Lecture 12: Predictive Coding of Speech at Low Bit Rates

Mark Hasegawa-Johnson

ECE 537: Speech Processing, Fall 2022

Outline

Review: LPC

Inverse filter:

$$H(z) = \frac{C_1}{1 - p_1 z^{-1}} + \frac{C_1^*}{1 - p_1^* z^{-1}}$$
$$h[n] = C_1 p_1^n u[n] + C_1^* (p_1^*)^n u[n]$$

Orthogonality principle: a_k minimizes

$$\sum_{n=p+1}^{N+p} e^2[n] = \sum_{n=p+1}^{N+p} \left(s[n] - \sum_{m=1}^{p} a_m s[n-m] \right)^2$$

if and only if $e[n] \perp s[n-k]$, meaning

$$\sum_{n=p+1}^{N+p} e[n] s[n-k] = 0$$

p linear equations in p unknowns:

$$\vec{c} = \Phi \vec{a}$$

Outline

Solving for the LPC Coefficients: The Standard Covariance Method

$$\Phi\vec{a} = \vec{c}$$

1. **Cholesky decomposition:** Since Φ is symmetric, find lower-triangular L s.t. $\Phi = LL^T$. (An $\mathcal{O}\{p^3\}$ operation, but it can be done iteratively).
2. **Parcor coefficients:** Find $\vec{q}$ s.t. $L\vec{q} = \vec{c}$. (An $\mathcal{O}\{p^2\}$ operation that can be done iteratively).
3. **Predictor coefficients:** Find $\vec{a}$ s.t. $\vec{q} = L^T\vec{a}$ (An $\mathcal{O}\{p^2\}$ operation that can't be done iteratively).

Covariance Method: The one-step predictor

Consider trying to solve for the first-order predictor that minimizes $\sum e^2[n]$:

$$e[n] = s[n] - a_1^{(1)} s[n-1]$$

This is done by solving $\Phi\vec{a} = \vec{c}$ where

$$\Phi = \phi(1,1), \quad \vec{a} = a_1^{(1)}, \quad \vec{c} = \phi(0,1)$$

Thus

$$a_1^{(1)} = \frac{\phi(0,1)}{\phi(1,1)} = \frac{\sum_{n=p+1}^{N+p} s[n]s[n-1]}{\sum_{n=p+1}^{N+p} s^2[n-1]}$$

In other words, the first-order predictor is the one-step correlation, divided by the signal energy.

Covariance Method: Iterative Approach

Suppose that we've solved for the $(m-1)$-step predictor, and we want to find the m-step predictor:

$$\begin{bmatrix} \phi(1,1) & \cdots & \phi(1,m-1) \\ \vdots & \ddots & \vdots \\ \phi(1,m-1) & \cdots & \phi(m-1,m-1) \end{bmatrix} \begin{bmatrix} a_1^{(m-1)} \\ \vdots \\ a_{m-1}^{(m-1)} \end{bmatrix} = \begin{bmatrix} \phi(0,1) \\ \vdots \\ \phi(0,m-1) \end{bmatrix}$$

$$\begin{bmatrix} \phi(1,1) & \cdots & \phi(1,m) \\ \vdots & \ddots & \vdots \\ \phi(1,m)) & \cdots & \phi(m,m) \end{bmatrix} \begin{bmatrix} a_1^{(m)} \\ \vdots \\ a_m^{(m)} \end{bmatrix} = \begin{bmatrix} \phi(0,1) \\ \vdots \\ \phi(0,m) \end{bmatrix}$$

Notice that Φ and $\vec{c}$ just add one row to the previous step, but that's not true about $\vec{a}$!

Covariance Method: Iterative Approach

To find the m^{th}-order predictor coefficients, $\vec{a}_m = [a_1^{(m)}, \ldots, a_m^{(m)}]^T$, we need to solve

$$\Phi_m \vec{a}_m = \vec{c}_m,$$

where

$$\Phi_m = \begin{bmatrix} \Phi_{m-1} & \vec{\phi}_m \\ \vec{\phi}_m^T & \phi(m,m) \end{bmatrix}, \quad \vec{a}_m = \begin{bmatrix} a_1^{(m)} \\ \vdots \\ a_m^{(m)} \end{bmatrix}, \quad \vec{c}_m = \begin{bmatrix} \vec{c}_{m-1} \\ \phi(0,m) \end{bmatrix}$$

Covariance Method: Iterative Approach, Step 1, Cholesky Decomposition

Cholesky Decomposition: find L_m such that $\Phi_m = L_m L_m^T$.

$$\begin{bmatrix} \Phi_{m-1} & \vec{\phi}_m \\ \vec{\phi}_m^T & \phi(m,m) \end{bmatrix} = \begin{bmatrix} L_{m-1} & \vec{0} \\ \vec{\ell}_m^T & L_{m,m} \end{bmatrix} \begin{bmatrix} L_{m-1} & \vec{\ell}_m \\ \vec{0} & L_{m,m} \end{bmatrix}$$

Notice that if you multiply the first row by the first column, you get

$$\Phi_{m-1} = L_{m-1} L_{m-1}^T$$

. . . In other words, the first $(m-1)$ rows of L_m are the solution from the previous step! So L can be computed iteratively, row by row.

Covariance Method: Iterative Approach, Step 2, Parcor Coefficients

Parcor coefficients: find $\vec{q}_m$ s.t. $L_m\vec{q}_m = \vec{c}_m$:

$$\left[\begin{array}{cc} L_{m-1} & \vec{0} \\ \vec{\ell}_m^T & L_{m,m} \end{array}\right]\left[\begin{array}{c} \vec{q}_{m-1} \\ q_m \end{array}\right] = \left[\begin{array}{c} \vec{c}_{m-1} \\ \phi(0,m) \end{array}\right]$$

Notice that if you multiply the first $m-1$ rows by the first $m-1$ columns, you get

$$L_{m-1}\vec{q}_{m-1} = \vec{c}_{m-1}$$

... In other words, the first $(m-1)$ elements of $\vec{q}_m$ are the solution from the previous step! So $\vec{q}$ can be computed iteratively, row by row.

New Concept: The Partial Correlation (parcor) Coefficients

In some sense, the coefficient q_m (the last element in the $\vec{q}_m$ vector) is the "new information" added at step m of this process. The **partial correlation** coefficients, k_i, usually called the **parcor** coefficients, are the "new information," scaled by the square root of the residual energy:

$$k_i = \frac{q_i}{\sqrt{\epsilon_i}}$$

New Concept: The Partial Correlation (parcor) Coefficients

Let's be a little more specific. The signal energy is

$$\epsilon_1 = \phi(0,0) = \sum_{n=p+1}^{N+p} s[n]^2$$

The error energy of the $(m-1)$-step predictor is

$$\begin{aligned}\epsilon_m &= \sum_{n=p+1}^{N+p} \left(s[n] - \sum_{i=1}^{m-1} a_i^{(m-1)} s[n-i] \right)^2 \\ &= \epsilon_1 - \sum_{i=1}^{m-1} |q_i|^2,\end{aligned}$$

where the proof of the last line is left for you to do in the homework.

LPC is Stable if and only if Parcor Coefficients are Bounded

The error energy of the $(m-1)$-step predictor is

$$\begin{aligned}\epsilon_m &= \sum_{n=p+1}^{N+p}\left(s[n]-\sum_{i=1}^{m-1}a_i^{(m-1)}s[n-i]\right)^2\\ &= \epsilon_1-\sum_{i=1}^{m-1}|q_i|^2,\end{aligned}$$

It turns out that the LPC filter is stable if and only if, each time we increase the strength of the predictor, the error energy always decreases. In other words,

$$\epsilon_1 > \epsilon_2 > \cdots > \epsilon_{m-1} > \epsilon_m > \cdots 0$$

LPC is Stable if and only if Parcor Coefficients are Bounded

So we have that LPC is stable if and only if

$$\epsilon_1 > \epsilon_2 > \cdots > \epsilon_{m-1} > \epsilon_m > \cdots 0$$

This can be guaranteed if we define $k_i = \frac{q_i}{\sqrt{\epsilon_i}}$, and if we quantize k_i using quantization levels that guarantee $-1 < k_i < 1$. In that case, each of the following lines is smaller than the one before it:

$$\begin{aligned}
\epsilon_2 &= \epsilon_1 - |q_1|^2 = \epsilon_1 - |k_1|^2\epsilon_1 \\
\epsilon_3 &= \epsilon_2 - |q_2|^2 = \epsilon_2 - |k_2|^2\epsilon_2 \\
\epsilon_4 &= \epsilon_3 - |q_3|^2 = \epsilon_3 - |k_3|^2\epsilon_3 \\
&\vdots
\end{aligned}$$

Atal's Modified Covariance Method

So Atal's modified covariance method is like this:

1. Quantize the parcor coefficients, k_i, using a quantizer that only represents reconstruction levels in the range $-1 < k_i < 1$, then
2. Compute the predictor coefficients, a_i, using the bounded parcor coefficients!

Outline

The Periodic Structure of Speech

There are two main types of periodicities in speech:

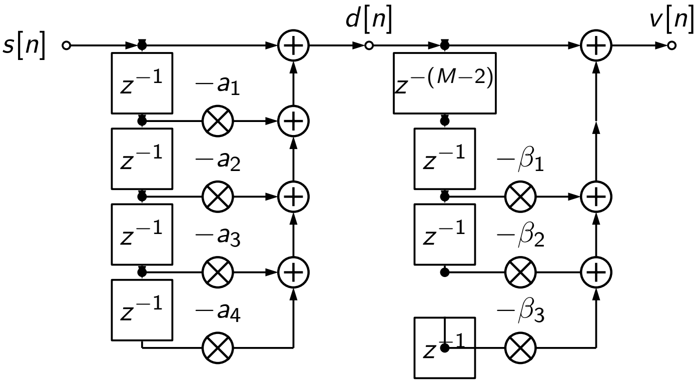
Each **formant resonance** appears as a damped sine wave. The samples of each damped sine wave can be predicted from two previous samples, thus to predict the resonant patterns of 4 formants, you need LPC with $p \approx 8 - 10$ previous samples:

$$d[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k]$$

The **pitch periodicity** shows up as a glottal closure once per T_0. This is not sinusoidal at all! Therefore, to predict, you really need to say that $d[n] \approx d[n-M]$:

$$v[n] = d[n] - \beta_1 d[n-M+1] - \beta_2 d[n-M] - \beta_3 d[n-M-1]$$

Predictive Analysis: LPC Plus Pitch Prediction



Predictive Analysis: LPC Plus Pitch Prediction

The LPC (short-time) predictor can be expressed as follows, where $P_s(z)$ is Atal's notation:

$$D(z) = (1 - P_s(z))\, S(z) = \left(1 - \sum_{k=1}^{p} a_k z^{-k}\right) S(z)$$

The pitch (long-time) predictor can be expressed as follows, where $P_d(z)$ is Atal's notation:

$$V(z) = (1 - P_d(z))\, D(z) = \left(1 - z^{-(M-2)} \sum_{k=1}^{3} \beta_k z^{-k}\right) D(z)$$

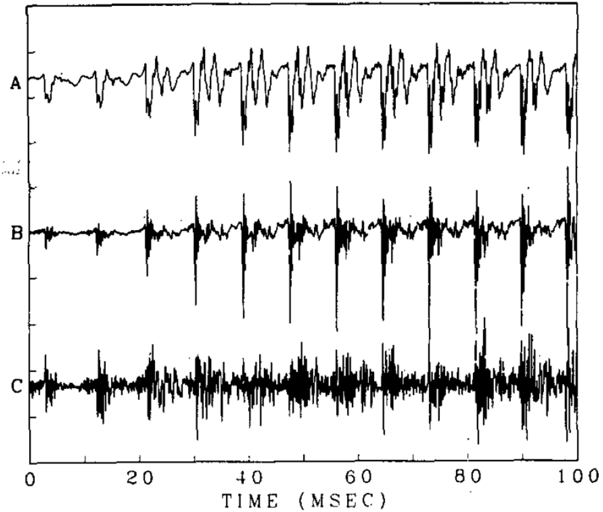


Fig. 6. (A) Speech waveform. (B) Difference signal after prediction based on spectral envelope (amplified 10 dB relative to the speech waveform). (C) Difference signal after prediction based on pitch periodicity (amplified 20 dB relative to the speech waveform).

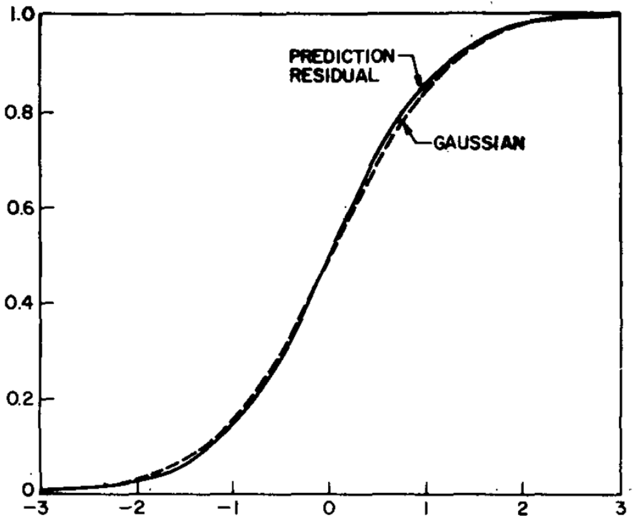
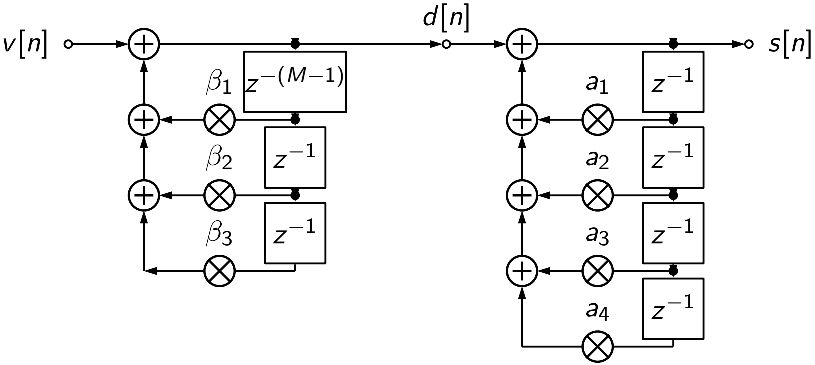


Fig. 7. First-order cumulative amplitude distribution function for the prediction residual samples (solid curve). The corresponding Gaussian distribution function with the same mean and variance is shown by the dashed curve.

Synthesis: LPC + Pitch Prediction



Predictive Synthesis: LPC Plus Pitch Prediction

Pitch predictive synthesis can be expressed as follows, where $P_d(z)$ is Atal's notation:

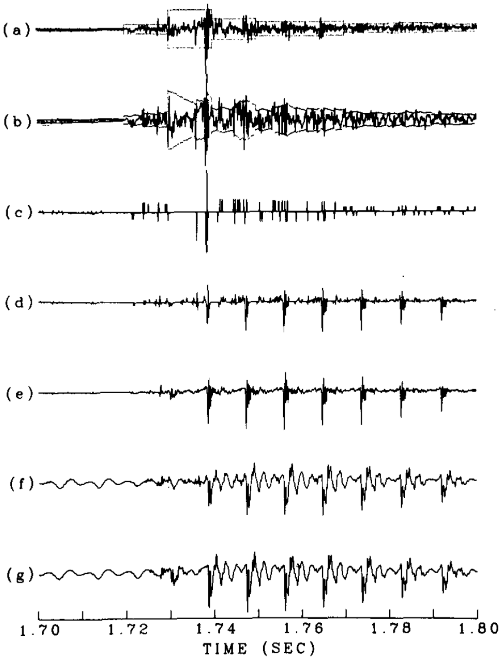
$$\hat{D}(z) = \frac{1}{1 - P_d(z)} \hat{V}(z) = \frac{1}{1 - z^{-(M-2)} \sum_{k=1}^{3} \beta_k z^{-k}} \hat{V}(z)$$

LPC (short-time) predictive synthesis can be expressed as follows, where $P_s(z)$ is Atal's notation:

$$\hat{S}(z) = \frac{1}{1 - P_s(z)} \hat{D}(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} \hat{D}(z)$$

Atal (1982), Figure 18:

(a) Prediction residual, $v[n]$, w/frame-wise center-clipping threshold
(b) Quantizer input w/sample-wise center-clipping threshold
(c) Quantized residual, $\hat{v}[n]$
(d) Reconstructed $\hat{d}[n]$
(e) Original $d[n]$
(f) Reconstructed $\hat{s}[n]$
(g) Original $s[n]$



Outline

Quantization Strategy

Let's define the predictor $P(z)$ as the sequence of both LPC and pitch predictors. We can express that by defining some coefficients α_k in terms of the a_k and β_k, thus:

$$V(z) = (1 - P_d(z))(1 - P_s(z))S(z) = (1 - P(z))S(z)$$

$$v[n] = s[n] - \sum_{k=1}^{M+p+1} \alpha_k s[n-k]$$

Quantization Strategy

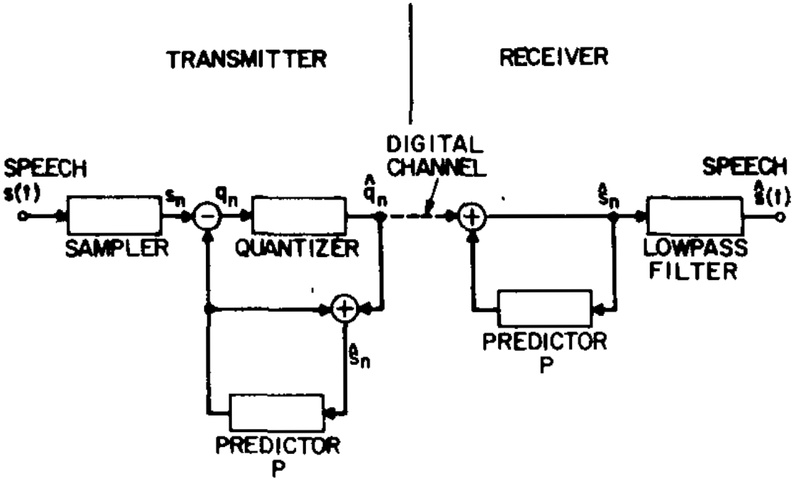
Atal proposed the following quantization strategy.
The **encoder** first constructs the **decoded** signal,

$$\hat{S}(z) = \frac{1}{1 - P(z)}\hat{Q}(z),$$

and then create the following signal, which is the one we should quantize:

$$Q(z) = S(z) - P(z)\hat{S}(z)$$

$$q[n] = s[n] - \sum_{k=1}^{M+p+1} \alpha_k \hat{s}[n-k]$$


TRANSMITTER
RECEIVER
DIGITAL
CHANNEL
SPEECH
s(t)
SAMPLER
s_n
q_n
QUANTIZER
$\hat{q}_n$
$\hat{s}_n$
PREDICTOR
P
LOWPASS
FILTER
SPEECH
$\hat{s}(t)$

What's the Error Spectrum?

Now we have the following strategy:

$$q[n] = s[n] - \sum_{k=1}^{M+p+1} \alpha_k \hat{s}[n-k]$$

$$\hat{q}[n] = q[n] + \epsilon[n]$$

$$\hat{s}[n] = \hat{q}[n] + \sum_{k=1}^{M+p+1} \alpha_k \hat{s}[n-k]$$

$$= s[n] + \epsilon[n],$$

where

- $\epsilon[n]$ is a random error, uniformly distributed between $-\frac{\Delta}{2}$ and $\frac{\Delta}{2}$, where Δ is the quantizer step size.
- If the quantizer step size is small enough, then $\epsilon[n]$ is uncorrelated with $\epsilon[n-m]$.
- In other words, $\epsilon[n]$ is white noise!

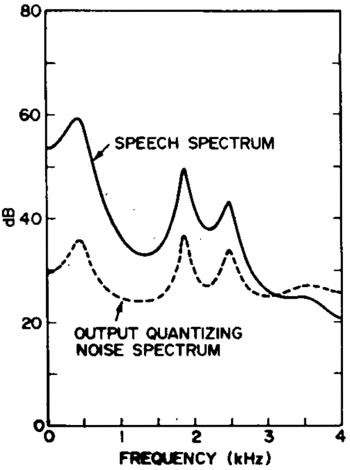
Do We Want White Noise?

Peaks in the speech spectrum can mask noise at the same frequencies.

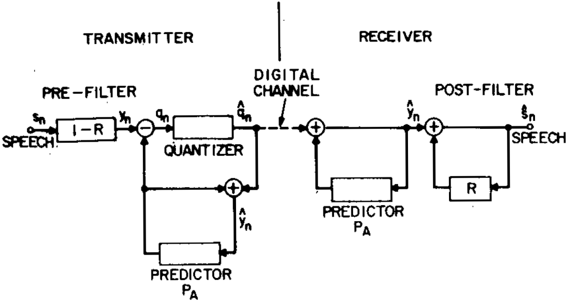
The total amount of noise energy is fixed.

Therefore, it would be better to have a noise spectrum that was shaped so that more of its energy is near the formants.

That way, there would be less noise energy at other frequencies.



The Noise-Shaping Filter



The structure above shapes the noise by $\frac{1}{|1-R(e^{j\omega})|^2}$:

$$Y(z) = (1 - R(z))S(z)$$

$$\hat{y}[n] = y[n] + \epsilon[n]$$

$$\hat{S}(z) - S(z) = \frac{1}{1 - R(z)}\epsilon(z)$$

$$E\left[\left|\hat{S}(e^{j\omega}) - S(e^{j\omega})\right|^2\right] = \left|\frac{1}{1 - R(e^{j\omega})}\right|^2$$

Criteria for a Noise-Shaping Filter

It should have peaks near the speech peaks $\Rightarrow$ it should have the same pole frequencies!

At frequencies far from those peaks, it should be flat $\Rightarrow$ it should be an all-pass filter!

All-Pass Filter: A Pole for Every Zero

The basic idea of an all-pass filter is to have a pole for every zero.

$$H(z) = \frac{1 - rz^{-1}}{1 - pz^{-1}}, \quad |H(\omega)| = \frac{|1 - re^{-j\omega}|}{|1 - pe^{-j\omega}|}$$

and then choose $r = be^{j\omega_c}$ and $p = ae^{j\omega_c}$. If $|b| > |a|$, it's an all-pass filter with a notch at ω_c.

When $\omega = \omega_c$, $|H(\omega)|$ is exactly

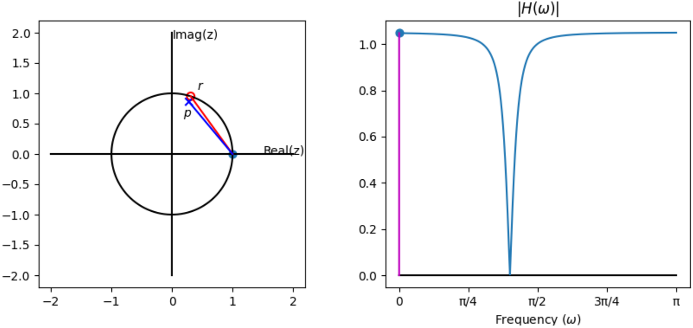
$$\frac{|1 - be^{j(\omega_c - \omega_c)}|}{|1 - ae^{j(\omega_c - \omega_c)}|} = \frac{|1 - b|}{|1 - a|} < 1$$

When $\omega \neq \omega_c$,

$$|e^{j\omega} - r| \approx |e^{j\omega} - p|, \quad \text{so} \quad |H(\omega)| \approx 1$$

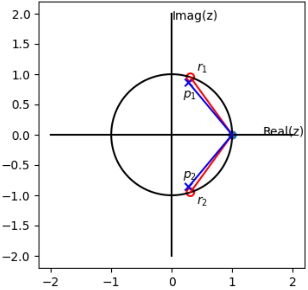
All-Pass Filter: A Pole for Every Zero

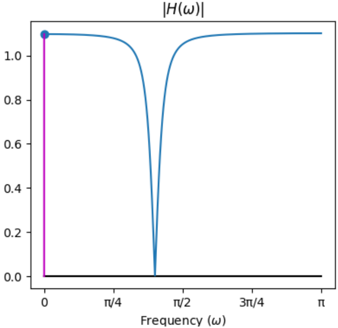
The red line is $|e^{j\omega} - r|$ (distance to the zero on the unit circle). The blue line is $|e^{j\omega} - p|$ (distance to the pole inside the unit circle). They are almost the same length.



All-Pass Filter with Conjugate-Pair Zeros and Poles

$$|H(\omega)| = \frac{|e^{j\omega} - r_1| \times |e^{j\omega} - r_2|}{|e^{j\omega} - p_1| \times |e^{j\omega} - p_2|}$$





All-Pass Noise Shaping Filter

Suppose that the LPC filter is

$$\frac{1}{1-P_A(z)} = \frac{1}{\prod_{k=1}^{p}(1-p_k z^{-1})}$$

We want the noise-shaping filter, $1/(1-R(z))$, to have a notch at every speech formant $\angle p_k$. Thus we want something like

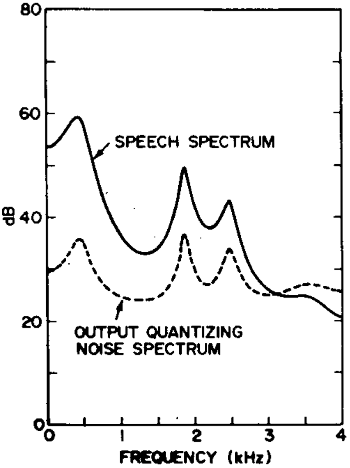
$$1-R(z) = \frac{1-P_A(z)}{1-P_B(z)} = \frac{\prod_{k=1}^{p}(1-p_k z^{-1})}{\prod_{k=1}^{p}(1-\alpha p_k z^{-1})}$$

where the bandwidth-expanding factor α is $0 \leq \alpha \leq 1$.

All-Pass Noise Shaping Filter

Thus the noise spectrum is

$$E\left[\left|\hat{S}(\omega)-S(\omega)\right|^2\right]=\frac{1}{|1-R(\omega)|^2}$$
$$=\frac{\prod_{k=1}^{p}\left|1-\alpha p_k e^{-j\omega}\right|^2}{\prod_{k=1}^{p}\left|1-p_k e^{-j\omega}\right|^2}$$



Outline

Conclusions

Atal's modified covariance-LPC guarantees a stable LPC filter by finding the parcor coefficients,

$$k_m = \frac{q_m}{\sqrt{\epsilon_m}},$$

and then quantizing them with quantization levels such that $|k_m| < 1$.

The pitch predictor turns a Gaussian white-noise-like signal, $v[n]$, into a signal with pitch periodicity.

Noise can be perceptually weighted, with a notch at each formant frequency, so that the quantizer is encouraged to shift noise toward the formants:

$$1 - R(z) = \frac{1 - P_A(z)}{1 - P_B(z)} = \frac{\prod_{i=1}^{p}(1 - p_i z^{-1})}{\prod_{i=1}^{p}(1 - \alpha p_i z^{-1})},$$