LPC Analysis and Synthesis for Speech Synthesis

Arthur R. Toth

March 25, 2004 and April 1, 2004

Overview of Linear Predictive Coding (LPC) (Roughly based on parts of Ch. 8 of “Digital Processing of Speech Signals” by L. R. Rabiner and R. W. Schafer)

M. Hunt, D. Zwierzynski, and R. Carr, “Issues in High Quality LPC Analysis and Synthesis”.

Some people argue for a discrete-time speech model where:

- Input signal for voiced speech is an impulse train with impulses placed according to pitch periods.
- Input signal for unvoiced speech is random noise.
- The input signal is multiplied by a gain factor, G, which modifies the overall amplitude.
- The resulting signal is passed through a time-varying filter to produce the final speech waveform.

In a Linear Predictive Model, the time-varying filter has a steady-state system function of the form:

$$H(z) = \frac{G}{1 - \sum_{k=1}^{p} a_k z^{-k}}$$

If we call the excitation (i.e. the pulse train or random noise input) $x(n)$, then its relationship to the output speech, $s(n)$, is

$$s(n) = \sum_{k=1}^{p} a_k s(n-k) + Gx(n)$$

The problem of Linear Predictive Analysis is to determine the parameters of such a model given a speech signal.

In practice, this tends to

- Give a relatively compact representation
- Handle most speech sounds well (perhaps not as good for nasals)
- Produce a parameterization which includes a separate term for the fundamental frequency

Analysis is accomplished by considering a linear predictor:

$$s'(n) = \sum_{k=1}^{p} \alpha_k s(n-k)$$

and its prediction error:

$$e(n) = s(n) - s'(n) = s(n) - \sum_{k=1}^{p} \alpha_k s(n-k)$$

If the signal is actually generated by model mentioned previously:

$$e(n) = \sum_{k=1}^{p} a_k s(n-k) + Gx(n) - \sum_{k=1}^{p} \alpha_k s(n-k)$$

So, if the α_ks are chosen correctly, the error term is simply the gain multiplied by the input signal (impulse train for voiced speech, random noise for unvoiced speech)

But how do you estimate the α_ks from the speech signal?

One approach (basis for Autocorrelation and Covariance Methods):

- Consider short segment of speech signal (so time varying parameters are fairly constant)
- Attempt to minimize the mean-square error (or equivalently, the sum of the squared errors)

In the following, using the subscript, n, corresponds to a segment of the unsubscripted signal. Its meaning varies in different approaches.

$$\begin{aligned} E_n &= \sum_m e_n^2(m) \\ &= \sum_m (s_n(m) - s'_n(m))^2 \\ &= \sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right)^2 \end{aligned}$$

Minimum Mean-Squared Error

Set $\partial E_n / \partial \alpha_i = 0$ for $i = 1, 2, \ldots, p$

$$
\begin{aligned}
\frac{\partial E_n}{\partial \alpha_i} &= \frac{\partial}{\partial \alpha_i} \sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right)^2 = 0 \\
&= \sum_m 2 \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) \frac{\partial}{\partial \alpha_i} \sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) \\
&= 2 \sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) \sum_m \frac{\partial}{\partial \alpha_i} \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) \\
&= 2 \sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) \sum_m (-s_n(m-i)) \\
&= 2 \sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) (-m s_n(m-i))
\end{aligned}
$$

$$-2m \sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) s_n(m-i) = 0$$

$$\sum_m \left(s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right) s_n(m-i) = 0$$

$$\sum_m s_n(m) s_n(m-i) = \sum_m \sum_{k=1}^{p} \alpha_k s_n(m-k) s_n(m-i)$$

$$\sum_m s_n(m-i) s_n(m) = \sum_{k=1}^{p} \alpha_k \sum_m s_n(m-i) s_n(m-k)$$

Now define the function:

$$\phi_n(i,k) = \sum_m s_n(m-i)s_n(m-k)$$

which looks very similar to the Autocorrelation function – more on this later...

Then, the minimum mean-square prediction error leads to the equations:

$$\sum_{k=1}^{p} \alpha_k \phi_n(i,k) = \phi_n(i,0), \qquad i = 1,2,\ldots,p$$

These equations can be expressed in matrix form and various matrix symmetries can be exploited to improve the efficiency of determining a solution.

$$\begin{bmatrix} \phi_n(1,1) & \phi_n(1,2) & \phi_n(1,3) & \dots & \phi_n(1,p) \\ \phi_n(2,1) & \phi_n(2,2) & \phi_n(2,3) & \dots & \phi_n(2,p) \\ \phi_n(3,1) & \phi_n(3,2) & \phi_n(3,3) & \dots & \phi_n(3,p) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_n(p,1) & \phi_n(p,2) & \phi_n(p,3) & \dots & \phi_n(p,p) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} \phi_n(1,0) \\ \phi_n(2,0) \\ \phi_n(3,0) \\ \vdots \\ \phi_n(p,0) \end{bmatrix}$$

Up to this point, the summation range for m and the precise meaning of the subscript, n have been left unspecified.

These choices affect the efficiency of solving the equations.

In the Autocorrelation Method, $s_n(m) = s(m+n)w(m)$, where $w(m)$ is a finite-length window such that $s_n(m) = 0$ when $m < 0$ or $m \geq N$ for some window length, N.

This leads to:

$$\phi_n(i,k) = \sum_{m=0}^{N+p-1} s_n(m-i)s_n(m-k)$$

This leads to the leftmost matrix being Toeplitz, and solvable by an $\mathcal{O}(n^2)$ algorithm called Levinson-Durbin recursion.

As an aside, calling this the “Autocorrelation Method” seems a bit confusing as the “Covariance Method” also uses autocorrelation estimates.

In the Covariance Method, different choices are made for the summation range for m and the precise meaning of the subscript, n.

In this approach, the interval for computing the mean-squared error is fixed so

$$E_n = \sum_{m=0}^{N-1} e_n^2(m)$$

It follows that

$$\phi_n(i,k) = \sum_{m=0}^{N-1} s_n(m-i)s_n(m-k)$$

where $1 \leq i \leq p$ and $0 \leq k \leq p$

Or

$$\phi_n(i,k) = \sum_{m=-i}^{N-i-1} s_n(m)s_n(m+i-k)$$

Or

$$\phi_n(i,k) = \sum_{m=-k}^{N-k-1} s_n(m)s_n(m+k-i)$$

This leads to the leftmost matrix being symmetric, but not necessarily Toeplitz.

Cholesky decomposition can be used to make the solution more efficient.

As an aside, the name of this technique can also lead to confusion. It is called the “Covariance Method” because the matrix has the same symmetry as a covariance matrix, but in DSP, covariance is also used to name a function that is related to the autocorrelation function.

Partial Correlation (Parcorr or Lattice) Method

This method involves estimating the parameters (a.k.a. reflection coefficients) for a Lattice filter.

(This technique is not mentioned in the paper to be discussed, but is described in Rabiner and Schaeffer.)

Synthesis - reconstructing a signal from a parametric representation.

If speech has been analyzed by one of the above methods, then the parameters can be used to recreate a signal (hopefully) similar to the original signal.

The parameters can also be modified in an attempt to change the prosody.

The basic idea for synthesizing speech from LPC parameters is to construct the system mentioned on an earlier slide.

For each frame:

- Use voicing decision to choose between impulse train/noise
- If impulse train, use pitch period to place impulses
- Multiply result by gain
- Use linear predictive filter to output final signal

In practice, there are extra details to be aware of.

e.g.: The analysis phase often uses overlapping frames. During synthesis, you don’t want to end up with a speech signal that is twice (or some other multiple) as long because you didn’t take this into account.

Which method (Autocorrelation/Covariance) is best for speech?

Empirical results... (Discuss Paper)

Paper: “Issues in High Quality LPC Analysis and Synthesis”

- Non-real-time LPC Analysis
- Pitch Synchronous Covariance Method
- Work to improve voicing decision, better F_0 determination, better voiced excitation waveform
- Lower limit on value of B_1
- Analysis frame location for covariance method unimportant
- Modified autocorrelation method gave results that were at least as good
- Method of resynthesizing using modified LPC residual

Paper Section 1: “Introduction”

- Unlike other problem domains, real-time analysis not required
- Evaluating performance very difficult

Paper Section 2: “Original System”

- Recording conditions
- Preemphasis filter
- Frame sizes and locations
- Covariance method instabilities fixed by reflecting poles inside unit circle
- Overall: 10 pole parameters, F_0, and power parameter

Paper Section 3: “Determination of Voicing and Fundamental Frequency”

- Errors in F_0 and voicing estimation lead to pops and clicks
- At ends of voiced regions, the laryngograph shows no activity even though the speech signal is periodically excited
- Modify voicing decision as follows:
  - For each frame judged voiceless by laryngograph, examine ratio of 1st to 0th autocorrelation coefficient of speech signal
  - If low-frequency energy predominates, judge frame voiced and derive F_0 from autocorrelation function

Paper Section 3 (cont.)

- Voiced /h/ sounds
- Comparison of laryngograph with LPC residual

Paper Section 4: “Voiced Excitation”

- Using a single impulse vs. residual from glottal cycle
- Tried a few things with glottal cycle
- Single glottal cycle worked better, more complicated approaches didn’t help
- Also experimented with temporal structure

Paper Section 5: “Treatment of Voiced Fricatives”

- Problem with “buzziness” in prolonged voiced fricatives
- Adding white noise to excitation helped
- Attempt to automatically detect voiced fricatives using:
  - total power
  - proportion of power above 3kHz
  - first formant bandwidth (B_1)
- Different classifier attempts:
  - Quadratic classifier not good (parameters not multivariate normal)
  - Multi-layer perceptron better, but still noticeable synthesis errors
  - Histograms fit with simple functions or sequences of straight lines worked better

Paper Section 6: “Adjustment of Bandwidth of First Formant”

- Certain vowels had unpleasant resonance
- Fixed by keeping B_1 above 40Hz
- Used formula $B_1' = \sqrt{B_1^2 + 40^2}$
- More important for pitch-asynchronous tests

Paper Section 7: “The Need for Pitch Synchrony”

- Varied location of analysis period relative to glottal excitation
- Advancing up to 7ms by 1ms increments: no perceivable differences
- Tried pitch-asynchronous analyses
  - OK when analysis period length similar to glottal cycle
  - Bad when analysis period outside of 7ms-16 ms range

Paper Section 8: “Windowing and the Autocorrelation Method”

- Windowing with covariance method didn’t seem to help
- “Typical” real-time autocorrelation system performed worse
- Using analysis phase from one glottal pulse to next helped autocorrelation method, but still not as good as covariance method
- Symmetric windows tapered at ends seemed to work best with autocorrelation method
- Starting analysis period 4ms before glottal closure gave quality at least as good as covariance method

Paper Section 9: “Excitation with a Modified Residual”

- Using residual for excitation synthesizes the original waveform in principle (but welcome to reality...)
- Tried modifying F_0 by deleting samples or adding zeros about 80% into the glottal cycle
- F_0 could be increased up to around 25% or decreased by more without causing noise
- Glottal cycles had to be removed or added to restore duration
- Changing formant frequencies also worked OK
- Listening tests

Paper Section 10: “Summary and Conclusions”

- Improved quality of LPC speech by modifying determination of voicing and F_0 and putting a lower bound on B_1
- Pitch-synchronous analysis didn’t seem to help covariance method, but did seem to help autocorrelation method
- Using modified residual for synthesis after pitch synchronous LPC analysis gives good results for modifying prosody for speech synthesis