Escola de Enxeñaría de Telecomunicación

Multimedia Communications

Distributed Source Coding Using Syndromes (DISCUS)

Author:

Blanco Caamaño, Ramón

July 9, 2018

Contents

1 Introduction

In this report, we consider the related problem of compressing a source which is correlated with another source that is available only at the decoder. This problem has been studied in the information theory literature under the name of the Slepian–Wolf source coding problem for the lossless coding case, and as "rate-distortion with side-information" for the lossy coding case.

In this work, a practical implementation in Matlab is provided based on algebraic trellis codes dubbed as Distributed Source Coding Using Syndromes (DISCUS), that can be applicable in a variety of settings. Construction detail are shown in paper *Distributed Source Coding Using Syndromes (DISCUS): Design and Construction* (see References), in particular, the section III.C.3 (using the trellis described in figure 4) that is what concerns us in this report.

Finally, main results are shown concluding with the study and comparison of the following proposed scenarios:

- Case 1: Coding and decoding of X when Y is known at reception.
- Case 2: Coding and decoding of X when Y is independent.
- Case 3: Coding and decoding of X when Y is known at both sides.
- Case 4: Estimation of X exclusively based on Y.

1.1 Architecture

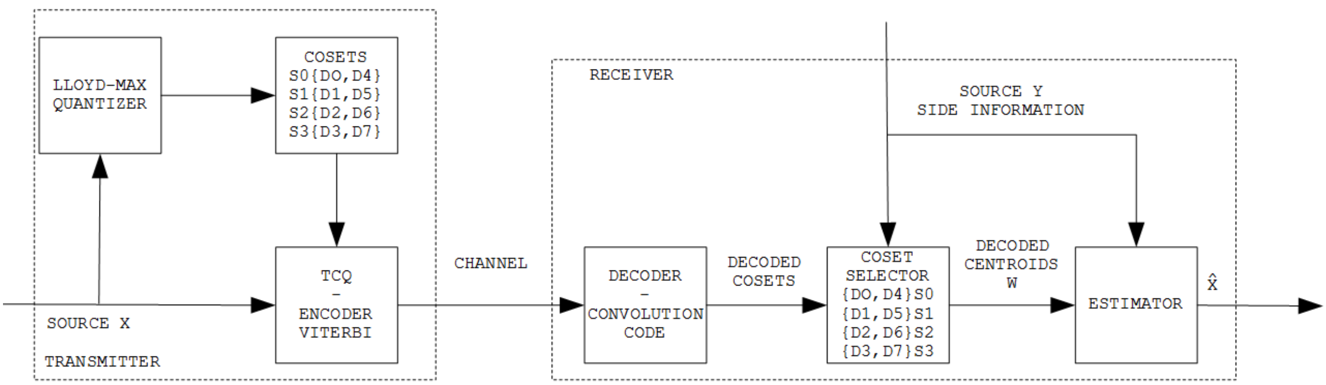

Figure 1: General scheme of the system architecture.

1.2 Trellis-coded quantization for source coding

Trellis-coded quantization is a particular kind of vector quantization of the dimension of the sequence length. It aims at reaching the rate-distortion bound for vector quantizer by partitioning a multidimensional grid into cosets by means of a convolucional code. The latter is chosen so that the coset lattices show good euclidean distance properties. The quantizing algorithm applies the Viterbi algorithm on the trellis of a convolutional code to find the best sequence of bits minimizing the quantization distortion.

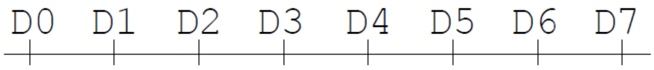

Figure 2: Lattice

Let $\mathbf{x} \in R^N$ a vector to be quantized. Each component x_i, is partitioned in n codebooks of 2^n codewords. These codebooks are grouped in 4 cosets $S_0\{D_0 \cup D_4\}$, $S_1\{D_1 \cup D_5\}$, $S_2\{D_2 \cup D_6\}$ and $S_3\{D_3 \cup D_7\}$, only one of which can be used for quantization at a given instant i. A total of n bits per sample is transmitted. For each sample, one bit is used to select a codebook inside the allowed coset whereas the $n-1$ other bits are used to index a codeword within this codebook. The codebook are chosen so the each coset is still a good quantizer for the source.

A trellis branch is assigned a metric which correspond to the distortion introduced by quantizing on the close codeword in the codebook corresponding to the branch. The path with the minimum total distortion is found by applying the Viterbi algorithm. The convolutional code input sequence which generates this minimum distortion path determines the sequence of codebooks.
In the dequantizer, the bits defining the minimum distortion path are encode with the convolutional code to recover the sequence of codebooks. The sample is reconstructed by indexing the codebook with the $n-1$ bits for each instant i.

In table [1], Trellis-coded quantization performance for the memoryless Gaussian source is shown. In order to verify the implementation, obtained results must match those shown in Table IV of the paper *Trellis coded quantization of memoryless and Gauss—Markov Sources* (see References). As can be seen, variance quantization is close to the results presented in paper. Note: assume that $\sigma_x^2 = 1$ in computer simulations.

Tabla 1: Trellis Coded Quantization Performance for the memoryless Guassian source using rate(R+1) Lloyd-Max ouput points. (Values are listed as SNR in dB)

	IMPLEMENTATION		TCQ PAPER	
RATE	1 bit/sample	2 bits/sample	1 bit/sample	2 bit/sample
σ_q^2	4.66	10.19	4.64	10.19

1.3 Mapping of symbols to transitions

For each transition, a metric is assigned to each branch of the trellis. As shown in the previous section, this metric correspond to the distortion introduce by quantizing on the close codeword in the codebook corresponding to the branch. The arrangement of the codebooks in the branches of the trellis guarantee the maximum Euclidean distance between them. In other words, the selected cosets are complementary to each other and the forming lattices minimizes the possibility of obtaining an incorrect metric.
In essence, the arrangement of the cosets in both Trellis are the same, they only differ in the order used in each state. For example, the cosets assigned to branches of state 0 in figure 4.A are S_0 and S_2, while in figure 4.B are S_1 and S_3.

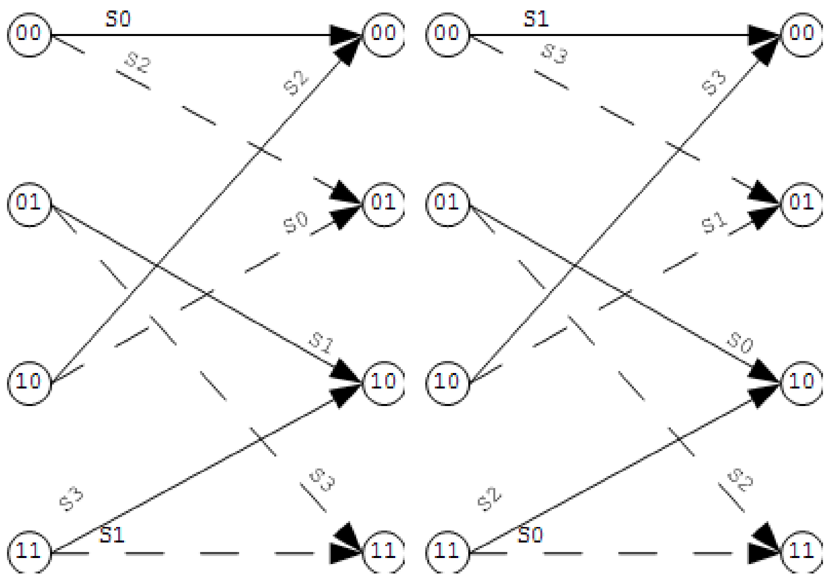

Figure 3: Trellis diagrams proposed in paper, figure 4.A and 4.B respectively.

Tabla 2: Distance between centroids/cosets of Figure 4.A (paper).

STATES	COSET	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
00	S_0	X				X			
	S_2			X				X	
01	S_1		X				X		
	S_3				X				X
10	S_2			X				X	
	S_0	X				X			
11	S_3				X				X
	S_1		X				X		

Tabla 3: Distance between centroids/cosets of Figure 4.B (paper).

STATES	COSET	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
00	S_1		X				X		
	S_3				X				X
01	S_0	X				X			
	S_2			X				X	
10	S_3				X				X
	S_1		X				X		
11	S_2			X				X	
	S_0	X				X			

2 Coding and decoding of X when Y is known at reception

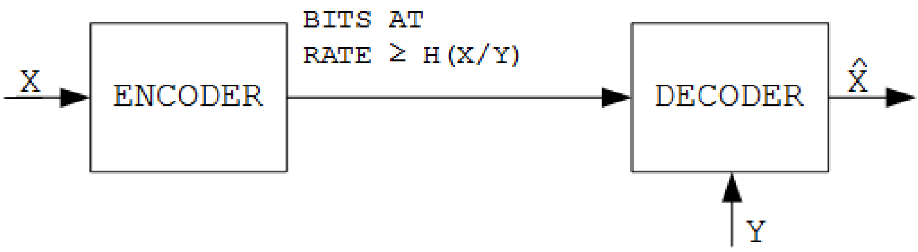

Figure 4: Defined scenario.

Transmission.

- Source: X.
- Lloyd-Max quantizer, 3 bits.
- Quantization noise: $Q_X = X - W_X$, where W_X quantized sample of X.
- Encoder: $f(X)$, where f is the encoding function.
- Rate: 1 bit/sample.

Reception.

- Decoded sample: $W = index(g(f(X)), Y)$, where g is the decoding function and Y indexes the centroids within the selected cosets.
- Side-information: Y, where $Y = X + N$.
- Estimator: $\widehat{X} = Y \frac{\sigma_{qx}^2}{\sigma_{qx}^2 + \sigma_n^2} + W \frac{\sigma_n^2}{\sigma_{qx}^2 + \sigma_n^2}$.

2.1 Description

At transmission side, TCQ uses 2 bits/sample to quantify Gaussian source X. Quantization levels are obtained from Lloyd-Max algorithm using 3 bits/sample. This alphabet is partitioned into 4 coset as follows:

- Rate 1 bit/sample: $S_0\{D_0,D_4\}$, $S_1\{D_1,D_5\}$, $S_2\{D_2,D_5\}$ and $S_3\{D_3,D_7\}$.

Where S and D are cosets and centroids(quantization levels) respectively. Then, encoder generates the best sequence of bits minimizing the quantization distortion, which specifies the cosets to quantify each sample of X.

At reception side, a convolutional code recovers the sequence of cosets chosen as best path of the Trellis diagram. The sample W is reconstructed by indexing the coset for each instant i with a bit extra provided by Gaussian side-information Y. This scheme reduce 1 bit/sample in transmission.

Finally, a MSE estimator is used to obtain $\widehat{X}$ estimated from decoded sample W and side-information Y.

2.2 Estimator

Let Y and W_X, such that $Y = X + N$ where N is White Gaussian Noise and $W_X = X - Q_X$ where Q_X is quantization noise, being X and N Gaussian samples i.i.d. and also Q_X and W_X. A linear estimator is used in order to obtain $\widehat{X}$ estimator.

$$\widehat{X} = \alpha Y + \beta W. \tag{1}$$

2.2.1 Maths

Minimizing mean squared error, α and β values are obtained, consequently the mathematical definition of $\widehat{X}$ estimator.

$$\min_{\alpha,\beta} E\{(X - (\alpha Y + \beta W))^2\} \tag{2}$$

Doing maths, results shown in the paper are obtained.

$$\widehat{X} = Y \frac{\sigma_{q_x}^2}{\sigma_{q_x}^2 + \sigma_n^2} + W \frac{\sigma_n^2}{\sigma_{q_x}^2 + \sigma_n^2} \tag{3}$$

2.2.2 Estimator response versus SNR values

If $Y = X + N$, it is assumed that for:

- Low SNR values: $Y \approx N \rightarrow \sigma_x^2 \ll \sigma_n^2$.
- High SNR values: $Y \approx X \rightarrow \sigma_x^2 \gg \sigma_n^2$

Approximations of the estimator versus SNR values.

- Low SNR values.

$$\lim_{\sigma_n^2 \to +\infty} \widehat{X} = \lim_{\sigma_n^2 \to +\infty} Y \frac{\sigma_{q_x}^2}{\sigma_{q_x}^2 + \sigma_n^2} + \lim_{\sigma_n^2 \to +\infty} W \frac{\sigma_n^2}{\sigma_{q_x}^2 + \sigma_n^2} \rightarrow \lim_{\sigma_n^2 \to +\infty} \widehat{X} = W \tag{4}$$

- High SNR values.

$$\lim_{\sigma_n^2 \to 0} \widehat{X} = \lim_{\sigma_n^2 \to 0} Y \frac{\sigma_{q_x}^2}{\sigma_{q_x}^2 + \sigma_n^2} + \lim_{\sigma_n^2 \to 0} W \frac{\sigma_n^2}{\sigma_{q_x}^2 + \sigma_n^2} \rightarrow \lim_{\sigma_n^2 \to 0} \widehat{X} = Y \tag{5}$$

Distortion of the estimator versus SNR values. Distortion is measured as the average of the squares of the errors(MSE) between X and $\widehat{X}$. $D = E\{(X - \widehat{X})^2\}$, where D is the measure of distortion obtained.

- Low SNR values:

$$D_{LOW} = E\{(X - \widehat{X})^2\} \approx E\{(X - W)^2\} = E\{Q^2\} \rightarrow D_{LOW} = \sigma_q^2 \tag{6}$$
$$\tag{7}$$

- Medium SNR values: Assume that side-information Y is able to index the cosets, therefore $W \approx W_X$.

$$\begin{aligned} D_{MEDIUM} &= E\{(X - \widehat{X})^2\} = E\{(X - (Y \tfrac{\sigma_{q_x}^2}{\sigma_{q_x}^2 + \sigma_n^2} + W \tfrac{\sigma_n^2}{\sigma_{q_x}^2 + \sigma_n^2}))^2 && (8) \\ &\approx E\{(X - (Y \tfrac{\sigma_{q_x}^2}{\sigma_{q_x}^2 + \sigma_n^2} + W_X \tfrac{\sigma_n^2}{\sigma_{q_x}^2 + \sigma_n^2}))^2\} && (9) \end{aligned}$$

Replace $c_1 = \frac{\sigma_{q_x}^2}{\sigma_{q_x}^2 + \sigma_n^2}$ and $c_2 = \frac{\sigma_n^2}{\sigma_{q_x}^2 + \sigma_n^2}$.

$$\begin{aligned} D_{MEDIUM} &= E\{(X - (Yc_1 + W_X c_2))^2\} = E\{c_1^2 Y^2 + 2c_1 c_2 Y W_X - 2c_1 XY + c_2^2 W_X^2 && (10) \\ &-2c_2 X W_X + X^2\} = c_1^2 E\{Y^2\} + 2c_1 c_2 E\{Y W_X\} - 2c_1 E\{XY\} && (11) \\ &+c_2^2 E\{W_X^2\} - 2c_2 E\{X W_X\} + E\{X^2\} && (12) \\ & && (13) \end{aligned}$$

Note: $E\{XY\} = \sigma_x^2$, $E\{XW_X\} = \sigma_{w_x}^2$, $E\{Y^2\} = \sigma_y^2$, $E\{YW_X\} = \sigma_{w_x}^2$, $E\{W_X^2\} = \sigma_{w_x}^2$ and $E\{X^2\} = \sigma_x^2$.

Expected values:

- $E\{XY\} = E\{X(X+N)\} = \{X^2+XN\} = E\{X^2\}+E\{XN\} = \sigma_x^2+E\{X\}E\{N\} = \sigma_x^2$
- $E\{XW_X\} = E\{(Q_X + W_X)W_X\} = E\{Q_XW_X + W_X^2\} = E\{Q_XW_X\} + E\{W_X^2\} = E\{Q_X\}E\{W_X\} + \sigma_{w_x}^2 = \sigma_{w_x}^2$
- $E\{NW_X\} = E\{N(X-Q_X)\} = E\{NX-NQ_X\} = E\{NX\}-E\{NQ_X\} = E\{N\}E\{X\}-E\{N\}E\{Q_X\} = 0$
- $E\{YW_X\} = E\{(X + N)W_X\} = E\{XW_X + NW_X\} = E\{XW_X\} + E\{NW_X\} = E\{XW_X\} + E\{NW_X\} = \sigma_{w_x}^2$

$$D_{MEDIUM} = c_1^2\sigma_y^2 + 2c_1c_2\sigma_{w_x}^2 - 2c_1\sigma_x^2 + c_2^2\sigma_{w_x}^2 - 2c_2\sigma_{w_x}^2 + \sigma_x^2 \tag{14}$$

Solve constants c_1 and c_2.

$$D_{MEDIUM} = \frac{\sigma_y^2\sigma_{q_x}^4}{(\sigma_{q_x}^2+\sigma_n^2)^2} + \frac{2\sigma_{w_x}^2\sigma_n^2\sigma_{q_x}^2}{(\sigma_{q_x}^2+\sigma_n^2)^2} - \frac{2\sigma_x^2\sigma_{q_x}^2}{\sigma_{q_x}^2+\sigma_n^2} + \frac{\sigma_{w_x}^2\sigma_n^4}{(\sigma_{q_x}^2+\sigma_n^2)^2} - \frac{2\sigma_{w_x}^2\sigma_n^2}{\sigma_{q_x}^2+\sigma_n^2} + \sigma_x^2 \tag{15}$$

$$= \frac{1}{(\sigma_{q_x}^2+\sigma_n^2)^2}(\sigma_y^2\sigma_{q_x}^4 + 2\sigma_{w_x}^2\sigma_n^2\sigma_{q_x}^2 + \sigma_{w_x}^2\sigma_n^4 + \sigma_x^2\sigma_n^4 + 2\sigma_x^2\sigma_n^2\sigma_{q_x}^2 + \sigma_x^2\sigma_{q_x}^4 \tag{16}$$

$$-2\sigma_x^2\sigma_{q_x}^4 - 2\sigma_x^2\sigma_{q_x}^2\sigma_n^2 - 2\sigma_{w_x}^2\sigma_n^2\sigma_{q_x}^2 - 2\sigma_{w_x}^2\sigma_n^4) \tag{17}$$

$$= \frac{1}{(\sigma_{q_x}^2+\sigma_n^2)^2}(\sigma_y^2\sigma_{q_x}^4 + \sigma_x^2\sigma_n^4 - \sigma_x^2\sigma_{q_x}^4 - \sigma_{w_x}^2\sigma_n^4) \tag{18}$$

$$= \frac{1}{(\sigma_{q_x}^2+\sigma_n^2)^2}(\sigma_{q_x}^4(\sigma_y^2 - \sigma_x^2) + \sigma_n^4(\sigma_x^2 - \sigma_{w_x}^2)) = \frac{1}{(\sigma_{q_x}^2+\sigma_n^2)^2}(\sigma_{q_x}^4\sigma_n^2 + \sigma_n^4\sigma_{q_x}^2) \tag{19}$$

$$= \frac{\sigma_{q_x}^2\sigma_n^2}{(\sigma_{q_x}^2+\sigma_n^2)^2}(\sigma_{q_x}^2 + \sigma_n^2) \rightarrow D_{MEDIUM} = \frac{\sigma_{q_x}^2\sigma_n^2}{\sigma_{q_x}^2+\sigma_n^2} \tag{20}$$

- High SNR values:

$$D_{HIGH} = E\{(X - \widehat{X})^2\} \approx E\{(X - Y)^2\} = E\{(X - X)^2\} \rightarrow D_{HIGH} = 0 \tag{21}$$

$$\tag{22}$$

2.3 Simulation results

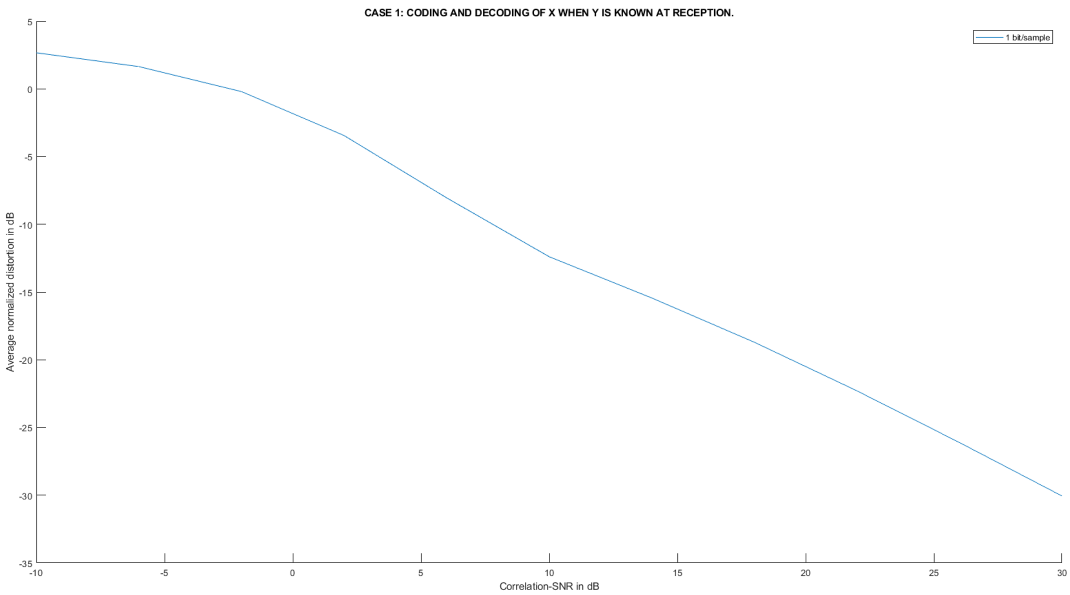

Figure 5: Coding and decoding of X when Y is known at reception. Average normalized distortion versus correlation-SNR.

When side-information Y is known at reception, $\widehat{X}$ estimated is obtained form decoded samples W and also from side information Y, where $Y = X + N$.. For this reason, the estimator response is dependent of SNR values. For low SNR values, $W \neq W_X$ due to side-information Y is unable to index the cosets. As it can been seen in figure [2.3], the distortion results are worse than those shown in the table [1], $\sigma_q \gg \sigma_{q_x}$.
System performance can be summarized in the following table [4]. Note: assume that $\sigma_x^2 = 1$ in computer simulations.

Tabla 4: Approximations/distortion of the estimator versus SNR values.

CASE_1 - 1bit/sample $\rightarrow \widehat{X} = Y\frac{\sigma_{q_x}^2}{\sigma_{q_x}^2+\sigma_n^2} + W\frac{\sigma_n^2}{\sigma_{q_x}^2+\sigma_n^2}$			
SNR	LOW	MEDIUM	HIGH
APPROXIMATIONS	W	$Y\frac{\sigma_{q_x}^2}{\sigma_{q_x}^2+\sigma_n^2} + W\frac{\sigma_n^2}{\sigma_{q_x}^2+\sigma_n^2}$	Y
DISTORTION[NATURAL UNITS]	σ_q^2	$\frac{\sigma_{q_x}^2\sigma_n^2}{\sigma_{q_x}^2+\sigma_n^2}$	0

3 Coding and decoding of X independent of Y

Figure 6: Defined scenario.

Transmission.

- Source: X.
- Lloyd-Max quantizer: 2-3 bits.
- Quantization noise: $Q_X = X - W_X$.
- Encoder: $f(X)$, where f is the encoding function.
- Rate: 1-2 bits/sample.

Reception.

- Decoded sample: $W = g(f(X))$, where g is the decoding function.
- Side-information: none.
- Estimator: $\widehat{X} = W$.

3.1 Description

At transmission side, TCQ uses 1-2 bits/sample to quantify Gaussian source X. Quantization levels are obtained from Lloyd-Max algorithm using 2-3 bits/sample. This alphabet is partitioned into 4 coset as follows:

- Case 2.1, rate 1 bit/sample: $S_0\{D_0\}$, $S_1\{D_1\}$, $S_2\{D_2\}$ and $S_3\{D_3\}$.
- Case 2.2, rate 2 bits/sample: $S_0\{D_0,D_4\}$, $S_1\{D_1,D_5\}$, $S_2\{D_2,D_5\}$ and $S_3\{D_3,D_7\}$.

Where S and D are cosets and centroids(quantization levels) respectively. Then, encoder generates the best sequence of bits minimizing the quantization distortion, which specifies the quantization levels for each sample of X.

At reception side, a convolutional code recovers the sequence of quantization levels chosen as best path of the Trellis diagram. Finally, a MSE estimator is used to obtain $\widehat{X}$ estimated from the decoded sample W. The estimator obtain is $\widehat{X} = W$ due to the lack of side-information.

3.2 Estimator

Let Y and W_X, such that $Y = X + N$ where N is White Gaussian Noise and $W_X = X - Q_X$ where Q_X is quantization noise, being X and N Gaussian samples i.i.d. and also Q_X and W_X. A linear estimator is used in order to obtain $\widehat{X}$ estimator.

$$\widehat{X} = \alpha W. \tag{23}$$

Minimizing mean squared error, α value is obtained, consequently the mathematical definition of $\widehat{X}$ estimator.

$$\min_{\alpha} E\{(X - \alpha W)^2\} \tag{24}$$

Doing accounts, this is the resulting estimator.

$$\widehat{X} = W \tag{25}$$

3.2.1 Maths

Assume that $W = W_X$.

$$
\begin{aligned}
E\{(X - \alpha W)^2\} &= E\{X^2 - 2X\alpha W_X + \alpha^2 W_X^2\} && (26)\\
&= E\{X^2\} - 2\alpha E\{X W_X\} + \alpha^2 E\{W_X^2\} && (27)\\
&= \sigma_x^2 + \alpha^2 \sigma_{w_x}^2 - 2\alpha E\{X W_X\} && (28)\\
&= \sigma_x^2 + \alpha^2 \sigma_{w_x}^2 - 2\alpha E\{(Q_X + W_X) W_X\} && (29)\\
&= \sigma_x^2 + \alpha^2 \sigma_{w_x}^2 - 2\alpha E\{Q_X W_X\} - 2\alpha E\{W_X^2\} && (30)\\
&= \sigma_x^2 + \alpha^2 \sigma_{w_x}^2 - 2\alpha \sigma_{w_x}^2 - 2\alpha E\{Q_X\} E\{W_X\} && (31)\\
&= \sigma_x^2 + \alpha^2 \sigma_{w_x}^2 - 2\alpha \sigma_{w_x}^2 && (32)
\end{aligned}
$$

Minimize MSE function.

$$
\begin{aligned}
\frac{d}{d_\alpha} E\{(X - \alpha W)^2\} &= \tfrac{d}{d_\alpha}\{\sigma_x^2 + \alpha^2 \sigma_{w_x}^2 - 2\alpha \sigma_{w_x}^2\} && (33)\\
&= 2\alpha \sigma_{w_x}^2 - 2\sigma_{w_x}^2 && (34)\\
& && (35)
\end{aligned}
$$

$$2\alpha \sigma_{w_x}^2 - 2\sigma_{w_x}^2 = 0 \rightarrow \alpha = 1 \tag{36}$$

Results:

$$\widehat{X} = W \tag{37}$$

3.2.2 Estimator response versus SNR values

If $Y = X + N$, it is assumed that for:

- Low SNR values: $Y \approx N \rightarrow \sigma_x^2 \ll \sigma_n^2$.

- High SNR values: $Y \approx X \rightarrow \sigma_x^2 \gg \sigma_n^2$

Approximations of the estimator versus SNR values.

- Low SNR values.

$$\lim_{\sigma_n^2 \rightarrow +\infty} \widehat{X} = \lim_{\sigma_n^2 \rightarrow +\infty} W \rightarrow \lim_{\sigma_n^2 \rightarrow +\infty} \widehat{X} = W \tag{38}$$

- High SNR values.

$$\lim_{\sigma_n^2 \rightarrow 0} \widehat{X} = \lim_{\sigma_n^2 \rightarrow 0} W \rightarrow \lim_{\sigma_n^2 \rightarrow 0} \widehat{X} = W \tag{39}$$

Obviously, the estimator is the same for all SNR values.

Distortion of the estimator versus SNR values. Distortion is measured as the average of the squares of the errors(MSE) between X and $\widehat{X}$. $D = E\{(X - \widehat{X})^2\}$, where D is the measure of distortion obtained. Assume that $W = W_X$.

$$D = E\{(X - \widehat{X})^2\} = E\{(X - W)^2\} = E\{(X - W_X)^2\} = E\{Q_X^2\} \rightarrow D = \sigma_{qx}^2 \tag{40}$$

3.3 Simulation results

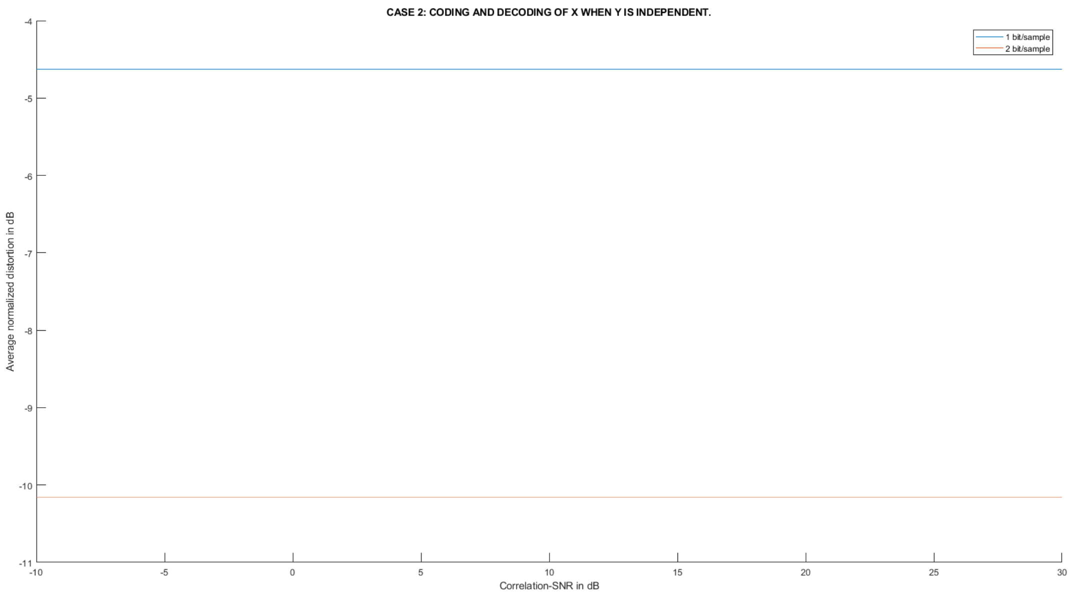

Figure 7: Coding and decoding of X independent of Y. Average normalized distortion versus correlation-SNR.

When there is no side-information, the only way to obtain $\widehat{X}$ estimated is through the decoded samples W, $\widehat{X} = W$. For this reason, the estimator response is constant and independent of SNR values.

Tabla 5: Approximations/distortion of the estimator.

	APPROXIMATIONS	DISTORTION[NATURAL UNITS]
$\widehat{X}_{CASE\_2.1}$ - 1bit/sample	W	$\sigma^2_{q_x}$
$\widehat{X}_{CASE\_2.2}$ - 2bits/sample	W	$\sigma^2_{q_x}$

We must not forget, the close relationship between the number of quantization levels and distortion. A greater number of quantization levels implies the use of a finer lattices, with the consequent reduction of distortion measurement between X and $\widehat{X}$ estimated.

System performance can be summarized in the following points, see tables [1] and [5]. Note: assume that $\sigma^2_x = 1$ in computer simulations.

- Case 2.1 - 1bit/sample, 4 quantization levels $\rightarrow \sigma^2_{q_x} \approx -4.64dB$.
- Case 2.2 - 2bits/sample, 8 quantization levels $\rightarrow \sigma^2_{q_x} \approx -10.19dB$.

4 Coding and decoding of X when Y is known at both sides

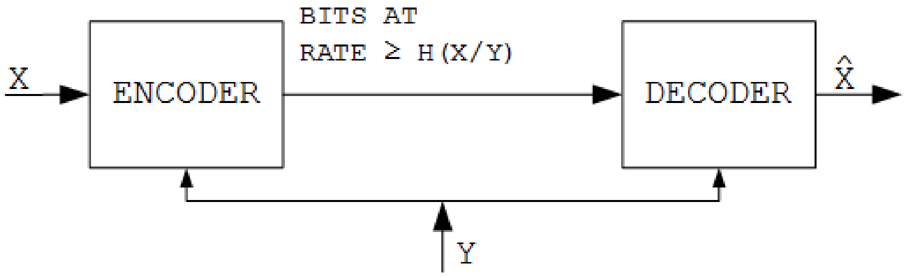

Figure 8: Defined scenario.

Transmission.

- Source: X and Y, where $Y = X + N$.
- Lloyd-Max quantizer, 2 bits.
- Quantization noise: $Q_N = N - W_N$.
- Encoder: $f(N)$, where f is the encoding function and N is the noise parameter.
- Rate: 1 bit/sample.

Reception.

- Decoded sample: $W = g(f(N))$, where g is the decoding function.
- Side-information: Y, where $Y = X + N$.
- Estimator: $\widehat{X} = (Y - W)\frac{\sigma_x^2}{\sigma_x^2+\sigma_{qn}^2}$.

4.1 Description

When Y is known at transmission and reception and assuming $\sigma_n^2 \leq \sigma_x^2$, noise parameter N is quantized instead of source X. Since the variance of N is smaller, the reconstruction levels of the Lloyd Max quantizer will be more optimal, resulting in a lower quantification error.

At transmission side, TCQ uses 1 bits/sample to quantify Gaussian noise N. Quantization levels are obtained from Lloyd-Max algorithm using 2 bits/sample. This alphabet is partitioned into 4 coset as follows:

- Rate 1 bits/sample: $S_0\{D_0\}$, $S_1\{D_1\}$, $S_2\{D_2\}$ and $S_3\{D_3\}$.

Where S and D are cosets and centroids(quantization levels) respectively. Then, encoder generates the best sequence of bits minimizing the quantization distortion, which specifies the quantization levels for each sample of N.

At reception side, a convolutional code recovers the sequence of quantization levels chosen as best path of the Trellis diagram. Finally, a MSE estimator is used to obtain $\widehat{X}$ estimated from decoded sample W and side-information Y.

4.2 Estimator

Let Y and W_N, such that $Y = X + N$ where N is White Gaussian Noise and $W_N = N - Q_N$ where Q_N is quantization noise, being X and N Gaussian samples i.i.d. and also Q_N and W_N. A linear estimator is used in order to obtain $\widehat{N}$ estimator.

$$\widehat{N} = \alpha Y + \beta W_N. \tag{41}$$

Minimizing mean squared error, α and β values are obtained, consequently the mathematical definition of $\widehat{N}$ estimator.

$$\min_{\alpha,\beta} E\{(N - (\alpha Y + \beta W_N))^2\} \tag{42}$$

Doing accounts, this is the resulting estimator.

$$\widehat{X} = Y - \widehat{N},\ \textit{where}\ \widehat{N} = Y\frac{\sigma_{q_n}^2}{\sigma_x^2 + \sigma_{q_n}^2} + W\frac{\sigma_x^2}{\sigma_x^2 + \sigma_{q_n}^2} \rightarrow \widehat{X} = (Y - W)\frac{\sigma_x^2}{\sigma_x^2 + \sigma_{q_n}^2} \tag{43}$$

4.2.1 Maths

Assume that $W = W_N$.

$$\begin{aligned}
E\{(N - (\alpha Y + \beta W))^2\} &= E\{N^2 - 2N(\alpha Y + \beta W_N) + (\alpha Y + \beta W_N)^2\} = E\{N^2 - 2\alpha NY & (44)\\
&+2\beta N W_N + \alpha^2 Y^2 + 2\alpha\beta Y W_N + \beta^2 W_N^2\} = E\{N^2\} - 2\alpha E\{NY\} & (45)\\
&-2\beta E\{N W_N\} + \alpha^2 E\{Y^2\} + 2\alpha\beta E\{Y W_N\} + \beta^2 E\{W_N^2\} & (46)\\
&= \sigma_n^2 + \alpha^2\sigma_y^2 + \beta^2\sigma_{w_n}^2 + 2\alpha\beta E\{Y W_N\} - 2\alpha E\{NY\} & (47)\\
&-2\beta E\{N W_N\} = \sigma_n^2 + \alpha^2\sigma_y^2 + \beta^2\sigma_{w_n}^2 + 2\alpha\beta\sigma_{w_n}^2 - 2\alpha\sigma_n^2 - 2\beta\sigma_{w_n}^2 & (48)\\
& & (49)
\end{aligned}$$

Expected values:

- $E\{XW_N\}$

$$\begin{aligned} E\{(X(N-Q_N))\} &= E\{XN\} - E\{XQ_N\} && (50)\\ &= E\{X\}E\{N\} - E\{X\}E\{Q_N\} && (51)\\ &= 0 && (52) \end{aligned}$$

- $E\{YW_N\}$

$$\begin{aligned} E\{(X+N)W_N\} &= E\{(X+Q_N+W_N)W_N\} && (53)\\ &= E\{XW_N\} + E\{Q_NW_N\} + E\{W_N^2\} && (54)\\ &= E\{XW_N\} + E\{Q_N\}E\{W_N\} + \sigma_{w_n}^2 && (55)\\ &= \sigma_{w_n}^2 && (56) \end{aligned}$$

- $E\{NY\}$

$$\begin{aligned} E\{N(X+N)\} &= E\{NX+N^2\} && (57)\\ &= E\{NX\} + E\{N^2\} && (58)\\ &= E\{N\}E\{X\} + \sigma_n^2 && (59)\\ &= \sigma_n^2 && (60) \end{aligned}$$

- $E\{NW_N\}$

$$\begin{aligned} E\{(Q_N+W_N)W_N\} &= E\{QW_N + W_N^2\} && (61)\\ &= E\{QW_N\} + E\{W_N^2\} && (62)\\ &= E\{Q_N\}E\{W_N\} + \sigma_{w_n}^2 && (63)\\ &= \sigma_{w_n}^2 && (64) \end{aligned}$$

Minimize MSE function.

- First derivative α

$$\begin{aligned} \frac{d}{d_\alpha}E\{(N-(\alpha Y+\beta W))^2\} &= \tfrac{d}{d_\alpha}\{\sigma_n^2 + \alpha^2\sigma_y^2 + \beta^2\sigma_{w_n}^2 + 2\alpha\beta\sigma_{w_n}^2 - 2\alpha\sigma_n^2 - 2\beta\sigma_{w_n}^2\} && (65)\\ &= 2\alpha\sigma_y^2 + 2\beta\sigma_{w_n}^2 - 2\sigma_n^2 = 0 && (66)\\ & && (67) \end{aligned}$$

- First derivative β

$$\begin{aligned} \frac{d}{d_\beta}E\{(N-(\alpha Y+\beta W))^2\} &= \tfrac{d}{d_\beta}\{\sigma_n^2 + \alpha^2\sigma_y^2 + \beta^2\sigma_{w_n}^2 + 2\alpha\beta\sigma_{w_n}^2 - 2\alpha\sigma_n^2 - 2\beta\sigma_{w_n}^2\} && (68)\\ &= 2\beta\sigma_{w_n}^2 + 2\alpha\sigma_{w_n}^2 - 2\sigma_{w_n}^2 = 0 && (69)\\ & && (70) \end{aligned}$$

Solve system of linear equations.

$$\begin{Bmatrix} \alpha\sigma_y^2 + \beta\sigma_{w_n}^2 = \sigma_n^2 \\ \alpha\sigma_{w_n}^2 + \beta\sigma_{w_n}^2 = \sigma_{w_n}^2 \end{Bmatrix} \tag{71}$$

$$\alpha\sigma_y^2 + \beta\sigma_{w_n}^2 = \sigma_n^2 \rightarrow \alpha = \frac{\sigma_n^2 - \beta\sigma_{w_n}^2}{\sigma_y^2} \tag{72}$$

β value:

$$\alpha\sigma_{w_n}^2 + \beta\sigma_{w_n}^2 = \sigma_{w_n}^2 \tag{73}$$

$$\tfrac{\sigma_n^2 - \beta\sigma_{w_n}^2}{\sigma_y^2}\sigma_{w_n}^2 + \beta\sigma_{w_n}^2 = \sigma_{w_n}^2 \tag{74}$$

$$\tfrac{\sigma_n^2\sigma_{w_n}^2}{\sigma_y^2} + \beta(\sigma_{w_n}^2 - \tfrac{\sigma_{w_n}^4}{\sigma_y^2}) = \sigma_{w_n}^2 \tag{75}$$

$$\tfrac{\sigma_n^2\sigma_{w_n}^2}{\sigma_y^2} + \beta\tfrac{\sigma_{w_n}^2\sigma_y^2 - \sigma_{w_n}^4}{\sigma_y^2} = \sigma_{w_n}^2 \tag{76}$$

$$\beta\tfrac{\sigma_{w_n}^2\sigma_y^2 - \sigma_{w_n}^4}{\sigma_y^2} = \sigma_{w_n}^2 - \tfrac{\sigma_n^2\sigma_{w_n}^2}{\sigma_y^2} \tag{77}$$

$$\beta\tfrac{\sigma_{w_n}^2\sigma_y^2 - \sigma_{w_n}^4}{\sigma_y^2} = \tfrac{\sigma_{w_n}^2\sigma_y^2 - \sigma_n^2\sigma_{w_n}^2}{\sigma_y^2} \tag{78}$$

$$\beta = \tfrac{\sigma_{w_n}^2\sigma_y^2 - \sigma_n^2\sigma_{w_n}^2}{\sigma_{w_n}^2\sigma_y^2 - \sigma_{w_n}^4} = \tfrac{\sigma_y^2 - \sigma_n^2}{\sigma_y^2 - \sigma_{w_n}^2} \tag{79}$$

$$\rightarrow \beta = \tfrac{\sigma_x^2}{\sigma_x^2 + \sigma_{qn}^2} \tag{80}$$

$$\tag{81}$$

α value:

$$\alpha = \frac{\sigma_n^2 - \beta\sigma_{w_n}^2}{\sigma_y^2} \tag{82}$$

$$= \tfrac{1}{\sigma_y^2}(\sigma_n^2 - \tfrac{\sigma_x^2\sigma_{w_n}^2}{\sigma_x^2 + \sigma_{qn}^2}) = \tfrac{\sigma_n^2(\sigma_x^2 + \sigma_{qn}^2) - \sigma_x^2\sigma_{w_n}^2}{(\sigma_x^2 + \sigma_{qn}^2)\sigma_y^2} = \tfrac{\sigma_n^2\sigma_x^2 + \sigma_n^2\sigma_{qn}^2 - \sigma_x^2\sigma_{w_n}^2}{(\sigma_x^2 + \sigma_{qn}^2)\sigma_y^2} \tag{83}$$

$$= \tfrac{\sigma_x^2(\sigma_n^2 - \sigma_{w_n}^2) + \sigma_n^2\sigma_{qn}^2}{(\sigma_x^2 + \sigma_{qn}^2)\sigma_y^2} = \tfrac{\sigma_x^2\sigma_{qn}^2 + \sigma_n^2\sigma_{qn}^2}{(\sigma_x^2 + \sigma_{qn}^2)\sigma_y^2} = \tfrac{\sigma_{qn}^2(\sigma_x^2 + \sigma_n^2)}{(\sigma_x^2 + \sigma_{qn}^2)\sigma_y^2} = \tfrac{\sigma_{qn}^2\sigma_y^2}{(\sigma_x^2 + \sigma_{qn}^2)\sigma_y^2} \tag{84}$$

$$\rightarrow \alpha = \tfrac{\sigma_{qn}^2}{\sigma_x^2 + \sigma_{qn}^2} \tag{85}$$

Results:

$$\widehat{X} = Y - \widehat{N},\ where\ \widehat{N} = Y\tfrac{\sigma_{qn}^2}{\sigma_x^2 + \sigma_{qn}^2} + W\tfrac{\sigma_x^2}{\sigma_x^2 + \sigma_{qn}^2} \rightarrow \widehat{X} = (Y - W)\tfrac{\sigma_x^2}{\sigma_x^2 + \sigma_{qn}^2} \tag{86}$$

4.2.2 Estimator response versus SNR values

If $Y = X + N$, it is assumed that for:

- Low SNR values: $Y \approx N \to \sigma_x^2 \ll \sigma_n^2$.
- High SNR values: $Y \approx X \to \sigma_x^2 \gg \sigma_n^2$

Approximations of the estimator versus SNR values. Assume that $snr = \frac{\sigma_x^2}{\sigma_n^2} \to \sigma_x^2 = \sigma_n^2 snr$.

$$\widehat{X} = (Y - W)\frac{\sigma_x^2}{\sigma_x^2 + \sigma_{q_n}^2} = \widehat{X} = (Y - W)\frac{\sigma_n^2 snr}{\sigma_n^2 snr + \sigma_{q_n}^2} \tag{87}$$

- Low SNR values:

$$\lim_{snr \to 0} \widehat{X} = \lim_{snr \to 0} Y(Y - W)\frac{\sigma_n^2 snr}{\sigma_n^2 snr + \sigma_{q_n}^2} \tag{88}$$

$$\to \lim_{snr \to 0} \widehat{X} = 0 \tag{89}$$

- High SNR values:

$$\lim_{snr \to \infty} \widehat{X} = \lim_{snr \to \infty} (Y - W)\frac{\sigma_n^2 snr}{\sigma_n^2 snr + \sigma_{q_n}^2} \tag{90}$$

$$\to \lim_{snr \to \infty} \widehat{X} = Y - W \tag{91}$$

Distortion of the estimator versus SNR values. Distortion is measured as the average of the squares of the errors(MSE) between X and $\widehat{X}$. $D = E\{(X - \widehat{X})^2\}$, where D is the measure of distortion obtained. Assume that $W = W_N$.

- Low SNR values:

$$D_{LOW} = E\{(X - \widehat{X})^2\} \approx E\{X^2\} = \sigma_x^2 \to D_{LOW} = \sigma_x^2 \tag{92}$$

- Medium SNR values:

$$D_{MEDIUM} = E\{(X - \widehat{X})^2\} = E\{(X - \frac{(Y - W)\sigma_x^2}{\sigma_x^2 + \sigma_{q_n}^2})^2\} = E\{(X - \frac{(Y - W_N)\sigma_x^2}{\sigma_x^2 + \sigma_{q_n}^2})^2\} \tag{93}$$

Replace $c = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_{q_n}^2}$.

$$D_{MEDIUM} = E\{(X - (Y - W_N)c)^2\} = E\{c^2Y^2 - 2c^2YW_N + c^2W_N^2 - 2cXY + \tag{94}$$

$$2cXW_N + X^2\} = c^2E\{Y^2\} - 2c^2E\{YW_N\} \tag{95}$$

$$+c^2E\{W_N^2\} - 2cE\{XY\} + 2cE\{XW_N\} + E\{X^2\} \tag{96}$$

$$\tag{97}$$

Note: $E\{Y^2\} = \sigma_y^2$, $E\{YW_N\} = \sigma_{w_n}^2$, $E\{W_N^2\} = \sigma_{w_n}^2$, $E\{XY\} = \sigma_x^2$, $E\{XW_N\} = 0$ and $E\{X^2\} = \sigma_x^2$.

$$\begin{aligned} D_{MEDIUM} &= c^2\sigma_y^2 - 2c^2\sigma_{w_n}^2 + c^2\sigma_{w_n}^2 - 2c\sigma_x^2 + \sigma_x^2 = c^2\sigma_y^2 - c^2\sigma_{w_n}^2 - 2c\sigma_x^2 + \sigma_x^2 && (98)\\ &= c^2(\sigma_y^2 - \sigma_{w_n}^2) - 2c\sigma_x^2 + \sigma_x^2 = c^2(\sigma_x^2 + \sigma_n^2 - (\sigma_n^2 - \sigma_{q_n}^2)) - 2c\sigma_x^2 + \sigma_x^2 && (99)\\ &= c^2(\sigma_x^2 + \sigma_{q_n}^2) - 2c\sigma_x^2 + \sigma_x^2 && (100)\\ & && (101)\end{aligned}$$

Solve constant c.

$$\begin{aligned} D_{MEDIUM} &= c^2(\sigma_x^2 + \sigma_{q_n}^2) - 2c\sigma_x^2 + \sigma_x^2 = \frac{\sigma_x^4}{(\sigma_x^2+\sigma_{q_n}^2)^2}(\sigma_x^2 + \sigma_{q_n}^2) - 2\frac{\sigma_x^4}{\sigma_x^2+\sigma_{q_n}^2} + \sigma_x^2 && (102)\\ &= \frac{\sigma_x^4}{\sigma_x^2+\sigma_{q_n}^2} - 2\frac{\sigma_x^4}{\sigma_x^2+\sigma_{q_n}^2} + \sigma_x^2 = \sigma_x^2 - \frac{\sigma_x^4}{\sigma_x^2+\sigma_{q_n}^2} && (103)\\ &\to D_{MEDIUM} = \frac{\sigma_x^2\sigma_{q_n}^2}{\sigma_x^2+\sigma_{q_n}^2} && (104)\\ & && (105)\end{aligned}$$

- High SNR values:

$$\begin{aligned} D_{HIGH} &= E\{(X - \widehat{X})^2\} \approx E\{(X - (Y - W))^2\} = E\{(X - (X - W_N))^2\} && (106)\\ &= E\{W_N^2\} = \sigma_{w_n}^2 = -\sigma_{q_n}^2 \to D_{HIGH} = 0 && (107)\\ & && (108)\end{aligned}$$

$D_{SQ}(R) = \sigma_x^2 2^{-2R}$, where σ_x^2 is the variance of X, and R is the encoding rate in bits per sample. If $\sigma_x^2 \approx 0$, then $D_{SQ}(R) \approx 0$. *Trellis coded quantization of memoryless and Gauss—Markov Sources* (see References).

4.3 Simulation results

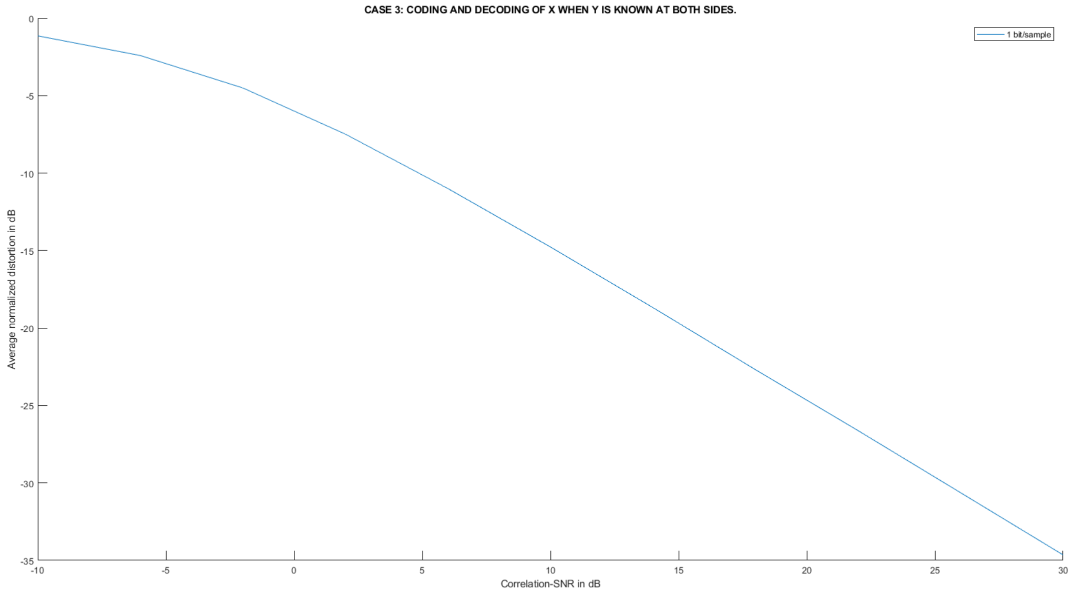

Figure 9: Coding and decoding of X when Y is known at both sides. Average normalized distortion versus correlation-SNR.

When Y is known at both sides, side-information Y may be used in reception and in transmission. $\widehat{X}$ estimated is obtained from decoded samples W and also from side-information Y, where $Y = X + N$. For this reason, the estimator response is dependent of SNR values.
System performance can be summarized in the following table [6]. Note: assume that $\sigma_x^2 = 1$ in computer simulations.

Tabla 6: Approximations/distortion of the estimator versus SNR values.

CASE_3 - 1bit/sample $\rightarrow \widehat{X} = Y\frac{\sigma_x^2}{\sigma_x^2+\sigma_{qn}^2} - W\frac{\sigma_x^2}{\sigma_x^2+\sigma_{qn}^2}$			
SNR	LOW	MEDIUM	HIGH
APPROXIMATIONS	0	$Y\frac{\sigma_x^2}{\sigma_x^2+\sigma_{qn}^2} - W\frac{\sigma_x^2}{\sigma_x^2+\sigma_{qn}^2}$	$Y - W$
DISTORTION[NATURAL UNITS]	σ_x^2	$\frac{\sigma_x^2\sigma_{qn}^2}{\sigma_x^2+\sigma_{qn}^2}$	0

5 Estimation of X exclusively based on Y

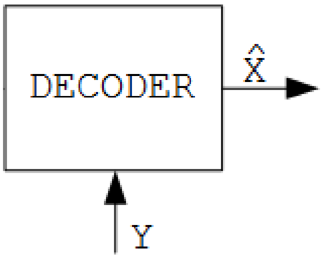

Figure 10: Defined scenario.

Reception

- Side-information: Y, where $Y = X + N$.
- Estimator: $\widehat{X} = Y\frac{\sigma_x^2}{\sigma_x^2+\sigma_n^2}$

5.1 Description

There is no communication, the only information which can be exploited to obtain $\widehat{X}$ estimated is the side-information Y. This case is consider as a math problem, MSE estimator.

5.2 Estimator

Let X, such that $Y = X + N$ where N is White Gaussian Noise, being X and N Gaussian samples i.i.d.. A linear estimator is used in order to obtain $\widehat{X}$ estimator.

$$\widehat{X} = \alpha Y \tag{109}$$

Mean squared error of an estimator measure the average of the squares of the error or deviation. To obtain the estimator, it's necessary to find value of α which minimize MSE between X and $\widehat{X}$, αY .

$$\min_{\alpha} E\{(X - \alpha Y)^2\} \tag{110}$$

Doing accounts, this is the resulting estimator.

$$\widehat{X} = Y\frac{\sigma_x^2}{\sigma_x^2 + \sigma_n^2} \tag{111}$$

5.2.1 Maths

$$
\begin{aligned}
E\{(X-\alpha Y)^2\} &= E\{X^2 - 2X\alpha Y + \alpha^2 Y^2\} && (112)\\
&= E\{X^2\} - 2\alpha E\{XY\} + \alpha^2 E\{Y^2\} && (113)\\
&= \sigma_x^2 + \alpha^2\sigma_y^2 - 2\alpha E\{X(X+N)\} && (114)\\
&= \sigma_x^2 + \alpha^2\sigma_y^2 - 2\alpha E\{X^2 + XN\} && (115)\\
&= \sigma_x^2 + \alpha^2\sigma_y^2 - 2\alpha(E\{X^2\} + E\{XN\}) && (116)\\
&= \sigma_x^2 + \alpha^2\sigma_y^2 - 2\alpha(\sigma_x^2 + E\{X\}E\{N\}) && (117)\\
&= \sigma_x^2 + \alpha^2(\sigma_x^2 + \sigma_n^2) - 2\alpha\sigma_x^2 && (118)\\
& && (119)
\end{aligned}
$$

Minimize MSE function.

$$
\begin{aligned}
\frac{d}{d_\alpha}E\{(X-\alpha Y)^2\} &= \tfrac{d}{d_\alpha}\{\sigma_x^2 + \alpha^2(\sigma_x^2 + \sigma_n^2) - 2\alpha\sigma_x^2\} && (120)\\
&= 2\alpha(\sigma_x^2 + \sigma_n^2) - 2\sigma_x^2 && (121)\\
& && (122)
\end{aligned}
$$

$$2\alpha(\sigma_x^2 + \sigma_n^2) - 2\sigma_x^2 = 0 \rightarrow \alpha = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_n^2} \tag{123}$$

Results:

$$\widehat{X} = Y\frac{\sigma_x^2}{\sigma_x^2 + \sigma_n^2} \tag{124}$$

5.2.2 Estimator response versus SNR values

If $Y = X + N$, it is assumed that for:

- Low SNR values: $Y \approx N \rightarrow \sigma_x^2 \ll \sigma_n^2$.
- High SNR values: $Y \approx X \rightarrow \sigma_x^2 \gg \sigma_n^2$

Approximations of the estimator versus SNR values.

- Low SNR values:

$$\lim_{\sigma_n^2 \rightarrow +\infty} \widehat{X} = \lim_{\sigma_n^2 \rightarrow +\infty} Y\frac{\sigma_x^2}{\sigma_x^2 + \sigma_n^2} \rightarrow \lim_{\sigma_n^2 \rightarrow +\infty} \widehat{X} = 0 \tag{125}$$

- High SNR values:

$$\lim_{\sigma_n^2 \rightarrow 0} \widehat{X} = \lim_{\sigma_n^2 \rightarrow 0} Y\frac{\sigma_x^2}{\sigma_x^2 + \sigma_n^2} \rightarrow \lim_{\sigma_n^2 \rightarrow 0} \widehat{X} = Y \tag{126}$$

Distortion of the estimator versus SNR values. Distortion is measured as the average of the squares of the errors(MSE) between X and $\widehat{X}$. $D = E\{(X - \widehat{X})^2\}$, where D is the measure of distortion obtained.

- Low SNR values:

$$D_{LOW} = E\{(X - \widehat{X})^2\} \approx E\{(X - 0)^2\} = E\{X^2\} = \sigma_x^2 \rightarrow D_{LOW} = \sigma_x^2 \tag{127}$$

- Medium SNR values:

$$\begin{aligned}
D_{MEDIUM} &= E\{(X - \widehat{X})^2\} = E\{(X - Y\tfrac{\sigma_x^2}{\sigma_x^2+\sigma_n^2})^2\} && (128)\\
&= E\{X^2 - 2XY\tfrac{\sigma_x^2}{\sigma_x^2+\sigma_n^2} + Y^2(\tfrac{\sigma_x^2}{\sigma_x^2+\sigma_n^2})^2\} && (129)\\
&= E\{X^2\} + \tfrac{\sigma_x^4}{(\sigma_x^2+\sigma_n^2)^2}E\{Y^2\} - \tfrac{2\sigma_x^2}{\sigma_x^2+\sigma_n^2}E\{XY\} && (130)\\
&= \sigma_x^2 + \tfrac{\sigma_x^4\sigma_y^2}{(\sigma_x^2+\sigma_n^2)^2} - \tfrac{2\sigma_x^2}{\sigma_x^2+\sigma_n^2}E\{X(X+N)\} && (131)\\
&= \sigma_x^2 + \tfrac{\sigma_x^4\sigma_y^2}{(\sigma_x^2+\sigma_n^2)^2} - \tfrac{2\sigma_x^2}{\sigma_x^2+\sigma_n^2}E\{X^2+XN)\} && (132)\\
&= \sigma_x^2 + \tfrac{\sigma_x^4\sigma_y^2}{(\sigma_x^2+\sigma_n^2)^2} - \tfrac{2\sigma_x^2}{\sigma_x^2+\sigma_n^2}E\{X^2\} - \tfrac{2\sigma_x^2}{\sigma_x^2+\sigma_n^2}E\{X\}E\{N\} && (133)\\
&= \sigma_x^2 + \tfrac{\sigma_x^4\sigma_y^2}{(\sigma_x^2+\sigma_n^2)^2} - \tfrac{2\sigma_x^4}{\sigma_x^2+\sigma_n^2} = \sigma_x^2 + \tfrac{\sigma_x^4(\sigma_x^2+\sigma_n^2)}{(\sigma_x^2+\sigma_n^2)^2} - \tfrac{2\sigma_x^4}{\sigma_x^2+\sigma_n^2} && (134)\\
&= \sigma_x^2 + \tfrac{\sigma_x^4}{\sigma_x^2+\sigma_n^2} - \tfrac{2\sigma_x^4}{\sigma_x^2+\sigma_n^2} = \sigma_x^2 - \tfrac{\sigma_x^4}{\sigma_x^2+\sigma_n^2} = \tfrac{\sigma_x^4-\sigma_x^4+\sigma_x^2\sigma_n^2}{\sigma_x^2+\sigma_n^2} && (135)\\
&\rightarrow D_{MEDIUM} = \tfrac{\sigma_x^2\sigma_n^2}{\sigma_x^2+\sigma_n^2} && (136)\\
& && (137)
\end{aligned}$$

- High SNR values:

$$D_{HIGH} = E\{(X - \widehat{X})^2\} \approx E\{(X - Y)^2\} = E\{(X - X)^2\} = 0 \rightarrow D_{HIGH} = 0 \tag{138}$$

5.3 Simulation results

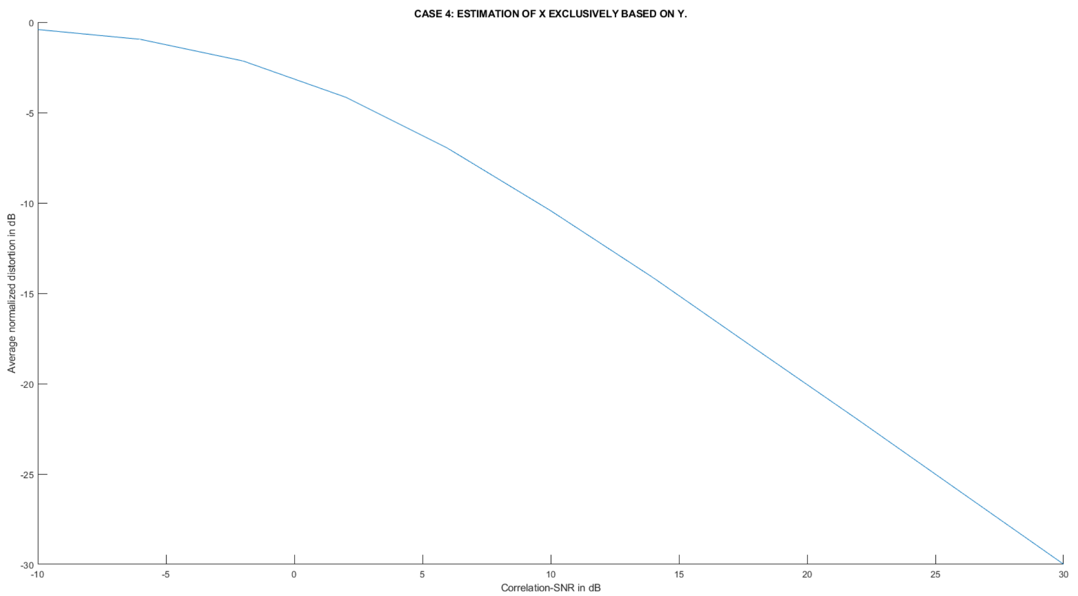

Figure 11: Estimation of X exclusively based on Y. Average normalized distortion versus correlation-SNR.

When there is no transmission information, or decoded samples W, the only way to obtain $\widehat{X}$ estimated is through the side-information Y, where $Y = X + N$. For this reason, the estimator response is dependent of SNR values. This scenario is not really a communication system consisting of a transmitter and a receiver, rather it is an mathematical estimator that provides an approximate version $\widehat{X}$ of X exclusively based on Y.
System performance can be summarized in the following table [7]. Note: assume that $\sigma_x^2 = 1$ in computer simulations.

Tabla 7: Approximations/distortion of the estimator versus SNR values.

CASE_4 $\rightarrow \widehat{X} = Y\frac{\sigma_x^2}{\sigma_x^2+\sigma_n^2}$			
SNR	LOW	MEDIUM	HIGH
APPROXIMATIONS	0	$Y\frac{\sigma_x^2}{\sigma_x^2+\sigma_n^2}$	Y
DISTORTION[NATURAL UNITS]	σ_x^2	$\frac{\sigma_x^2\sigma_n^2}{\sigma_x^2+\sigma_n^2}$	0

6 Conclusions

In this section, a final conclusion is proposed from the different scenarios studied at the previous sections.

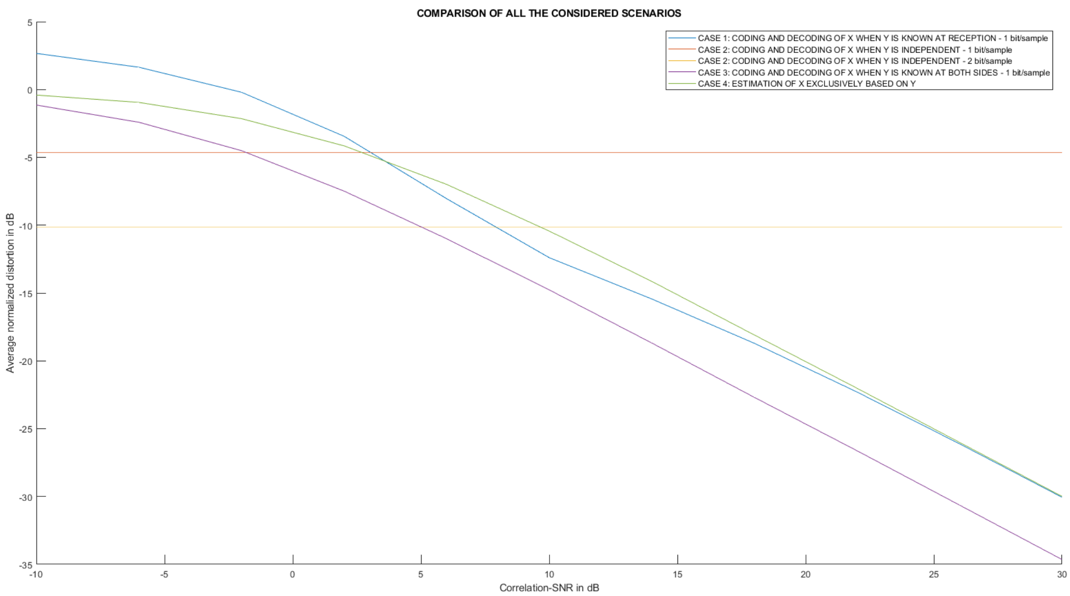

Figure 12: Comparison of all the considered scenarios. Average normalized distortion versus correlation-SNR.

Assume the following points:

- A fixed value of $\sigma_x = 1$ has been used in all of simulations.
- The distortion for σ_{q_x} at 1bit/sample $-4.64dB$.
- The distortion for σ_{q_x} at 2bits/sample $-10.19dB$.
- In case 1, for low SNR values $W \neq W_X \to \sigma_q \gg \sigma_{q_w}$.
- If $Y = X + N$, it is assumed that for:
  - Low SNR values: $Y \approx N \to \sigma_x^2 \ll \sigma_n^2$.
  - High SNR values: $Y \approx X \to \sigma_x^2 \gg \sigma_n^2$

Tabla 8: Approximations of the different kinds of estimators versus SNR values.

SNR	LOW	MEDIUM	HIGH
$\widehat{X}_{CASE\_1}$ - 1bit/sample	W	$Y\frac{\sigma_{q_x}^2}{\sigma_{q_x}^2+\sigma_n^2} + W\frac{\sigma_n^2}{\sigma_{q_x}^2+\sigma_n^2}$	Y
$\widehat{X}_{CASE\_2.1}$ - 1bit/sample	W	W	W
$\widehat{X}_{CASE\_2.2}$ - 2bits/sample	W	W	W
$\widehat{X}_{CASE\_3}$ - 1bit/sample	0	$(Y-W)\frac{\sigma_x^2}{\sigma_x^2+\sigma_{q_n}^2}$	$Y-W$
$\widehat{X}_{CASE\_4}$	0	$Y\frac{\sigma_x^2}{\sigma_x^2+\sigma_n^2}$	Y

System performance can be summarized in the following table[9]:

Tabla 9: Distortion of the different kinds of estimators versus SNR values (in natural units).

SNR	LOW	MEDIUM	HIGH
$\widehat{X}_{CASE\_1}$ - 1bit/sample	σ_q^2	$\frac{\sigma_{qx}^2\sigma_n^2}{\sigma_{qx}^2+\sigma_n^2}$	0
$\widehat{X}_{CASE\_2.1}$ - 1bit/sample	σ_{qx}^2	σ_{qx}^2	σ_{qx}^2
$\widehat{X}_{CASE\_2.2}$ - 2bits/sample	σ_{qx}^2	σ_{qx}^2	σ_{qx}^2
$\widehat{X}_{CASE\_3}$ - 1bit/sample	σ_x^2	$\frac{\sigma_x^2\sigma_{qn}^2}{\sigma_x^2+\sigma_{qn}^2}$	0
$\widehat{X}_{CASE\_4}$	σ_x^2	$\frac{\sigma_x^2\sigma_n^2}{\sigma_x^2+\sigma_n^2}$	0

References

[1] W. Marcellin, Michael Fischer, T.R., *Trellis coded quantization of memoryless and Gauss—Markov Sources. Communications.*, IEEE Transactions on. 38. 82 - 93. 1990

[2] Pradhan, S Ramchandran, Kannan, *Distributed Source Coding Using Syndromes (DISCUS): Design and Construction*, IEEE Transactions on Information Theory - TIT. 49. 158-167. 1999

[3] V. Chappelier, C. Guillemot S. Marinkovic, *Turbo Trellis-Coded Quantization*, IRISA/Université de Rennes.

[4] Aditya K. Jagannatham, *Optimal Lloyd-Max quantizer design*, IIT Kanpur.

A Gaussian random variables X and Y

Let X and N two independent Gaussian random variables:

$$X \sim N(0, \sigma_x^2), \text{ where E[X]} = 0 \text{ and V[X]} = \sigma_x^2 \tag{139}$$

$$N \sim N(0, \sigma_n^2), \text{ where E[N]} = 0 \text{ and V[N]} = \sigma_n^2 \tag{140}$$

Y is defined as shown:

$$Y = X + N \sim N(0, \sigma_x^2 + \sigma_n^2), \text{ where E[Y]} = 0 \text{ and V[Y]} = \sigma_x^2 + \sigma_n^2 \tag{141}$$

h(X;Y)

The probability density function of X and Y, is:

$$\phi(x,y) = \frac{1}{\sqrt{(2\pi)^2|\Sigma|}} exp(-\frac{1}{2}(m-\mu_x)^T\Sigma^{-1}(m-\mu_x)) \tag{142}$$

Where:

- $m = [x \ y]^T$
- $\mu = [\mu_x \ \mu_y]^T$
- Σ covariance matrix.

Do math to obtain the covariance matrix.

$$\begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}, \text{ where } \rho \text{ is the correlation coefficient between X and Y.} \tag{143}$$

$$\begin{aligned} \rho\sigma_x\sigma_y = E[(x-\mu_x)(y-\mu_y)] = E[xy - \mu_x y - \mu_y x + \mu_x\mu_y] = E[xy] - \mu_x E[y] - \mu_y E[x] + \mu_x\mu_y = \\ E[x(x+y)] - \mu_x E[y] - \mu_y E[x] + \mu_x\mu_y = E[x^2] + E[xy] - \mu_x E[y] - \mu_y E[x] + \mu_x\mu_y = \\ \sigma_x^2 + \sigma_x^2 + E[x]E[y] - \mu_x E[y] - \mu_y E[x] + \mu_x\mu_y = \sigma_x + \sigma_y \end{aligned} \tag{144}$$

The covariance matrix obtained is:

$$\begin{bmatrix} \sigma_x^2 & \sigma_x + \sigma_y \\ \sigma_x + \sigma_y & \sigma_y^2 \end{bmatrix} \tag{145}$$

Then calculating the joint differential entropy in bits, it is obtained:

$$h(X,Y) = \frac{1}{2}\log_2((2\pi e)^2|\Sigma|) \tag{146}$$

B Lloyd-Max Quantizer Algorithm

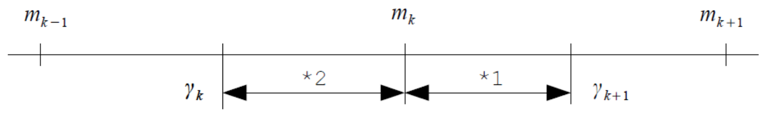

Figure 13: Lloyd-Max quantizer

where:

- *1 Quantized to γ_k
- *2 Quantized to γ_{k+1}

Algorithm: Iterative repeat steps A and B until convergence to get optimal quantizer. Minimize the mean squared quantization error.

- Step A: Levels-Intervals. $\gamma_k = \frac{\sqrt{\frac{\sigma^2}{2\pi}}[e^{-\frac{m_{k-1}^2}{2\sigma^2}} - e^{-\frac{m_k^2}{2\sigma^2}}]}{qfunc(\frac{m_k}{\sigma}) - qfunc(\frac{m_{k-1}}{\sigma})}$
- Step B: Intervals-levels. $\mathrm{m}_k = \frac{1}{2}(\gamma_{k+1} + \gamma_k)$

B.1 Step A: Levels-Intervals

Calculation of quantification level γ, where $f(x)$ is probability density function.

$$\gamma = \frac{\int_{m_k}^{m_{k-1}} xf(x)dx}{\int_{m_k}^{m_{k-1}} f(x)dx},\ where\ f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}} \tag{147}$$

Numerator:

$$\int_{m_k}^{m_{k-1}} xf(x)dx \quad = \int_{m_k}^{m_{k-1}} \frac{x}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} dx \quad = -\sqrt{\frac{\sigma^2}{2\pi}} \int_{m_k}^{m_{k-1}} -\frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}} dx \tag{148}$$

$$= -\sqrt{\frac{\sigma^2}{2\pi}} e^{-\frac{x^2}{2\sigma^2}} \Big|_{m_k}^{m_{k-1}} \quad = \sqrt{\frac{\sigma^2}{2\pi}} [e^{-\frac{m_k^2}{2\sigma^2}} - e^{-\frac{m_{k-1}^2}{2\sigma^2}}]$$

(149)

Note: trick to clear accounts.

$$g(x) = e^{-\frac{x^2}{2\sigma^2}}\ and\ \frac{dg(x)}{dx} = -\frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}} \tag{150}$$

Denominator:

$$\int_{m_k}^{m_{k-1}} f(x)dx = qfunc(\tfrac{m_k}{\sigma}) - qfunc(\tfrac{m_{k-1}}{\sigma}) \tag{151}$$

$$= \int_{m_k}^{m_{k-1}} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} dx \tag{152}$$

$$= qfunc(\tfrac{m_k}{\sigma}) - qfunc(\tfrac{m_{k-1}}{\sigma}) \tag{153}$$

C Trellis state diagram

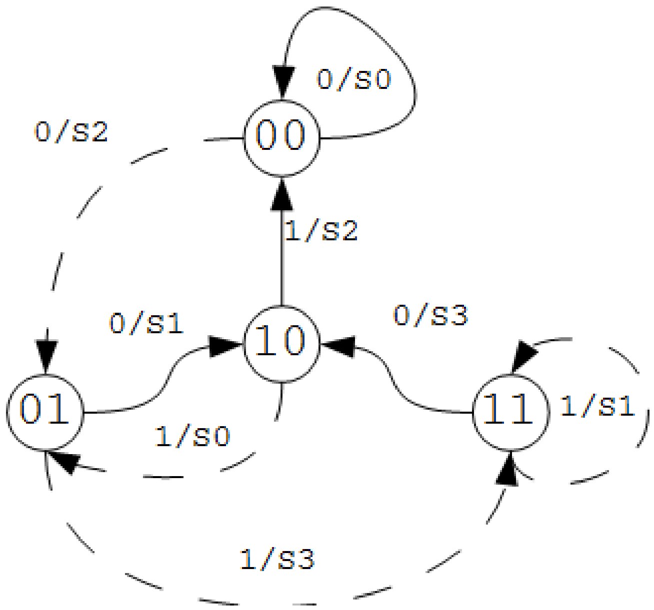

Figure 14: State diagram

START STATE	END STATE	CODE	COSET
0	0	0	S0
1	2	0	S1
2	0	0	S2
3	2	0	S3
0	1	1	S2
1	3	1	S3
2	1	1	S0
3	3	1	S1

D Viterbi algorithm

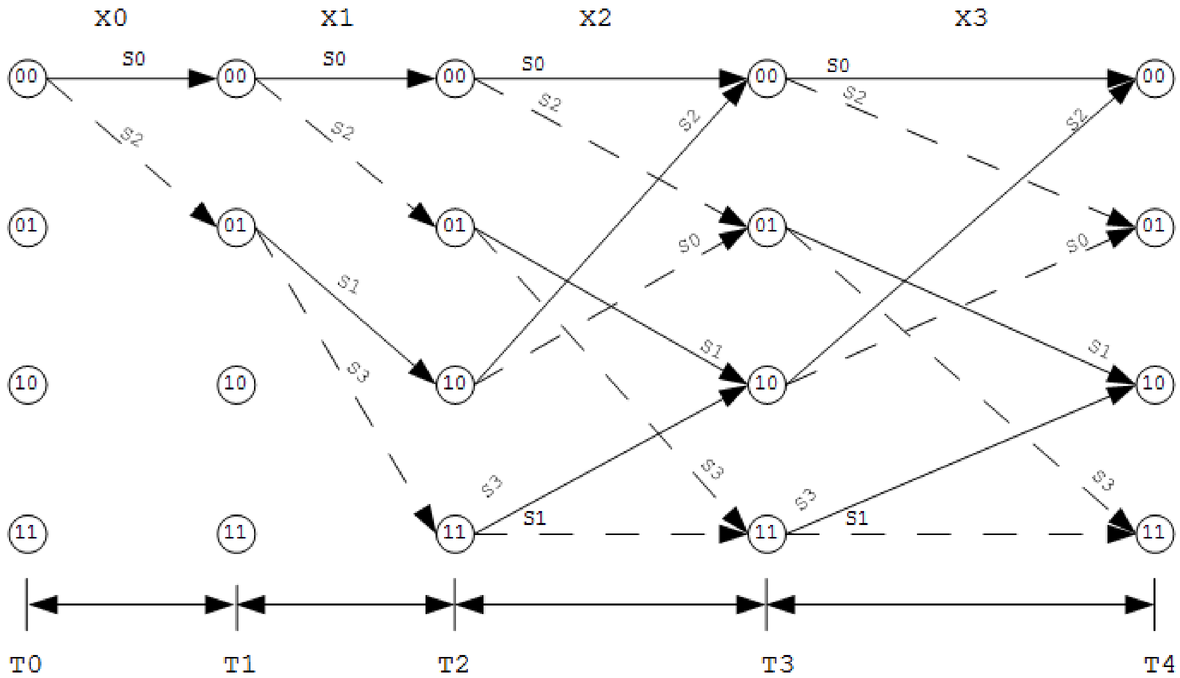

Figure 15: Representation of the transitions along the different iterations of trellis diagram. See figure ([3]).

Tabla 10: Description of the solution adopted for the implementation of the Viterbi algorithm.

T	1		2		index-i		
ST	METRIC	CODE	METRIC	CODE	METRIC		CODE
00	X1,S1	0	M1.00+X2,S0	C1.00&'0'	MIN	$M_{i-1}.00 + X_i, S0$	$C_{i-1}.00\&'0'$
						$M_{i-1}.10 + X_i, S2$	$C_{i-1}.10\&'0'$
01	X1,S2	1	M1.00+X2,S2	C1.00&'1'	MIN	$M_{i-1}.00 + X_i, S2$	$C_{i-1}.00\&'1'$
						$M_{i-1}.10 + X_i, S0$	$C_{i-1}.10\&'1'$
10	NONE	NONE	M1.01+X2,S1	C1.01&'0'	MIN	$M_{i-1}.01 + X_i, S1$	$C_{i-1}.01\&'0'$
						$M_{i-1}.11 + X_i, S3$	$C_{i-1}.11\&'0'$
11	NONE	NONE	M1.01+X2,S3	C1.01&'1'	MIN	$M_{i-1}.01 + X_i, S3$	$C_{i-1}.01\&'1'$
						$M_{i-1}.11 + X_i, S1$	$C_{i-1}.11\&'1'$

E MATLAB CODE

E.1 TCQ TEST

```
%%
```

```
% @file: main.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: TRELLIS CODED QUANTIZER TEST.
%%
clearvars;

%% DEFINE PARAMATERS.
tcq_rate = 1:2;
frame_length = 10000;

% RESULTS.
tcq = zeros(length(tcq_rate),1);

% MONTECARLO.
MC = 100;
err_q = zeros(MC,1);

% VARIANCES.
variance_x = 1;

for index_rate = 1:length(tcq_rate)
    %% LLOYD-MAX ALGORITHM.
    [~,levels] = lloyd_max_quantizer(variance_x,tcq_rate(index_rate)+1);

    %% SET PARTITIONING.
    coset = set_partitioning(levels);

    tic;
    for index_mc = 1:MC

        %% GENERATE X AND Y.
        x = sqrt(variance_x)*randn(1,frame_length);

        %% TRELLIS CODE QUANTIZER
        w_tcq = tcq_encoder(coset,x);
        err_q(index_mc) = mse_calculation(x,w_tcq);

    end
```

```
    tcq(index_rate) = mean(err_q);
    toc;
end

%% RESULTS.
tcq = 10*log10(tcq);

for index = 1: length(tcq_rate)
    fprintf('TCQ_RATE %d -> SNR: %f dB\n', tcq_rate(index), tcq(index));
end
```

E.2 CASE 1: CODING AND DECODING OF X WHEN Y IS KNOWN AT RECEPTION.

```
 %%
% @file: main.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: DISTRIBUTED SOURCE CODING USING SYNDROMES.
%         CASE 1: CODING AND DECODING OF X WHEN Y IS KNOWN AT RECEPTION.
%%
clearvars;

%% DEFINE PARAMATERS.
% NOTE: LLOYD-MAX 8 LEVELS.
%       TCQ RATE 2 BITS/SAMPLE.
%       RATE 1 BIT/SAMPLE.
quantizer = 3;
frame_length = 1000;
SNR = -10:4:30;

% RESULTS.
dist = zeros(length(SNR),1);

% MONTECARLO.
MC = 100;
dist_mc = zeros(MC,1);

% VARIANCES.
```

```
variance_x = 1;
variance_n = variance_x./(10.^(SNR/10));

%% LLOYD-MAX ALGORITHM.
[~,levels] = lloyd_max_quantizer(variance_x,quantizer);

%% SET PARTITIONING.
coset = set_partitioning(levels);

for index_snr = 1:length(SNR)
    tic;
    for index_MC = 1:MC

        %% GENERATE X AND Y.
        x = sqrt(variance_x)*randn(1,frame_length);
        n = sqrt(variance_n(index_snr))*randn(1,frame_length);
        y = x + n;

        %% TRELLIS CODE QUANTIZER
        [frame,w_tcq] = tcq_encoder1(coset,x);

        %% DECODER.
        w = decoder_side_info(coset,frame,y);

        %% QUANTIZATION NOISE.
        variance_q = mse_calculation(x, w_tcq);

        %% ESTIMATION.
        x_est = estimator_case1(y,w,variance_n(index_snr),variance_q);

        %% EQUATIONS.
        dist_mc(index_MC) = mse_calculation(x,x_est);

    end
    dist(index_snr) = mean(dist_mc);
    toc;
end

%% RESULTS.
```

```
dist = 10*log10(dist/variance_x);

figure(1)
hold on;
plot(SNR,dist);
legend('1 bit/sample');
hold off;
xlabel('Correlation-SNR in dB');
ylabel('Average normalized distortion in dB');
title('CASE 1: CODING AND DECODING OF X WHEN Y IS KNOWN AT RECEPTION.');
```

E.3 CASE 2: CODING AND DECODING OF X WHEN Y IS INDEPENDENT.

```
%%
% @file: main.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: DISTRIBUTED SOURCE CODING USING SYNDROMES.
%         CASE 2: CODING AND DECODING OF X WHEN Y IS INDEPENDENT.
%%
clearvars;

%% DEFINE PARAMATERS.
% NOTE: LLOYD-MAX 4:8 LEVELS.
%       TCQ RATE 1:2 BITS/SAMPLE.
%       RATE 1:2 BIT/SAMPLE.
rate = [1 2];
quantizer = rate + 1;
frame_length = 1000;
SNR = [-10 30];

% RESULTS.
dist = zeros(length(rate),1);

% MONTECARLO.
MC = 100;
dist_mc = zeros(MC,1);
```

```
% VARIANCES.
variance_x = 1;

for index_rate = 1:length(rate)
    %% LLOYD-MAX ALGORITHM.
    [~,levels] = lloyd_max_quantizer(variance_x,quantizer(index_rate));

    %% SET PARTITIONING.
    coset = set_partitioning(levels);

    tic;
    for index_MC = 1:MC

        %% GENERATE X AND Y.
        x = sqrt(variance_x)*randn(1,frame_length);

        switch index_rate
            case 1
                %% TRELLIS CODE QUANTIZER
                [frame,w_tcq] = tcq_encoder1(coset,x);

                %% DECODER.
                w = decoder1(coset,frame);
            case 2
                %% TRELLIS CODE QUANTIZER
                [frame,w_tcq] = tcq_encoder2(coset,x);

                %% DECODER.
                w = decoder2(coset,frame);
        end

        %% ESTIMATION.
        x_est = w;

        %% EQUATIONS.
        dist_mc(index_MC) = mse_calculation(x,x_est);

    end
    dist(index_rate) = mean(dist_mc);
```

```
    toc;
end

%% RESULTS.
dist = 10*log10(dist/variance_x);

rate1 = [dist(1) dist(1)];
rate2 = [dist(2) dist(2)];

figure(1)
hold on;
plot(SNR,rate1);
plot(SNR,rate2)
legend('1 bit/sample','2 bit/sample');
hold off;
xlabel('Correlation-SNR in dB');
ylabel('Average normalized distortion in dB');
title('CASE 2: CODING AND DECODING OF X WHEN Y IS INDEPENDENT.');
```

E.4 CASE 3: CODING AND DECODING OF X WHEN Y IS KNOWN AT BOTH SIDES.

```
%%
% @file: main.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: DISTRIBUTED SOURCE CODING USING SYNDROMES.
%         CASE 3: CODING AND DECODING OF X WHEN Y IS KNOWN AT BOTH SIDES.
%%
clearvars;

%% DEFINE PARAMATERS.
% NOTE: LLOYD-MAX 4 LEVELS.
%       TCQ RATE 1 BITS/SAMPLE.
%       RATE 1 BIT/SAMPLE.
quantizer = 2;
frame_length = 1000;
SNR = -10:4:30;
```

```
% RESULTS.
dist = zeros(length(SNR),1);

% MONTECARLO.
MC = 100;
dist_mc = zeros(MC,1);

% VARIANCES.
variance_x = 1;
variance_n = variance_x./(10.^(SNR/10));

for index_snr = 1:length(SNR)

    %% LLOYD-MAX ALGORITHM.
    [~,levels] = lloyd_max_quantizer(variance_n(index_snr),quantizer);

    %% SET PARTITIONING.
    coset = set_partitioning(levels);

    tic;
    for index_MC = 1:MC

        %% GENERATE X AND Y.
        x = sqrt(variance_x)*randn(1,frame_length);
        n = sqrt(variance_n(index_snr))*randn(1,frame_length);
        y = x + n;

        %% TRELLIS CODE QUANTIZER
        [frame,w_tcq] = tcq_encoder1(coset,n);

        %% DECODER.
        w = decoder1(coset,frame);

        %% QUANTIZATION NOISE.
        variance_q = mse_calculation(n,w_tcq);

        %% ESTIMATION.
        x_est = estimator_case3(y,w,variance_x,variance_q);
```

```
        %% EQUATIONS.
        dist_mc(index_MC) = mse_calculation(x,x_est);

    end
    dist(index_snr) = mean(dist_mc);
    toc;
end

%% RESULTS.
dist = 10*log10(dist/variance_x);

figure(1)
hold on;
plot(SNR,dist);
legend('1 bit/sample');
hold off;
xlabel('Correlation-SNR in dB');
ylabel('Average normalized distortion in dB');
title('CASE 3: CODING AND DECODING OF X WHEN Y IS KNOWN AT BOTH SIDES.');
```

E.5 CASE 4: ESTIMATION OF X EXCLUSIVELY BASED ON Y.

```
%%
% @file: main.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: DISTRIBUTED SOURCE CODING USING SYNDROMES.
%         CASE 4: ESTIMATION OF X EXCLUSIVELY BASED ON Y.
%%
clearvars;

%% DEFINE PARAMATERS.
frame_length = 1000;
SNR = -10:4:30;

% RESULTS.
dist = zeros(length(SNR),1);

% MONTECARLO.
```

```
MC = 100;
dist_mc = zeros(MC,1);

% VARIANCES.
variance_x = 1;
variance_n = variance_x./(10.^(SNR/10));

for index_snr = 1:length(SNR)
    tic;
    for index_MC = 1:MC

        %% GENERATE X AND Y.
        x = sqrt(variance_x)*randn(1,frame_length);
        n = sqrt(variance_n(index_snr))*randn(1,frame_length);
        y = x + n;

        %% ESTIMATION.
        x_est = estimator_case4(y,variance_x,variance_n(index_snr));

        %% EQUATIONS.
        dist_mc(index_MC) = mse_calculation(x,x_est);

    end
    dist(index_snr) = mean(dist_mc);
    toc;
end

%% RESULTS.
dist = 10*log10(dist/variance_x);

figure(1)
hold on;
plot(SNR,dist);
hold off;
xlabel('Correlation-SNR in dB');
ylabel('Average normalized distortion in dB');
title('CASE 4: ESTIMATION OF X EXCLUSIVELY BASED ON Y.');
```

E.6 FUNCTIONS

E.6.1 lloyd_max_quantizer

```
%%
% @file: lloyd_max_quantizer.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: ITS OBTAINS OPTIMAL QUANTIZER, MINIMIZE THE MEAN SQUARED
% QUANTIZER ERROR.
%%
function [intervals,levels] = lloyd_max_quantizer(variance,n)

    % INITIALIZATION.
    intervals = [-inf,-6:12/(2^n-2):6,+inf]*sqrt(variance);
    levels = zeros(1,2^n);

    % LLOYD-MAX ALGORITHM
    for iter=1:10000
        % REPRESENTATIVE LEVELS.
        for i=1:length(levels)
            levels(i)=(sqrt(variance/(2*pi))*(exp(-(intervals(i)^2)/(2*variance))-exp(-(inte
        end
        % DECISION THRESHOLDS.
        for i=2:length(levels)
            intervals(i) = 1/2 *(levels(i) + levels(i-1));
        end
    end

end
```

E.6.2 set_partitioning

```
%%
% @function: set_partitioning.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: COSETS ASSOCIATED TO CENTROIDS(LEVELS).
% EXAMPLE: S0{D0,D4},S1{D1,D5},S2{D2,D6},S3{D3,D7}
%%
```

E.6.3 tcq_encoder

```
%%
% @file: tcq_encoder.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: TRELLIS CODE QUANTIZER.
%%
function [w_tcq] = tcq_encoder(coset,input)
```

E.6.4 tcq_encoder1

```
%%
% @file: tcq_encoder1.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: TRELLIS CODE QUANTIZER(1bit).
%%
```

E.6.5 tcq_encoder2

```
%%
% @file: tcq_encoder2.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: TRELLIS CODE QUANTIZER(2bit).
%%
```

E.6.6 decoder_side_info

```
%%
% @function: decoder_side_info.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: CONVOLUTIONAL CODE + SIDE-INFORMATION.
%%
```

E.6.7 decoder1

```
%%
% @function: decoder1.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: CONVOLUTIONAL CODE(1bit).
%%
```

E.6.8 decoder2

```
%%
% @function: decoder2.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: CONVOLUTIONAL CODE(2bit).
%%
```

E.6.9 mse_calculation

```
%
% @function: mse_calculation.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: MEAN SQUARED ERROR.
%%
```

E.6.10 estimator_case1

```
%%
% @function: estimator_case1.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: CODING AND DECODING OF X WHEN Y IS KNOWN AT RECEPTION.
%%
```

E.6.11 estimator_case3

```
%%
```

```
% @function: estimator_case3.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: CODING AND DECODING OF X WHEN Y IS KNONW AT BOTH SIDES.
%%
```

E.6.12 estimator_case4

```
%%
% @function: estimator_case4.m
% @author: BLANCO CAAMANO, RAMON.
%
% @about: ESTIMATION OF X EXCLUSIVELY BASED ON Y.
%%
```