

CECS401
Fundamentals of Spoken Language Processing

Note-7
Tuesday 9/21/99

C. Linear predictive coding (LPC) of speech

Alternative representations of LPC parameters

Optimal representations depend on specific applications of speech compression, synthesis, and recognition.

1. LPC parameters $\{a_1, a_2, \dots, a_p\}$
 - relative small changes in the prediction coefficients could result in large changes in the pole positions of the vocal tract filter model,
 - dynamic range of prediction coefficients is large.

2. Roots of the predictor polynomial $\{z_1, z_2, \dots, z_p\}$

$$A(z) = 1 - \sum_{k=1}^p a_k z^{-k} = \prod_{k=1}^p (1 - z_k z^{-1})$$

—from the roots $z_i = z_{ir} + jz_{ii} = e^{s_i T}$ with $s_i = \sigma_i + j\Omega_i$, the parameters of resonant frequencies can be solved as

$$\Omega_i = \frac{1}{T} \tan^{-1} \frac{z_{ii}}{z_{ir}}$$

$$\sigma_i = \frac{1}{2T} \log (z_{ir}^2 + z_{ii}^2)$$

- the parameters $\{(\Omega_i, \sigma_i), i = 1, 2, \dots, p\}$ are useful for formant analysis,
- the stability of vocal tract filter can be guaranteed by restricting the roots to lie within the unit-circle
- the solution of roots is computational expensive.

3. Reflection coefficients or Partial-correlation coefficients (PARCOR) $\{k_1, k_2, \dots, k_p\}$

—computed from Durbin's method

$$k_i = \left\{ r(i) - \sum_{j=1}^{i-1} a_j^{(i-1)} r(|i-j|) \right\} / E^{(i-1)}$$

or the Lattice method

$$k_i = \frac{\sum_{m=0}^{N-1} e^{(i-1)}(m) b^{(i-1)}(m-1)}{\left[\sum_{m=0}^{N-1} (e^{(i-1)}(m))^2 \sum_{m=0}^{N-1} (b^{(i-1)}(m-1))^2 \right]^{1/2}}$$

- dynamic range is limited to $|k_i| \leq 1$
- stability of the vocal tract filter is guaranteed.
- when k_i 's are close to ± 1 , small quantization errors results in large spectral distortions.

4. Log area ratio $\{g_1, g_2, \dots, g_p\}$
 — g_i 's are obtained from a nonlinear transform of k_i 's, i.e.,
- $$g_i = \log \frac{1 - k_i}{1 + k_i} \quad i = 1, 2, \dots, p$$
- the transform expands the scale near $k_i = \pm 1$, so that an equivalent nonuniform quantizer on k_i 's is achieved using a uniform quantizer on the transformed coefficients g_i 's.
 — define A_i and A_{i+1} to be two consecutive areas in the equal-length lossless tube model, then

$$g_i = \log \frac{A_{i+1}}{A_i} \quad i = 1, 2, \dots, p$$

$$k_i = \frac{A_i - A_{i+1}}{A_i + A_{i+1}} = \frac{1 - e^{g_i}}{1 + e^{g_i}} \quad i = 1, 2, \dots, p$$

Example of uniform scalar quantization

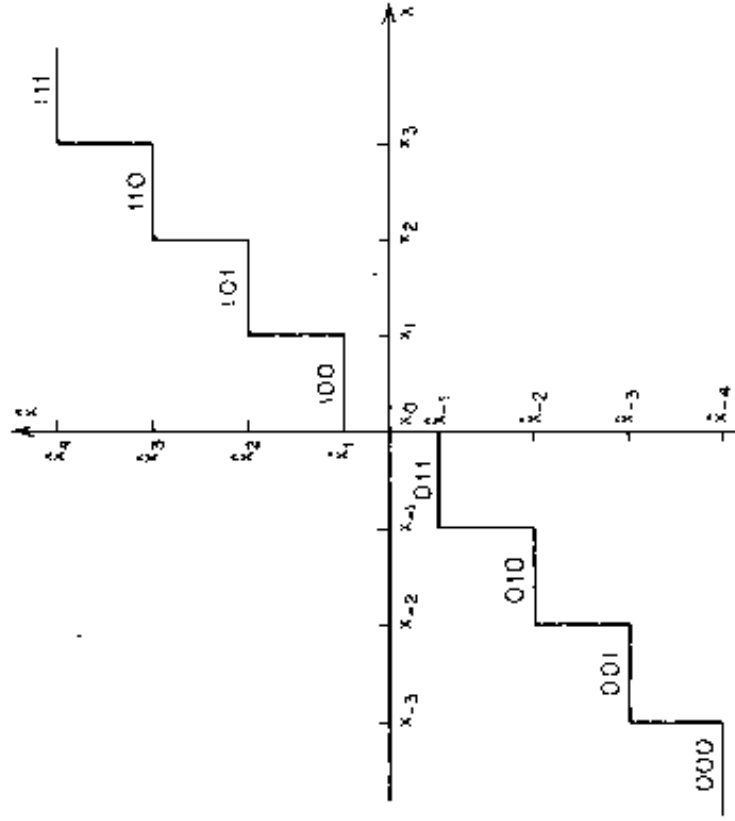
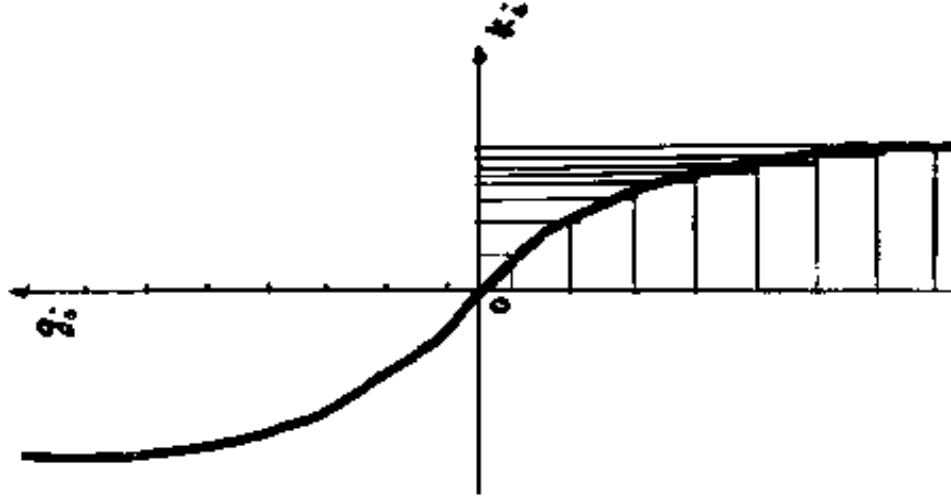


Fig. 5.9 Input-output characteristic of a 3-bit quantizer.

Illustration of uniform quantization on a log area ratio parameter and the equivalent nonuniform quantization on the PARCOR parameter.



5. LPC cepstral coefficients $\{c_1, c_2, \dots, c_p, \dots\}$
- log power spectrum compresses the dynamic range of linear spectrum and is a useful representation for speech recognition, cepstral coefficients are parameters representing log spectrum.
 - define LPC power spectrum and log spectrum as

$$S(\omega) = \frac{\sigma^2}{|A(e^{j\omega})|^2} = \frac{\sigma^2}{|A(z)|^2} \Big|_{z=e^{j\omega}}$$

$$\log S(\omega) = \sum_{n=-\infty}^{\infty} c_n e^{-jn\omega}$$

- c_i 's are called cepstral coefficients.
- when the zeros of $A(z)$ are all inside the unit circle, the following equation holds by Laurent expansion:

$$\log \frac{1}{1 - \sum_{k=1}^p a_k z^{-k}} = \sum_{n=1}^{\infty} c_n z^{-n}$$

—by differentiating both sides of the equation w.r.t. z^{-1} , one can derive the equation

$$\sum_{k=1}^p k a_k z^{-k+1} = \left(1 - \sum_{k=1}^p a_k z^{-k} \right) \sum_{n=1}^{\infty} n c_n z^{-n+1}$$

—equating the coefficients of like powers of z^{-1} leads to the solution

$$\begin{aligned} c_1 &= a_1 \\ c_n &= \sum_{k=1}^{n-1} \left(1 - \frac{k}{n} \right) a_k c_{n-k} + a_n \quad 1 < n \leq p \\ c_n &= \sum_{k=1}^{n-1} \left(1 - \frac{k}{n} \right) a_k c_{n-k} \quad n > p \end{aligned}$$

—cepstral coefficients are symmetric, i.e. $c_n = c_n$ and $c_0 = \log \sigma^2$

—LPC coefficients can be computed from cepstral coefficients as

$$a_n = c_n - \sum_{k=1}^{n-1} \frac{k}{n} c_k a_{n-k} \quad 1 \leq n \leq p$$

—statistical properties of cepstral coefficients can be approximated

as $E[c_n] = 0$ and $E[c_n^2] = \frac{1}{n^2}$

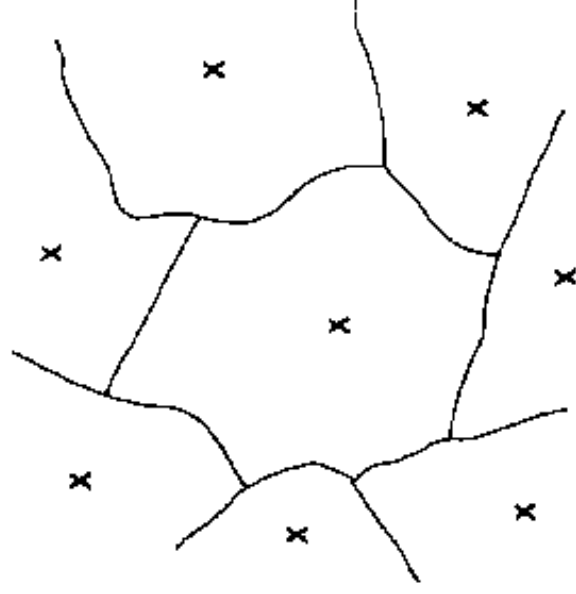
—the cepstral sequence for representing a log spectrum is normally truncated as $\{c_1, c_2, \dots, c_Q\}$ with

$$p \leq Q \leq \frac{3}{2}p$$

D. Vector Quantization

Vector quantization (VQ) is an efficient source coding technique: signal vectors are encoded by a small set of representative vectors called codewords, and the collection of codewords is called the codebook of the vector quantizer.

Example: partition of a vector space into VQ cells with each cell represented by a codeword



Example: assuming speech is sampled at 10 KHz,

—using a short-time 10th order LPC analysis with a frame shift of 10 ms, each LPC parameter is coded by 7 bits, leads to

$$\textit{bit rate} = 100 \times 10 \times 7 = 7,000 \textit{ bits per second (bps)}$$

—using 25 codewords per phoneme for 40 English phonemes, leads to

$$\textit{codebook size} = 25 \times 40 = 1000 < 2^{10}$$

$$\rightarrow 10 \textit{ bits/codeword}$$

$$\rightarrow \textit{bit rate} = 100 \times 10 = 1,000 \textit{ bps}$$

Procedure of VQ training and classification

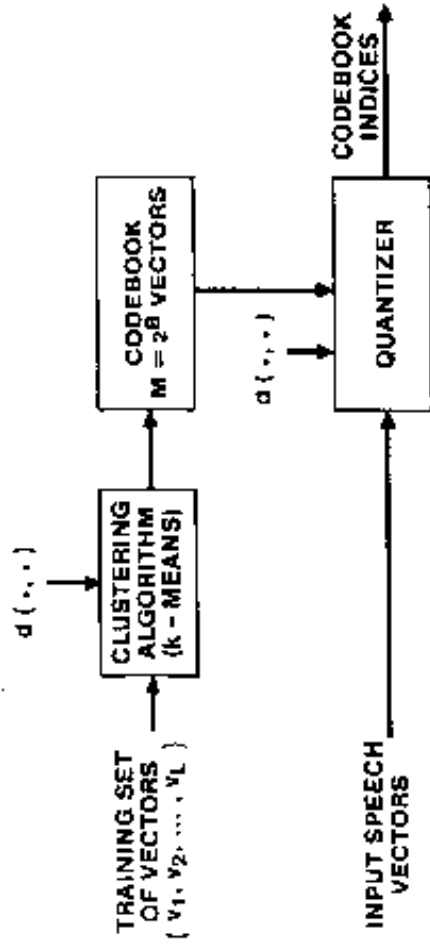


Figure 3.40 Block diagram of the basic VQ training and classification structure.

Quantization

Assume available the codebook $C = \{y_1, y_2, \dots, y_M\}$. The quantization of a vector v_t uses the nearest-neighbor decision:

$$\hat{v}_t = \arg \min_{y_i \in C} d(v_t, y_i)$$

$d(v_t, y_i)$ satisfies the property of distance measure:

$$\begin{aligned} d(v_t, y_i) &= 0 && \text{if } v_t = y_i \\ d(v_t, y_i) &> 0 && \text{otherwise} \end{aligned}$$

e.g., $d(v_t, y_i) = \|v_t - y_i\|^2$

The total distortion incurred from quantizing v_1, v_2, \dots, v_L by $\hat{v}_1, \hat{v}_2, \dots, \hat{v}_L$ is

$$D = \frac{1}{L} \sum_{t=1}^L d(v_t, \hat{v}_t)$$

Criterion for optimal codebook design

The designed codebook $C = \{y_1, y_2, \dots, y_M\}$ should minimize the total distortion D , i.e.,

$$C = \arg \min_{C'} D(C')$$

or

$$\{y_1, y_2, \dots, y_M\} = \arg \min_{\{y'_1, y'_2, \dots, y'_M\}} D(y'_1, y'_2, \dots, y'_M)$$

In practice, often only suboptimal codebooks are attainable.