OPTIMIZED FILTERING AND RECONSTRUCTION IN PREDICTIVE QUANTIZATION WITH LOSSES

Alyson K. Fletcher,[1] Sundeep Rangan,[2] Vivek K Goyal,[3] and Kannan Ramchandran[1]

[1]University of California, Berkeley
[2]Flarion Technologies
[3]Massachusetts Institute of Technology

alyson@eecs.berkeley.edu, s.rangan@flarion.com, vgoyal@mit.edu, kannanr@eecs.berkeley.edu

ABSTRACT

Consider a communication system in which a filtered and quantized signal is sent over a channel with erasures and (potentially) additive noise. Linear MMSE estimation is achieved in such a system by Kalman filtering. Allowing any Markov erasure process and any Markov-state jump linear signal generation model, it is shown that the estimation performance at the receiver can be computed as a deterministic optimization with linear matrix inequality (LMI) constraints rather than a pseudorandom simulation. Furthermore, in contrast to the case without erasures, the filtering in the transmitter should not necessarily be MMSE prediction (whitening); a procedure is given to find a locally optimal prefilter. The main tools are recent LMI characterizations of asymptotic state estimation error covariance and output estimation error variance for discrete-time jump linear systems in which the discrete portion of the system state is a Markov chain. As another application of this framework, a novel analysis and optimization of a "streaming" version of multiple description coding based on subsampling is outlined.

1. INTRODUCTION

Prediction is fundamental to source coding, and in particular, predictive quantization is one of the simplest ways to exploit the memory of a random process or real-world source in lossy source coding. Predictive quantization is used in almost all practical speech coding [1], and in most current video coding via motion compensation.[1] However, prediction is problematic when there are uncorrectable losses between the encoder and the decoder. In this case, the decoder is forced to "predict" based on data it simply does not have. There are many techniques for concealing the errors; see, e.g., [3] for a survey of techniques used for video.

If losses are very likely, then it may be advantageous to not do predictive coding, but rather to allow the decoder to exploit the memory of the source in joint source–channel decoding. In fact, the difficulty caused by losses in predictive quantization is an example of the general principle that compression increases sensitivity to channel errors.[2] With existing tools, it is difficult is analyze the overall performance of predictive quantization with losses, i.e., to quantify the effect on mean-squared error (MSE) of the losses. In particular, note that the optimal estimates at the decoder are given by a Kalman filter. Thus, it is difficult to assess whether the prediction is desirable, let alone to improve upon the standard encoding.

In the first stage of this work, we show how to compute—via a deterministic optimization with linear matrix inequality (LMI) constraints, rather than a pseudorandom simulation—the overall performance of a predictive quantization system with losses. This computation relies on a framework introduced in [5]. The signal to be compressed is modeled as an output of a jump linear system driven by a white noise input. Losses of encoded prediction errors can be modeled by any Markov chain, and in fact any linear degradation can also be included. Thus, this approach has considerable generality. Two weaknesses should be noted, however: a) we must adopt the technically incorrect (but common and useful) stochastic model for quantization error whereby the error is independent of the input; and b) we assume a linear reconstruction at the decoder even though this may not be MSE minimizing for non-Gaussian signals.

The second stage of this work provides a methodology for optimizing the "prediction filter" for overall performance. The quotes are to emphasize that the solution may not be an optimal prediction filter in the conventional sense because it does not minimize the variance of the prediction error sequence; rather, the optimizing filter is the key component of a type of joint source–channel code that intentionally leaves some redundancy in the quantized sequence.

The loss model that we consider allows the possibility of a set of quantized samples being either all received or all lost, as when they are packetized together. We demonstrate the types of systems that can be analyzed and optimized by considering a multiple description coding system that puts odd- and even-numbered samples in separate streams of packets as in [6, 7]. Notably, the analysis is not for the case that *all* of one packet stream or the other is received. Such an analysis would be difficult without the framework presented here.

1.1. Example

For illustration, consider the encoding of a Gaussian first-order autoregressive process $\{s_k\}$ where

$$s_k = \alpha s_{k-1} + n_k$$

with $\alpha \in [0, 1)$ and the n_ks are i.i.d. $\mathcal{N}(0, 1)$ random variables. (The n_ks are "hidden," observed only through the s_ks.) Note that the power of the $\{s_k\}$ sequence is given by $(1 - \alpha^2)^{-1}$. Roughly speaking, one would rather send the lower-power sequence $\{n_k\}$

[1]For a concise yet detailed history of predictive quantization, see [2].
[2]See, for example, [4] for a good exposition and a memorable title.

than the higher-power sequence $\{s_k\}$. Thus rather than sending quantized samples $s_k^{\mathrm{Q}} = Q(s_k)$, one could send a quantized sequence $n_k^{\mathrm{Q}} = Q(n_k) = Q(s_k - \alpha s_{k-1})$. The decoder could then form the estimate $\hat{s}_k = \alpha \hat{s}_{k-1} + n_k^{\mathrm{Q}}$. A flaw in this approach is that quantization errors accumulate. Thus, in practice one forms a prediction from quantized data and quantizes the prediction error:

$$q_k = Q(s_k - \beta \hat{s}_{k-1}) \tag{1}$$
$$\hat{s}_k = \beta \hat{s}_{k-1} + q_k \tag{2}$$

This is commonly referred to as "putting the quantizer in the loop." An intuitive choice for the prediction gain β is $\beta = \alpha$.

Losses of quantized prediction errors q_k complicate the decoding (or estimation) process. If q_k is missing for some k, then motivated by $E[q_k] = 0$ (under reasonable assumptions) one could simply set $q_k = 0$ in (2). The approach we take here is to model the quantization error $v_k = (s_k - \beta \hat{s}_{k-1}) - q_k$ as a stochastic process with mean zero and known variance (determined by the coarseness of the quantizer Q) that is independent of $\{n_k\}$. Then the optimal linear estimate of $\{s_k\}$ is computed by a Kalman filter. Regardless of the estimation technique, the effect of a loss persists in $\hat{s}_{k+1}, \hat{s}_{k+2}, \ldots$, with a decay determined by β. Note in particular that there is no propagation of error if $\beta = 0$.

For this toy problem with a simple source and simple loss process, it is intuitive that lowering the prediction gain β increases the robustness to loss. Thus, the optimal β must be a decreasing function of the loss probability p. However, there is no evident way to find optimal βs without resorting to pseudorandom simulations. The computation methodology presented here—which applies much more generally—gives the performance as a function of β and hence allows overall system optimization. In the general case, the performance for any given β is given by the result of an efficient numerical optimization involving linear matrix inequalities (LMIs).

2. STATE-SPACE MODELING OF PREDICTIVE QUANTIZATION

2.1. Predictive Quantization Model

We consider the standard predictive quantization system in Figure 1(a). The signal to be quantized is denoted $s(k)$. In predictive quantization, the signal $s(k)$ is typically slowly varying and is quantized to a sequence of quantized prediction errors $q(k)$. The prediction errors are the differences, $s(k) - \widehat{s}(k)$, where $\widehat{s}(k)$ is a *prediction* of $s(k)$ based on past quantized values $q(k)$. The predicted values $\widehat{s}(k)$ are generated by a prediction filter. The idea of predictive quantization is to exploit the slowly-varying nature of $s(k)$ to "subtract out" past information through the term $\widehat{s}(k)$ and reduce the quantizer input variance.

In this paper, we study the predictive quantization in the presence of losses. Specifically, the quantized prediction errors are sent on a lossy channel, and a decoder attempts to estimate $s(k)$ from whatever is received.

2.2. State-Space Modeling

To analyze the predictive quantizer, we model the system as a linear state-space system. To this end, we assume that the signal $s(k)$ is a finite-order wide-sense stationary random process. Any such

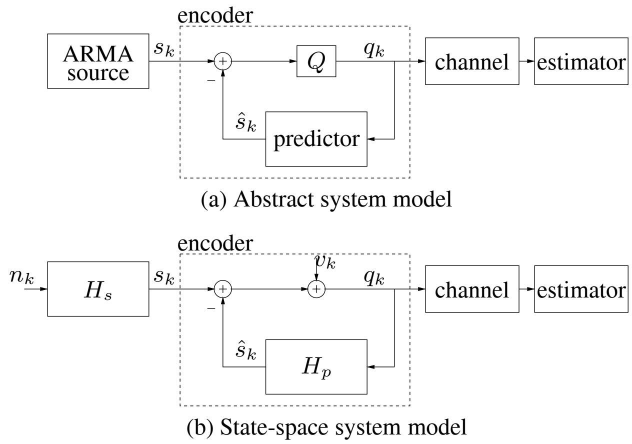

Fig. 1. Schematic representations of communication system with predictive quantizer.

signal can be modeled as the output of a state-space system,

$$H_s : \quad \begin{aligned} x_s(k+1) &= A_s x_s(k) + B_s n(k) \\ s(k) &= C_s x_s(k) + D_s n(k), \end{aligned} \tag{3}$$

where $x_s(k)$ is a model state and $n(k)$ is unit-variance white noise. The state-space matrices (A_s, B_s, C_s, D_s) depend on the statistics of $s(k)$. The abstract state space signal model is shown in Figure 1(b) as H_s.

The quantizer Q in Figure 1(a) is modeled as a linear gain with additive noise,

$$Q : \quad \begin{aligned} q(k) &= Q(s(k) - \widehat{s}(k)) \\ &= \rho(s(k) - \widehat{s}(k)) + \sigma v(k), \end{aligned} \tag{4}$$

where $v(k)$ represents the quantization error with a scaling factor σ. The parameter ρ is the quantizer gain and depends, among other things, on the number of quantizer bits. Although technically incorrect, we will make the common and useful assumption that the quantization errors $v(k)$ can be treated a white noise sequence independent of the quantizer input [8]. We will assume $v(k)$ has unit variance, so that σ^2 is the quantization variance.

Given the state-space model for $s(k)$ and additive white noise model for the quantizer, the optimal linear predictor has a well-known and standard state-space form,

$$H_p : \quad \begin{aligned} x_p(k+1) &= A_s x_p(k) + L q(k) \\ \widehat{s}(k) &= C_s x_p(k), \end{aligned} \tag{5}$$

where $q(k)$ is the quantized prediction error and L is a *prediction filter gain matrix*. The gain matrix L will be the key parameter in the predictive quantizer design and will be discussed in detail below.

2.3. Channel Losses and Decoding

Having described the predictive quantizer encoder, let us now consider the transmission and decoding/estimation. Referring back to Figure 1, the quantized prediction errors $q(k)$ are sent over a channel to the decoder. The decoder must estimate the original signal $s(k)$ from whatever it receives.

We will model the channel as a simple erasure channel: each sample $q(k)$ is either received without errors or completely lost. Let $\widehat{q}(k)$ denote the channel output,

$$\widehat{q}(k) = \begin{cases} q(k) & \text{when sample } k \text{ is not lost;} \\ 0 & \text{when sample } k \text{ is lost.} \end{cases}$$

In addition to $\widehat{q}(k)$, we will assume that the decoder knows which samples are lost. This assumption is natural, for example, with either numbered packets or limited jitter.

In the absence of losses, the decoder receives $\widehat{q}(k) = q(k)$. In this case, the standard predictive quantizer decoder simply reruns the prediction filter (5), and thereby reconstructs the linear predictor state $x_p(k)$ and predictor output $\widehat{s}(k)$. The predictor output $\widehat{s}(k)$ is used as the decoded value for the original signal $s(k)$. This reconstruction technique is usually optimal, since the prediction filter output $\widehat{s}(k)$ is usually designed to be the best estimate of $s(k)$ given the quantized values $q(k)$.

To deal with losses of samples $q(k)$ at the decoder, one simple method is to still use the prediction filter, but with $\widehat{q}(k)$ as the input, instead of $q(k)$. Using $\widehat{q}(k)$ inserts zeros for the unknown samples, which is reasonable since the prediction errors are typically zero mean. Based on the prediction filter (5), the resulting decoder can be described by the state-space system

$$\begin{aligned} x_d(k+1) &= A_s x_d(k) + L\widehat{q}(k) \\ \widehat{s}_d(k) &= C_s x_d(k), \end{aligned} \tag{6}$$

where $x_d(k)$ is the decoder state and $\widehat{s}_d(k)$ is the decoder's reconstruction of the original signal $s(k)$. Comparing (6) with (5), observe that when $q(k) = \widehat{q}(k)$ (i.e. there are no losses), $x_d(k) = x_p(k)$ and $\widehat{s}_d(k) = \widehat{s}(k)$.

2.4. Markov Erasure Model

The performance of the decoder depends, of course, on the channel losses. In this paper, we model the losses as the result of a finite, hidden Markov chain. Specifically, we will assume that there is some hidden Markov state $\theta(k)$ which can take on one of M values, $\theta(k) \in \{1, \ldots, M\}$. The sample $q(k)$ is lost when $\theta(k)$ falls into some subset of states $I_{\text{loss}} \subseteq \{1, \ldots, M\}$. That is, using the notation above,

$$\widehat{q}(k) = \begin{cases} q(k) & \text{when } \theta(k) \notin I_{\text{loss}}; \\ 0 & \text{when } \theta(k) \in I_{\text{loss}}. \end{cases}$$

Let p_{ij} denote the transition probabilities

$$p_{ij} = \mathbf{Pr}(\theta(k{+}1) = j \mid \theta(k) = i).$$

We will assume that the chain is aperiodic and irreducible, so that it admits a unique stationary distribution

$$q_i = \mathbf{Pr}(\theta(k) = i).$$

The hidden Markov chain model is extremely general and can incorporate a wide range of loss processes. Simple examples include independent losses, Gilbert-Elliot losses, and fixed-length burst losses.

3. ANALYSIS AND OPTIMIZATION

3.1. Distortion Estimation

The key performance measure for any quantization system is the distortion as a function of the bit rate. One of the appealing features of the state-space modeling described above is that the distortion can be estimated through a standard convex programming method known as *linear matrix inequalities (LMIs)*.

As a measure of the distortion, we consider the *asymptotic mean square error (MSE)* between the original signal $s(k)$ and the decoder estimate $\widehat{s}_d(k)$,

$$\mathbf{D} := \lim_{k\to\infty} \mathbf{E}|s(k) - \widehat{s}_d(k)|^2. \tag{7}$$

The limit here is used to remove transients effects of initial conditions in either the encoder or decoder.

To estimate the distortion, define the error signals

$$\begin{aligned} e_x(k) &= [(x(k) - x_p(k))' \ \ (x(k) - x_d(k))']' \\ e_s(k) &= s(k) - \widehat{s}_d(k) \end{aligned}$$

and define the noise vector $w(k) = [n(k) \ v(k)]'$. Combining equations (3), (4), (5) and (6), $e_x(k)$ and $e_s(k)$ can be written as the state and output of a time-varying system,

$$\begin{aligned} e_x(k+1) &= A(k)e_x(k) + B(k)w(k) \\ e_s(k) &= Ce_x(k) + Dw(k), \end{aligned} \tag{8}$$

where $C = [C_s \ 0]$, $D = [D_s \ 0]$, and

$$\begin{aligned} A(k) &= A_{\text{no loss}} = \begin{bmatrix} A_s - \rho LC & 0 \\ -\rho LC & A_s \end{bmatrix} \\ B(k) &= B_{\text{no loss}} = \begin{bmatrix} B_s - \rho LD & \sigma L \\ B_s - \rho LD & \sigma L \end{bmatrix}, \end{aligned}$$

when there is no sample loss and

$$\begin{aligned} A(k) &= A_{\text{loss}} = \begin{bmatrix} A_s - \rho LC & 0 \\ 0 & A_s \end{bmatrix} \\ B(k) &= B_{\text{loss}} = \begin{bmatrix} B_s - \rho LD & \sigma L \\ B_s & 0 \end{bmatrix}, \end{aligned}$$

when there is a sample loss.

Using the Markov model for the losses described earlier, for each state $i \in \{1, 2, \ldots, M\}$, let $A_i = A_{\text{loss}}$ when $i \in I_{\text{loss}}$ and $A_i = A_{\text{no loss}}$ for $i \notin I_{\text{loss}}$. Define B_i similarly. Also, let $C_i = C$ and $D_i = D$ for all i. With these definitions, we can rewrite (8) as

$$\begin{aligned} e_x(k+1) &= A_{\theta(k)}e_x(k) + B_{\theta(k)}w(k) \\ e_s(k) &= C_{\theta(k)}e_x(k) + D_{\theta(k)}w(k). \end{aligned} \tag{9}$$

The system (9) is an example of a *jump linear system*, namely, a time-varying system whose matrices are determined by a Markov parameter. Jump linear systems have been extensively studied. In particular, since the distortion $\mathbf{D}$ in (7) is simply the mean square output of the jump linear system (9), the distortion can be computed as described in the following theorem.

Theorem 1 ([9]) *Consider the predictive quantizer encoder and decoder and channel loss model described in Section 2. The distortion* $\mathbf{D}$ *in (7) is given by*

$$\mathbf{D} = \min \sum_{i=1}^{M} q_i \operatorname{tr}(B_i' P_i B_i + D_i' D_i)$$

where the minimization is over matrices $P_i \geq 0$, $i = 1, \dots, M$ *satisfying the* coupled Lyapunov equations

$$P_i \geq A_i' \overline{P}_i A_i + C_i' C_i,$$

and $\overline{P}_j = \sum_i p_{ij} P_i$.

The minimization in Theorem 1 is an example of a *linear matrix inequality* (LMI) optimization. The decision variables are matrices, P_i, and, similar to linear programming, appear linearly in the constraints and objective functions. However, unlike linear programming, the constraints are matrix-valued. Nevertheless, LMI optimization is convex and can be performed easily even with a large number of variables [10].

3.2. Gain Matrix Optimization

The previous section described how, given a predictive quantizer system and loss model, we can estimate the distortion. Ultimately, we wish to optimize the predictive quantizer to minimize this distortion. In the predictive encoder/decoder system of Section 2, this amounts to selecting the optimal gain matrix L.

In the standard lossless predictive encoder, the gain matrix L is chosen to minimize the prediction error power $\mathbf{E}[|s(k) - \widehat{s}(k)|^2]$. As discussed in the introduction, this maximally exploits the redundancy of the source, but also makes the quantized values least robust to losses. In the presence of losses, we wish to optimally tradeoff prediction gain and robustness.

Using Theorem 1, the optimal gain can be found by minimizing the distortion $\mathbf{D}$, where the minimization is taken over the matrices P_i *and* the gain matrix L. The gain matrix appears implicitly through the matrices A_i and B_i. Unfortunately, this joint minimization is non-convex. However, as discussed earlier, for a fixed L the optimal P_is can be found as an LMI. Similarly, it is easy to show that for matrices P_is, the optimal L can also be found by an LMI. Such optimization problems are often called *bilinear matrix inequalities* or BLMIs. While the optimal solution for a BLMI is, in general, difficult, there is a natural iterative algorithm for finding suboptimal solutions: alternatively minimize over the matrices P_i and the gain matrix L, performing each minimization as an LMI. The alternating procedure will converge to a local minimum, and if it is started sufficiently close to the global minimum, it will converge to the global minimum as well.

4. EXTENSION TO MULTIPLE DESCRIPTION CODING

One interesting extension of the communication model shown in Figure 1 is for (potentially overlapping) subsets of q_ks to be assigned to separate "descriptions" of the source, thus making the predictive quantizer a multiple description (MD) encoder [11]. The MD coding paradigm applies to both point-to-point and point-to-multipoint communication.

In the former case, the analysis and optimization framework of this paper applies in a straightforward manner when the performance criterion is the overall MSE at the receiver, averaging over the reception patterns for the packets of each description. The latter case is more interesting, as losses within descriptions imply that each receiver sees a different sequence of C matrices. As long as the loss processes for the packets of each description have the Markov property, the results of this paper give an MSE performance for each receiver. The *vector* of MSEs representing the steady-state performances of the receivers could then be the subject of optimization of the encoder filters.

5. CONCLUDING COMMENTS

This paper introduces a framework for optimization of predictive quantization for Markov channels that may erase or linearly degrade quantized prediction errors. The practical application of these ideas would be advanced by the invention of an on-line adaptive version. This seems entirely feasible, as the contribution here allows concrete numerical computations given a wide variety of signal and channel models.

6. REFERENCES

[1] R. V. Cox, "Speech coding," in *The Digital Signal Processing Handbook*, chapter 45, pp. 45.1–45.19. CRC and IEEE Press, 1998.

[2] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inform. Th.*, vol. 44, no. 6, pp. 2325–2383, Oct. 1998.

[3] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[4] G. Buch, F. Burkert, J. Hagenauer, and B. Kukla, "To compress or not to compress?," in *IEEE Globecom*, London, Nov. 1996, pp. 196–203.

[5] A. K. Fletcher, S. Rangan, and V. K Goyal, "Estimation from lossy sensor data: Jump linear modeling and Kalman filtering," in *Information Proc. in Sensor Netw.*, Berkeley, CA, Apr. 2004, pp. 251–258.

[6] N. S. Jayant, "Subsampling of a DPCM speech channel to provide two "self-contained" half-rate channels," *Bell Syst. Tech. J.*, vol. 60, no. 4, pp. 501–509, Apr. 1981.

[7] N. S. Jayant and S. W. Christensen, "Effects of packet losses in waveform coded speech and improvements due to an odd–even sample-interpolation procedure," *IEEE Trans. Comm.*, vol. COM-29, no. 2, pp. 101–109, Feb. 1981.

[8] R. M. Gray, "Quantization noise spectra," *IEEE Trans. Inform. Th.*, vol. 36, no. 6, pp. 1220–1244, Nov. 1990.

[9] O. L. V. Costa and R. P. Marques, "Robust H_2-control for discrete-time Markovian jump linear systems," *Int. J. Contr.*, vol. 73, no. 1, pp. 11–21, 2000.

[10] S. P. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia, PA, 1994.

[11] V. K Goyal, "Multiple description coding: Compression meets the network," *IEEE Sig. Proc. Mag.*, vol. 18, no. 5, pp. 74–93, Sept. 2001.