DISTRIBUTED SCALAR QUANTIZERS FOR NOISY CHANNELS

Johannes Karlsson, Niklas Wernersson, and Mikael Skoglund

School of Electrical Engineering, Royal Institute of Technology (KTH), SE-100 44 Stockholm, Sweden

ABSTRACT

Sensor nodes in wireless sensor networks should preferable be both cheap and energy efficient. To cope with these requirements an algorithm for designing distributed scalar quantizers optimized for noisy channels is proposed and evaluated. Applying the algorithm results in locally optimal systems. It is demonstrated that the correlation between the sources can be used to reduce the quantization distortion when the channel is close to error-free. If, on the other hand, there are a lot of disturbances on the channel the correlation can be used to introduce protection against channel errors.

Index Terms— Multisensor systems, distributed estimation, quantization, joint source–channel coding

1. INTRODUCTION

Wireless sensor networks (WSNs) are expected to play an important role in tomorrow's sensing and actuating systems. One important property of sensor networks is that there may be a high correlation between different sensor measurements due to high spatial density of sensor nodes. This justifies source coding of correlated sources, which has been analyzed in for instance [1] where the Slepian–Wolf theorem is stated. A practical solution for Slepian–Wolf coding termed DISCUS was presented in [2], allowing the use of powerful codes such as LDPC and Turbo codes in the context of distributed source coding, see e.g. [3, 4]. These schemes can be extended to the case of continuous sources, hence lossy coding, by adding a quantizer before the Slepian–Wolf encoder. Even though both LDPC and turbo codes have relatively low encoding complexity, these schemes, together with the quantizer, will require some data processing in the sensor nodes. This will therefore counteract one of the desired design criteria in sensor network design, namely low cost and energy efficient sensor nodes.

An alternative is to design sensor nodes of very low complexity. In this paper, this is accomplished by seeing the distributed source coding as a quantization problem. Similar approaches can be found in [5, 6], but whereas they consider the CEO problem this paper is focused on estimating the values of each of the individual sources. Furthermore, [5, 6] assume perfect communication between the source encoders and the decoder. Our work is targeted towards wireless sensor networks, therefore a non-ideal channel is introduced between the source encoders and the decoder to make the system more realistic. The related problem of distributed detection over non-ideal channels has previously been studied in [7]. In what follows, we propose a design algorithm that results in sensor nodes operating on a sample by sample basis in a similar fashion as the channel optimized scalar quantizer (COSQ) [8].

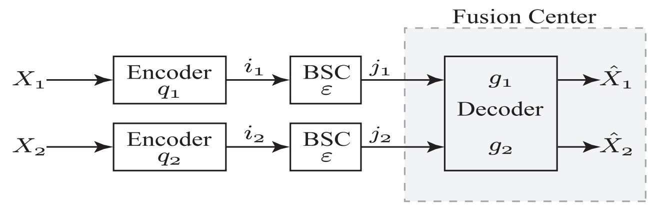


Fig. 1: System with two signals X_1 and X_2 that are encoded separately but decoded jointly at a fusion center.

2. PROBLEM FORMULATION

The system that will be studied can be seen in Figure 1. Two scalar signals X_1 and X_2 are encoded separately by the encoders q_1 and q_2. Each encoder outputs a codeword i_1, or i_2, which is a vector of R bits. The bits representing the codewords are transmitted over two binary symmetric channels (BSC) with equal crossover probability ε. The decoder at the fusion center outputs estimates $\hat{X}_1$ and $\hat{X}_2$ based on the received codewords j_1 and j_2. The data is modeled as

$$X_1 = Y + N_1 \tag{1}$$
$$X_2 = Y + N_2 \tag{2}$$

where Y, N_1 and N_2 are three independent zero-mean Gaussian distributed random variables with variances σ_Y^2, $\sigma_{N_1}^2$ and $\sigma_{N_2}^2$, respectively. We will further make the assumption $\sigma_{N_1}^2 = \sigma_{N_2}^2 = \sigma_N^2$. For measuring the correlation between X_1 and X_2, the correlation-SNR is defined as

$$C_{SNR} = 10 \log_{10} \left(\frac{\sigma_Y^2}{\sigma_N^2} \right). \tag{3}$$

Hence, $C_{SNR} = -\infty$ dB means that X_1 and X_2 are uncorrelated and $C_{SNR} = \infty$ dB means that they are fully correlated.

Given this system, the objective is to find the optimal pair of R bit scalar quantizers, q_1 and q_2, and the corresponding joint decoders, g_1 and g_2. As optimization criterion the sum of the mean squared error (MSE) for the two signals is used

$$D = D_1 + D_2 = E[(X_1 - \hat{X}_1)^2] + E[(X_2 - \hat{X}_2)^2]. \quad (4)$$

3. ANALYSIS

We will only consider the design of q_1 and g_1 since the problem is symmetric. Let $N = 2^R$ define the number of codewords available for each encoder. The encoder q_1 is a mapping from $x_1 \in \mathbb{R}$ to a codeword $i_1 \in \{1, 2, \ldots, N\}$. To define this mapping we let $\mathcal{A}_{i_1}$ be a set containing all points x_1 that are mapped to codeword i_1. Hence,

$$x_1 \in \mathcal{A}_{i_1} \Rightarrow q_1(x_1) = i_1 \quad (5)$$

where $i_1 \in \{1, 2, \ldots, N\}$. Similarly, the decoder g_1 is a mapping from a pair of received codewords (j_1, j_2) described as

$$\hat{x}_1 = g_1(j_1, j_2) = r_{j_1 j_2} \quad (6)$$

where $(j_1, j_2) \in \{1, 2, \ldots, N\}^2$ and $\mathcal{R} = \{r_{11}, r_{12}, \ldots, r_{NN}\}$ is the set of possible reconstruction points. With these definitions the distortion D_1 can be written as

$$D_1 = \int_{x_1} f_{X_1}(x_1) \sum_{j_1=1}^{N} P(j_1 \,|\, q_1(x_1)) \sum_{i_2=1}^{N} P(i_2 \,|\, x_1) \sum_{j_2=1}^{N} P(j_2 \,|\, i_2)(x_1 - g_1(j_1, j_2))^2 \, \mathrm{d}x_1 \quad (7)$$

and D_2 can be expressed as

$$D_2 = \int_{x_1} f_{X_1}(x_1) \sum_{j_1=1}^{N} P(j_1 \,|\, q_1(x_1)) \int_{x_2} f_{X_2}(x_2 \,|\, x_1) \sum_{j_2=1}^{N} P(j_2 \,|\, q_2(x_2))(x_2 - g_2(j_1, j_2))^2 \, \mathrm{d}x_2 \, \mathrm{d}x_1 \quad (8)$$

where $f_{X_1}(x_1)$ and $f_{X_2}(x_2 \,|\, x_1)$ are pdf:s. $P(j_1 \,|\, q_1(x_1))$ is the probability of receiving j_1 given that $q_1(x_1)$ is sent and $P(i_2 \,|\, x_1)$ is the probability that x_2 is encoded to i_2 given x_1.

As in traditional Lloyd-Max training [9] we will optimize the system in an iterative fashion. The optimal encoder q_1 will depend not only on the decoder g_1 but also on the encoder q_2 and the decoder g_2. Similarly the optimal decoder g_1 will depend on both of the encoders q_1 and q_2. Because of these interdependencies they will be updated in the following order: q_1, g_1 and g_2, q_2 and finally g_1 and g_2 again.

3.1. Finding the Optimal Encoder

We want to design the optimal encoder q_1 in the minimum MSE (MMSE) sense, given a fixed encoder q_2 and fixed decoders g_1 and g_2. Since $f_{X_1}(x_1)$ is non-negative, to minimize (7) and (8) it is sufficient to minimize the MSE for each value of x_1. Hence, the objective is to, for each x_1, find i_1 to minimize

$$\begin{aligned} D(x_1, i_1) &= D_1(x_1, i_1) + D_2(x_1, i_1) \\ &= E[(x_1 - \hat{X}_1)^2 \,|\, x_1, i_1] + E[(X_2 - \hat{X}_2)^2 \,|\, x_1, i_1] \end{aligned} \quad (9)$$

where

$$D_1(x_1, i_1) = \sum_{j_1=1}^{N} P(j_1 \,|\, i_1) \sum_{i_2=1}^{N} P(i_2 \,|\, x_1) \sum_{j_2=1}^{N} P(j_2 \,|\, i_2)(x_1 - g_1(j_1, j_2))^2 \quad (10)$$

$$D_2(x_1, i_1) = \sum_{j_1=1}^{N} P(j_1 \,|\, i_1) \int_{x_2} f_{X_2}(x_2 \,|\, x_1) \sum_{j_2=1}^{N} P(j_2 \,|\, q_2(x_2))(x_2 - g_2(j_1, j_2))^2 \, \mathrm{d}x_2. \quad (11)$$

The set $\mathcal{A}_{i_1}$ can now be defined as

$$\mathcal{A}_{i_1} = \{x_1 : D(x_1, i_1) \leq D(x_1, \tilde{i}_1), \forall \tilde{i}_1 \neq i_1\}. \quad (12)$$

In [8] a similar analysis is made in the case of a single source. In that case the inequality corresponding to (12) was shown to be linear in x_1. This implied that the sets corresponding to each encoder region must be intervals and analytical expressions for finding the endpoints of these intervals were derived. However, (12) is not linear in x_1 and $\mathcal{A}_{i_1}$ *will in general not be an interval*, as illustrated in Section 4. Therefore, the set $\mathcal{A}_{i_1}$ is designed according to (12) by numerically evaluating the distortion when a large range of values of x_1 are encoded to each of the available codewords.

3.2. Finding the Optimal Decoder

Finding the optimal decoder g_1 is equivalent to computing the best reconstruction points $\mathcal{R} = \{r_{11}, r_{12}, \ldots, r_{NN}\}$. Keeping the encoders fixed it is a well known fact that the optimal (MMSE) reconstruction point of x_1 is given as

$$r_{j_1 j_2} = E[x_1 \,|\, j_1, j_2]. \quad (13)$$

3.3. Design Algorithm

To find encoders and decoders for a given error probability, ε, and a given C_{SNR} the following algorithm is proposed

1. Choose q_1 and q_2 to be two known initial encoders and compute the optimal decoders g_1 and g_2.
2. Set the iteration index $k = 0$ and $D^{(0)} = \infty$.
3. Set $k = k + 1$.

4. Find the optimal encoder q_1 by using (12).
5. Find the optimal decoders g_1 and g_2 by using (13).
6. Find the optimal encoder q_2 by using (12) (with x_1 replaced with x_2, i_1 replaced with i_2, etc.).
7. Find the optimal decoders g_1 and g_2 by using (13).
8. Evaluate the distortion $D^{(k)}$ for the system. If the relative improvement of $D^{(k)}$ compared to $D^{(k-1)}$ is less than some threshold $\delta > 0$ stop the iteration. Otherwise go to Step 3.

Each iteration will decrease the distortion. As in the case of the Lloyd-Max algorithm the training will result in a locally optimal system, and not necessarily the global optimum.

3.3.1. *Avoiding Poor Local Optima*

In [8] it was found that a good locally optimal system could be achieved by designing the encoder/decoder pair for a range of values of ε, $0 \leq \varepsilon \leq \varepsilon_{max}$. This is done by, in steps of $\Delta\varepsilon$, first stepping from 0 to ε_{max} and then back from ε_{max} to 0. In each step the system is initialized with the system from the previous value of ε. The new system obtained from the training algorithm is kept if it results in a lower distortion than the previous system for this ε. The process of stepping back and forth between 0 and ε_{max} is repeated until no further improvements are made. The reason why this method improves the systems is because it introduces a way for the algorithm to break free from poor local optima.

The idea of stepping back and forth is incorporated in our design algorithm in the following way. Decide the range of values of ε and C_{SNR} for which to design the system, for example $\varepsilon = 0, \ldots, \varepsilon_{max}$ and $C_{SNR} = -\infty, \ldots, \infty$ dB. For each value of C_{SNR} step in the ε-direction, that is start at $\varepsilon = 0$ and step to ε_{max} and then back to $\varepsilon = 0$. In each point, keep the system that results in the lowest distortion for that point and use this system as initialization for the next value of ε. After the stepping in the ε-direction is done, for each value of ε, step in the C_{SNR}-direction, that is from $C_{SNR} = \infty$ dB to $C_{SNR} = -\infty$ dB and then back to $C_{SNR} = \infty$ dB. As before, for each point, keep the system that results in the lowest distortion and use this system as initialization for the next value of C_{SNR}. This process of stepping back and forth was repeated two times with an additional stepping in the ε-direction in the end. Some of the benefits of stepping back and forth are that poor locally optimum systems are removed to some extent and also that it reduces the importance of the choice of initialization encoders.

4. NUMERICAL RESULTS

Systems with $R = 2, 3, 4$ bits have been designed and tested for $C_{SNR} = -\infty, 10, 20, 30, \infty$ dB and $\varepsilon \in [0, 0.1]$. The performance of the systems is evaluated using the signal-to-distortion ratio (SDR). Since there are two channels to take into consideration, the SDR is defined as

$$\mathrm{SDR} = 10\log_{10}\left(\frac{E[X_1^2] + E[X_2^2]}{E[(X_1 - \hat{X}_1)^2] + E[(X_2 - \hat{X}_2)^2]}\right). \tag{14}$$

As initial encoders we have used two traditional single source Lloyd-Max optimized scalar quantizers with the folded binary code [10] as initial codeword assignment.

In Figure 2 results are presented for the case where $R = 3$. When $C_{SNR} = -\infty$ dB the problem is reduced to two independent COSQs each using 3 bits. When $C_{SNR} = \infty$ dB, ideally, the systems should have the same performance as a single COSQ with twice as many bits since $X_1 = X_2 = Y$. Indeed, for the 2 bit systems (results not included) this is the case for all values of ε. However, when $R = 3$ bits this bound is only reached for $\varepsilon \geq 0.001$ and for $R = 4$ bits this bound is reached only for $\varepsilon \geq 0.003$.

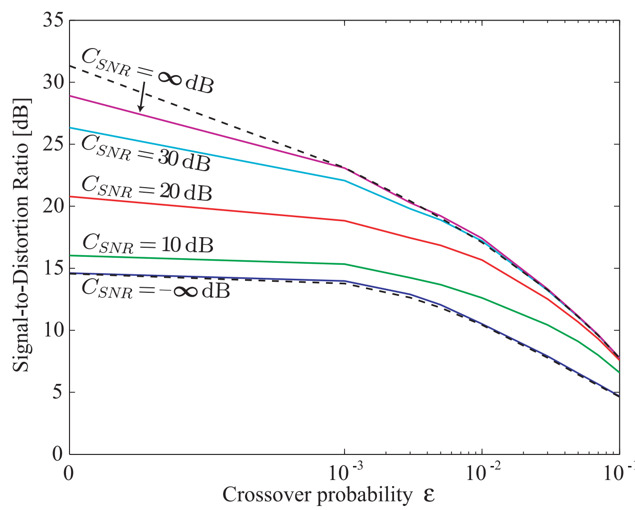
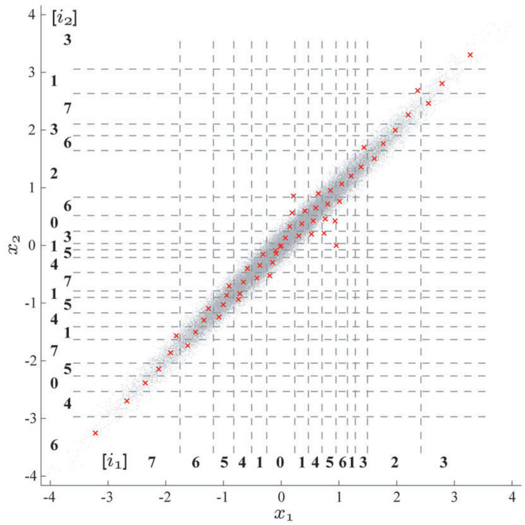
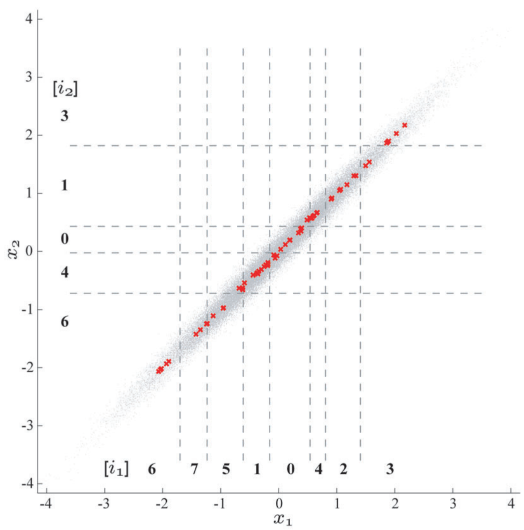


Fig. 2: Graph showing the performance of a 3 bit system in comparison to the COSQ (dashed) with 3 and 6 bits.

Let us take a deeper look at the 3 bit system for sources with $C_{SNR} = 20$ dB to see how the tradeoff between quantization distortion and robustness against channel errors works in practice. For $\varepsilon = 0$ the SDR is about 6 dB higher than for the COSQ [8]. To understand this we have to look at the structure of the two encoders q_1 and q_2. As can be seen in Figure 3(a) where $\varepsilon = 0$, some of the codewords are used for more than one quantization region, for example the codeword $i_2 = 5$ is used for 3 separated intervals such that $\mathcal{A}_{i_2=5} \approx (-2.3, -2.0] \cup (-1.2, -0.9] \cup (-0.2, -0.1]$. With information from only one of the channels it would not be possible to distinguish between which of these different intervals x_2 must have belonged to. However, with help from i_1 this can be done because $i_1 = 7$ is highly likely when $x_2 \in (-2.3, -2.0]$, $i_1 \in \{4, 5, 6\}$ is highly likely when $x_2 \in (-1.2, -0.9]$ and so on. Hence, i_1 will indicate which of the separated intervals x_2 belongs to. In this way the distributed coding is used to de-



(a) $\varepsilon = 0$



(b) $\varepsilon = 0.04$

Fig. 3: Encoder structures for 3 bit systems with $C_{SNR} = 20$ dB and $\varepsilon = 0$ in (a) and $\varepsilon = 0.04$ in (b). The small dots in the background show a sample distribution of (X_1, X_2), the dashed lines show the boundaries for the quantization regions and the small crosses mark reconstruction points, $(\hat{X}_1, \hat{X}_2)$.

crease the quantization distortion. Something to note is that the sets of separated intervals are created by the design algorithm despite the fact that the initial encoders are Lloyd-Max quantizers where all sets, $\mathcal{A}_{i_1}$ and $\mathcal{A}_{i_2}$, are single intervals.

When the same system is designed for $\varepsilon = 0.04$ the resulting encoders get the structure shown in Figure 3(b). In this case not even all available codewords are used by the encoders. Instead some codewords that would make the system more sensitive to channel errors are removed by the design algorithm. The removed codewords are those that are never optimal according to (12).

5. CONCLUSIONS

A design algorithm for a joint source-channel code that works on a sample by sample basis for correlated sources is presented. The resulting encoders use the same codeword for several separated intervals so that the quantization distortion is reduced when there is a high correlation between the sources and the channel is close to error free. When there are more disturbances on the channel the correlation is instead used for protection against channel errors.

Depending on the correlation between the sources, different gains are achieved in comparison to the case where the two sources are encoded and decoded separately using channel optimized scalar quantization (COSQ) [8]. For the case of $C_{SNR} = 10$ dB and a binary symmetric channel with crossover probability $\varepsilon = 0.03$ the gain is as much as 2.5 dB in comparison to the single source COSQ.

References

[1] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. on Information Theory*, vol. IT-19, no. 4, pp. 471–480, July 1973.

[2] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Trans. on Information Theory*, vol. 49, no. 3, pp. 626–643, March 2003.

[3] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Communications Letters*, vol. 6, no. 10, pp. 440–442, October 2002.

[4] J. Garcia-Frias and Y. Zhao, "Compression of correlated binary sources using turbo codes," *IEEE Commnications Letters*, vol. 5, no. 10, pp. 417–419, October 2001.

[5] T. J. Flynn and R. M. Gray, "Encoding of correlated observations," *IEEE Trans. on Information Theory*, vol. 33, no. 6, pp. 773–787, November 1987.

[6] W.M. Lam and A. R. Reibman, "Design of quantizers for decentralized estimation systems," *IEEE Trans. on Communications*, vol. 41, no. 11, pp. 1602–1605, November 1993.

[7] B. Liu and B. Chen, "Channel-optimized quantizers for decentralized detection in sensor networks," *IEEE Trans. on Information Theory*, vol. 52, no. 7, pp. 3349–3358, July 2006.

[8] N. Farvardin and V. Vaishampayan, "Optimal quantizer design for noisy channels: an approach to combined source-channel coding," *IEEE Trans. on Information Theory*, vol. 33, no. 6, pp. 827–838, November 1987.

[9] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Press, Dordrecht, The Netherlands, 1992.

[10] A. Mehes and K. Zeger, "Binary lattice vector quantization with linear block codes and affine index assignments," *IEEE Trans. on Information Theory*, vol. 44, no. 1, pp. 79–94, January 1998.