(\mathbf{j}_0^t) = \hat{x}_{t|t} = E\{x_t | \mathbf{j}_0^t\} = q_{j_0^t} + \sum_{s=0}^{t-1} a^{t-1-s} u_s, \quad (18)$$

where $q_{j_0^t} = E\{\bar{x}_t | \mathbf{j}_0^t\}$ is the *reconstruction point* at time t , stored in a codebook at the decoder. The task for the decoder is thus to estimate \bar{x}_t based on \mathbf{j}_0^t . (It needs not to know x_0 and \mathbf{v}_0^{t-1} separately.)

The *optimal encoder* needs to take the impact of the predicted future state evolutions into account. Hence, for fixed decoders and controllers, and given the encoder mappings f_t

for $t = 0, \dots, t-1$ and $t+1, \dots, T-1$, we get the optimal encoder $f_t = f_t(x_0, \mathbf{v}_0^{t-1}, \mathbf{k}_0^{t-1})$ as

$$i_t = \arg \min_{i \in \mathcal{I}_L} E \left\{ \sum_{s=t}^T x_s^2 + \rho u_s^2 \middle| x_0, \mathbf{v}_0^{t-1}, \mathbf{k}_0^{t-1}, i_t = i \right\}, \quad (19)$$

which is closely related to \mathbf{k}_0^{t-1} , the available side-information. Also, observe that the channel transition probability $p(c_j | c_i)$, recall (8), is implicitly included in (19). The encoding rules are updated once the reconstruction points are recalculated. Therefore, both the encoder and decoder are specified by the set of reconstruction points $\{q_{j_0^t}\}$.

We summarize the previous discussion in the following design algorithm.

Encoder–Decoder Design Algorithm

- 1) Initialize the encoder–decoder mappings $\{f_t\}$ and $\{g_t\}$.
- 2) Compute the controller parameters $\{\ell_t\}$ using (16).
- 3) a) For each $t = 0, \dots, T-1$,
 - Update the encoder function f_t using (19).
 - Update the decoder function d_t and the control u_t using (15).
- b) If J_T has not converged, go to step 3a), otherwise stop.

Convergence is monitored by updating J_T in each step. Unfortunately, the design algorithm does not guarantee global optimality. The algorithm converges to a local minimum, which has shown to work well in practice.

VI. SEQUENTIAL MONTE CARLO SAMPLING

The encoder–decoder design presented in previous section is computationally demanding. Therefore we resort to a Monte Carlo approach to handle the resulting nonlinear filtering. The most computationally intensive part is the computation of

$$E\{\bar{x}_t | \mathbf{j}_0^t\} \quad \text{and} \quad E\{x_s^2 + \rho u_s^2 | x_0, \mathbf{v}_0^{t-1}, \mathbf{k}_0^{t-1}, i_t\}, \quad s > t.$$

When the encoder–decoder pairs are known, as well as the pdfs p_{x_0} and p_{v_t} , transition probabilities $p(c_j | c_i)$, and the side-information mapping γ_t , the conditional pdf $p(\bar{x}_t | \mathbf{j}_0^t)$, $t < T$, can be derived. Similarly, given the pdf of the current state, the pdfs of future estimates can also be derived. These estimation problems are solved through a sequential Monte Carlo approach.

To obtain $E\{\bar{x}_t | \mathbf{j}_0^t\}$, the following steps are performed. At each time t :

- 1) Generate a set of samples according to $p(\bar{x}_{t-1} | \mathbf{j}_0^{t-1})$.
- 2) Generate a set of process noise samples v_{t-1} according to $p(v_{t-1})$.
- 3) Compute \bar{x}_t based on (14) and the samples from steps 1 and 2.
- 4) Encode the samples in step 3 according to (19).
- 5) Simulate the transmission over the channel and create a sequence of symbols j_t .
- 6) Estimate $E\{\bar{x}_t | \mathbf{j}_0^t\}$.

Since the transition probability of a BSC has a closed-form expression, steps 4 and 5 can be replaced by

$$E\{\bar{x}_t | \mathbf{j}_0^{t-1}, j_t = n\} = \sum_{i_t} E\{\bar{x}_t | \mathbf{j}_0^{t-1}, i_t\} \frac{\Pr(j_t = n | i_t) P(i_t)}{\Pr(j_t = n)},$$

where $P(i_t)$ and $P(j_t)$ are estimated based on sampling.

VII. NUMERICAL RESULTS

In this section we present simulation results from applying the encoder–decoder design of Sec. V.

The random magnitudes of the initial state and the process noise are both modeled using the generalized Gaussian distribution (GGD) (e.g., [7]), since it provides a wide coverage of pdfs from narrow-tailed to broad tailed. We use the notation $GGD(\alpha, \beta)$, where α describes the exponential rate of decay, and β is the standard deviation. We assume for simplicity that full side-information is available at the encoder.

Fig. 4 illustrates simulation results for a numerical example with $a = 0.8$, $T = 3$, $R = 2$, $\rho = 1$, $p_{x_0} = GGD(2, 1)$ and $p_{v_t} = GGD(2, 0.25)$. In Fig. 4, we show \bar{J}_T as a function of the channel transition probability ε . This performance measure is computed by normalizing J_T in (9) with the cost obtained when no control action is taken, cf. the horizontal dashed line in Fig. 4. A performance comparison between the proposed coding scheme and a time-invariant uniform quantizer is depicted in Fig. 4. In particular, the time-invariant uniform quantizer is optimized by a grid search. At low ε , control using quantized feedback provides improvements more than 30%, compared to the case without feedback control. When ε increases, the proposed encoder–decoder outperforms the time-invariant uniform quantizer apparently. At the scenario of 35% bit errors, the improvement compared to using no control strategy, is around 16% for the proposed encoder–decoders, while only 3% for using uniform quantizers. Moreover, the figure demonstrates that there is still some advantage to feedback state observation on a noisy channel even with 35% bit errors.

Since the contribution $x_t^2 + \rho u_t^2$ to the total cost (9) typically decreases with time, the reconstruction points at $t = 0$, i.e., $\{q_{j_0=n}\}_{n=0}^{L-1}$, are important. Fig. 5 shows the resulting encoding boundaries and the reconstruction points at $t = 0$ versus ε . We can note that the number of reconstruction points chosen by the encoder decreases with increasing ε . This phenomenon is well known in quantization for noisy channels and is attributed to the varying abilities of binary codewords in combating channel errors. For very noisy channels, it is beneficial to transmit only the “stronger” codewords [6], providing true redundancy for error protection. However observe that the number of reconstruction points does not decrease with increasing ε . In Fig. 5, for $\varepsilon = 0.3$ and $\varepsilon = 0.35$, the two middle reconstruction points are close, so that additional zooming would be needed to distinguish them. Another impact of increasing ε is that the ranges of the reconstruction points decrease, so the encoding boundaries and the reconstruction points are all moved closer to zero, indicating that control magnitude is small.

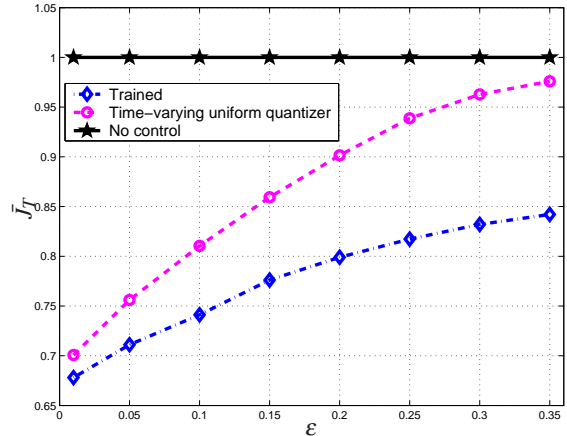


Fig. 4. A performance comparison between using the proposed encoder–decoder of this paper and using a uniform quantizer. Performance measure \bar{J}_T is shown as a function of channel transition probability ε , and it is obtained by normalizing J_T in (9) with the cost obtained with no control action. The horizontal line corresponds to the resulting cost when no control action ($\rho = \infty$) is taken.

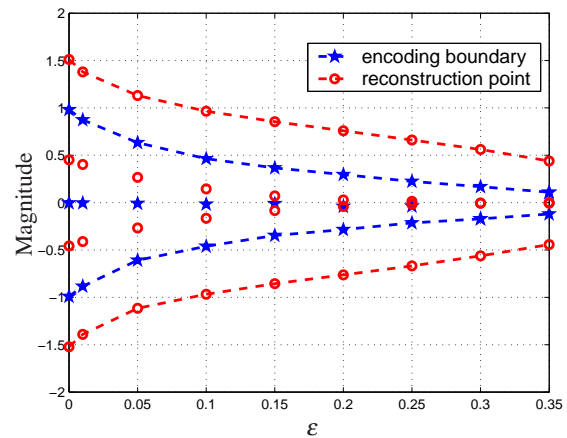


Fig. 5. The iteration results for the encoder–decoder pair at $t = 0$. On y-axis it shows the reconstruction points q_{j_0} and the encoding boundaries versus the channel transition probability ε . \star markers an encoding boundary, and o markers a reconstruction point.

The proposed iterative design requires an initial set-up of reconstruction points. Two initialization methods are compared in Fig. 6. The first method, referred to as **init_{uni}**, exploits properly scaled uniform quantizers, in which p_{x_0} and p_{v_t} have been given consideration, but not the channel transition probability ε . The second method, referred to as **init_{chop}**, applies channel optimized quantizers [6], designed for each pdf $p(\bar{x}_t | \mathbf{j}_0^t)$, $t = 0, \dots, T$. Note that the channel transition probability has been taken into account in this initialization. Fig. 6 illustrates that with iterative improvements, both initializations converge to quite similar final results, i.e., **trained_{uni}**, and **trained_{chop}**, although the performances of **init_{uni}** is notably worse than the performance of **init_{chop}**.

VIII. CONCLUSION

This paper has introduced an approach that jointly optimizes the encoder and decoder in closed-loop control of a linear plant with low-rate feedback over a binary symmetric channel. The variability of the plant is modeled as process noise. Having argued that a fixed sequence of

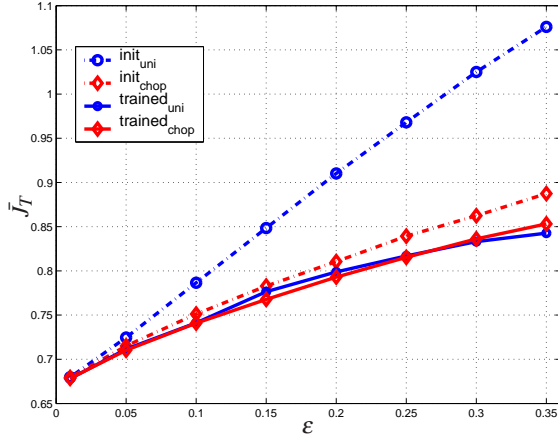


Fig. 6. Normalized performance measure \bar{J}_T as function of channel transition probability ϵ for encoder–decoder pairs obtained by using various initializations. **init_{uni}**: uniform initialization; **init_{chop}**: Channel Optimized initialization; **iterated_{uni}**: iteratively trained with **init_{uni}** as initialization; **iterated_{chop}**: iteratively trained with **init_{chop}** as initialization.

encoder mappings will result in the certainty equivalence condition being fulfilled, we fixed the receiver structure to implement separate decoding (estimation) and control. Our main contribution is to introduce optimized encoders given this structure on the estimation–control. We have performed various numerical investigations of the proposed optimization algorithm and illustrated quite promising results.

Even though our results were presented for a scalar system with full state observation, the overall problem was formulated under quite general assumptions, allowing for extensions, e.g., to systems of multiple dimensions and with partial observations. Another interesting extension of our work is to examine, the trade-off between the channel transition probability ϵ and how coarse the feedback information can be, given a certain control performance requirement.

REFERENCES

- [1] K. J. Åström. *Introduction to Stochastic Control Theory*. Academic Press, INC., 1970.
- [2] L. Bao, M. Skoglund, and K. H. Johansson. Encoder-decoder design for event-triggered feedback control. In *Proc. of 2006 American Control Conference*, pages 4183–4188, 2006.
- [3] L. Bao, M. Skoglund, and K. H. Johansson. Encoder-decoder design for feedback control over the binary symmetric channel. In *Proc. of 2006 IEEE International Symposium on Information Theory*, pages 2481–2485, 2006.
- [4] Y. Bar-Shalom and E. Tse. Dual effect, certainty equivalence, and separation in stochastic control. *IEEE Transactions on Automatic Control*, AC-19(5):494–500, Oct. 2004.
- [5] V. S. Borkar and S. K. Mitter. LQG control with communication constraints. In *Proc. of the 38th Conference on Decision and Control*, Dec. 1999.
- [6] N. Farvardin. A study of vector quantization for noisy channels. *IEEE Transactions on Information Theory*, 36(4):799–809, July 1990.
- [7] N. Farvardin and J. W. Modestino. Optimum quantizer performance for a class of non-gaussian memoryless sources. *IEEE Transactions on Information Theory*, 30(3):485–497, May 1984.
- [8] A. Mahajan and D. Teneketzis. Fixed delay optimal joint source–channel coding for finite-memory systems. In *Proc. of 2006 IEEE International Symposium on Information Theory*, pages 2319–2323, July 2006.

- [9] A. S. Matveev and A. V. Savkin. An analogue of shannon information theory for networked control systems. stabilization via a noisy discrete channel. In *Proc. of the 43rd IEEE Conference on Decision and Control*, volume 4, pages 14–17, Dec. 2004.
- [10] G. N. Nair and R. J. Evans. Stabilizability of stochastic linear systems with finite feedback data rates. *SIAM Jour. Contr. Opt.*, 43(2):413–436.
- [11] A. Sahai. The necessity and sufficiency of anytime capacity for control over a noisy communication link. In *Proc. of the 43rd IEEE Conference on Decision and Control*, pages 1896–1901, Dec. 2004.
- [12] T. Simsek, R. Jain, and P. Varaiya. Scalar estimation and control with noisy binary observation. *IEEE Transactions on Automatic Control*, 49(9):1598–1603, Sept. 2004.
- [13] S. Tatikonda, A. Sahai, and S. Mitter. Stochastic linear control over a communication channel. *IEEE Transactions on Automatic Control*, 49(9):1549–1561, Sept. 2004.
- [14] G. G. Walsh, O. Beldiman, and L. G. Bushnell. Error encoding algorithms for networked control systems. *Automatica*, pages 261–267, Sept. 2002.

APPENDIX

In this section, we will show that the certainty equivalence controller in (15) is optimal for the system (7), and the design criterion (9). As in [1], one can show that

$$\begin{aligned}
 x_T^2 &= x_0^2 P_0 + \sum_{t=0}^{T-1} (x_{t+1}^2 P_{t+1} - x_t^2 P_t) \\
 &= P_0 x_0^2 + \sum_{t=0}^{T-1} \left(2P_{t+1} (ax_t + u_t) v_t + P_{t+1} v_t^2 \right. \\
 &\quad \left. + (l_t^2 (P_{t+1} + \rho) - 1) x_t^2 + 2ax_t u_t P_{t+1} + u_t^2 P_{t+1} \right)
 \end{aligned}$$

Accordingly, the cost J_T can be written as

$$\begin{aligned}
 J_T &= E \left\{ x_T^2 + \sum_{t=0}^{T-1} (x_t^2 + \rho u_t^2) \right\} \\
 &= E \left\{ P_0 x_0^2 + \sum_{t=0}^{T-1} (2P_{t+1} (ax_t + u_t) v_t + P_{t+1} v_t^2 \right. \\
 &\quad \left. + (P_{t+1} + \rho) (x_t l_t + u_t)^2) \right\} \quad (20)
 \end{aligned}$$

Since v_t is white noise, and uncorrelated with x_t and u_t , the contribution from the first term within the sum in (20) is zero. The control u_t that minimizes J_T , is hence obtained as

$$\begin{aligned}
 &\min_{u_t} \left(E \{ (\rho + P_{t+1}) (u_t + l_t x_t)^2 \} \right) \\
 &= E_{\mathbf{j}_0^t} \left(\min_{u_t} \left(E \{ (\rho + P_{t+1}) (u_t + l_t x_t)^2 | \mathbf{j}_0^t \} \right) \right) \\
 &= E_{\mathbf{j}_0^t} \left(\min_{u_t} \left(E \{ (\rho + P_{t+1}) (u_t + l_t \hat{x}_{t|t})^2 | \mathbf{j}_0^t \} \right. \right. \\
 &\quad \left. \left. + (P_{t+1} + \rho) l_t^2 E \{ (\hat{x}_{t|t} - x_t)^2 | \mathbf{j}_0^t \} \right) \right), \quad (21)
 \end{aligned}$$

where $E_{\mathbf{j}_0^t}$ denotes expectation over \mathbf{j}_0^t . By assuming the encoder has the structure in (10), the received symbol j_t depends on x_0 , \mathbf{v}_0^{t-1} , the potential channel errors and side-information \mathbf{k}_0^{t-1} , but not on the control signals \mathbf{u}_0^{t-1} . This implies that the last term in (21) is not a function of the control signals. Hence, optimal control is obtained by minimizing the first term in (21), which gives $u_t = -l_t \hat{x}_{t|t}$.