1. Linear Predictive Coding (LPC)
2. LPC is based on AR signal modeling
3. LPC is the basis of speech compression for cell phones, digital answering machines, etc.
4. LPC is a lossy compression scheme.
5. LPC is specifically tailored for speech. It does not work well for audio in general.

Human speech

1. Parts of speech: vowels, consonants, semivowels, and dipthongs.
2. voiced sounds: generated by vocal cords.
3. unvoiced sounds: do not involve vocal cords (uses mouth and nasal cavities).

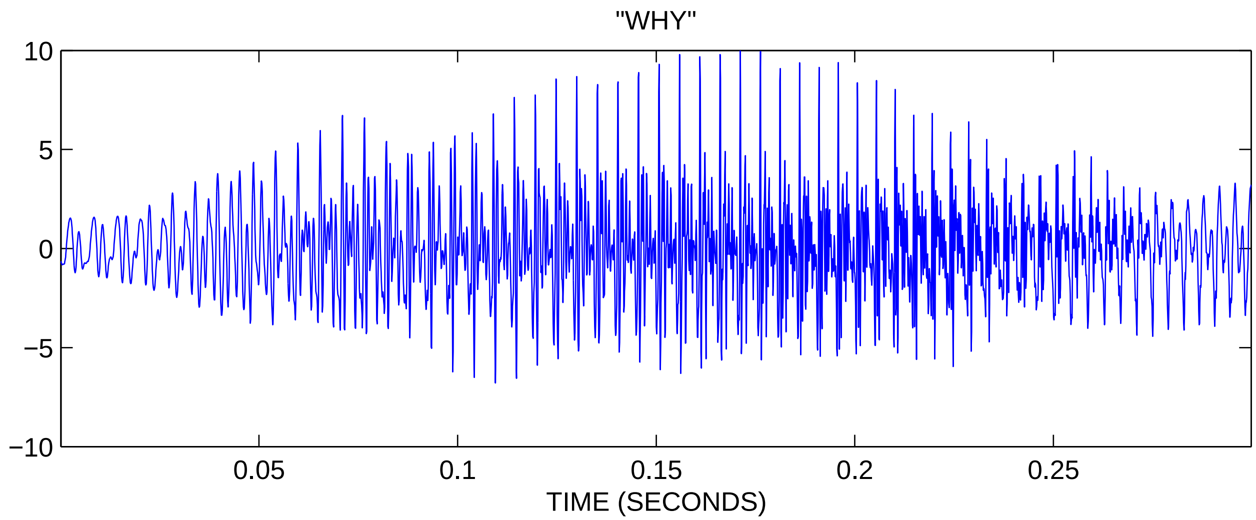
Over short intervals (30 milliseconds), voiced speech:

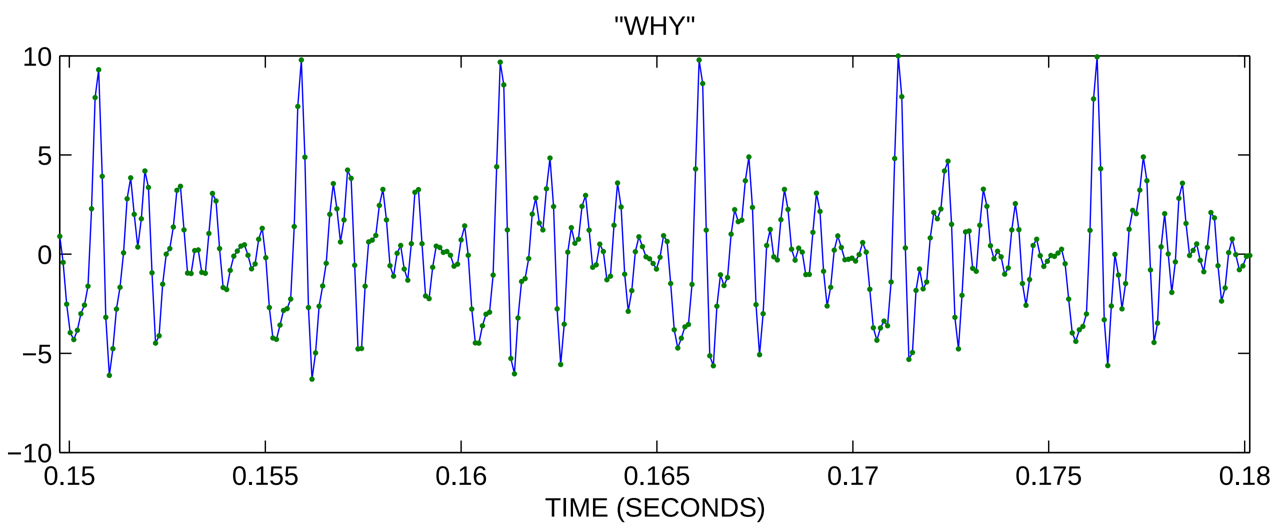
1. resembles a quasi-periodic pulse train;
2. the interval between successive pulses are not exactly the same.
3. the amplitudes of successive pulse are not exactly the same.

Pitch frequency:

1. The *pitch frequency* is the reciprocal of the average period of the quasi-periodic pulse train.
2. Different people have different pitch frequencies. The pitch will vary slightly as a person speaks. A question often ends with a higher pitch (for some cultures).
3. Adult males: 80 - 120 Hz.
4. Adult females: 150 - 300 Hz.
5. Children: higher pitch than adults.

A speech signal ('why') was sampled at 11025 samples/seconds and 3305 samples are collected.





HUMAN SPEECH MODEL

The vocal tract is commonly modeled as a time-varying linear system.

$$\text{vocal cords} \quad \longrightarrow \quad \boxed{\text{vocal tract}} \quad \longrightarrow \quad \text{speech}$$

The vocal tract contains

1. larynx
2. pharynx
3. mouth cavity
4. nasal cavity

Generation of voiced speech:

1. The vocal cords generate the *excitation signal* and the vocal tract affects the sound of the speech.
2. As the mouth cavity changes shape, the sound of the speech changes.
3. The changing shape of the mouth cavity requires that it be modeled using a time-varying system.
4. However, over short intervals (10-30 milliseconds) it can be modelled as a time-invariant system.

The most commonly used model for the vocal tract is an all-pole LTI system.

$$H(z) = \frac{1}{1 + a_1 \, z^{-1} + a_2 \, z^{-2} + \cdots + a_p \, z^{-p}}$$

p is the model order.

Typical values of p are from 8 to 12.

$$e(n) \quad \longrightarrow \quad \boxed{H(z)} \quad \longrightarrow \quad y(n)$$

$e(n)$ is called the *excitation signal*.

$$y(n) = e(n) - a_1 \, y(n-1) - a_2 \, y(n-2) - \cdots - a_p \, y(n-p).$$

1. This is an *autoregressive* (AR) model for the signal $y(n)$.
2. The problem: Given a signal (eg: a speech signal) $s(n)$, find coefficients a_k and a simple excitation signal $e(n)$ so that the output $y(n)$ is close to the given signal $s(n)$.
3. Then to represent $s(n)$ it is necessary only to save the coefficients a_k and the parameters of the excitation signal $e(n)$.

In spectral estimation, an AR system driven by a white noise is used to model a wide sense stationary random signal. In speech coding, the driving signal (excitation signal) is instead a quasi-periodic impulse train. However, we can still use the Yule-Walker equations to estimate the coefficients a_k for $1 \leq k \leq p$.
LPC Speech compression consists of parts.

1. Segment the sampled speech signal into short intervals (10-30 milliseconds long). These segments are called *frames* and can be overlapping or nonoverlapping.

2. For each frame, compute the LPC parameters (a_k for $1 \leq k \leq p$) from the data. This can be done by solving the Yule-Walker equations, or by other related methods.

3. Compute the excitation signal $e(n)$.

4. Model the excitation with a small number of parameters (its pitch and amplitude during the frame). Sometimes a secondary excitation signal is used as well.

5. Quantize and code the parameters:

   (a) the LPC coefficients (a_k for $1 \leq k \leq p$)

   (b) the parameters of the excitation signal

   (c) the parameters of the secondary excitation signal (if used)

To compute the LPC coefficients of a frame:

1. Let $b(n)$ denote the current frame.

2. Compute the autocorrelation of the frame $b(n)$

$$r(n) = b(n) * b(-n)$$

   (We only need to compute the values $r(n)$ for $0 \le n \le p$.)

3. Solve the Yule-Walker equations. When $p = 4$ the Yule-Walker equations are:

$$\begin{bmatrix} r(0) & r(1) & r(2) & r(3) \\ r(1) & r(0) & r(1) & r(2) \\ r(2) & r(1) & r(0) & r(1) \\ r(3) & r(2) & r(1) & r(0) \end{bmatrix} \begin{bmatrix} a(1) \\ a(2) \\ a(3) \\ a(4) \end{bmatrix} = - \begin{bmatrix} r(1) \\ r(2) \\ r(3) \\ r(4) \end{bmatrix}$$

For each frame there will be p LPC parameters.

The LPC parameters describe a different system $H(z)$ for each frame,

$$H(z) = \frac{1}{1 + a_1\, z^{-1} + a_2\, z^{-2} + \cdots + a_p\, z^{-p}}$$

In the following Matlab code, each successive 160-point frame is extracted from the sampled speech signal, its autocorrelation function is computed, the Yule-Walker equations are solved, and the frequency response of the all-pole filter is plotted.

```
N = 160;
p = 10;
for k = 1:20
   n = (k-1)*N+[1:N];
   frame = s(n);
   corr = xcorr(frame)/N;
   r = corr(N:N+p);
   R = toeplitz(r(1:p));
   a = [1; -R\r(2:p+1)'];
   [H,w] = freqz(1,a);
   plot(w/pi*Fs/2,20*log10(abs(H)))
   title('FREQUENCY RESPONSE OF H(z) in dB')
   xlabel('FREQUENCY (HERZ)')
end
```

This Matlab code can be shortened by using the Matlab function `a = lpc(frame,p)`, which efficiently solves the Yule-Walker equations using the Levinson-Durbin algorithm. Other related Matlab functions are: `levinson` and `aryule`.

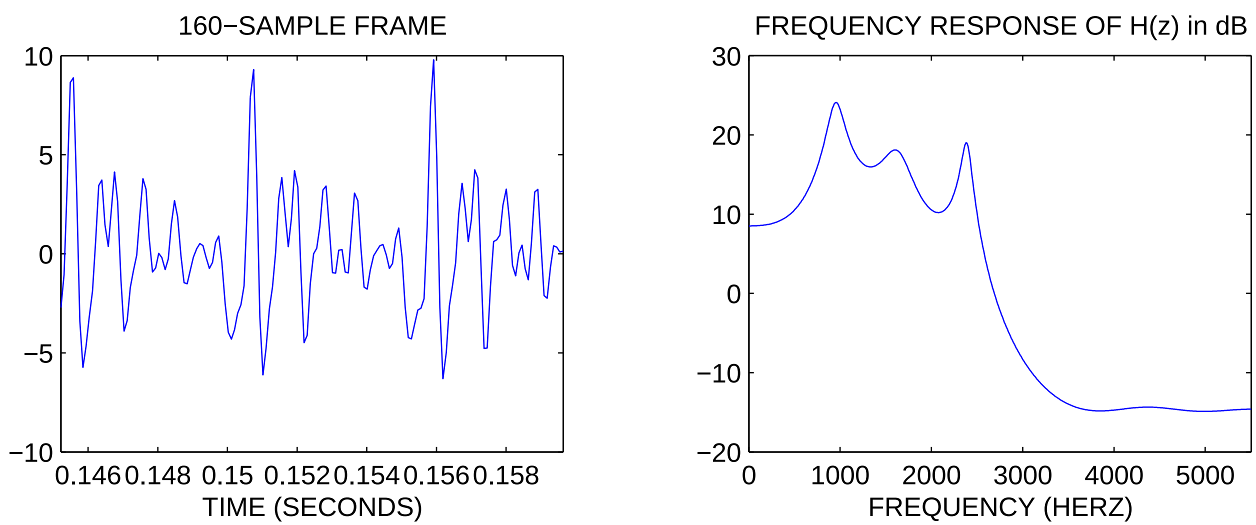
Example

1. Segment the signal 'why' into segments of 160 samples.

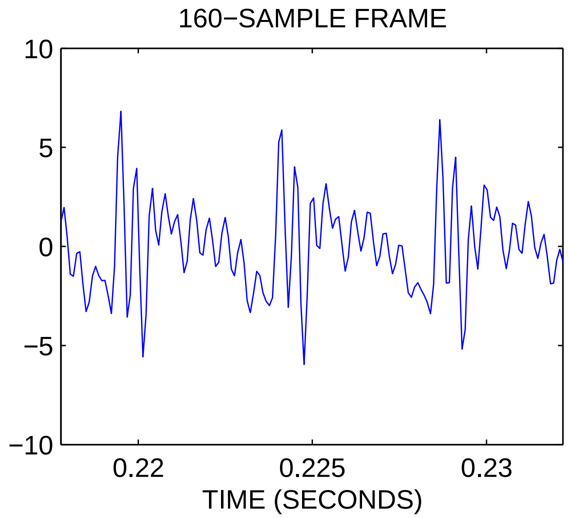
2. In this case, because $F_s = 11025$, the duration of each segment is about 14.5 milliseconds.

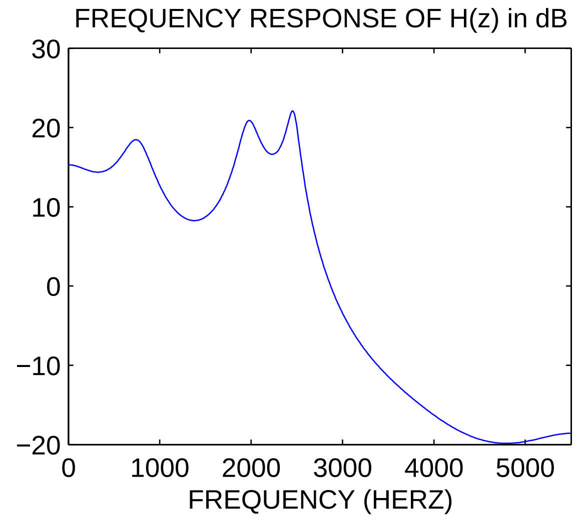
3. Use $p = 10$.

Frame 11



Frame 16





The frequencies where peaks occur are called *formants*.

COMPUTING THE EXCITATION SIGNAL

$$y(n) = h(n) * e(n)$$

$$Y(z) = H(z)\,E(z)$$

$$E(z) = Y(z)/H(z)$$

$$y(n) \quad \longrightarrow \quad \boxed{1/H(z)} \quad \longrightarrow \quad e(n)$$

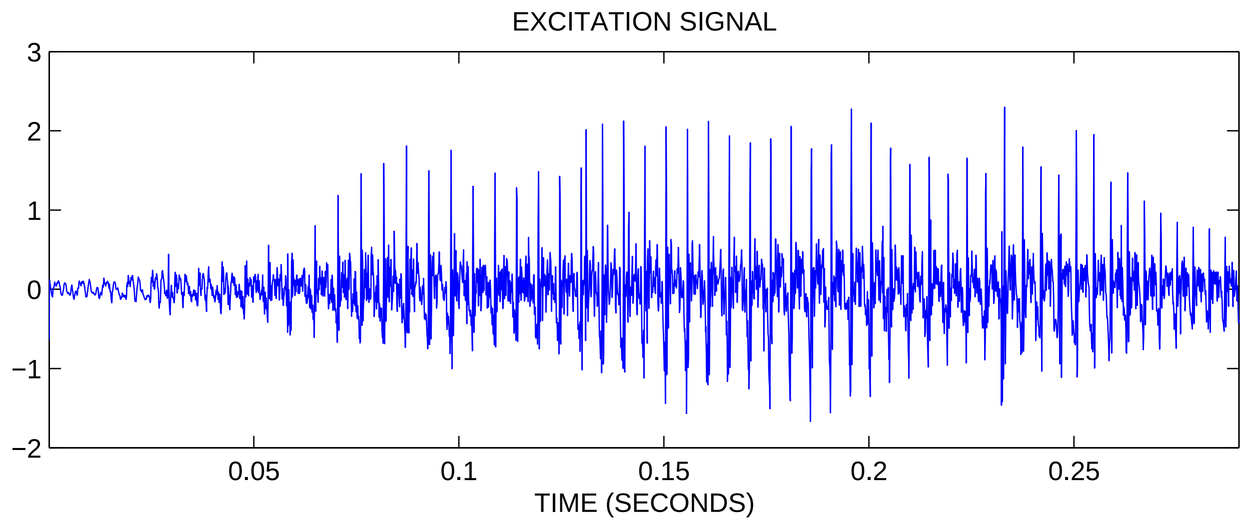
$1/H(z)$ is an FIR filter,

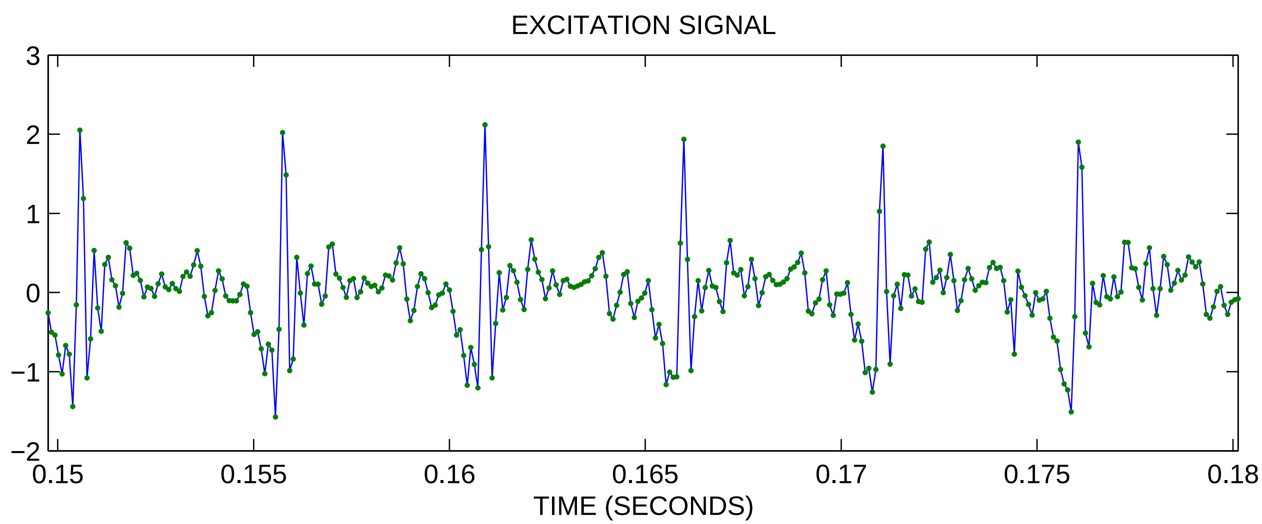
$$\frac{1}{H(z)} = 1 + a_1\,z^{-1} + a_2\,z^{-2} + \cdots + a_p\,z^{-p}$$

To find an excitation signal that makes $y(n) = s(n)$, the excitation signal $e(n)$ can be found by filtering the $s(n)$ with the FIR filter $1/H(z)$.

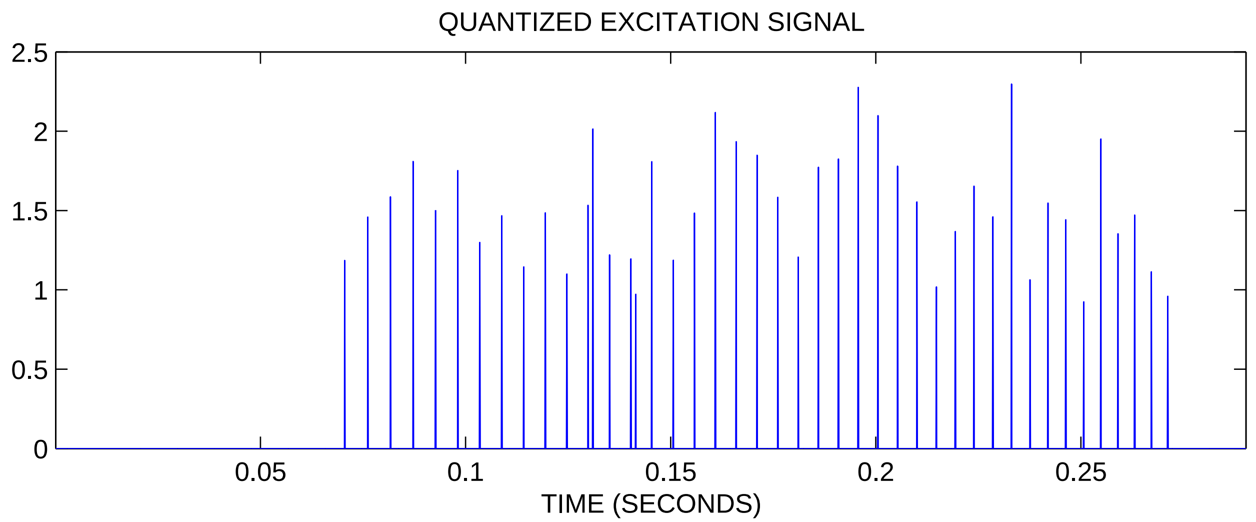
The quantized excitation signal consists of a sequence of amplitudes and intervals.

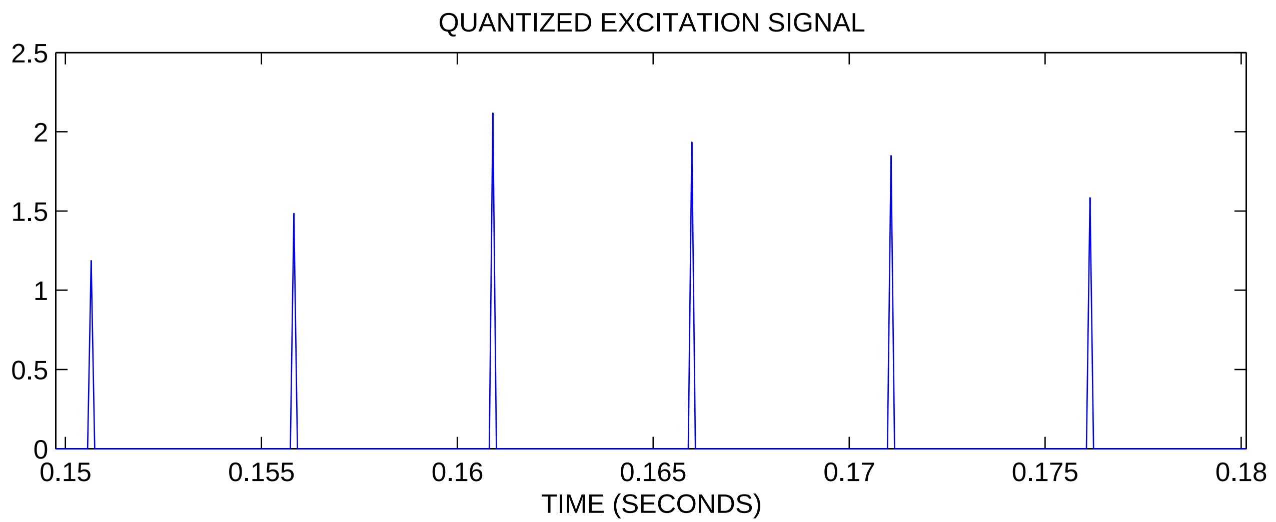
COMPUTING THE EXCITATION SIGNAL





THE QUANTIZED EXCITATION SIGNAL





RECONSTRUCTED SIGNAL