Source-filter analysis/synthesis

- Separate
  - **Source/excitation** fine time/frequency structure (e.g. pitch)
  - **Filter** broad spectral shape (resonances)
- Similar to subtractive synthesis
- Satisfying physical interpretation for real-world signals
- Easier to make sense of than e.g. phase
Human speech production

Reasonable approximation to speech signals:

- **Source is oscillation of vocal chords**
  - e.g. normal speech (varying pitches) vs whispering

- **Filtered by vocal tract (throat + tongue + lips)**
  - e.g. “oooh” vs “aaah”
  - resonances = formants

- Both are time-varying
Source filter model

Excitation source $t$

Resonance filter $f$

$X(f)$ and $|G \cdot H(f)|$ in dB

Magnitude spectra $|X(f)|$ and $|G \cdot H(f)|$ in dB

Time signal of pred. error $e(n)$

$n$
Formants in speech

has a watch thin as a dime

E85.2607: Lecture 8 – Source-Filter Processing
How to separate the source and filter?

Short-time analysis

For each frame, estimate spectral envelope (filter response)

1. Channel vocoder (frequency-domain)
2. Linear Predictive Coding (LPC) (time-domain)
3. Cepstral analysis

Source signal is what's left over (residual) after “whitening”
Channel vocoder

- Wideband STFT filterbank
- but using relatively few filters
  - Linearly spaced with equal bandwidth (STFT)
  - Logarithmically spaced (constant-Q filter bank)
- Take RMS energy in each frequency band

![Diagram of channel vocoder](image-url)
Channel vocoder using FFT

- Lowpass filter magnitude of each STFT frame
  - i.e. filter columns of the spectrogram
Linear predictive coding

- **Predict** next input sample as linear combination of previous samples

\[
x(n) \xrightarrow{z^{-1}} a_1 \xrightarrow{z^{-1}} a_2 \xrightarrow{z^{-1}} a_p \xrightarrow{+} \hat{x}(n) \xrightarrow{-} e(n)
\]

- Filter is described by a few filter coefficients for each frame

\[
x^m[n] \approx \hat{x}[n] = \sum_{k=1}^{p} a_k x[n - k]
\]

- Excitation is what's left after filtering (residual aka prediction error)

\[
e[n] = x[n] - \hat{x}[n] = x[n] - \sum_{k=1}^{p} a_k x[n - k]
\]
LPC analysis/synthesis

(a) LPC analysis

(b) LPC synthesis

- \( P(z) \) is just an FIR filter: \( P(z) = \sum_{k=1}^{p} a_k z^{-k} \)
- Excitation is still a filtered version of the input:

\[
E(x) = X(z) (1 - P(z))
\]

- For synthesis, pass (approximate) excitation through the inverse filter:

\[
Y(z) = \tilde{E}(z) H(z)
\]

\[
H(z) = \frac{1}{1 - P(z)}
\]

- all-pole “autoregressive” (AR) modeling
LPC - varying filter order

- LPC filter $H(z)$ models the spectrum of $x[n]$
- Minimizing the energy of the residual $e[n]$ gives optimal coefficients

$$\{a_k\} = \arg\min_{a_k} \sum_{n} \left( x[n] - \sum_{k} a_k x[m - k] \right)^2$$

- The approximation improves with increasing filter order $p$

![spectra of original and LPC filters]
Estimating LPC parameters

- Set derivative of $\sum_n e^2[n]$ w.r.t. $a_k$ zero and solve for $a_k$:
  \[
  \frac{\partial}{\partial a_k} \sum_n e^2[n] = 0
  \]

- End up with $p$ linear equations involving autocorrelations of $x$:
  \[
  \sum_m x[m]x[m-k] = \sum_i a_k \sum_m x[m-i]x[m-k]
  \]

- Solve using Levinson-Durbin recursion
LPC example

![Graph showing LPC example](image)

- **Windowed original**
- **LPC residual**
- **Original spectrum**
- **LPC spectrum**
- **Residual spectrum**

Filter poles in the z-plane.
Short-time LPC analysis

Solve LPC for each ~20 ms frame.
Cepstral analysis

- cepstrum = String.reverse("spec") + "trum"
  - Entire lexicon of funny anagrams
- Insight: source and filter add in the log spectral domain

\[ X(z) = E(z)H(z) \]
\[ \log X(z) = \log E(z) + \log H(z) \]

- Makes them easy to separate

![Diagram of source-filter processing](image)
Lifting example

By low-pass “liftering” the cepstrum we obtain the spectral envelope of the signal.
Lifting example 2

- Original waveform has excitation fine structure convolved with resonances
- DFT shows harmonics modulated by resonances
- Log DFT is sum of harmonic ‘comb’ and resonant bumps
- IDFT separates out resonant bumps (low quefrency) and regular, fine structure (‘pitch pulse’)
- Selecting low-n cepstrum separates resonance information (deconvolution / ‘liftering’)

---

![Waveform and min. phase IR](image1)
![abs(dft) and liftered](image2)
![log(abs(dft)) and liftered](image3)
![real cepstrum and lifter](image4)

---

E85.2607: Lecture 8 – Source-Filter Processing
4. LPC Synthesis

- LP analysis on ~20ms frames gives prediction filter and residual recombining them should yield perfect coding applications further
  - e.g. simple pitch tracker
  - "buzz-hiss" encoding

\[ e[n] = \frac{1}{1 - \sum a[i] z^{-i}} \]

- Low bitrate speech codec used in cell phones is based on LPC
- Quantize LPC filter parameters, use crude approximation to residual
  - Many different ways to represent filter params:
    - prediction coefficients \( \{a_k\} \)
    - roots of \( 1 - P(z) \)
    - line spectral frequencies
  - Switch between noise and pulse train for excitation

Use codebook of excitations (CELP: Code Excited Linear Prediction)
Applications - Cross-synthesis/Vocoding

- Reconstruct using excitation from one sound and filter from another
- Whisperization: replace excitation with white noise
Still more applications

- Process formants independent of pitch
  - Pitch-shifting while preserving formants
  - Shift formants while preserving pitch

- Voice transformation
- Pitch-analysis
DAFX 9.1 – 9.3 - Source-Filter Processing