WAVEFORM CODING TECHNIQUES

CONTENTS

REFERENCES:

N. S. JAYANT AND P. NOLL,

DIGITAL CODING OF WAVEFORMS

ENGLEWOOD CLIFFS, NJ: PRENTICE-HALL, INC., 1989.

1. INTRODUCTION

In waveform coding, we try our best to approximate the waveform of the input signal using some distortion measure (失真测度) such as mean squared error.

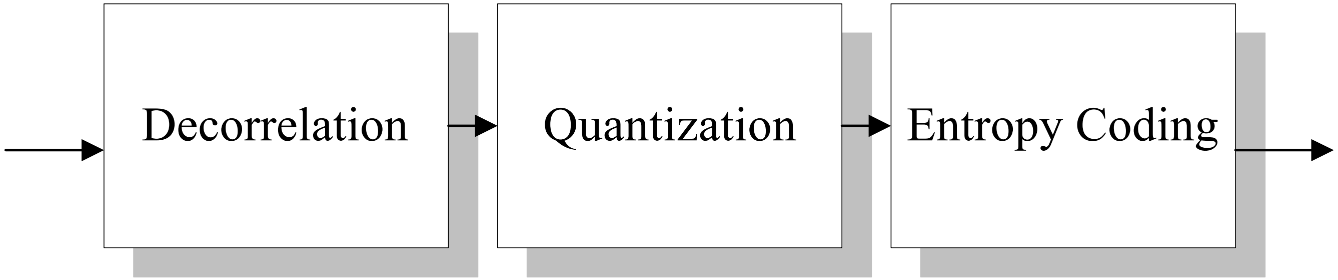


Fig. 1.1 General structure of a waveform coding system.

Decorrelation

Decorrelation techniques such as transformation or prediction are frequently used to reduce the redundancy (e.g. statistical) in the input signal.

Quantization

The output of the decorrelator is usually of small amplitude. Significant compression can be achieved by quantizing the signal to small number of discrete symbols (Lossy).

Entropy Coding

The symbols are then losslessly encoded using entropy coding (Huffman, arithmetic codes, etc).

2. QUANTIZATION TECHNIQUES

Quantization (such as A/D convertion) is the process of transforming a given (signal) value into a finite set of possible output values.

- **The *L*-level scalar quantizer *Q* is a mapping**

$$Q : R \to C, \qquad \text{(2-1)}$$

which maps the real line R to a finite set of reconstruction values or representation levels, y_k, called the codebook:

$$C = \{y_1, y_2, ..., y_L\} \subset R. \qquad \text{(2-2)}$$

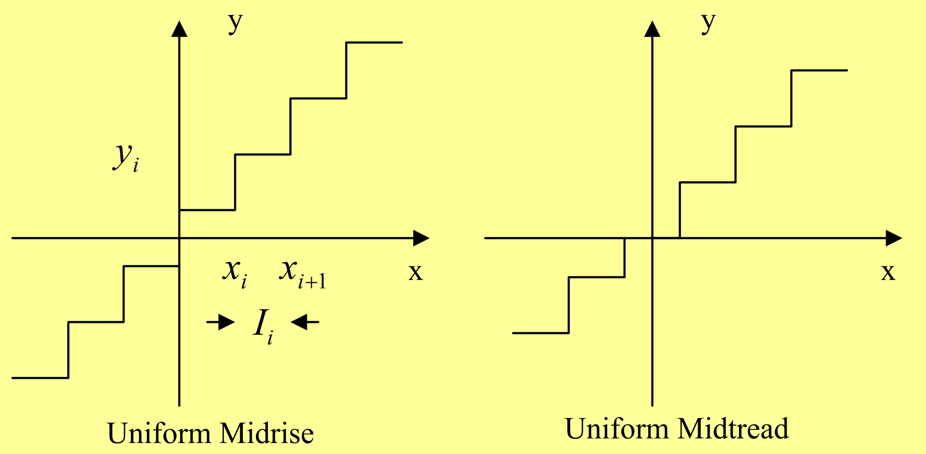


Fig 2.1 Midrise and Midtread Quantizers.

The quantizer partitions R into L cells, I_k,

$$I_k = \{x \in (x_k, x_{k+1}] : Q(x) = y_k\} \quad k = 1,2,..., L. \qquad \text{(2-3)}$$

x_k **are called the decision levels.**

- **The *L*-ary number *k* is transmitted to the receiver. If $L = 2^R$, then a *bit rate* of**

$$R = \log_2 L \quad \text{bits/samples}$$

is needed. At the receiver, y_k is recovered from the index using a lookup table.

- **The quantizer characteristic consists of *N treads*, or horizontal steps, with a *riser* or vertical segment jointing successive levels. For a *symmetric quantizer* and with *L* even, we obtain a *mid-rise* quantizer while for *L* odd, we obtain a *mid-tread* quantizer.**
- **Expression for quantizer output,**

$$Q(x) = \sum_{i=1}^{L} y_i S_i(x) \quad \text{where} \quad S_i(x) = \begin{cases} 1 & x \in I_i \\ 0 & otherwise \end{cases}. \qquad \text{(2-4)}$$

Quantization Error

Quantization error is the error between input x and output y:

$$q(x) = x - Q(x). \tag{2-5}$$

Common performance measure of quantizer are mean squared error (MSE) and mean absolute error (MAE).

$$E[q^2(x)] \quad \text{(MSE)}$$

$$E[|q(x)|] \quad \text{(MAE)}$$

Variance of Quantization Error

If *x* is a zero-mean random variable with variance σ_x^2 and probability density function (pdf) $p_x(.)$, the quantization error is also a random variable, $q(x)$, with pdf $p_q(.)$ and variance σ_q^2:

$$\sigma_q^2 = E[q^2(x)] = \int_{-\infty}^{\infty} [x - Q(x)]^2 p_x(x) dx. \tag{2-6}$$

Breaking up the integration into L intervals I_k, we obtain:

$$\sigma_q^2 = \sum_{k=1}^{L} \int_{x_k}^{x_{k+1}} (x - y_k)^2 p_x(x) dx. \tag{2-7}$$

The performance of the quantizer is measured by the quantizer performance factor,

$$\varepsilon_q^2 = \sigma_q^2 / \sigma_x^2. \tag{2-8}$$

$\left(\varepsilon_q^2\right)^{-1}$ is the signal to quantization noise ratio (SNR):

信号比噪声 $$SNR(dB) = 10 \cdot \log_{10}(\sigma_x^2 / \sigma_q^2) = 10 \cdot \log_{10}(1/\varepsilon_q^2). \tag{2-9}$$

$$\sigma_q^2 = \varepsilon_*^2 2^{-2R} \sigma_x^2. \tag{2-10}$$

It is possible to compute the necessary bit rate for a given SNR as,

$$R = \frac{1}{2}\log_2(\sigma_x^2/\sigma_q^2) + \frac{1}{2}\log_2 \varepsilon_*^2 \qquad \textbf{bits/sample.} \qquad \textbf{(2-11)}$$

ε_*^2 are either determined analytically or numerically during the design of the quantizer.

Granular Noise and Overload Distortion

- When the signal lies in the bounded cells of the quantizer, the quantization error is usually small and is referred to as the granular noise.
- For input signal that lies in the unbounded cells of the quantizer, large distortion called overload distortion is experienced. x_{ol} is called the overload amplitude above which overload will occur.

The trade-off between granular and overload noise is measured by the loading factor, f_l, typically in the range of 2-4:

$$f_l = x_{ol} / \sigma_x . \tag{2-12}$$

QUANTIZATION DESIGN

The Optimal Encoder for a given Decoder

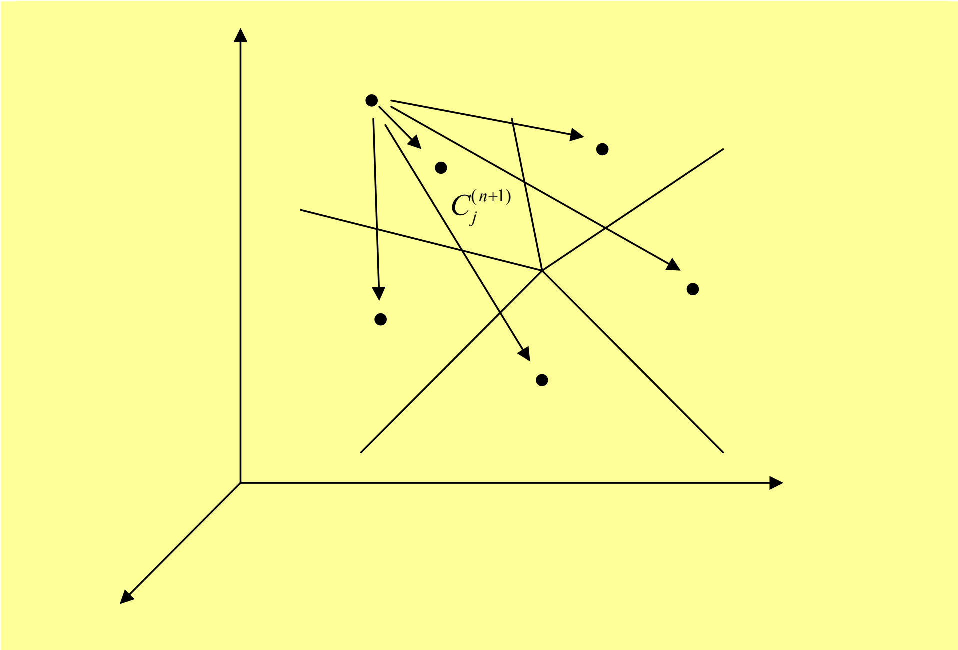
The best encoder for a given codebook satisfies the nearest neighbor condition which requires that the ith region of the partition should consist of all input value closer to y_i than to any other output level.

Proof: For a given codebook, *C*, *Q(x)* takes on values in *C*, so that

$$D = \int d(x, Q(x))\, p_X(x)dx.$$

$$\geq \int [\min_{y_i \in C} d(x, y_i)] f_X(x)dx \qquad \textbf{(2-13)}$$

and this lower bound is indeed attained when *Q(x)* performs the nearest neighbor mapping with the given codebook *C*.

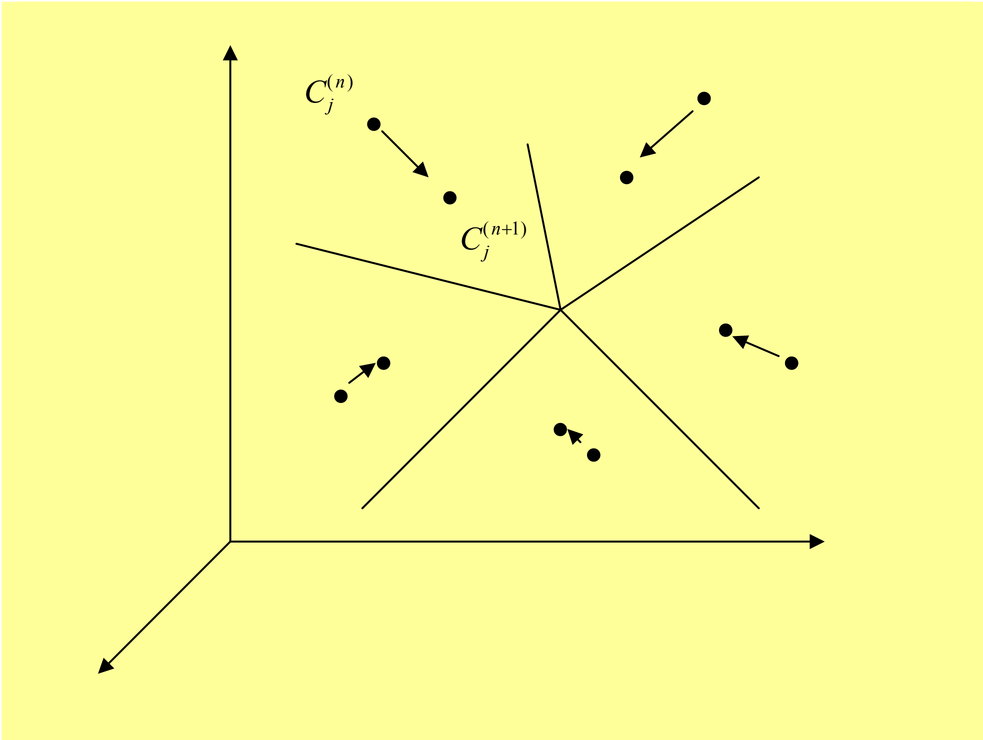


Optimal Decoder for a given Encoder

Given a nondegenerate partition $\{I_k\}$, the unique optimal codebook for a random variable X wrt the mean squared error is given by:

$$y_k = E[X \mid X \in I_k]. \qquad \text{(2-14)}$$

This is known as the centroid condition.



The Lloyd Quantizer Design Algorithm

Step 1: Begin with an initial codebook C_1.

Step 2: Given the codebook, C_m, perform the Lloyd iteration to generate an improved Codebook, C_{m+1}:

Lloyd iteration:

- If the pdf of the distribution is known, then we can find the optimal partition into cells using nearest neighbor condition from the given codebook C_m:

  $$R_i = \{x : d(x, y_i) \le d(x, y_j); j \ne i\}.$$

  Using the centroid condition, find the improved codebook C_{m+1}.

- If the pdf is unknown, a large set of observations are taken known as the *training set*. The training data are them clustered (quantized) by the codebook C_m. The centroid of the clustered data are then computed to obtain the optimal reconstruction levels $y_i s$ and hence C_{m+1}.

Step 3: Compute the average distortion for C_{m+1}. If it has changed by a small enough amount since the last iteration, stop. Otherwise set $m+1 \rightarrow m$ and go to step 2. A common stopping criteria is to test if the fractional drop in the distortion, $(D_m - D_{m+1})/D_m$, is below or above a suitable threshold.

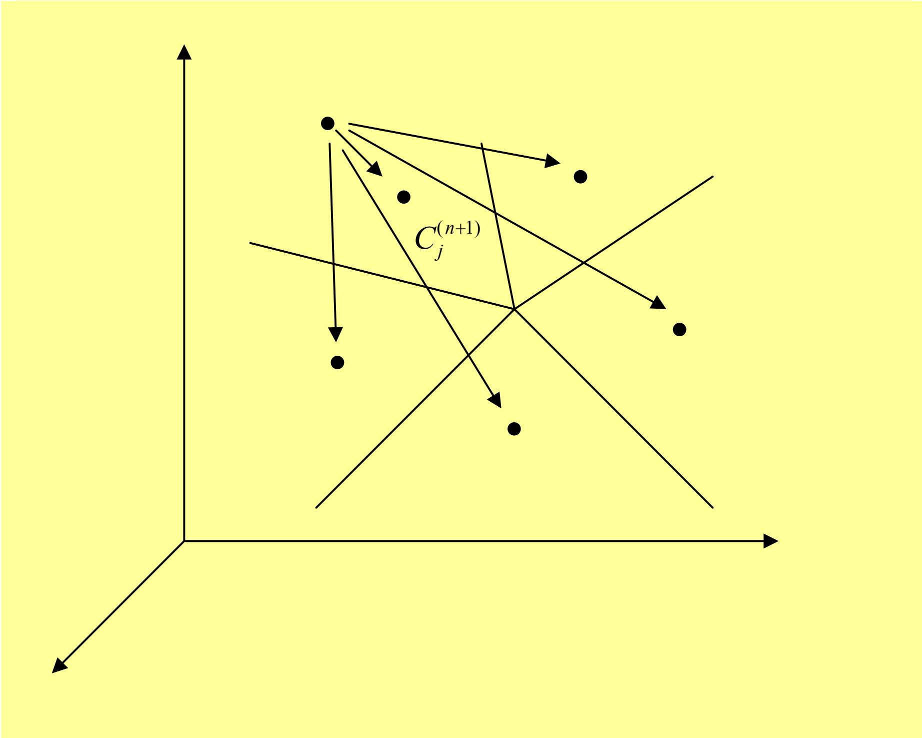


Figure 2.2 Nearest Neighbor Rule

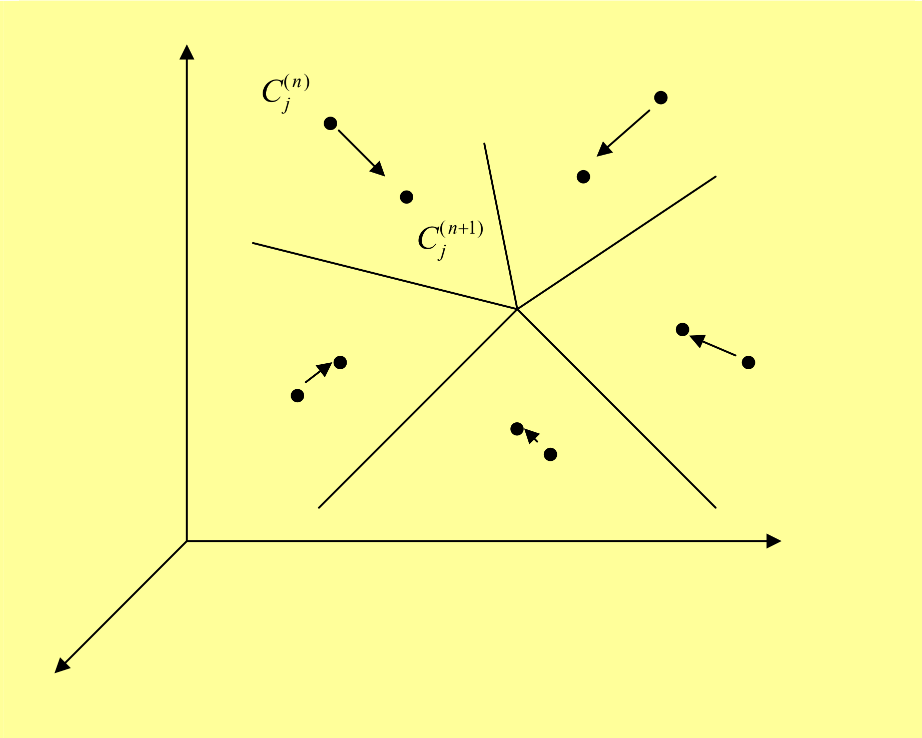


Figure 2.3 Codebook Generation using GLA

3. DECORRELATION TECHNIQUES

Subband Coding

Subband and transform coding decompose the input signal into a number of frequency components for encoding.

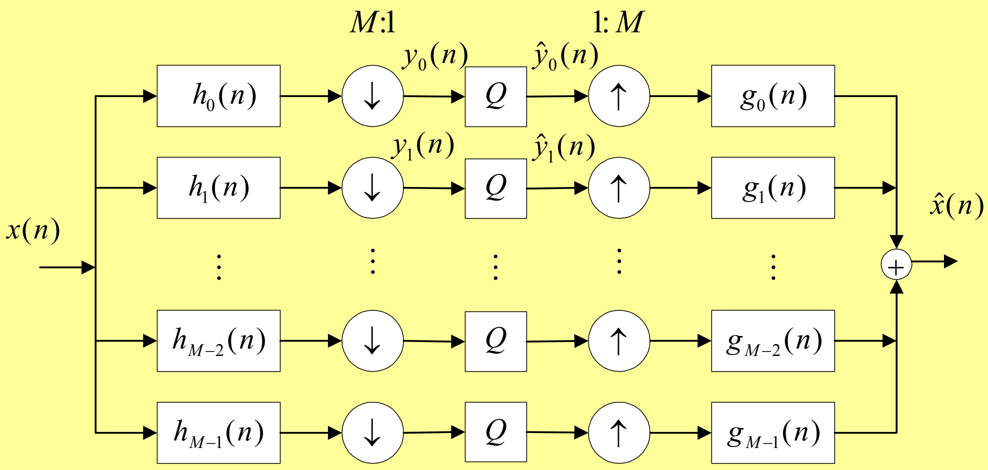


Fig 3.1 Analysis-synthesis system (or filterbank)

Advantages:

- provide arbitrary forms of noise shaping.
- the eye and the ear, seem to operate in a transform domain. It is easier to incorporate psychoacoustic or pyschovisual models into the codec.

- The input signal is first filtered by a set of frequency selective analysis filters, $h_i(n)$, i=1,...,*M* .
- The filtered signal are nearly bandlimited to π / M . It can be downsampled (decimated) by a factor of *M* (critical sampling).
- The total number of samples after downsampling are preserved.

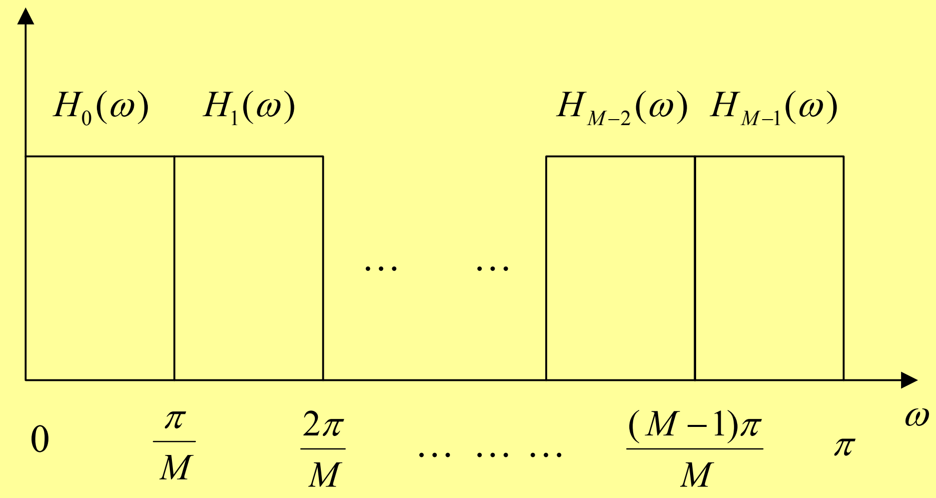


Fig 3.2 Frequency Responses of analysis and synthesis filters.

- The subband coefficients, $y_i(n)$, are quantized and transmitted to the receiver. At the receiver, the quantized coefficients, $\hat{y}_i(n)$, are upsampled, interpolated by the synthesis *filters*, $g_i(n)$, and summed to yield the reconstructed signal, $\hat{x}(n)$.

Perfect reconstruction filterbanks

- Due to downsampling, aliasing will occur.
- $h_i(n)$ and $g_i(n)$ can be designed to perfectly reconstruct $x(n)$ up a certain delay n_d. That is

$$\hat{x}(n) = x(n - n_d), \tag{3-1}$$

  in absence of quantization.
- Important issues are the design of the filterbanks, adaptive quantization procedures, and the incorporation of the pychoacoustic or pyschovisual models.
- Commonly used FBs are either tree-structured or with a uniform bandwidth. Nonuniform FBs are difficult to design.

Coding Gain

- In case of *M*-channel uniform PR filter bank, it can be shown that for identical quantizers for each subband, and idea filters that

  the maximum coding gain of the M-channel FB is the ratio of arithmetic and geometric mean of the subband variance $\sigma_{y_k}^2$:

$$\max\{G_{SBC}\} = \frac{\sigma_x^2}{M\left[\prod_{k=1}^{M}\sigma_{y_k}^2\right]^{1/M}} = \frac{\frac{1}{M}\sum_{k=1}^{M}\sigma_{y_k}^2}{\left[\prod_{k=1}^{M}\sigma_{y_k}^2\right]^{1/M}}. \qquad \text{(3-2)}$$

  Note that $\sigma_{x_k}^2 = \sigma_{y_k}^2$ and $\sigma_{\hat{x}_k}^2 = \sigma_{\hat{y}_k}^2$. The same result is also true when the filter bank is paraunitary.

Transform Coding

- **Transform coding is a special case of subband coding. The analysis and synthesis filters are of length *M*.**
- **A block of *M* input samples $\boldsymbol{x} = \{x(0),\ldots,x(M-1)\}$ is linearly transformed by a matrix *A* into a set of *M* transformed coefficients $\boldsymbol{X} = \{X(0),\ldots,X(M-1)\}$:**

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{x}. \tag{3-3}$$

Matrix *A* is usually an orthogonal matrix satisfying:

$$\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}_M. \tag{3-4}$$

- **The idea is to pack the input signal energy into fewer dimensions (energy compaction). Transform coefficients are then adaptively quantized to achieve compression.**

The Optimal Transformation (KLT)

The coding gain is maximized when the geometric mean is minimized.

Such optimal transform is called the Karhunen Loeve Transform (KLT).

$$\boldsymbol{R}_{XX} = E[\boldsymbol{X}\boldsymbol{X}^T]$$ **(transformed correlation matrix)**

$$\boldsymbol{R}_{xx} = E[\boldsymbol{x}\boldsymbol{x}^T]$$ **(correlation matrix)**

From matrix theory, we have for any matrix *B*:

$$\left|\boldsymbol{B}\right| \le \prod_{i=1}^{M} b(i,i). \qquad \text{(3-5)}$$

For an orthogonal matrix *A*,

$$\left|\boldsymbol{R}_{XX}\right| = \left|\boldsymbol{A}\boldsymbol{R}_{xx}\boldsymbol{A}^T\right| = \left|\boldsymbol{R}_{xx}\right|\left|\boldsymbol{A}\boldsymbol{A}^T\right| = \left|\boldsymbol{R}_{xx}\right|. \qquad \text{(3-6)}$$

The *kth* diagonal entries of $\boldsymbol{R}_{xx}$ is equal to $\sigma_{x_k}^2$. Using (3-5) and (3-6) we obtain

$$\prod_{k=0}^{M-1} \sigma_{x_k}^2 \geq |\boldsymbol{R}_{XX}| = |\boldsymbol{R}_{xx}|. \tag{3-7}$$

The geometric mean of coefficient variances is minimum when the equality holds in (3-7).

The eigenvalues λ_k, $k = 1,\ldots,M-1$, and eigenvectors $\boldsymbol{l}_k$, $k = 1,\ldots,M-1$, of matrix $\boldsymbol{R}_{xx}$ is defined by:

$$\boldsymbol{R}_{xx}\boldsymbol{l}_k = \lambda_k \boldsymbol{l}_k. \tag{3-8}$$

If $\boldsymbol{R}_{xx}$ is real symmetric, the eigenvalues are real and there are exactly *M* orthonormal eigenvectors:

$$\boldsymbol{l}_k^T \boldsymbol{l}_l = \delta_{kl}. \tag{3-9}$$

The *KLT*, $\boldsymbol{L}$, is defined as

$$\boldsymbol{L} = \left[\boldsymbol{l}_0^T \quad \boldsymbol{l}_1^T \quad \cdots \quad \boldsymbol{l}_{M-1}^T\right]^T. \tag{3-10}$$

The corresponding $\boldsymbol{R}_{xx}$ is diagonalized as,

$$\boldsymbol{R}_{XX} = \boldsymbol{L}\boldsymbol{R}_{xx}\boldsymbol{L}^T = \begin{bmatrix} \boldsymbol{l}_0^T \\ \boldsymbol{l}_1^T \\ . \\ \boldsymbol{l}_{M-1}^T \end{bmatrix} [\lambda_0 \boldsymbol{l}_0 \quad \lambda_1 \boldsymbol{l}_1 \quad . \quad \lambda \boldsymbol{l}_{M-1}] = \begin{bmatrix} \lambda_0 & & & \\ & \lambda_1 & 0 & \\ & 0 & . & \\ & & & \lambda_{M-1} \end{bmatrix}.$$

As $\boldsymbol{R}_{XX}$ is diagonal, the equality in (3-7) is achieved by the *KLT*.

- **The KLT is data dependent. It is very time-consuming to estimate it in real-time (solving an eigenvalue problem). In coding applications, suboptimal transform like the discrete cosine transform (DCT) is used.**

Linear Prediction (LPC) and modeling

$$y(n) = \sum_{k=1}^{P} a_k x(n-k) \quad \textbf{(3-11)}$$

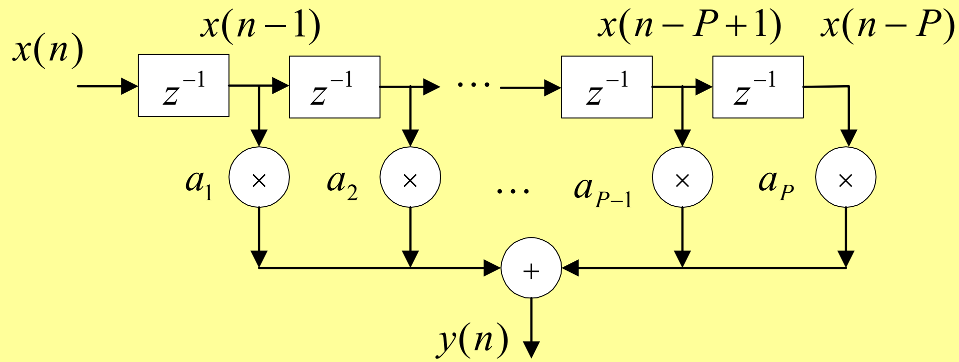


Fig 3.3 *P-th* order Linear Predictor

- **For linear prediction, y(n), which is a linear combination of the previous inputs, is used to predict the current input x(n). a_k's are called the linear prediction coefficients.**
- **In general, a_k's can be determined so that y(n) approximates a desired response d(n), which is a function of x(n)'s. Applications includes)**

To determine a_k's, the mean square error (MSE) criterion is usually minimized:

$$MSE = E[\varepsilon_n^2] = E[(x(n) - y(n))^2]$$

$$= E[(x(n) - \sum_{\ell=1}^{P} a_i x(n-k))^2]$$

$$= E\left[x^2(n)\right] - 2\sum_{k=1}^{P} a_k E\left[x(n)x(n-k)\right] + \sum_{k=1}^{P} a_k \sum_{j=1}^{P} a_j E\left[x(n-k)x(n-j)\right]. \qquad \textbf{(3-12)}$$

$$= r_{xx}(0) - 2\boldsymbol{a}^T \boldsymbol{r}_{xx} + 2\boldsymbol{a}^T \boldsymbol{R}_{xx} \boldsymbol{a}$$

where

$\boldsymbol{R}_{xx} = E[\boldsymbol{x}\boldsymbol{x}^T]$ **is the autocorrelation matrix of the input signal,**

$\boldsymbol{r}_{xx} = E\left[x(n)\boldsymbol{x}\right]$ **is the autocorrelation vector of the input signals.**

$\boldsymbol{a}^T = \left[a_1, \quad a_2, \quad \ldots, \quad a_{L-1}\right]$, $\boldsymbol{x}^T = \left[x(n-1) \quad x(n-2), \quad \ldots, \quad x(n-P)\right]$.

The gradient of the mean squared error function with respect to the filter coefficient vector is given by:

$$\frac{\partial}{\partial \boldsymbol{a}} E\left[e^2(n)\right] = -2\boldsymbol{r}_{xx}^T + 2\boldsymbol{a}^T \boldsymbol{R}_{xx}, \qquad \textbf{(3-13)}$$

where the gradient vector is defined as

$$\frac{\partial}{\partial \boldsymbol{a}} = \left[\frac{\partial}{\partial a_1}, \quad \frac{\partial}{\partial a_2}, \quad \frac{\partial}{\partial a_3}, \quad \ldots, \quad \frac{\partial}{\partial a_P},\right]^T.$$

The minimum is obtained by setting the gradient to zero, which yields

$$\boldsymbol{R}_{xx}\boldsymbol{a} = \boldsymbol{r}_{xx}. \qquad \textbf{(3-14)}$$

or equivalently,

$$\boldsymbol{a} = (\boldsymbol{R}_{xx})^{-1}\boldsymbol{r}_{xx}. \qquad \textbf{(3-15)}$$

(3-15) in matrix form is,

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_P \end{bmatrix} = \begin{bmatrix} r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \cdots & r_{xx}(P-1) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(P-2) \\ r_{xx}(2) & r_{xx}(1) & & \cdots & r_{xx}(L-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{xx}(L-1) & r_{xx}(L-2) & r_{xx}(L-3) & \cdots & r_{xx}(0) \end{bmatrix}^{-1} \begin{bmatrix} r_{xx}(1) \\ r_{xx}(2) \\ r_{xx}(3) \\ \vdots \\ r_{xx}(P) \end{bmatrix}.$$

Predictive quantization

The best minimum mean square error (mmse) predictor, $\hat{x}(n)$, based on previous input samples, $x(n-1), x(n-2),...$. is the conditional expectation:

$$\hat{x}(n) = E[x(n)|x(n-1), X(n-2),...]. \qquad \textbf{(3-16)}$$

- **This predictor is impractical because:**
  1. **the conditional pdf in (3-16) is generally unavailable**
  2. **the receiver only has the quantized values but not the original values of $x(n)$.**
- **Practical predictors are linear functions of x(n)'s.**

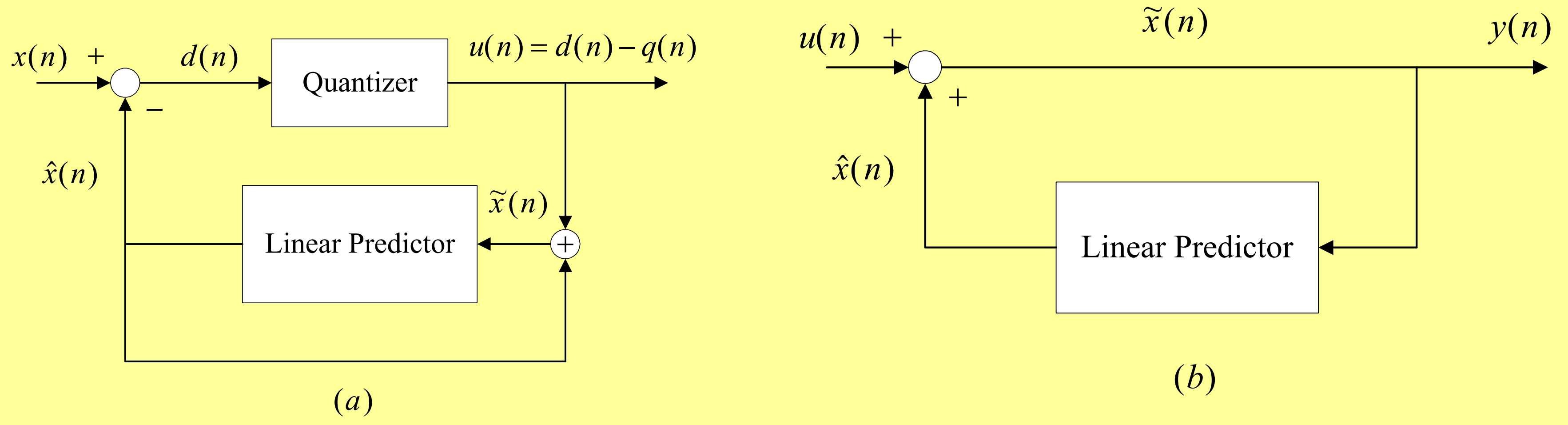


Fig. 3.4 Block diagram of differential PCM (DPCM) (a) Encoder (b) Decoder

$$\hat{x}(n) = \sum_{j=1}^{N} h_j \tilde{x}(n-j). \quad \textbf{(All pole prediction)} \qquad \textbf{(3-17)}$$

$$\hat{x}(n) = \sum_{j=1}^{N} g_j u(n-j). \quad \textbf{(All zero prediction)} \qquad \textbf{(3-18)}$$

The basic equations describing DPCM are:

$$d(n) = x(n) - \hat{x}(n). \tag{3-19a}$$

$$u(n) = d(n) - q(n). \tag{3-19b}$$

$$y(n) = \hat{x}(n) + v(n). \tag{3-19c}$$

where: *y(n)* is the DPCM approximation to coder input *x(n).*

***d(n)* is the prediction error, *q(n)* is the quantization error,**

***u(n)* is the quantized prediction error,**

and *v(n)* is the receiver copy of *u(n)*.

For error free transmission, *v(n)* = *u(n)*.

Forward predictor

$$\tilde{x}(n) = f[x(n-1), x(n-2), \ldots]. \qquad \text{(3-20)}$$

Use previous data samples to computer the predictor.

Better prediction gain but requires longer coding delay and side information for predictor coefficients (the decoder does not have the true samples, hence it is necessary to transmit the coefficients to the decoder).

Backward predictor

$$\hat{x}(n) = f[y(n-1), y(n-2), \ldots]. \qquad \text{(3-21)}$$

Use previous encoded samples to compute the predictor.

No need to send predictor coefficients, low-coding delay (no need to buffer original data samples), but sensitive to quantization error and lower prediction gain.

Slope overload and granular noise

- Slope overload occurs whenever the largest step size of the quantizer is too small to follow a fast changing input.
- Granular noise occurs when input variations are too small in comparison with the smallest step size of the quantizer.

Signal to noise ratio and prediction gain

The SNR gain of DPCM over PCM is

$$\sigma_{rP}^2 / \sigma_{rD}^2 = (\varepsilon_{qP}^2 \sigma_x^2)/(\varepsilon_{qD}^2 \sigma_d^2) = (\varepsilon_{qP}^2 / \varepsilon_{qD}^2)(\sigma_x^2 / \sigma_d^2). \qquad (3\text{-}22)$$

where subscripts *P* and *D* denote *PCM* and *DPCM*, respectively.

- **The term $(\varepsilon_{qP}^2/\varepsilon_{qD}^2)$ is in general not equal to unity. But for speech signal, Gaussian sources and logarithmic quantization with large number of levels, it is close to unity. In these cases, (3-22) is simplified to:**

$$\sigma_{rP}^2/\sigma_{rD}^2 = \sigma_x^2/\sigma_d^2 = G_P.$$

The prediction gain G_P is typically greater than 1.

- **In all pole backward predictor of order *N*, we have:**

$$\hat{x}(n) = \sum_{j=1}^{N} h_j y(n-j) = \sum_{j=1}^{N} h_j [x(n-j) - q(n-j)]. \tag{3-23}$$

It is difficult to analyze this predictor due to the noise feedback. But under the assumption of fine quantization, *q(n)* is small and (3-23) is approximated as follows: $\hat{x}(n) \approx \sum_{j=1}^{N} h_j x(n-j)$.