Lecture 5: Speech modeling

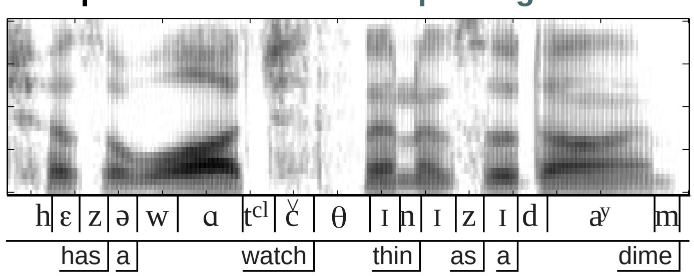
1. **Modeling speech signals**
2. **Spectral and cepstral models**
3. **Linear Predictive models (LPC)**
4. Other signal models
5. Speech synthesis

Dan Ellis <dpwe@ee.columbia.edu>
http://www.ee.columbia.edu/~dpwe/e6820/

Columbia University Dept. of Electrical Engineering
Spring 2006

1 The speech signal

- **Speech sounds in the spectrogram**



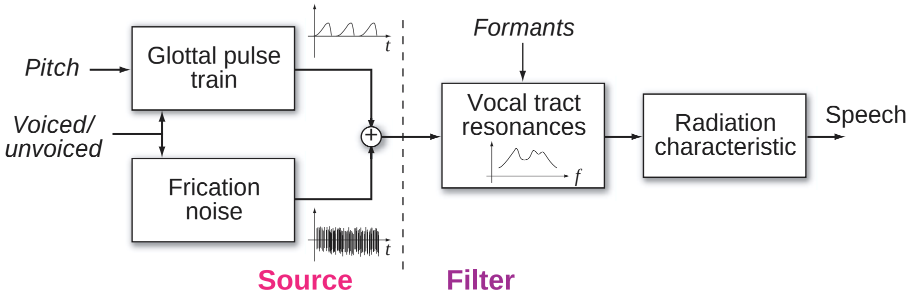
- **Elements of the speech signal:**
  - spectral resonances (formants, moving)
  - periodic excitation (voicing, pitched)
    + pitch contour
  - noise excitation (fricatives, unvoiced, no pitch)
  - transients (stop-release bursts)
  - amplitude modulation (nasals, approximants)
  - timing!

The source-filter model

- **Notional separation of:**

  source: excitation, fine time-frequency structure

  & filter: resonance, broad spectral structure



- **More a modeling approach than a single model**

Signal modeling

- **Signal models are a kind of representation**
  - to make some aspect explicit
  - for efficiency
  - for flexibility
- **Nature of model depends on goal**
  - classification: remove irrelevant details
  - coding/transmission: remove perceptual irrelevance
  - modification: isolate control parameters
- **But commonalities emerge**
  - perceptually irrelevant detail (coding) will also be irrelevant for classification
  - modification domain will usually reflect ‘independent’ perceptual attributes
  - getting at the abstract information in the signal

Different influences for signal models

- **Receiver:**
  - see how signal is treated by listeners
    → cochlea-style filterbank models ...
- **Transmitter (source)**
  - physical vocal apparatus can generate only a limited range of signals...
    → LPC models of vocal tract resonances
- **Making explicit particular aspects**
  - compact, separable correlates of resonances
    → cepstrum
  - modeling prominent features of NB spectrogram
    → sinusoid models
  - addressing unnaturalness in synthesis
    → Harmonic+Noise model

Applications of (speech) signal models

- **Classification / matching**
  Goal**: highlight important information**
  - speech recognition (lexical content)
  - speaker recognition (identity or class)
  - other signal classification
  - content-based retrieval
- **Coding / transmission / storage**
  Goal**: represent just enough information**
  - real-time transmission e.g. mobile phones
  - archive storage e.g. voicemail
- **Modification/synthesis**
  Goal**: change certain parts independently**
  - speech synthesis / text-to-speech
    (change the words)
  - speech transformation / disguise
    (change the speaker)

Outline

1. Modeling speech signals
2. **Spectral and cepstral models**
   - Auditorily-inspired spectra
   - The cepstrum
   - Feature correlation
3. Linear predictive models (LPC)
4. Other models
5. Speech synthesis

2 Spectral and cepstral models

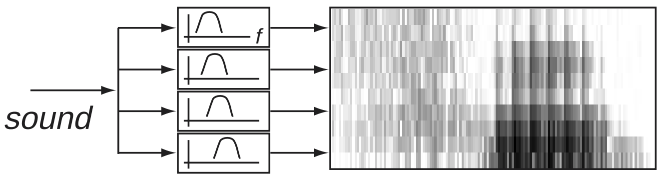
- **Spectrogram seems like a good representation**
  - long history
  - satisfying in use
  - experts can 'read' the speech
- **What is the information?**
  - intensity in time-frequency cells;
    typically 5ms x 200 Hz x 50 dB

→ **Discarded detail:**
  - phase
  - fine-scale timing

- **The starting point for other representations**

The filterbank interpretation of the short-time Fourier transform (STFT)

- **View spectrogram rows as coming from separate bandpass filters:**

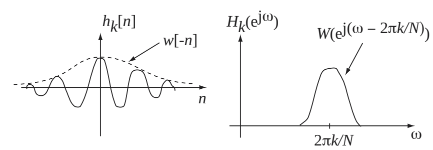


- **Mathematically:**

$$X[k, n_0] = \sum_n x[n] \cdot w[n - n_0] \cdot \exp - j\left(\frac{2\pi k(n - n_0)}{N}\right)$$

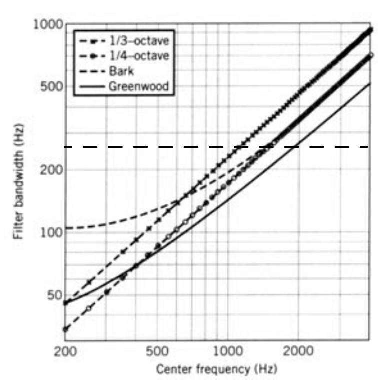
$$= \sum_n x[n] \cdot h_k[n_0 - n]$$

where $h_k[n] = w[-n] \cdot \exp j\left(\frac{2\pi k n}{N}\right)$



Spectral models: Which bandpass filters?

- **Constant bandwidth? (analog / FFT)**
- **But: cochlea physiology & critical bandwidths**
  → implement ear models with bandpass filters & choose bandwidths by e.g. CB estimates
- **Auditory frequency scales**
  - constant 'Q' (center freq/bandwidth), mel, Bark...

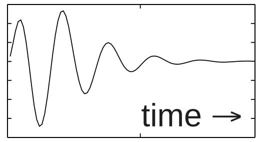


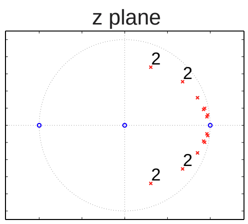
Gammatone filterbank

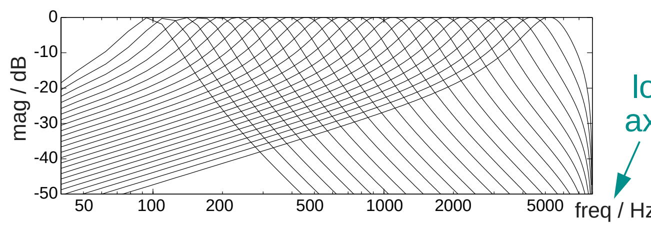
- **Given bandwidths, which filter shapes?**
  - match inferred temporal integration window
  - match inferred spectral shape (sharp hi-F slope)
  - keep it simple (since it's only approximate)

→ **Gammatone filters**

$$h[n] = n^{N-1} \cdot \exp{-bn} \cdot \cos(\omega_i n)$$





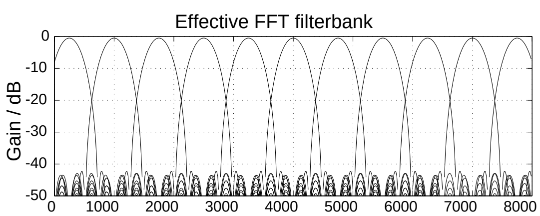


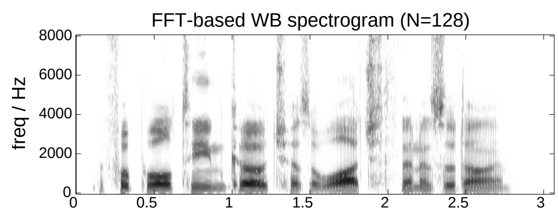
log axis!

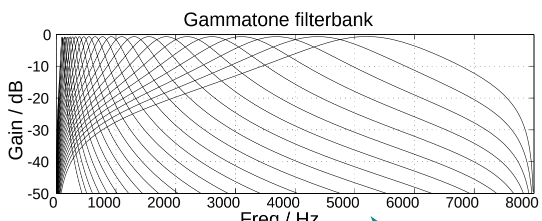
  - 2N poles, 2 zeros, low complexity
  - reasonable linear match to cochlea

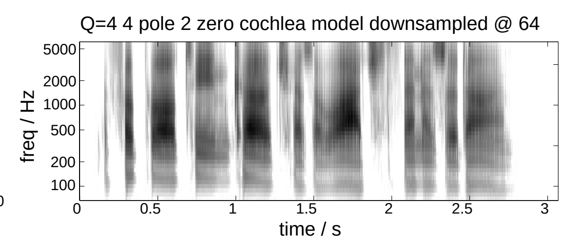
Constant-BW vs. cochlea model

- **Frequency responses:**
- **Spectrograms:**









linear axis

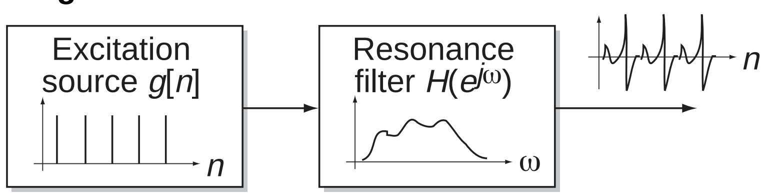
- **Magnitude smoothed over 5-20 ms time window**

Limitations of spectral models

- **Not much data thrown away**
  - just fine phase/time structure (smoothing)
  - little actual 'modeling'
  - still a large representation!
- **Little separation of features**
  - e.g. formants and pitch
- **Highly correlated features**
  - modifications affect multiple parameters
- **But, quite easy to reconstruct**
  - iterative reconstruction of lost phase

The cepstrum

- **Original motivation: Assume a source-filter model:**

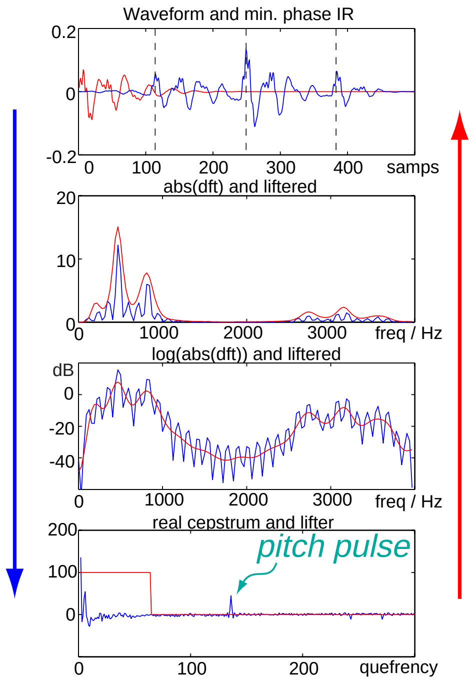


- **Define 'Homomorphic deconvolution':**
  - source-filter convolution: $g[n]*h[n]$
  - FT → product $G(e^{j\omega})\cdot H(e^{j\omega})$
  - log → sum: $\log G(e^{j\omega}) + \log H(e^{j\omega})$
  - IFT
    → separate fine structure: $c_g[n] + c_h[n]$
    = deconvolution
- **Definition:**

  Real cepstrum $c_n = \mathrm{idft}(\log|\mathrm{dft}(x[n])|)$

Stages in cepstral deconvolution

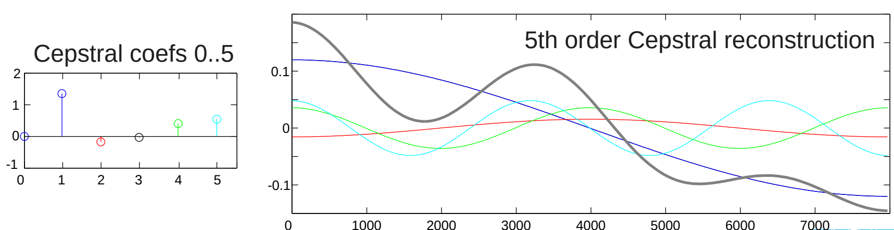
- Original waveform has excitation fine structure convolved with resonances
- DFT shows harmonics modulated by resonances
- Log DFT is sum of harmonic 'comb' and resonant bumps
- IDFT separates out resonant bumps (low quefrency) and regular, fine structure ('pitch pulse')
- Selecting low-n cepstrum separates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



Properties of the cepstrum

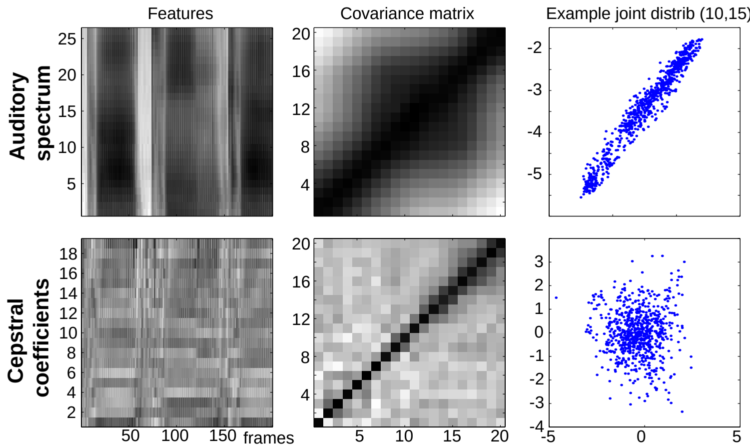
- Separate source (fine) & filter (broad structure)
  - smooth the log mag. spectrum to get resonances
- *Smoothing* spectrum is *filtering* along freq.
  - i.e. convolution applied in Fourier domain
    → *multiplication* in IFT ('liftering')
- Periodicity in time → harmonics in spectrum
  → 'pitch pulse' in high-n cepstrum
- Low-n cepstral coefficients are DCT of broad filter / resonance shape:

$$c_n = \int \log \left| X(e^{j\omega}) \right| \cdot (\cos n\omega + \cancel{j \sin n\omega}) d\omega$$



Aside: Correlation of elements

- **Cepstrum is a popular in speech recognition**
  - feature vector elements are decorrelated:



  - c_0 ‘normalizes out’ average log energy
- **Decorrelated pdfs fit diagonal Gaussians**
  - simple correlation is a waste of parameters
- **DCT is close to PCA for (mel) spectra?**

Outline

1. Modeling speech signals
2. Spectral and cepstral modes
3. **Linear Predictive models (LPC)**
   - The LPC model
   - Interpretation & application
   - Formant tracking
4. Other models
5. Speech synthesis

3 Linear predictive modeling (LPC)

- **LPC is a very successful speech model**
  - it is mathematically efficient (IIR filters)
  - it is remarkably accurate for voice (fits source-filter distinction)
  - it has a satisfying physical interpretation (resonances)
- **Basic math**
  - model output as linear function of prior outputs:

    $$s[n] = \left(\sum_{k=1}^{p} a_k \cdot s[n-k]\right) + e[n]$$

    ... hence "linear prediction" (p^{th} order)
  - $e[n]$ is excitation (input), a/k/a *prediction error*

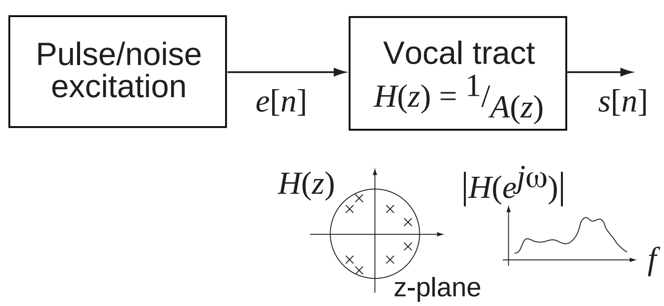
    $$\rightarrow \frac{S(z)}{E(z)} = \frac{1}{\left(1 - \sum_{k=1}^{p} a_k \cdot z^{-k}\right)} = \frac{1}{A(z)}$$

    ... *all-pole* modeling, 'autoregression' (AR) model

Vocal tract motivation for LPC

- **Direct expression of source-filter model:**

$$s[n] = \left(\sum_{k=1}^{p} a_k \cdot s[n-k]\right) + e[n]$$



- **Acoustic tube models suggest all-pole model for vocal tract**
- **Relatively slowly-changing**
  - update $A(z)$ every 10-20 ms
- **Not perfect: Nasals introduce zeros**

Estimating LPC parameters

- **Minimize short-time squared prediction error:**

$$E = \sum_{n=1}^{m} e^2[n] = \sum_n \left\{ s[n] - \sum_{k=1}^{p} a_k s[n-k] \right\}^2$$

Differentiate w.r.t. a_k to get eqns for each k:

$$\sum_n 2\left(s[n] - \sum_{j=1}^{p} a_j s[n-j]\right) \cdot (-s[n-k]) = 0$$

$$\sum_n s[n]s[n-k] = \sum_j a_j \cdot \sum_n s[n-j]s[n-k]$$

$$\phi(0,k) = \sum_j a_j \cdot \phi(j,k)$$

where $\phi(j,k) = \sum_{n=1}^{m} s[n-j]s[n-k]$

are correlation coefficients

- **p linear equations to solve for all a_js...**

Evaluating parameters

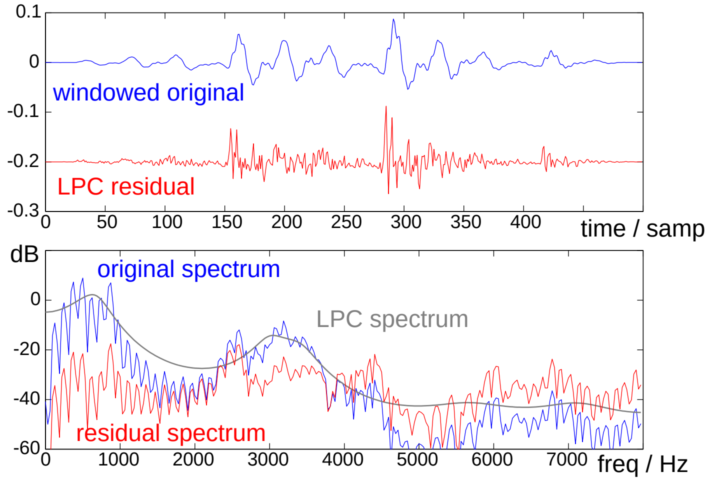
- **Linear equations** $\phi(0,k) = \sum_{j=1}^{p} a_j \cdot \phi(j,k)$
- **If $s[n]$ is assumed zero outside some window**
  $\phi(j,k) = \sum_n s[n-j]s[n-k] = r_{ss}(|j-k|)$
  - $r_{ss}(\tau)$ is autocorrelation

  Hence equations become:

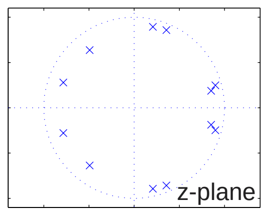
$$\begin{bmatrix} r(1) \\ r(2) \\ \dots \\ r(p) \end{bmatrix} = \begin{bmatrix} r(0) & r(1) & \dots & r(p-1) \\ r(1) & r(2) & \dots & r(p-2) \\ \dots & \dots & \dots & \dots \\ r(p-1) & r(p-2) & \dots & r(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_p \end{bmatrix}$$

- **Toeplitz matrix (equal antidiagonals)**
  → can use Durbin recursion to solve
- **(Solve full $\phi(j,k)$ via Cholesky)**

LPC illustration



- **Actual poles:**



Interpreting LPC

- **Picking out resonances**
  - if signal really was source + all-pole resonances, LPC should find the resonances
- **Least-squares fit to spectrum**
  - minimizing $e^2[n]$ in time domain is the same as minimizing $E^2(e^{j\omega})$ (by Parseval)

  →close fit to spectral *peaks*; valleys don't matter
- **Removing smooth variation in spectrum**
  - $1/A(z)$ is low-order approximation to $S(z)$
  - $$\frac{S(z)}{E(z)} = \frac{1}{A(z)}$$
  - hence, residual $E(z) = A(z)S(z)$ is 'flat' version of S
- **Signal whitening:**
  - white noise (independent $x[n]$s) has flat spectrum

  →whitening removes temporal correlation

Alternative LPC representations

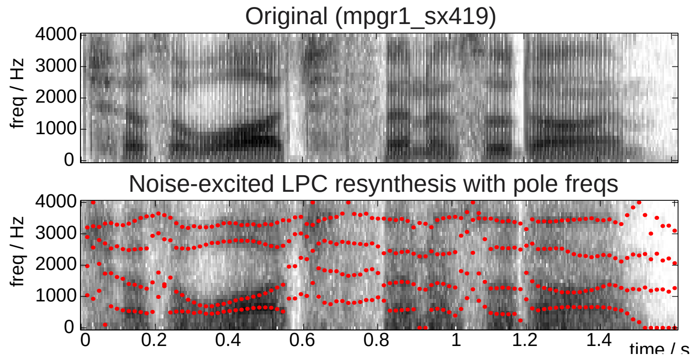
- **Many alternate p-dimensional representations:**
  - coefficients $\{a_i\}$
  - roots $\{\lambda_i\}$: $\prod(1 - \lambda_i z^{-1}) = 1 - \sum a_i z^{-1}$
  - line spectrum frequencies...
  - reflection coefficients $\{k_i\}$ from lattice form
  - tube model log area ratios $g_i = \log\left(\dfrac{1 - k_i}{1 + k_i}\right)$
- **Choice depends on:**
  - mathematical convenience/complexity
  - quantization sensitivity
  - ease of guaranteeing stability
  - what is made explicit
  - distributions as statistics

LPC Applications

- **Analysis-synthesis (coding, transmission):**
  - $S(z) = \dfrac{E(z)}{A(z)}$
    hence can reconstruct by filtering $e[n]$ with $\{a_i\}$s
  - whitened, decorrelated, minimized $e[n]$s
    are easy to quantize
  - .. or can model $e[n]$ e.g. as simple pulse train
- **Recognition/classification**
  - LPC fit responds to spectral peaks (formants)
  - can use for recognition (convert to cepstra?)
- **Modification**
  - separating source and filter supports cross-synthesis
  - pole / resonance model supports ‘warping’
    (e.g. male → female)

Aside: Formant tracking

- **Formants carry (most?) linguistic information**
- **Why not classify → speech recognition ?**
  - e.g. local maxima in cepstral-liftered spectrum pole frequencies in LPC fit
- **But: recognition needs to work in *all* circumstances**
  - formants can be obscure or undefined



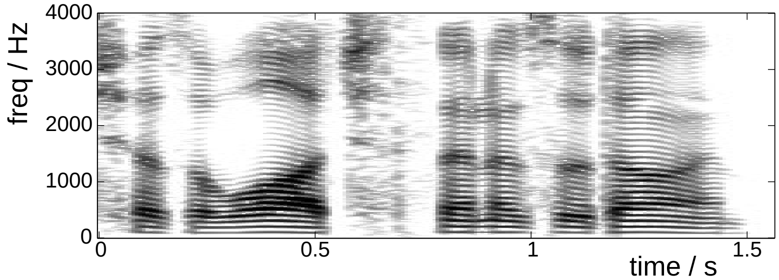
→ **Need more graceful, robust parameters**

Outline

1. Modeling speech signals
2. Spectral and cepstral modes
3. Linear predictive models (LPC)
4. **Other models**
   - Sinewave modeling
   - Harmonic+Noise model (HNM)
5. Speech synthesis

Other models: Sinusoid modeling

- **Early signal models required low complexity**
  - e.g. LPC
- **Advances in hardware open new possibilities...**
- **NB spectrogram suggests harmonics model:**



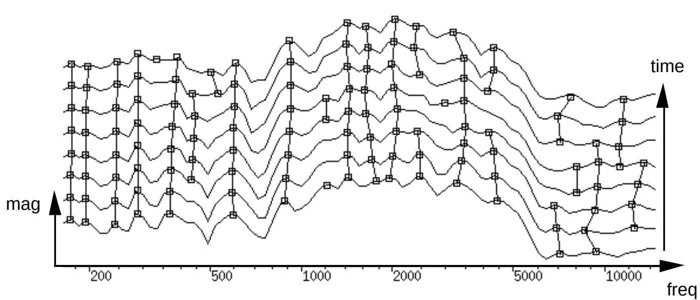
  - 'important' info in 2-D surface is set of tracks?
  - harmonic tracks have ~ smooth properties
  - straightforward resynthesis

Sine wave models

- **Model sound as sum of AM/FM sinusoids:**

$$s[n] = \sum_{k=1}^{N[n]} A_k[n]\cos(n \cdot \omega_k[n] + \phi_k[n])$$

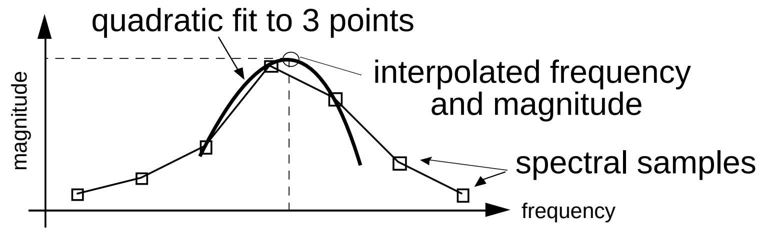
  - A_k, ω_k, ϕ_k piecewise linear or constant
  - can enforce harmonicity: $\omega_k = k.\omega_0$
- **Extract parameters directly from STFT frames:**



  - find local maxima of $|S[k,n]|$ along frequency
  - track birth/death & correspondence

Finding sinusoid peaks

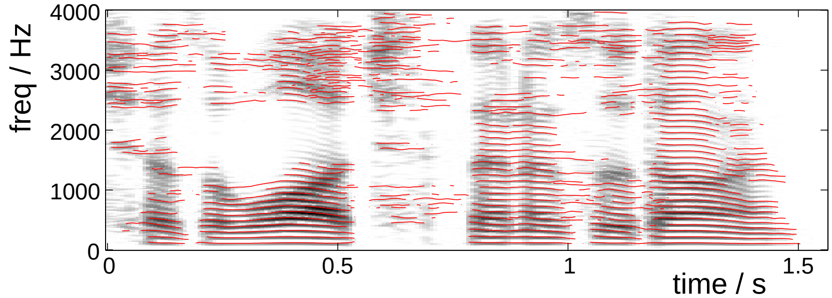
- **Look for local maxima along DFT frame**
  - i.e. $|S[k\text{-}1,n]| < |S[k,n]| > |S[k+1,n]|$
- **Want exact frequency of implied sinusoid**
  - DFT is normally quantized quite coarsely e.g. 4000 Hz / 256 bins = 15.6 Hz
  - interpolate at peaks via, e.g., quadratic fit



  - may also need interpolated unwrapped phase
- **Or, use differential of phase along time (pvoc):**
  - $\omega = \dfrac{a\dot{b} - b\dot{a}}{a^2 + b^2}$ where $S[k,n] = a + jb$

Sinewave modeling applications

- **Modification (interpolation) & synthesis**
  - connecting arbitrary ω & ϕ requires cubic phase interpolation (because $\omega = \dot{\phi}$)
- **Types of modification**
  - time & frequency scale modification .. with or without changing formant envelope
  - concatenation/smoothing boundaries
  - phase realignment (for crest reduction)
- **Non-harmonic signals? OK-ish**

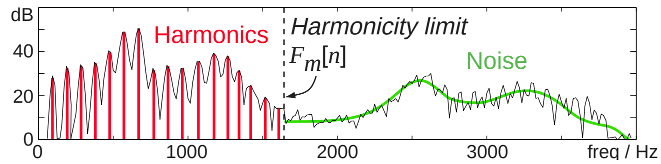


Harmonics + noise model

- **Motivation to improve sinusoid model because:**
  - problems with analysis of real (noisy) signals
  - problems with synthesis quality (esp. noise)
  - perceptual suspicions
- **Model:**

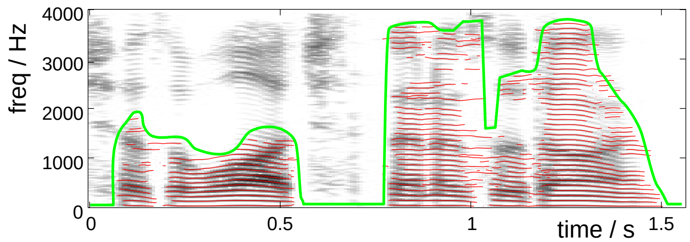
$$s[n] = \underbrace{\sum_{k=1}^{N[n]} A_k[n]\cos(n \cdot k \cdot \omega_0[n])}_{\text{Harmonics}} + \underbrace{e[n] \cdot (h_n[n] \otimes b[n])}_{\text{Noise}}$$

  - sinusoids are forced to be harmonic
  - remainder is filtered & time-shaped noise
- **'Break frequency' $F_m[n]$ between H and N:**

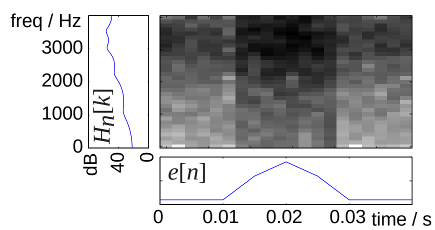


HNM analysis and synthesis

- **Dynamically adjust $F_m[n]$ based on 'harmonic test':**



- **Noise has envelopes in time $e[n]$ and freq H_n**



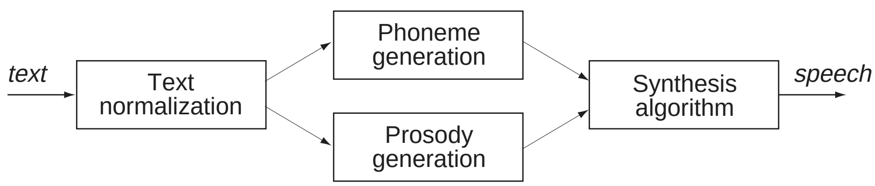
  - reconstruct bursts / synchronize to pitch pulses

Outline

1. Modeling speech signals
2. Spectral and cepstral modes
3. Linear predictive models (LPC)
4. Other models
5. **Speech synthesis**
   - Phone concatenation
   - Diphone synthesis

5 Speech synthesis

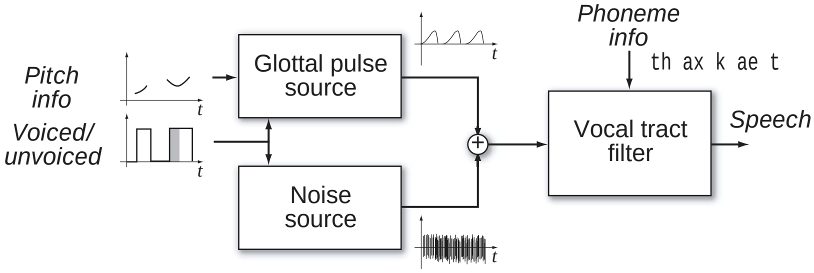
- **One thing you can do with models**
- **Synthesis easier than recognition?**
  - listeners do the work
  - .. but listeners are very critical
- **Overview of synthesis**



  - normalization disambiguates text (abbreviations)
  - phonetic realization from pronouncing dictionary
  - prosodic synthesis by rule (timing, pitch contour)
  - .. all controls waveform generation

Source-filter synthesis

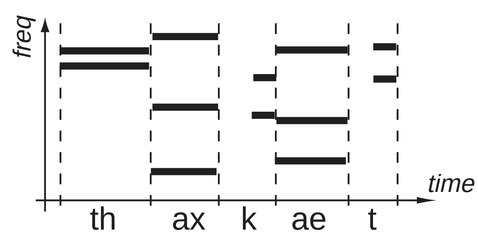
- **Flexibility of source-filter model is ideal for speech synthesis**



- **Excitation source issues:**
  - voiced / unvoiced / mixture ([th] etc.)
  - pitch cycle of voiced segments
  - glottal pulse shape → voice quality?

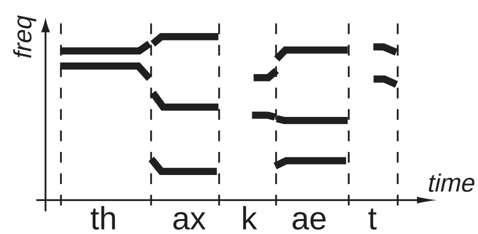
Vocal tract modeling

- **Simplest idea:**
  Store a single VT model for each phoneme



  - but: discontinuities are very unnatural

- **Improve by smoothing between templates**



  - trick is finding the right domain

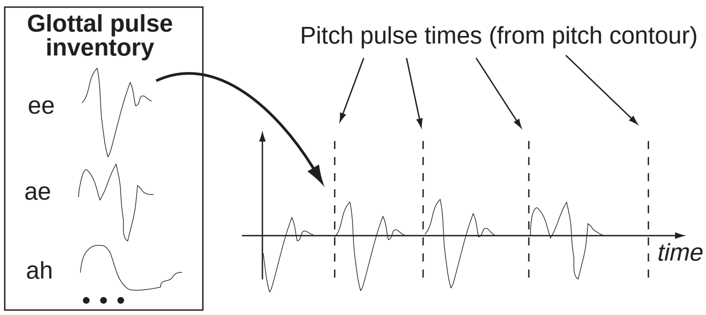
Cepstrum-based synthesis

- **Low-n cepstrum is compact model of target spectrum**
- **Can invert to get actual VT IR waveform:**

$$c_n = \mathrm{idft}(\log|\mathrm{dft}(x[n])|)$$

$$\rightarrow h[n] = \mathrm{idft}(\exp(\mathrm{dft}(c_n)))$$

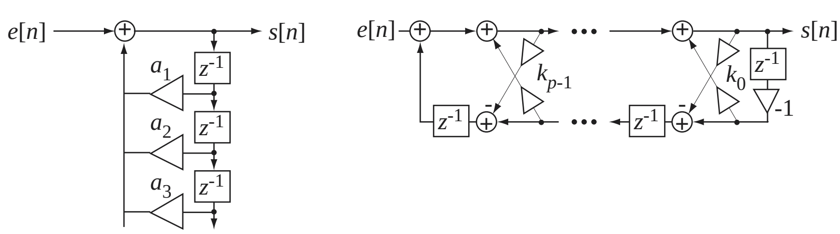
- **All-zero (FIR) VT response**
  → can pre-convolve with glottal pulses



  - cross-fading between templates is OK

LPC-based synthesis

- **Very compact representation of target spectra**
  - 3 or 4 pole pairs per template
- **Low-order IIR filter → very efficient synthesis**
- **How to interpolate?**
  - cannot just interpolate a_i in a running filter
  - but: lattice filter has better-behaved interpolation



- **What to use for excitation**
  - residual from original analysis
  - reconstructed periodic pulse train
  - parameterized residual resynthesis

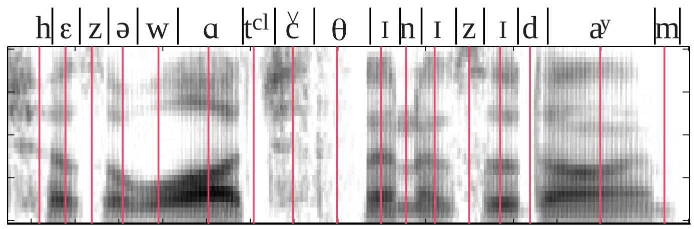
Diphone synthesis

- **Problems in phone-concatenation synthesis**
  - phonemes are context-dependent
  - coarticulation is complex
  - transitions are critical to perception

→ **store *transitions* instead of just phonemes**

Phones

Diphone segments

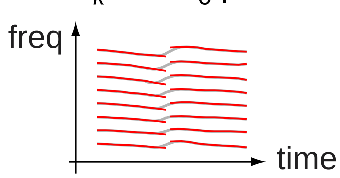


  - ~40 phones → 800 diphones
  - or even more context if have a larger database

- **How to splice diphones together?**
  - TD-PSOLA: align pitch pulses and cross-fade
  - MBROLA: normalized, multiband

HNM synthesis

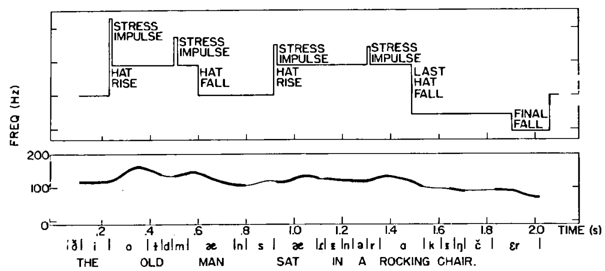
- **High quality resynthesis of real diphone units + parametric representation for modifications**
  - pitch, timing modifications
  - removal of discontinuities at boundaries
- **Synthesis procedure:**
  - linguistic processing gives phones, pitch, timing
  - database search gives best-matching units
  - use HNM to fine-tune pitch & timing
  - cross-fade A_k and ω_0 parameters at boundaries



- **Careful preparation of database is key**
  - sine models allow phase alignment of all units
  - larger database improves unit match

Generating prosody

- **The real factor limiting speech synthesis?**
- **Waveform synthesizers have inputs for**
  - intensity (stress)
  - duration (phrasing)
  - fundamental frequency (pitch)
- **Curves produced by superposition of (many) inferred linguistic rules**
  - phrase final lengthening, unstressed shortening..



- **Or learn rules from transcribed examples**

Summary

- **Range of models:**
  - spectral, cepstral
  - LPC, Sinusoid, HNM
- **Range of applications:**
  - general spectral shape (filterbank) → ASR
  - precise description (LPC+residual) → coding
  - pitch, time modification (HNM) → synthesis
- **Issues:**
  - performance vs. computational complexity
  - generality vs. accuracy
  - representation size vs. quality

Parting thought:

not all parameters are created equal ...