Speech Signal Analysis and Feature Extraction

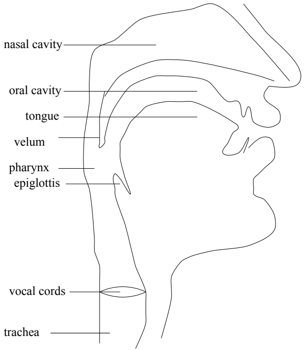
Dr. Mathew Magimai Doss

September 21, 2022

Outline

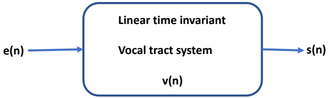
Outline

Speech signal production model



excitation: vibration of vocal cords
system: vocal tract (oral cavity) [sometimes nasal cavity]
response: speech

With in a short-term analysis window of 20-40 ms

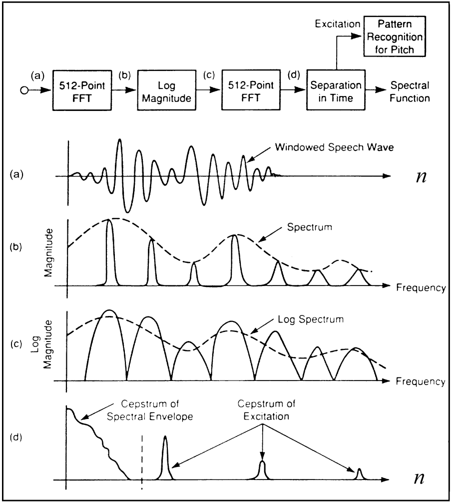


$$s(n) = e(n) * v(n)$$

, $*$ denotes convolution

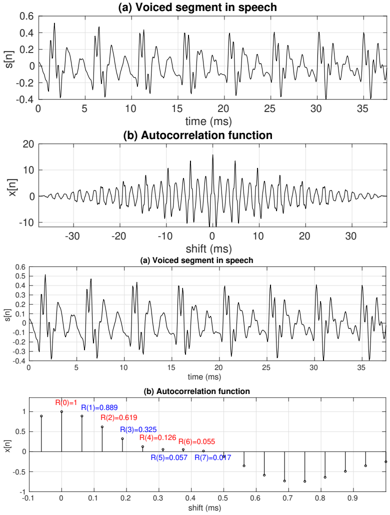
- frequency domain processing based source-system decomposition: cepstrum
- time domain processing-based source-system decomposition: linear prediction

Cepstral analysis



- **(a)** Windowed speech signal model
  $s(n) = e(n) * v(n)$
- **(b)** Apply DFT or FFT
  $S(\omega) = E(\omega) \cdot V(\omega)$
- **(c)** Logarithm of DFT or FFT
  $\log|S(\omega)| =$
  $\log|E(\omega)| + \log|V(\omega)|$
- **(d)** inverse DFT or FFT leads to cepstrum domain
  $c_s(n) = c_e(n) + c_v(n)$
  $c_e(n)$ - cepstrum of excitation (source)
  $c_v(n)$ - cepstrum of spectral envelop (system)

Linear prediction (1)



- Each sample with in the analysis window is modeled as a linear weighted sum of past p samples $\hat{s}(n) = \sum_{k=1}^{p} a_k \cdot s(n-k)$
- Error or residual signal $e(n) = s(n) - \hat{s}(n)$
- Estimate $\{a_k\}_{k=1}^{p}$ by minimizing the mean square error
- $\{a_k\}_{k=1}^{p}$ models the spectral envelop (system) and $e(n)$ mainly models excitation (source)

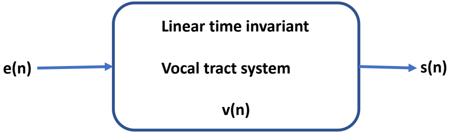
Linear prediction (2)

- signal model

$$
\begin{aligned}
s(n) &= \hat{s}(n) + e(n) \\
s(n) &= \sum_{k=1}^{p} a_k \cdot s(n-k) + e(n)
\end{aligned}
$$

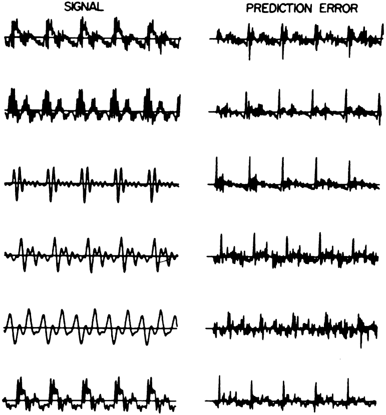
$$
s(n) - \sum_{k=1}^{p} a_k \cdot s(n-k) = e(n)
$$

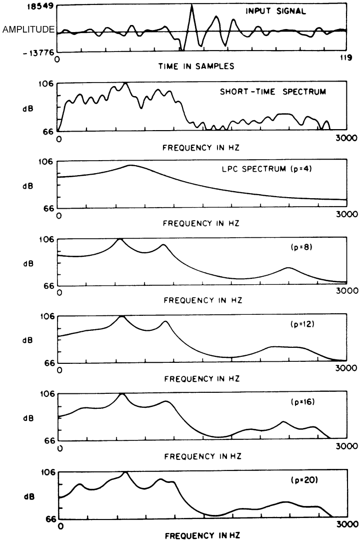
- Applying Z-transform
  $S(z) - \sum_{k=1}^{p} a_k \cdot z^{-k} S(z) = E(z)$
- All-pole transfer function
  $\frac{S(z)}{E(z)} = \frac{1}{(1 - \sum_{k=1}^{p} a_k \cdot z^{-k})} = V(z)$



$$
s(n) = e(n) * v(n)
$$

Linear prediction (3)





Thumb rule for choosing linear prediction order p:

$2 \times \#$ of formants to model $+2$

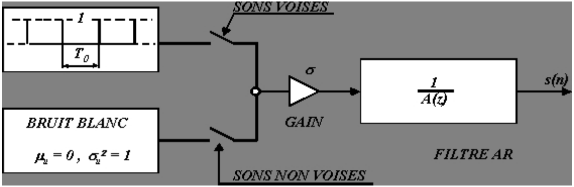
Outline

LP-based speech coding (1)

For each analysis window

- Transmitter side: perform linear prediction (LP) analysis
  - Estimate $\{a_k\}_{k=1}^{p}$
  - From the residual estimate, (a) whether signal is voiced or unvoiced (v/uv), (b) Fundamental frequency or pitch period T_0 and (c) gain σ

  Transmit $\{a_k\}_{k=1}^{p}$, v/uv, T_0 and σ
- Receiver side: Given $\{a_k\}_{k=1}^{p}$, v/uv, T_0 and σ, synthesize speech signal of window shift length



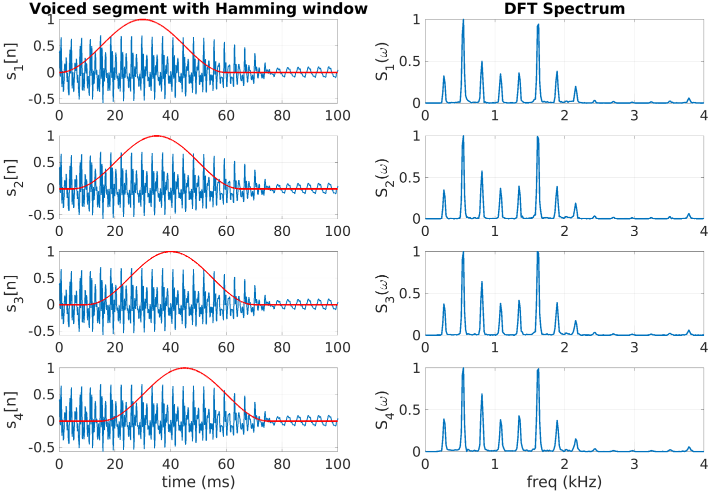
$A(z) = 1 - \sum_{k=1}^{p} a_k \cdot z^{-k}$

LP-based speech coding (2)

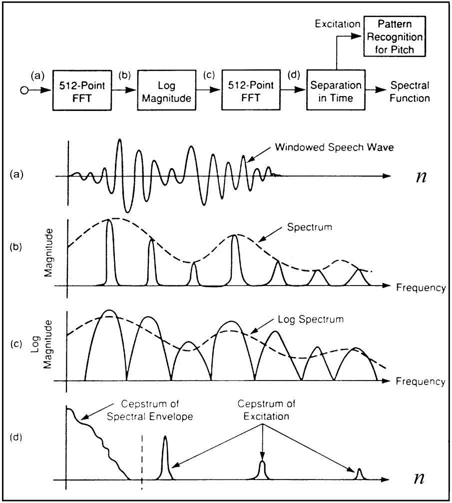
- Bit rate with μ-law or A-law in telephony
  64000 bits/second = 8 $\text{bits/sample} \times$ 8000 samples/second
- Bit rate with linear prediction coding
  - Window size: 30 ms
  - Window shift: 10 ms (i.e. 100 frames/second)
  - Linear prediction order: 10
  - Example bits per frame: 10×8 bits for $\{a_k\}_{k=1}^{p}$ + 8 bits for T_0 + 8 bits for σ + 1 bit for v/uv = 97 bits/frame
  - Example bit rate:
    97 $\text{bits/frame} \times$ 100 $\text{frames/second} =$ 9700 bits/second
- G.729 standard bit rate is 8000 bits/second
- LP-based speech coding is used in cell phones for speech transmission

Outline

Short-term spectral processing

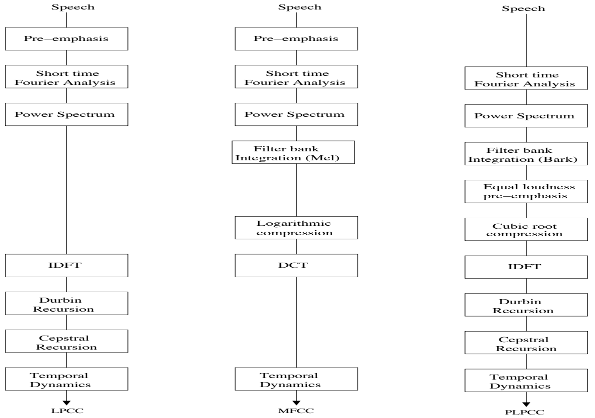


Linear frequency cepstral coefficients



- **(a)** Windowed speech signal model
  $s(n) = e(n) * v(n)$
- **(b)** Apply DFT or FFT
  $S(\omega) = E(\omega) \cdot V(\omega)$
- **(c)** Logarithm of DFT or FFT
  $\log |S(\omega)| =$
  $\log |E(\omega)| + \log |V(\omega)|$
- **(d)** inverse DFT or FFT leads to cepstrum domain
  $c_s(n) = c_e(n) + c_v(n)$
  $c_e(n)$ - cepstrum of excitation (source)
  $c_v(n)$ - cepstrum of spectral envelop (system)

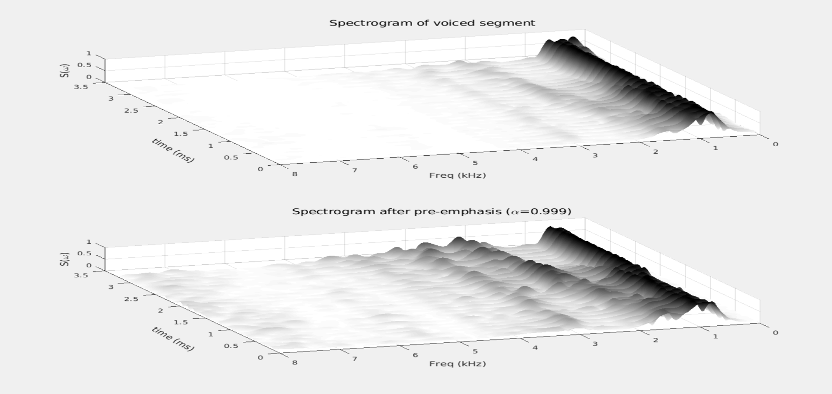
Other cepstral features



LPCC: Linear prediction cepstral coefficients, MFCC: Mel frequency cepstral coefficients, PLPCC: Perceptual linear prediction cepstral coefficients

$$c_m^k = -a_k + \frac{1}{N}\sum_{i=1}^{k-1}(k-i)\cdot a_i \cdot c_{k-i}$$

Pre-emphasis

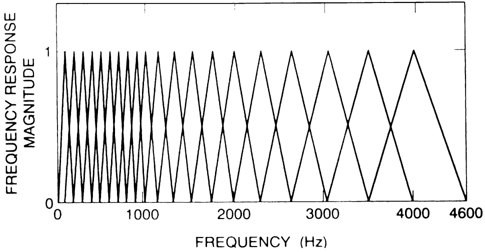


- -6dB tilt in the spectrum due to combination of glottal exication source (-12dB) and lip radiation (+6dB)
- High pass filter to lift high frequency components (liftering)

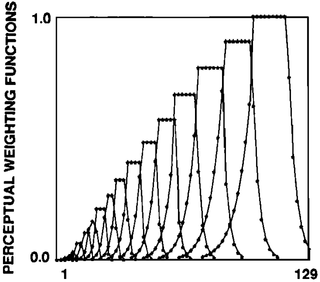
$$s(n) = s(n) - \alpha \cdot s(n-1)$$

Filter banks

- Mel scale (based on pitch perception)

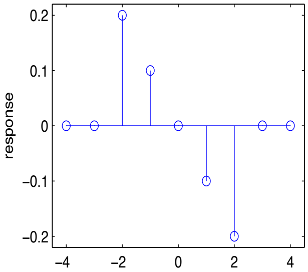


- Bark scale (based on loudness perception)

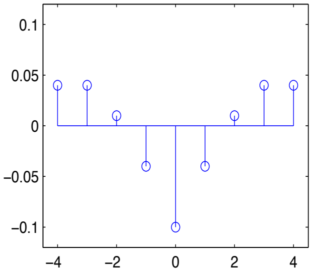


Temporal derivatives

$$\Delta_{c_m} = \frac{\sum_{k=1}^{K} k \cdot (c_{m+k} - c_{m-k})}{2 \cdot \sum_{k=1}^{K} k^2} \qquad (1)$$



Delta (first order derivative)



Delta-Delta (second order derivative)

- Savitzky-Golay filtering and temporal derivatives computation

Feature vector

- Cepstral features
  - Speech recognition: $C_1 - C_{12} + \Delta + \Delta\Delta$
  - Speaker recognition: $C_1 - C_{20} + \Delta + \Delta\Delta$
  - Speech synthesis using HMMs: $C_1 - C_{39} + \Delta + \Delta\Delta$

  Typically, in static features, e.g. $C_1 - C_{12}$, mean estimated over the utterance is removed to handle channel variation.
- log filter bank energies $+\Delta + \Delta\Delta$
- Energy: log energy (in the short-term analysis window) or C_0 $+\Delta + \Delta\Delta$
- Fundamental frequency: $\log F_0$ (typically) $+\Delta + \Delta\Delta$

Δ denotes first order temporal derivative

$\Delta\Delta$ denotes second order temporal derivative

Feature sequence $X = \{x_1, \cdots x_m, \cdots x_M\}$

Thank you for your attention!

Dr. Mathew Magimai Doss

Idiap Research Institute, Martigny, Switzerland