Lecture 1

Introduction/Signal Processing, Part I

Michael Picheny, Bhuvana Ramabhadran, Stanley F. Chen, Markus Nussbaum-Thom

Watson Group
IBM T.J. Watson Research Center
Yorktown Heights, New York, USA
{picheny,bhuvana,stanchen,nussbaum}@us.ibm.com

20 January 2016

Part I

Introduction

Three Questions

- Why are you taking this course?
- What do you think you might learn?
- How do you think this may help you in the future?

What Is Speech Recognition?

- Converting speech to text (STT).
  - a.k.a. automatic speech recognition (ASR).
- What it's not.
  - *Natural language understanding* — *e.g.*, Siri.
  - *Speech synthesis* — converting text to speech (TTS), *e.g.*, Watson.
  - *Speaker recognition* — identifying who is speaking.

Why Is Speech Recognition Important?

Because It's Fast

modality	*method*	*rate (words/min)*
sound	speech	150–200
sight	sign language; gestures	100–150
touch	typing; mousing	60
taste	dipping self in different flavorings	<1
smell	spraying self with perfumes	<1

Because it's easier to process text than audio

VS

Because It's Hands Free

Because It's a Natural Form of Communication

Key Applications

- Transcription: archiving/indexing audio.
  - Legal; medical; television and movies.
  - Call centers.
- Whenever you interact with a computer . . .
  - Without sitting in front of one.
  - *e.g.*, smart or dumb phone; car; home entertainment.
- Accessibility.
  - People who can't type, or type slowly.
  - The hard of hearing.

Why Study Speech Recognition?

- Learn a lot about many popular machine learning techniques.
  - They all originated in speech.
- Be exposed to a real problem with real data — no artificial ingredients.
- Learn how to build a complex end-to-end system.
  - Toto, we aren't in Kansas anymore!
- Not solved yet, so maybe you will be inspired to make it your life's work — like we have!

Where Are We?

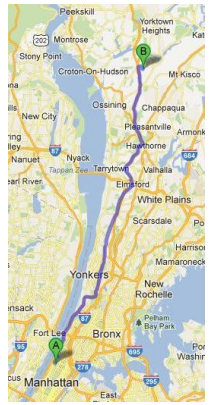
Who Are We?

- Stanley F. Chen: Productive Researcher
- Markus Nussbaum-Thom: Productive Researcher
- Bhuvana Ramabhadran: Useless Manager
- Michael Picheny: Even More Useless Senior Manager

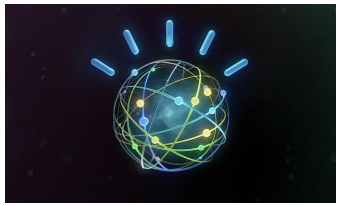
We are all from the Watson Multimodal Group located at the IBM T.J. Watson Research Center in Yorktown Heights, NY.







What is the Watson Group?



The application of technologies that allow:

- A business to load and process tremendous volumes of data developed in formats designed for human consumption
- Queries made to the data in human language in order to assist in .
  - Asking
  - Discovering
  - Deciding
- Responses to be developed through an "understanding" of natural language and provided with a degree of certainty of their accuracy.

In other words, Watson allows us to create the equivalent of a human expert. One that:

- Works 24 X 7 X 365
- Can talk to hundreds, if not thousands of people simultaneously
- Does not forget anything
- Will not leave and go to a competitor
- Will not win the lottery on Saturday and retire on Monday
- Can continue to learn forever

Why Four Professors?

- Too much knowledge to fit in one brain.
  - Signal processing.
  - Probability and statistics.
  - Phonetics; linguistics.
  - Natural language processing.
  - Machine learning; artificial intelligence.
  - Automata theory.
  - Optimization.

How To Contact Us

- **In E-mail, prefix subject line with "EECS E6870:"!!!.**
  - Michael Picheny — `picheny@us.ibm.com`.
  - Bhuvana Ramabhadran — `bhuvana@us.ibm.com`.
  - Stanley F. Chen — `stanchen@us.ibm.com`.
  - Markus Nussbaum-Thom — `nussbaum@us.ibm.com`.
- Office hours: right after class.
  - Before class by appointment.
- TA: TBD
- Courseworks.
  - For posting questions about labs.

Course Outline

week	lecture	topic	assigned	due
1	1	Introduction		
2	2	Signal processing; DTW	lab 1	
3	3	Gaussian mixture models		
4	4	Hidden Markov models	lab 2	lab 1
5	5	Language modeling 101		
6	6	Pronunciation modeling	lab 3	lab 2
7	7	Training Speech Recognition Systems		
8	8	The Search Problem	lab 4	lab 3
9	recess			
10	9	The Search Problem, continued		
11	10	Language Modeling 201	lab 5	lab 4
12	11	Robustness and Adaptation		
13	12	Discriminative Training, ROVER and Consensus		lab 5
14	13	Neural Networks 101		
15	14	Neural Networks 201		
16	study			
17		Project Presentations		project

Programming Assignments

- 80% of grade ($\sqrt{-}$, $\sqrt{}$, $\sqrt{+}$ grading).
- Some short written questions.
- Write key parts of basic large vocabulary continuous speech recognition system.
  - Only the "fun" parts.
  - C++ code infrastructure provided by us.
- Get account on ILAB computer cluster (x86 Linux PC's).
  - Login to cluster using `ssh`.
  - Can't run labs on PC's/Mac's.
- If not yet signed up for course, but going to add:
  - Fill out index card with name, UNI, and E-mail address.
  - Or E-mail this info to `stanchen@us.ibm.com`.

Final Project

- 20% of grade.
- Option 1: Reading project (individual).
  - Pick paper(s) from provided list, or propose your own.
  - Write 1500–2500 word paper reviewing + analyzing paper(s).
- Option 2: Programming/experimental project (group).
  - Pick project from provided list, or propose your own.
  - Group gives 10–15m presentation summarizing project *and* writes paper.
  - 40% of grade (if helps).

Readings

- PDF versions of readings will be available on the web site.
- Recommended text:
  - *Speech Synthesis and Recognition*, Holmes, 2nd edition (paperback, 256 pp., 2001) **[Holmes]**.
- Reference texts:
  - *Theory and Applications of Digital Signal Processing*, Rabiner, Schafer (hardcover, 1056 pp., 2010) **[R+S]**.
  - *Speech and Language Processing*, Jurafsky, Martin (2nd edition, hardcover, 1024 pp., 2000) **[J+M]**.
  - *Statistical Methods for Speech Recognition*, Jelinek (hardcover, 305 pp., 1998) **[Jelinek]**.
  - *Spoken Language Processing*, Huang, Acero, Hon (paperback, 1008 pp., 2001) **[HAH]**.

Web Site

`tinyurl.com/e6870s16` $\Rightarrow$
`www.ee.columbia.edu/~stanchen/spring16/e6870/`

- Syllabus.
- Slides from lectures (PDF).
  - Online *after* each lecture.
  - Save trees — no hardcopies!
- Lab assignments (PDF).
- Reading assignments (PDF).
  - Online by lecture they are assigned.
  - Username: *speech*, password: *pythonrules*.

Prerequisites

- Basic knowledge of probability and statistics.
- Willingness to implement algorithms in C++.
  - Only basic features of C++ used; $\sim$100 lines/lab.
- Basic knowledge of Unix or Linux.
- Knowledge of digital signal processing optional.
  - Helpful for understanding signal processing lectures; *i.e.*, CS majors may find signal processing material baffling!
  - Not needed for labs!

Help Us Help You

- Feedback questionnaire after each lecture (2 questions).
  - Feedback welcome any time.
- You, the student, are partially responsible . . .
  - For the quality of the course.
- Please ask questions anytime!
- EE's may find CS parts challenging, and vice versa.
- Together, we can get through this.
- Let's go!

Where Are We?

What is the basic goal?

- Recognize as many words correctly as possible.

Obvious Success Metric – Word Error Rate (WER):

– 100 x (Substitutions + Deletions + Insertions) / (Total Words in Reference transcripts)

Ref:	THE	CAT	IN	ON	THE		HAT	
Hyp:		CAT	IS		THE	GREEN		HAT
	Del		Sub	Ins		Ins		

Error rate = 100 x (1 S + 1 D + 2 I) / 5 = 80%

- Use those algorithms that lower the Word Error Rate
- Imperfect but very useful simple to measure objective criterion

Why is this difficult? (Part I)

Speaker Variation

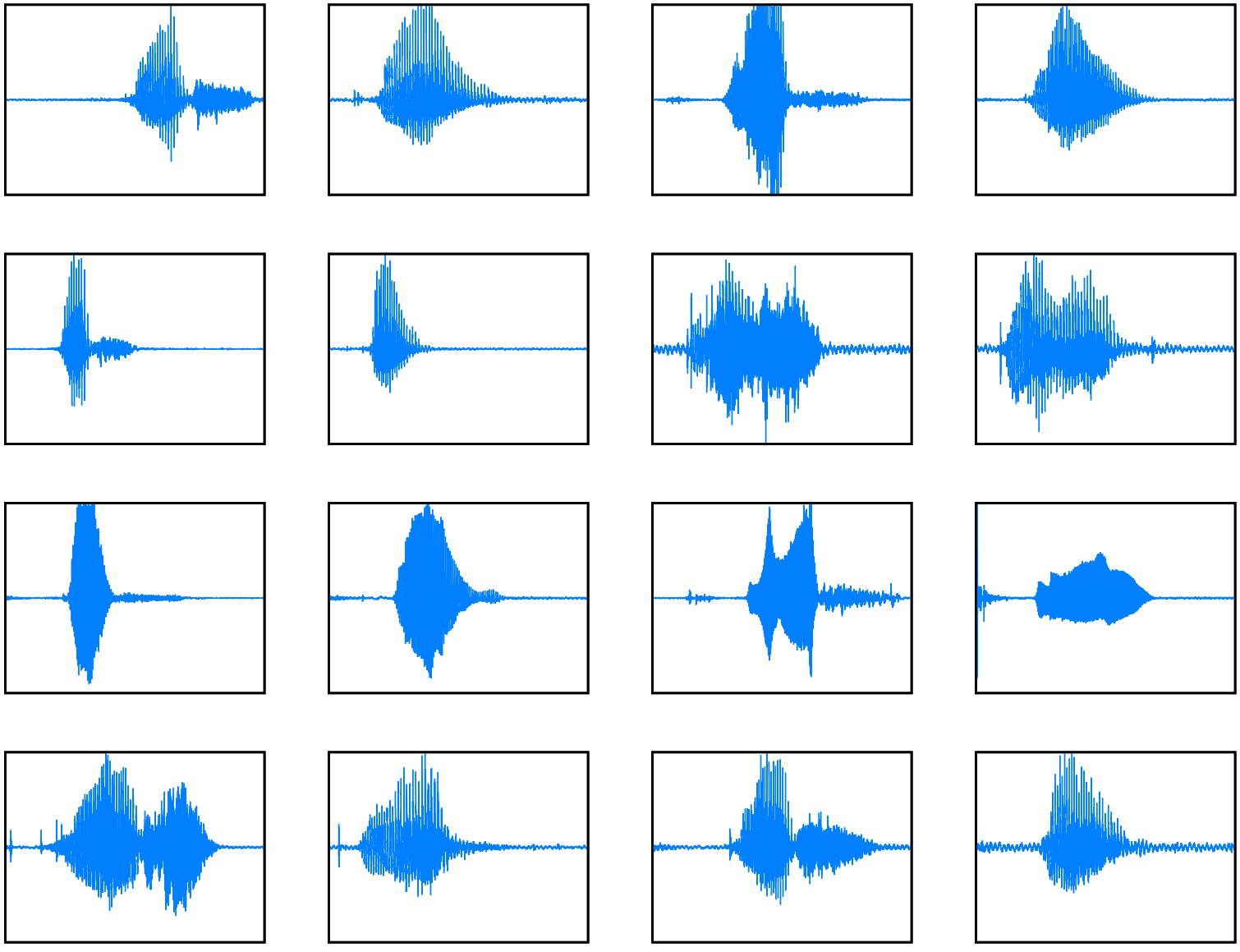
Channel Variation

Background Noise

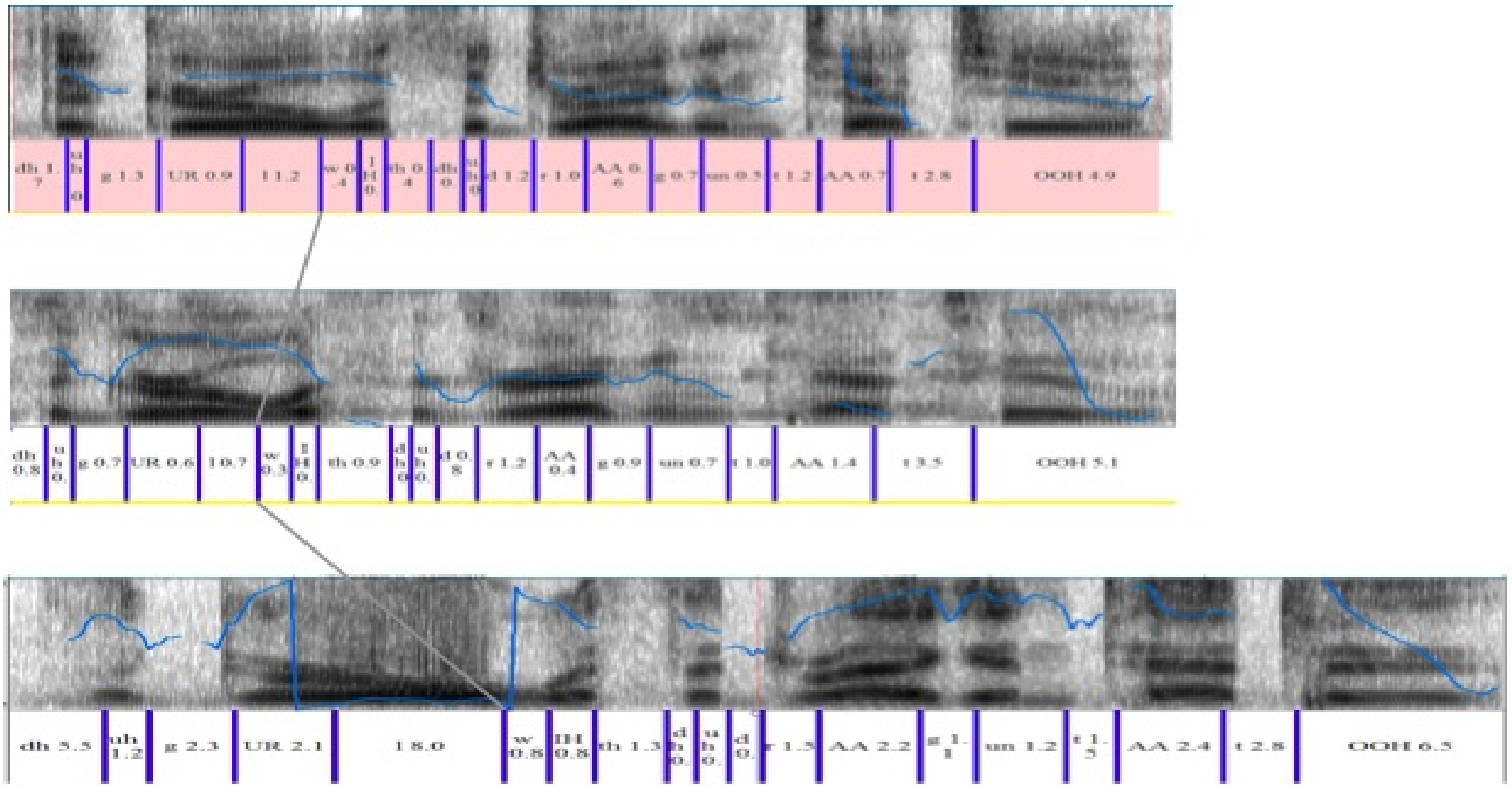
Accent

Speaking Style

A Thousand Times No!



Why is this difficult? (Part II)



Inherent variability of Speech biggest challenge

Basic Concepts

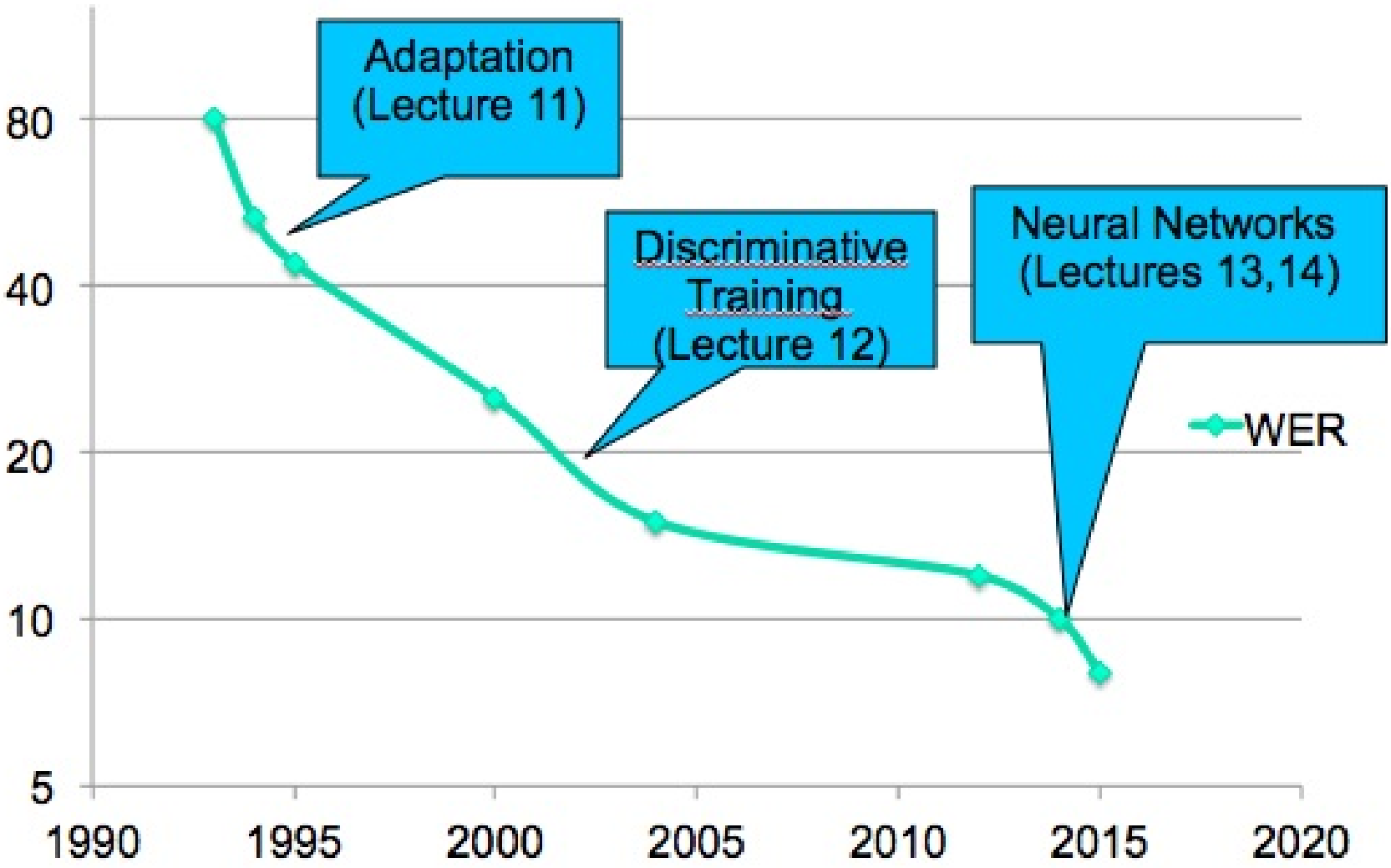
Choose W to maximize:

$$P(W|A) = \frac{P(A|W)\,P(W)}{\cancel{P(A)}}$$

W = vocabulary
A = extracted features from the speech signal (Lectures 1 and 2)
P(A|W) = Acoustic Model (Lectures 3 and 4, 6, 7)
P(W) = Language Model (Lecture 5)

Hypothesis Search (Lecture 8, 9)

Historical Developments



Where Are We?

The Early Years: 1950–1960's

- *Ad hoc* methods.
  - Many key ideas introduced; not used all together.
  - *e.g.*, spectral analysis; statistical training; language modeling.
- Small vocabulary.
  - Digits; yes/no; vowels.
- Not tested with many speakers (usually $<$10).

The Birth of Modern ASR: 1970–1980's

Every time I fire a linguist, the performance of the speech recognizer goes up.

—Fred Jelinek, IBM

- Ignore (almost) everything we know about phonetics, linguistics.
- View speech recognition as . . . .
  - Finding *most probable* word sequence given audio.
  - Train probabilities automatically w/ transcribed speech.

The Birth of Modern ASR: 1970–1980's

- Many key algorithms developed/refined.
  - Expectation-maximization algorithm; *n*-gram models; Gaussian mixtures; Hidden Markov models; Viterbi decoding; etc.
- Computing power still catching up to algorithms.
  - First real-time dictation system built in 1984 (IBM).
  - Specialized hardware required — had the computation power of a 60 MHz Pentium.

The Golden Years: 1990's–now

	1994	now
CPU speed	60 MHz	3 GHz
training data	$<$10h	10000h+
output distributions	GMM	NN /GMM hybrids
sequence modeling	HMM	HMM and/or NN
language models	*n*-gram	*n*-gram and NN

- Basic algorithms have remained similar but now seeing huge penetration of NN technologies.
- Significant performance gains can also be attributed to presence of more data, faster CPU's, and more run-time memory.

Person *vs.* Machine (Lippmann, 1997)

task	machine	human	ratio
Connected Digits[1]	0.72%	0.009%	80×
Letters[2]	5.0%	1.6%	3×
Resource Management	3.6%	0.1%	36×
WSJ	7.2%	0.9%	8×
Switchboard	43%	4.0%	11×

- For humans, one system fits all; for machine, not.
- Today: Switchboard WER $< 8\%$.
  - But that is with 2000 hours of SWB training data; can't assume this is always available.

[1] String error rates.

[2] Isolated letters presented to humans; continuous for machine.

Commercial Speech Recognition

- 1995 – 1998 — first large vocabulary speaker dependent dictation systems.
- 1996 – 2005 — first telephony- based customer assistance systems.
- 2003 – 2007 — first automotive interactive systems.
- 2008 – 2010 — first voice search systems.
- 2011 – today — growth of cloud-based speech services.

What's left?

- Accents
- Noise
- Far field microphones
- Informal speech

Are We Awake?

- Of the time you spend interacting with devices . . .
- What fraction do you use ASR?
- What fraction would it be if ASR were perfect?
- What are the biggest problems with current ASR performance?

The First Two Lectures

- A little background on speech production and perception.
- *signal processing* — Extract *features* from audio $A \Rightarrow A' \ldots$
  - That discriminate between different words.
  - Normalize for volume, pitch, voice quality, noise, ...
- *dynamic time warping* —Handling time/rate variation.

Where Are We?

Data-Driven *vs.* Knowledge-Driven

- Don't ignore *everything* we know about speech, language.

- Knowledge/concepts that have proved useful.
  - Words; phonemes.
  - A little bit of human production/perception.
- Knowledge/concepts that haven't proved useful (yet).
  - Nouns; vowels; syllables; voice onset time; . . .

Finding Good Features

- Extract features from audio . . .
  - That help determine word identity.
- What are good types of features?
  - Instantaneous air pressure at time t?
  - Loudness at time t?
  - Energy or phase for frequency ω at time t?
  - Estimated position of speaker's lips at time t?
- Look at human production and perception for insight.
  - Also, introduce some basic speech terminology.

Speech Production

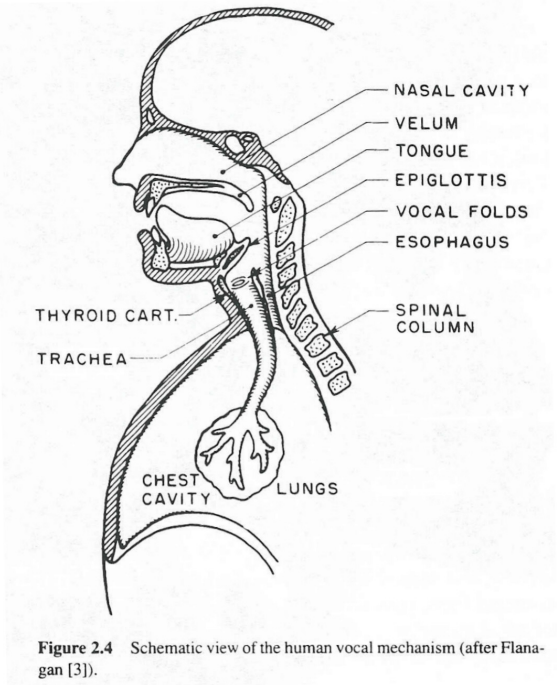


Figure 2.4 Schematic view of the human vocal mechanism (after Flanagan [3]).

- Air comes out of lungs.
- Vocal cords tensed (vibrate $\Rightarrow$ voicing) or relaxed (unvoiced).
- Modulated by vocal tract (glottis $\rightarrow$ lips); resonates.
  - Articulators: jaw, tongue, velum, lips, mouth.

Speech Consists Of a Few Primitive Sounds?

- Phonemes.
  - 40 to 50 for English.
  - Speaker/dialect differences.
  - *e.g.*, do MARY, MARRY, and MERRY rhyme?
  - Phone: acoustic realization of a phoneme.
- May be realized differently based on context.
  - *allophones*: different ways a phoneme can be realized.
  - *e.g.*, P in SPIN, PIN are two different allophones of P.

spelling	phonemes
SPIN	S P IH N
PIN	P IH N

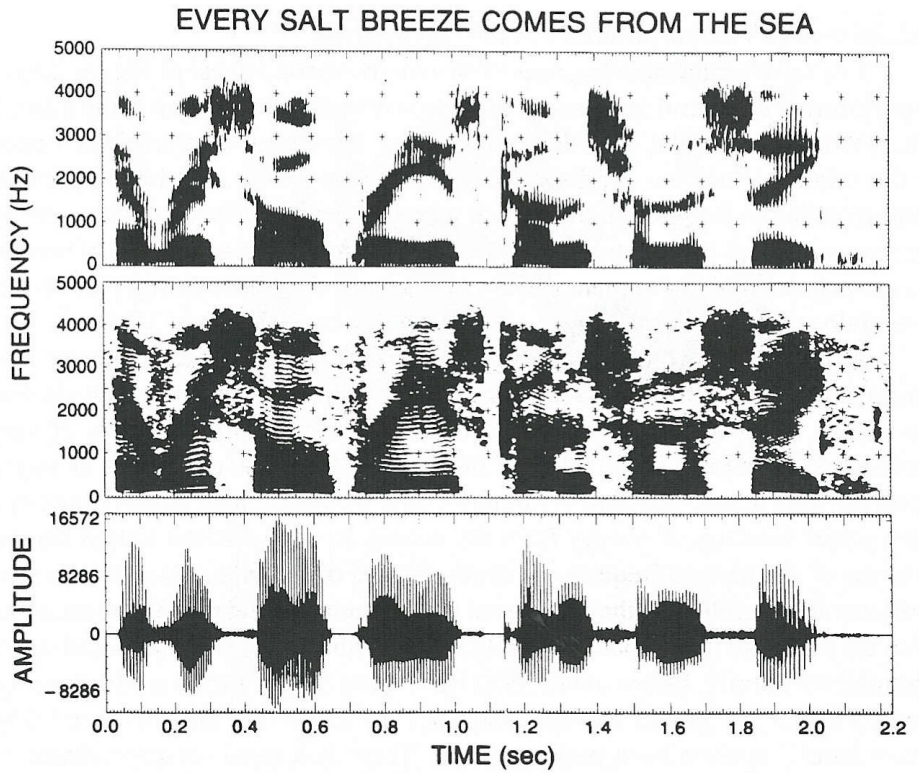
  - *e.g.*, T in BAT, BATTER; A in BAT, BAD.

Classes of Speech Sounds

- Can categorize phonemes by how they are produced.
- Voicing.
  - *e.g.*, F (unvoiced), V (voiced).
  - All vowels are voiced.
- Stops/plosives.
  - Oral cavity blocked (*e.g.*, lips, velum); then opened.
  - *e.g.*, P, B (lips).

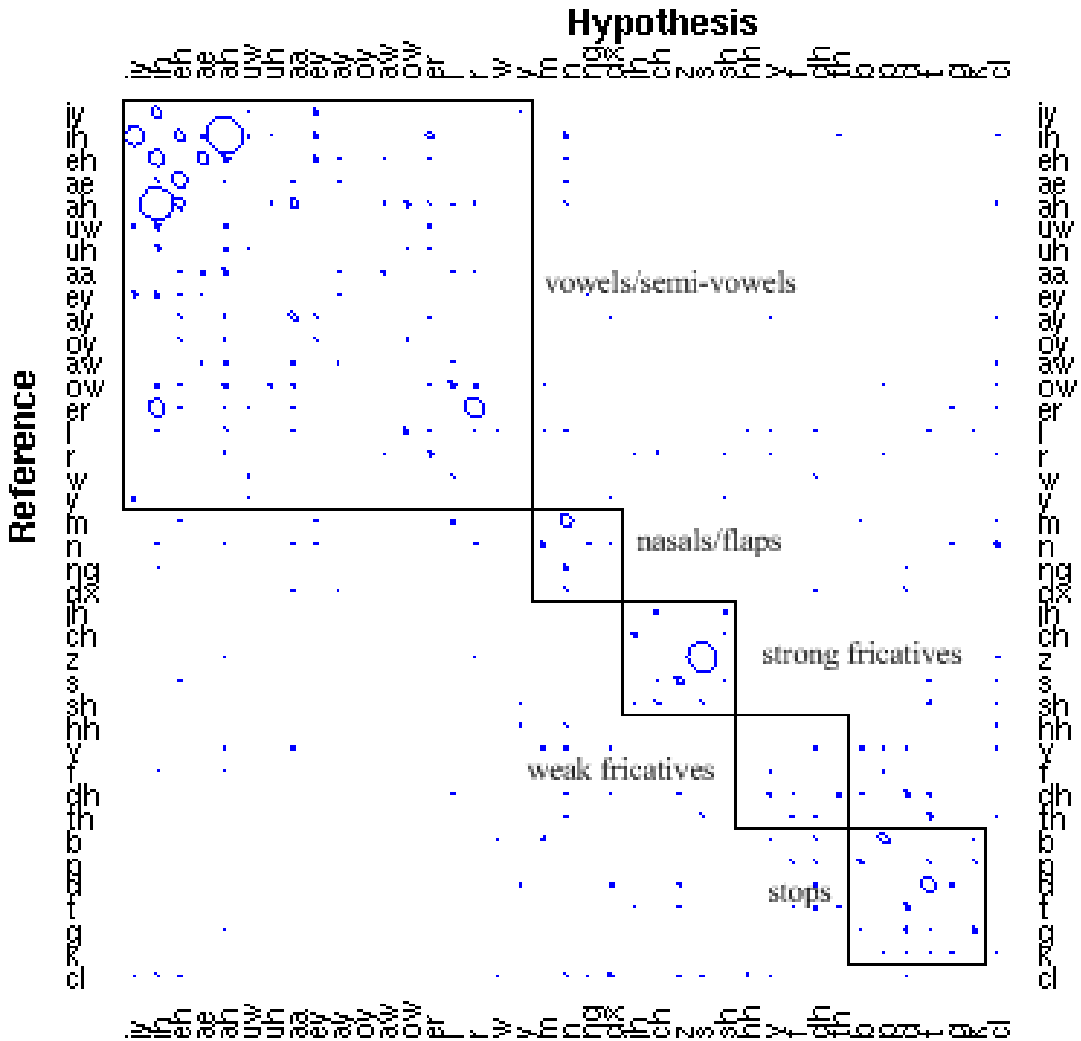
Classes of Speech Sounds

- Spectogram shows energy at each frequency over time.
- Voiced sounds have pitch (F0); formants (F1, F2, F3).
- Very highly trained humans can do recognition on spectrograms with high accuracy so this is a valid representation.



Classes of Speech Sounds

- What can the machine do? Here is a sample on TIMIT:



Classes of Speech Sounds

- Vowels — EE, AH, etc.
  - Differ in locations of formants.
  - Dipthongs — transition between two vowels (*e.g.*, COY, COW).
- Consonants.
  - Fricatives — F, V, S, Z, SH, J.
  - Stops/plosives — P, T, B, D, G, K.
  - Nasals — N, M, NG.
  - Semivowels (liquids, glides) — W, L, R, Y.

Coarticulation

- Realization of a phoneme can differ very much depending on context (allophones).
- Where articulators were for last phone affect how they transition to next.

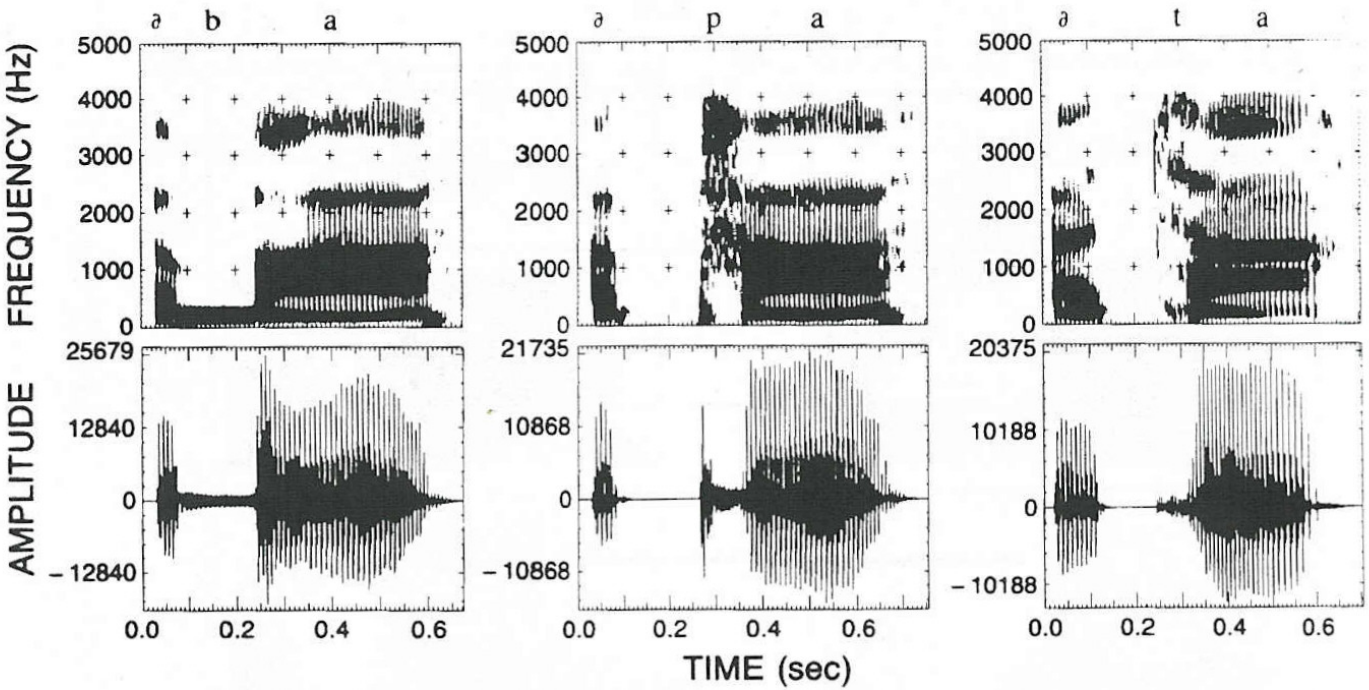


Figure 2.28 Spectrogram comparisons of the sequences of voiced (/ə-b-a/) and voiceless (/ə-p-a/ and /ə-t-a/) stop consonants.

Speech Production and ASR

- Directly use features from acoustic phonetics?
  - *e.g.*, (inferred) location of articulators; voicing; formant frequencies.
  - In practice, has not been made to work
- Still, influences how signal processing is done.
  - Source-filter model.
  - Separate excitation from modulation from vocal tract.
  - *e.g.*, frequency of excitation can be ignored (English).

Speech Perception and ASR

- As it turns out, the features that work well . . . .
  - Motivated more by speech perception than production.
- *e.g.*, Mel Frequency Cepstral Coefficients (MFCC).
  - Motivated by human perception of pitch.
  - Similarly for perceptual linear prediction (PLP).

Speech Perception — Physiology

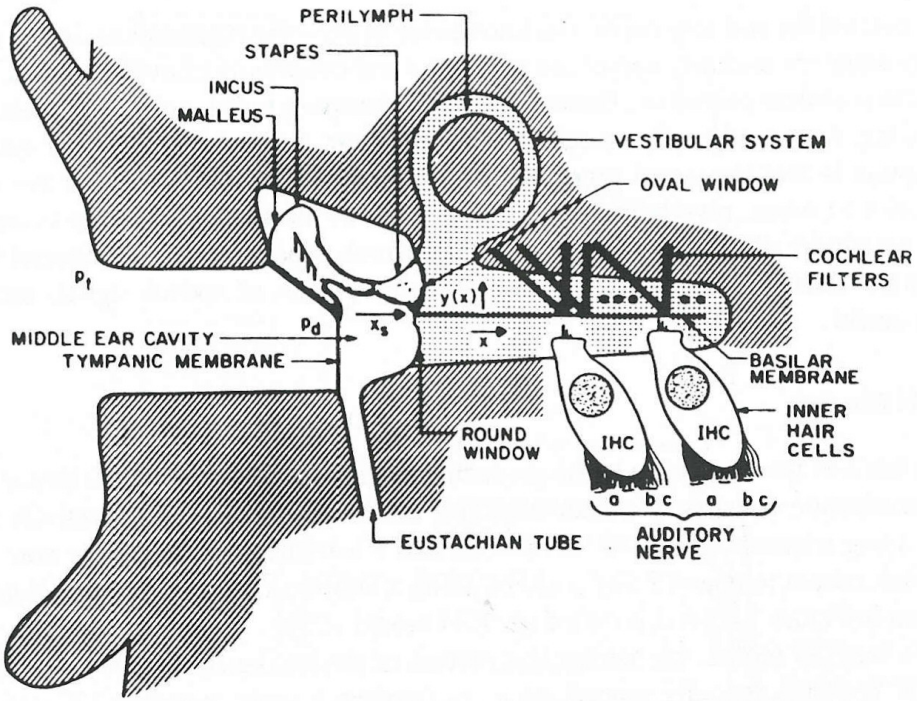


Figure 3.48 Expanded view of the middle and inner ear mechanics.

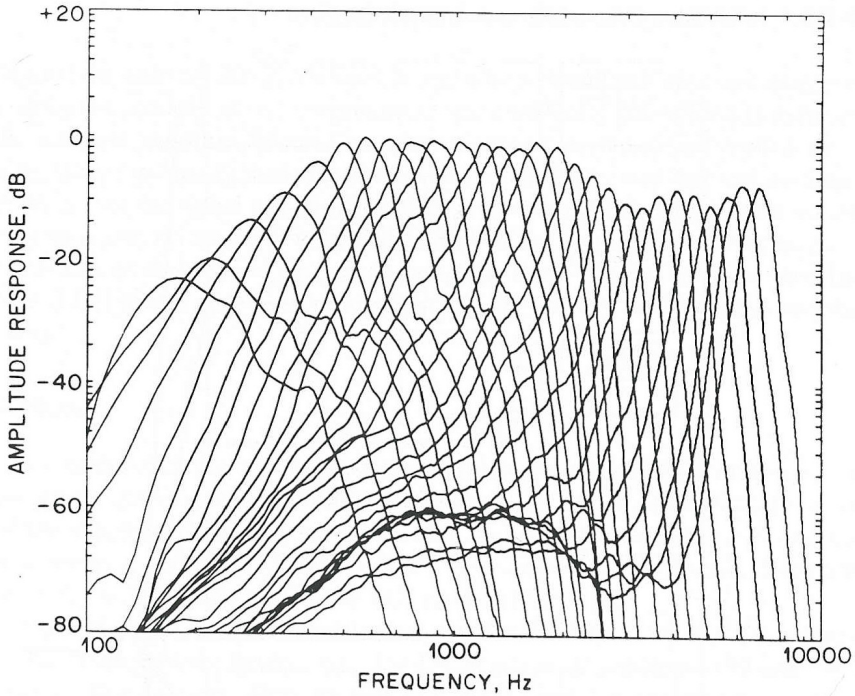


Figure 3.50 Frequency response curves of a cat's basilar membrane (after Ghitza [13]).

- Sound enters ear; converted to vibrations in cochlear fluid.
- In fluid is basilar membrane, with $\sim$30,000 little hairs.
  - Sensitive to different frequencies (band-pass filters).

Speech Perception — Physiology

- Human physiology used as justification for frequency analysis ubiquitous in speech processing.
- Limited knowledge of higher-level processing.
  - Can glean insight from psychophysical experiments (relationship between physical stimuli and human responses)

Speech Perception — Psychophysics

- Sound Pressure Level (SPL) in dB = $20 \log_{10} P/P_0$
- $P_0 \Leftrightarrow$ threshold of hearing at 1 KHz (it varies!)

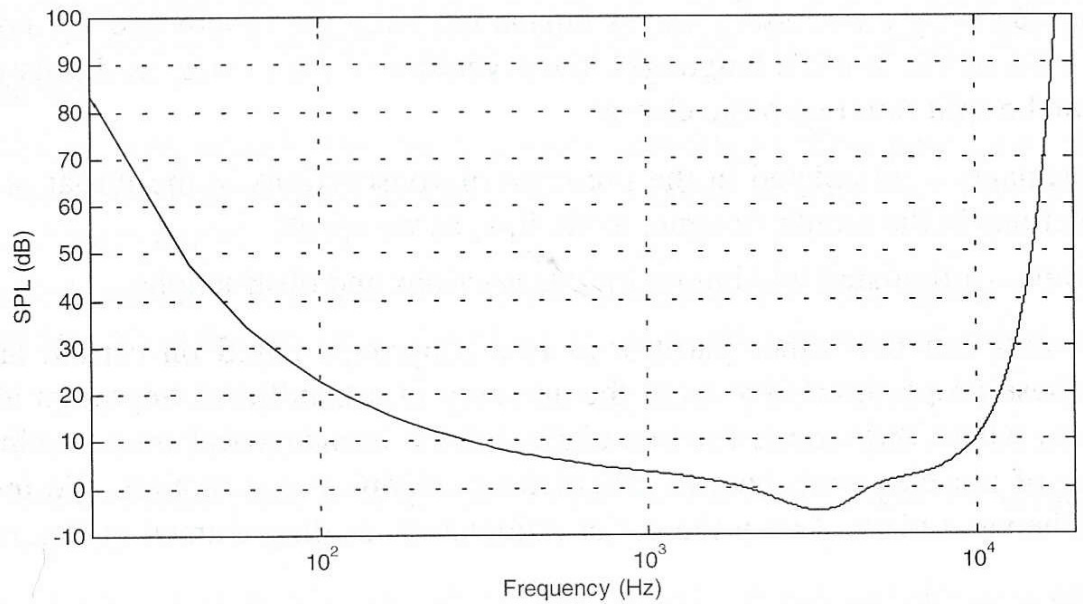
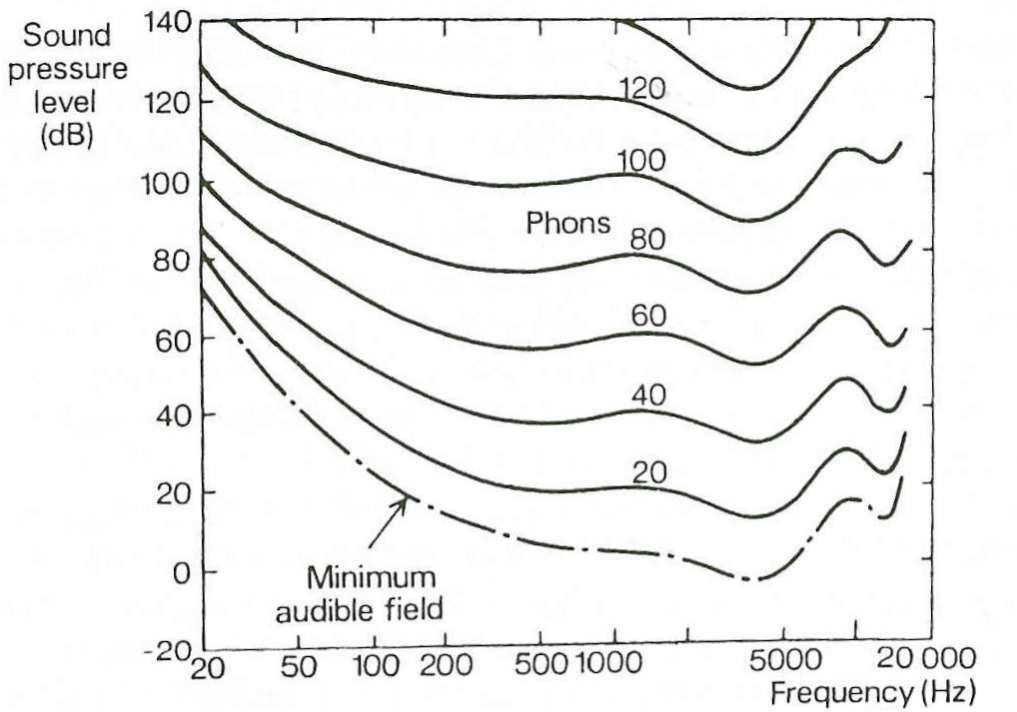


Figure 2.3 The sound pressure level (SPL) level in dB of the absolute threshold of hearing as a function of frequency. Sounds below this level are inaudible. Note that below 100 Hz and above 10 kHz this level rises very rapidly. Frequency goes from 20 Hz to 20 kHz and is plotted in a logarithmic scale from Eq. (2.3).

Speech Perception — Psychophysics

- Different sensitivity of humans to different frequencies.
- Equal loudness contours.
  - Subjects adjust volume of tone to match volume of another tone at different pitch.
- Tells us what range of frequencies may be good to focus on.



Speech Perception — Psychophysics

- Human perception of distance between frequencies.
- Adjust pitch of one tone until twice/half pitch of other tone.
- Mel scale — frequencies equally spaced in Mel scale are equally spaced according to human perception.

$$\text{Mel freq} = 2595 \log_{10}(1 + \text{freq}/700)$$

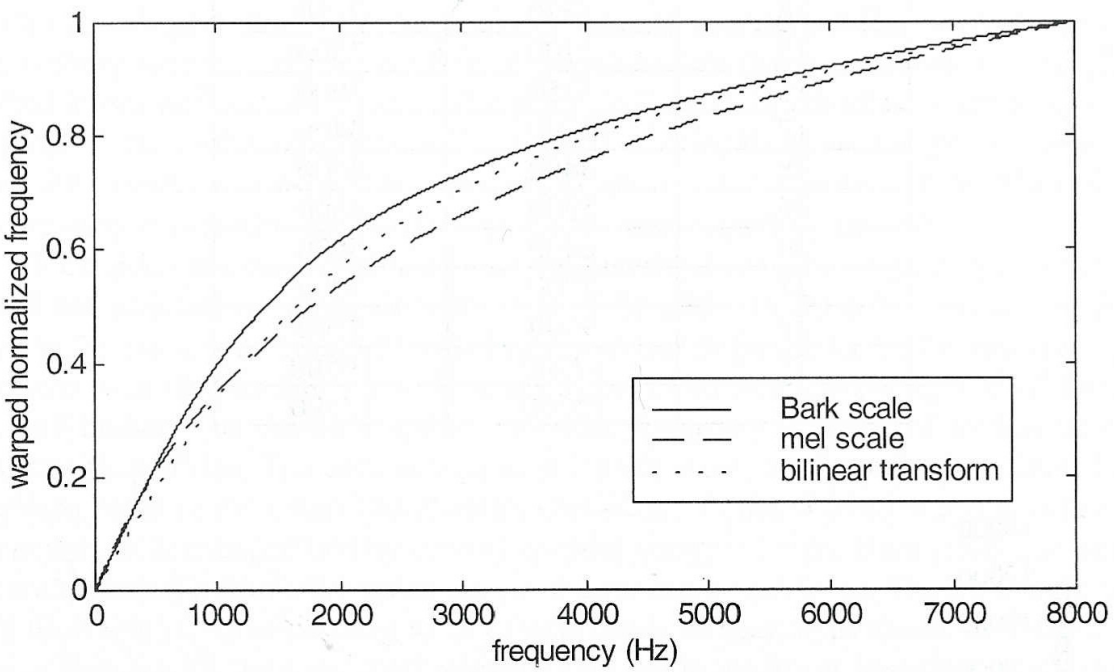


Figure 2.13 Frequency warping according to the Bark scale, ERB scale, mel-scale, and bilinear nsform for $\alpha = 0.6$: linear frequency in the x-axis and normalized frequency in the y-axis.

Speech Perception — Machine

- Just as human physiology has its quirks . . .
  - So does machine “physiology”.
- Sources of distortion.
  - Microphone — different response based on direction and frequency of sound.
  - Sampling frequency — *e.g.*, 8 kHz sampling for landlines throws away all frequencies above 4 kHz.
  - Analog/digital conversion — need to convert to digital with sufficient precision (8–16 bits).
  - Lossy compression — *e.g.*, cellular telephones, VOIP.

Speech Perception — Machine

- Input distortion can still be a significant problem.
  - Mismatched conditions between train/test.
  - Low bandwidth — telephone, cellular.
  - Cheap equipment — *e.g.*, mikes in handheld devices.
- Enough said.

Are We Awake?

- Sometimes it helps to mimic nature; sometimes not (*e.g.*, airplanes and flying).
  - Which way should be best for ASR in the long run?
- Does it make more sense to mimic human speech production or perception?
- Why do humans have two ears, and what does this mean for ASR?

Segue

- Now that we see what humans do.
- Let's discuss what signal processing has been found to work well empirically.
  - Has been tuned over decades.
- Start with some mathematical background.

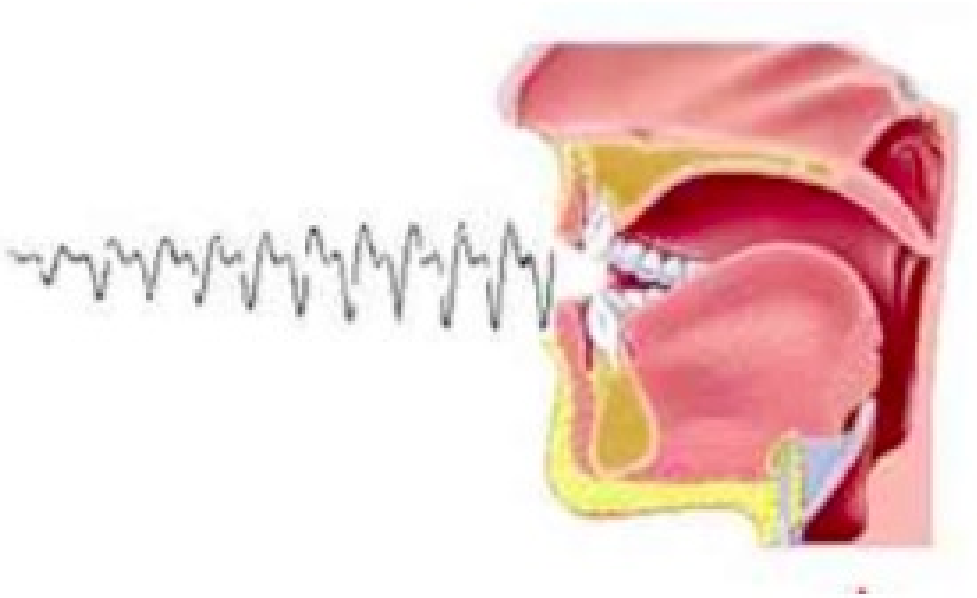
Part II

Signal Processing Basics

Overview

- Background material: how to mathematically model/analyze human speech production and perception.
  - Introduction to signals and systems.
  - Basic properties of linear systems.
  - Introduction to Fourier analysis.
- Next week: discussion of actual features used in ASR.
- Recommended readings: **[HAH]** pg. 201-223, 242-245. **[R+J]** pg. 69-91. All figures taken from these texts.

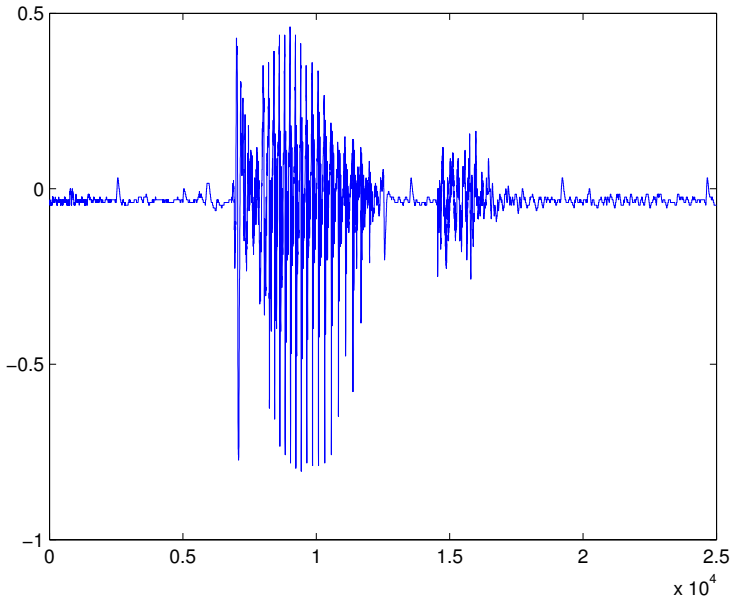
Speech Production



The sound pressure modulations can be captured by a microphone, converted to an electrical signal, and then digitized creating a sequence of numbers we call a "signal".

Signals and Systems

- Signal: a function $x[n]$ over time .
  - *e.g.*, output of microphone attached to an A/D converter.



- A digital *system* (or *filter*) H takes an input signal $x[n]$ and produces a signal $y[n]$:

$$y[n] = H(x[n])$$

What do we need to do to this signal to be useful for speech recognition?

- Model signal as being generated from a set of time-varying physiological variables (vocal tract geometry, glottal vibration, lip radiation, etc.) and extract these variables from the signal.
- **Operate on the signal to mimic some of the processing done in the auditory system, for example — frequency analysis**.

Either way we want to make things simple, so we will focus on *linear* processing.

Linear Time-Invariant Systems

- Calculating output of H for input signal x becomes very simple if digital system H satisfies two basic properties.
- H is *linear* if

$$H(a_1 x_1[n] + a_2 x_2[n]) = a_1 H(x_1[n]) + a_2 H(x_2[n])$$

- H is *time-invariant* if

$$y[n - n_0] = H(x[n - n_0])$$

i.e., a shift in the time axis of x produces the same output, except for a time shift.

Linear Time-Invariant (LTI) Systems

- Let H be a linear system. Define

$$h(n) = H(x(n) = \delta[n]), \delta(n = 0) = 1, \delta(n \neq 0) = 0$$

  $h(n)$ is called the *impulse response* of the system.
- Then, by the LTI properties $H(x[n])$ can be written as

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = \sum_{k=-\infty}^{\infty} x[n-k]h[k]$$

- The above is also known as *convolution* and is written as

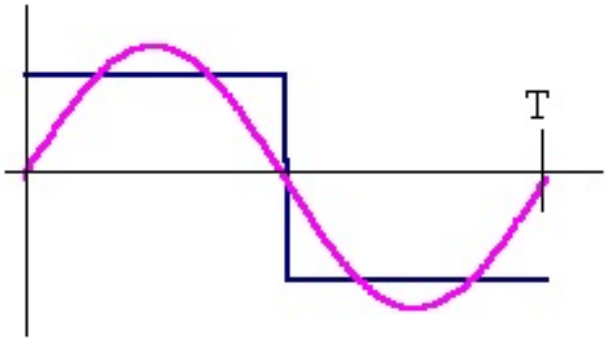
$$y[n] = x[n] * h[n]$$

- So if you know the impulse response of an LTI system it is easy to calculate the output for any input.

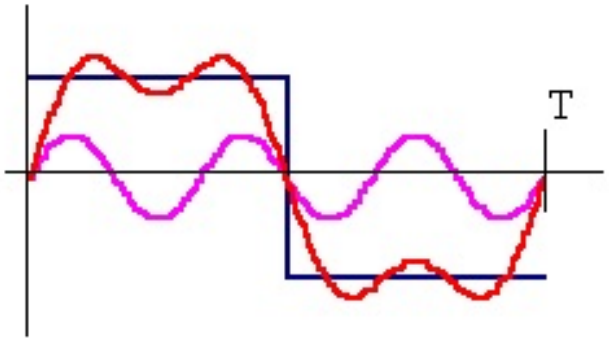
Fourier Analysis

- Moving towards more meaningful features.
  - Time domain: $x[n] \sim$ air pressure at time n.
  - Frequency domain: $X(\omega) \sim$ energy at frequency ω.
  - As we discussed earlier, energy as a function of frequency is what seems to be useful for speech recognition
  - This is very easy to compute when dealing with LTI systems
- Can express (almost) any signal $x[n]$ as sum of sinusoids.
  - Let $X(\omega)$ be the coefficient for sinusoid w/ frequency ω.
- Given $x[n]$, can compute $X(\omega)$ efficiently, and *vice versa*.
  - Time and frequency domain representations are equivalent.
- *Fourier transform* converts between representations.

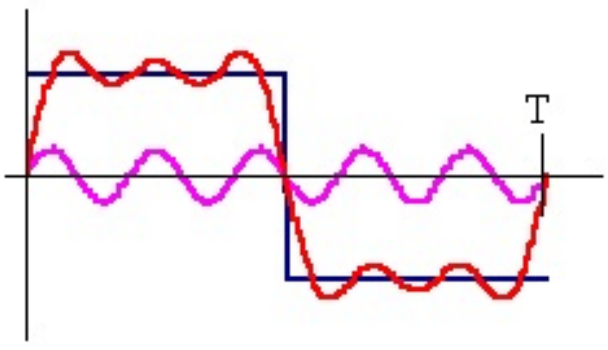
Fourier Series Illustration



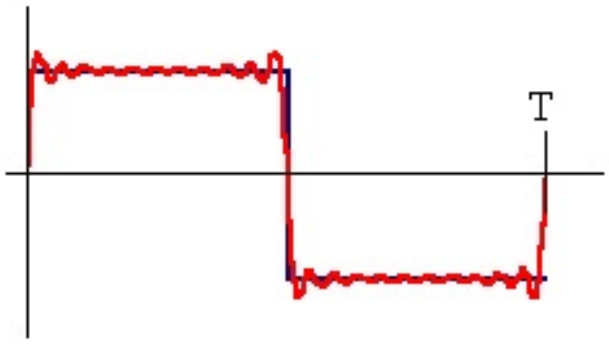
The first term in the Fourier series for the square wave shown in blue



The second term in the Fourier series for the square wave shown in blue. The sum of the first two terms already begins to look somewhat like the square wave



The third term in the Fourier series for the square wave shown in blue. The sum of the first three terms looks even more like the square wave



The sum of the first ten terms looks even very much like the square wave, but some bumpiness remains.

Review: Complex Exponentials

- Math is simpler using complex exponentials.
- Euler's formula.

$$e^{j\omega} = \cos\omega + j\sin\omega$$

- Sinusoid with frequency ω, phase ϕ.

$$\cos(\omega n + \phi) = \mathrm{Re}(e^{j(\omega n+\phi)})$$

The Fourier Transform

- The discrete-time Fourier transform (DTFT) is defined as

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}$$

  Note: this is a *complex* quantity.
- The inverse Fourier transform is defined as

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{j\omega n} d\omega$$

- Exists and is invertible as long as $\sum_{-\infty}^{\infty} |x[n]| < \infty$.
- Can apply DTFT to impulse response as well: $h[n] \Rightarrow H(\omega)$.

The Z-Transform

- One can generalize the discrete-time Fourier Transform to

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n}$$

  where z is any complex variable. The Fourier Transform is just the z-transform evaluated at $z = e^{-j\omega}$.
- The z-transform concept allows us to analyze a large range of signals, even those whose integrals are unbounded. We will primarily just use it as a notational convenience, though.

The Convolution Theorem

- Apply system H to signal x to get signal y: $y[n] = x[n] * h[n]$.

$$
\begin{aligned}
Y(z) &= \sum_{n=-\infty}^{\infty} y[n] z^{-n} = \sum_{n=-\infty}^{\infty} \left(\sum_{k=-\infty}^{\infty} x[k] h[n-k] \right) z^{-n} \\
&= \sum_{k=-\infty}^{\infty} x[k] \left(\sum_{n=-\infty}^{\infty} h[n-k] z^{-n} \right) \\
&= \sum_{k=-\infty}^{\infty} x[k] \left(\sum_{n=-\infty}^{\infty} h[n] z^{-(n+k)} \right) \\
&= \sum_{k=-\infty}^{\infty} x[k] z^{-k} H(z) = X(z) \cdot H(z)
\end{aligned}
$$

The Convolution Theorem (cont'd)

- Duality between time and frequency domains.

$$\begin{aligned} \mathrm{DTFT}(x[n] * y[n]) &= \mathrm{DTFT}(x) \cdot \mathrm{DTFT}(y) \\ \mathrm{DTFT}(x[n] \cdot y[n]) &= \mathrm{DTFT}(x) * \mathrm{DTFT}(y) \end{aligned}$$

- *i.e.*, convolution in time domain is same as multiplication in frequency domain, and *vice versa*.

The Discrete Fourier Transform (DFT)

- Preceding analysis assumes *infinite* signals: $n = -\infty, \ldots, +\infty$.
- In reality, can assume signals $x[n]$ are finite and of length N ($n = 0, \ldots, N-1$). Then, we can define the DFT as

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi kn}{N}}$$

  where we have replaced ω in the DTFT with $\frac{2\pi k}{N}$.
- The DFT is just a discrete-frequency version of the DTFT and is needed for any sort of digital processing.
- The DFT is equivalent to a Fourier series expansion of a periodic version of $x[n]$.

The Discrete Fourier Transform (cont'd)

- The inverse of the DFT is

$$\begin{aligned}\frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{j\frac{2\pi kn}{N}} &= \frac{1}{N}\sum_{k=0}^{N-1}\left[\sum_{m=0}^{N-1} x[m]e^{-j\frac{2\pi km}{N}}\right] e^{j\frac{2\pi kn}{N}} \\ &= \frac{1}{N}\sum_{m=0}^{N-1} x[m]\sum_{n=0}^{N-1} e^{j\frac{2\pi k(n-m)}{N}}\end{aligned}$$

- The last sum on the right is N for $m = n$ and 0 otherwise, so the entire right side is just $x[n]$.

The Fast Fourier Transform

- Note that the computation of

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi kn}{N}} \equiv \sum_{n=0}^{N-1} x[n] W_N^{nk}$$

  for $k = 0, \ldots, N-1$ requires $O(N^2)$ operations.
- Let $f[n] = x[2n]$ and $g[n] = x[2n+1]$. Then, we have

$$\begin{aligned} X[k] &= \sum_{n=0}^{N/2-1} f[n] W_{N/2}^{nk} + W_N^k \sum_{n=0}^{N/2-1} g[n] W_{N/2}^{nk} \\ &= F[k] + W_N^k G[k] \end{aligned}$$

  when $F[k]$ and $G[k]$ are the $N/2$ point DFT's of $f[n]$ and $g[n]$. To produce values for $X[k]$ for $N > k \geq N/2$, note that $F[k + N/2] = F[k]$ and $G[k + N/2] = G[k]$.
- The above process can be iterated to compute the DFT using only $O(N \log N)$ operations.

The Discrete Cosine Transform

- Instead of decomposing a signal into a sum of complex sinusoids, it is useful to decompose a signal into a sum of *real* sinusoids.
- The Discrete Cosine Transform (DCT) (a.k.a. DCT-II) is defined as

$$C[k] = \sum_{n=0}^{N-1} x[n] \cos\left(\frac{\pi}{N}(n + \frac{1}{2})k\right) \quad k = 0, \ldots, N-1$$

The Discrete Cosine Transform (cont'd)

We can relate the DCT and DFT as follows. If we create a signal

$$y[n] = x[n] \qquad n = 0, \ldots, N-1$$
$$y[n] = x[2N-1-n] \qquad n = N, \ldots, 2N-1$$

then we can compute $C(k)$ in terms of $Y[k]$, the DFT of $y[n]$, as

$$C(k) = Y(k)2e^{-j\frac{\pi k}{2N}} \qquad k = 0, \ldots, N-1$$

Long-Term *vs.* Short-Term Information

- Have infinite (or long) signal $x[n]$, $n = -\infty, \ldots, +\infty$.
  - Take DTFT or DFT of whole damn thing.
  - Is this interesting?
- Point: we want short-term information!
  - *e.g.*, how much energy at frequency ω over span $n = n_0, \ldots, n_0 + k$?
- Going from long-term to short-term analysis.
  - Windowing.
  - Filter banks.

Windowing: The Basic Idea

- Excise N points from signal $x[n]$, $n = n_0, \ldots, n_0 + (N-1)$ (*e.g.*, 0.02s or so).
- Perform DFT on truncated signal; extract some features.
- Shift n_0 (*e.g.*, by 0.01s or so) and repeat.

What's the Problem?

- Excising N points from signal $x \Leftrightarrow$ multiplying by rectangular window h.
- Convolution theorem: multiplication in time domain is same as convolution in frequency domain.
  - Fourier transform of result is $X(\omega) * H(\omega)$.
- Imagine original signal is periodic.
  - Ideal: after windowing, $X(\omega)$ remains unchanged $\Leftrightarrow$ $H(\omega)$ is delta function.
  - Reality: short-term window cannot be perfect.
  - How close can we get to ideal?

Rectangular Window

$$h[n] = \begin{cases} 1 & n = 0, \ldots, N-1 \\ 0 & \text{otherwise} \end{cases}$$

- The FFT can be written in closed form as

$$H(\omega) = \frac{\sin \omega N/2}{\sin \omega/2} e^{-j\omega(N-1)/2}$$

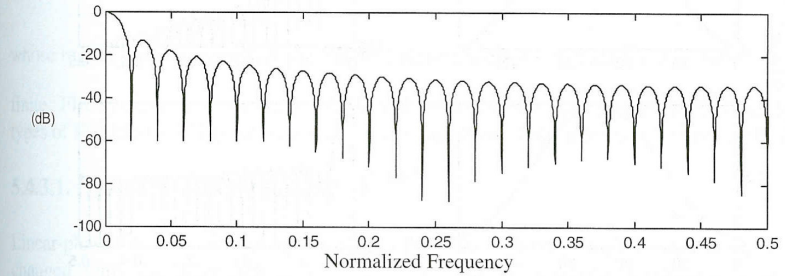


Figure 5.19 Frequency response (magnitude in dB) of the rectangular window with $N = 50$, which is a digital sinc function.

- Note the high sidelobes of the window. These tend to distort low energy components in the spectrum when there are significant high-energy components also present.

Hanning and Hamming Windows

- Hanning: $h[n] = .5 - .5\cos 2\pi n/N$
- Hamming: $h[n] = .54 - .46\cos 2\pi n/N$

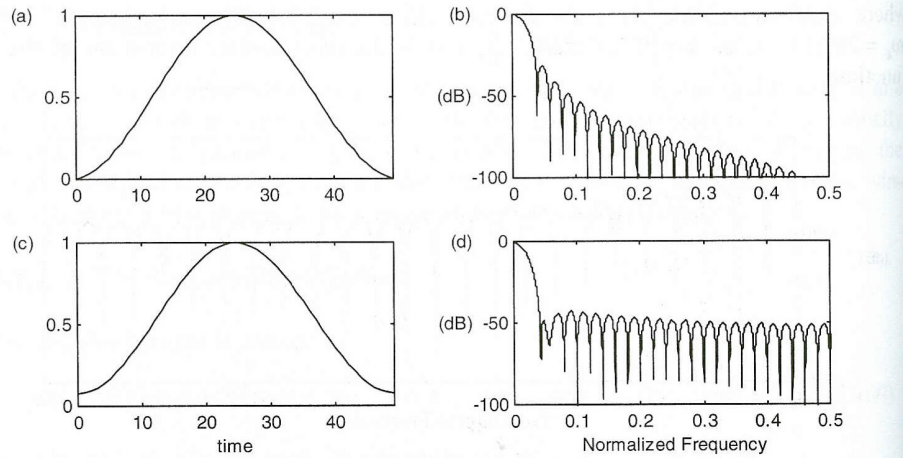
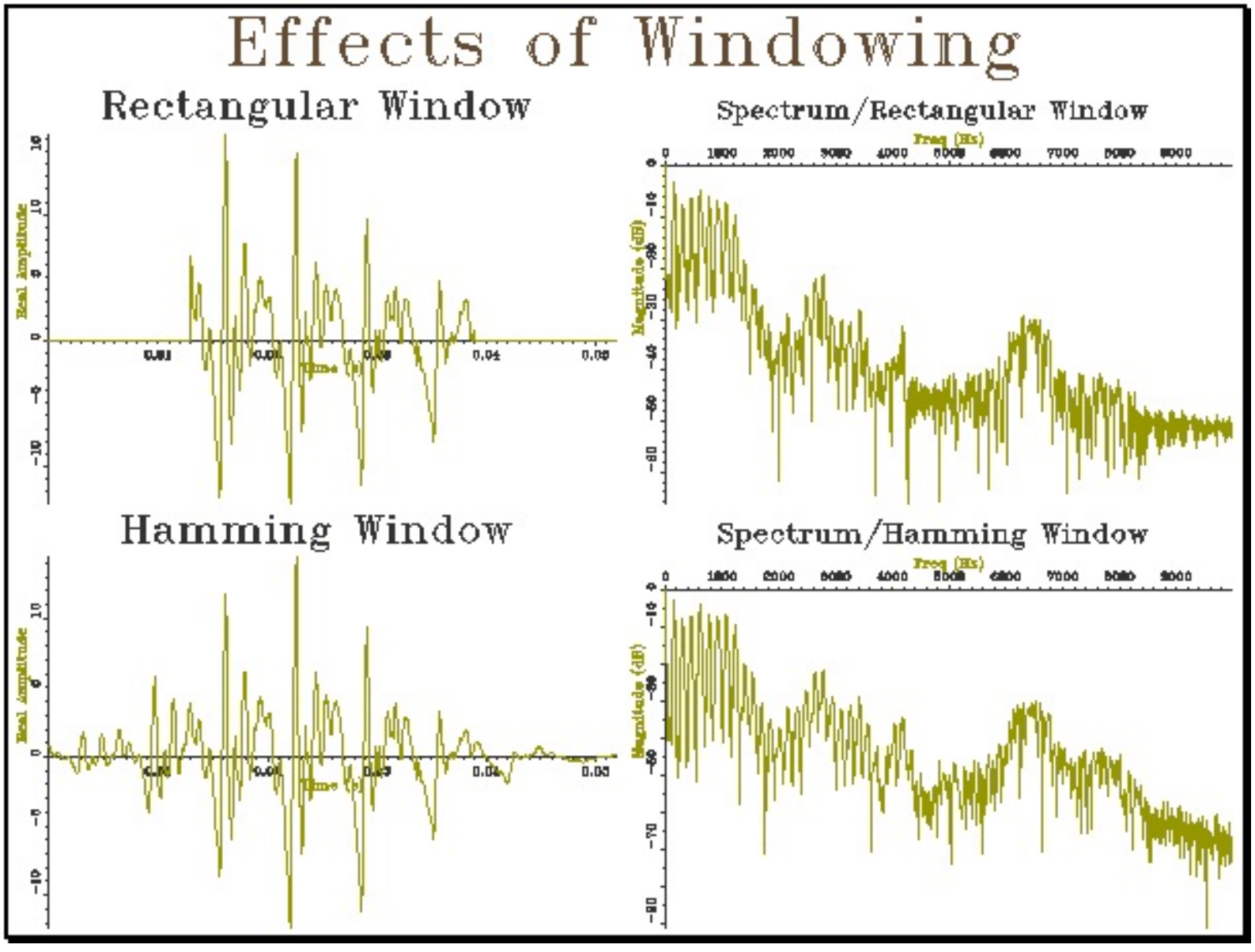


Figure 5.20 (a) Hanning window and (b) the magnitude of its frequency response in dB; (c) Hamming window and (d) the magnitude of its frequency response in dB for $N = 50$.

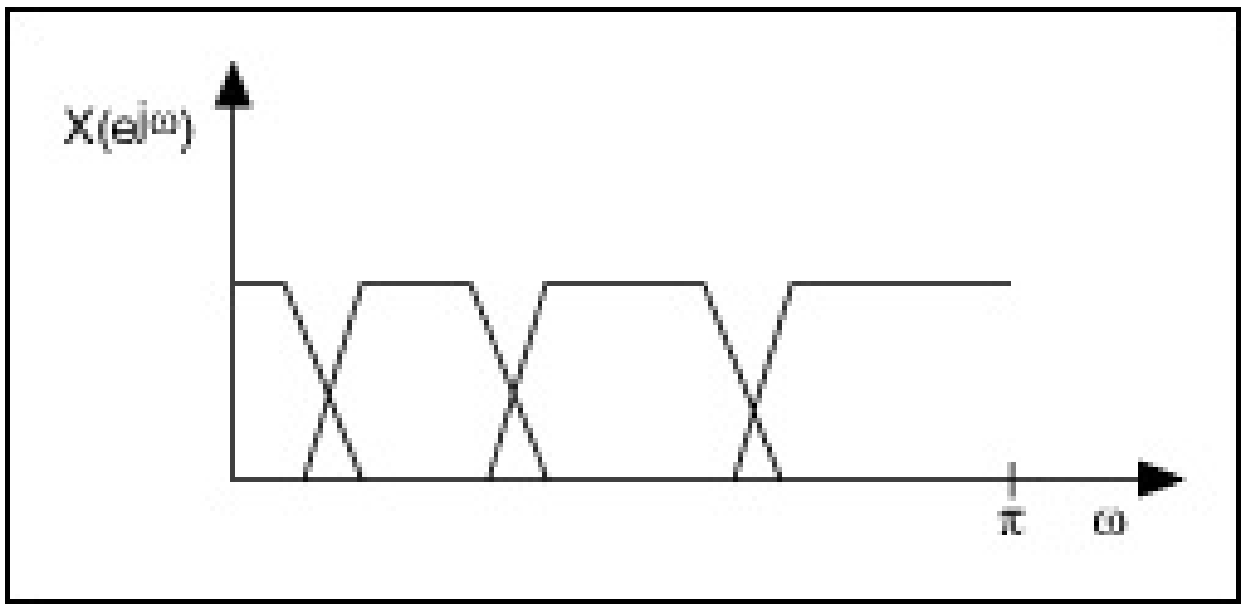
- Hanning and Hamming have slightly wider main lobes, much lower sidelobes than rectangular window.
- Hamming window has lower first sidelobe than Hanning; sidelobes at higher frequencies do not roll off as much.

Windowing Comparison



Using Signal Processing to do Frequency Analysis like the Auditory System

- Each cochlear hair does a frequency analysis at a different frequency.
  - Input signal: air pressure; output: hair displacement.
  - Each hair responds to different frequency.
  - Cochlea is a *filter bank*.



Bandpass Filters

- A filter H acts on each input frequency ω independently.
  - Scales component with frequency ω by $H(\omega)$.
- *Low-pass* filter.
  - "Lets through" all frequencies below cutoff frequency.
  - Suppresses all frequencies above.
- *High-pass* filter; *band-pass* filter.
- A *Filter Bank* is a collection of band-pass filters spanning a frequency range.
- Can implement filter bank via convolution.
  - For each output point n, computation for ith filter is on order of L_i (length of impulse response).

$$x_i[n] = x[n] * h_i[n] = \sum_{m=0}^{L_i-1} h_i[m]x[n-m]$$

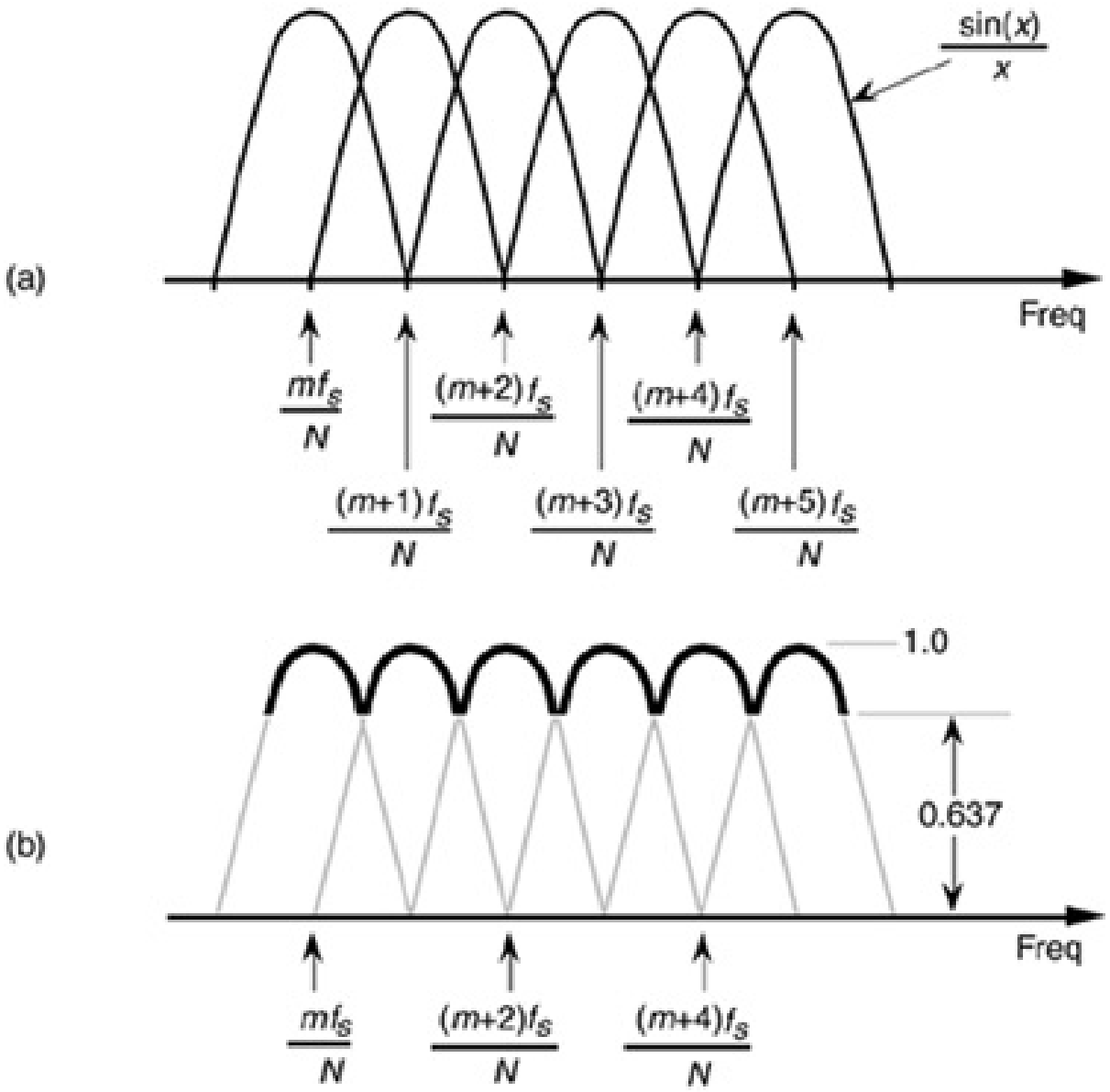
Implementation of Filter Banks

- Given low-pass filter $h[n]$, can create band-pass filter $h_i[n] = h[n]e^{j\omega_i n}$ via *heterodyning*.
  - Multiplication in time domain $\Rightarrow$ convolution in frequency domain $\Rightarrow$ shift $H(\omega)$ by ω_i.

$$\begin{aligned} x_i[n] &= \sum h[m]e^{j\omega_i m}x[n-m] \\ &= e^{j\omega_i n}\sum x[m]h[n-m]e^{-j\omega_i m} \end{aligned}$$

- The last term on the right is just $X_n(\omega)$, the Fourier transform of a windowed signal, where now the window is the same as the filter. So, we can interpret the FFT as just the instantaneous filter outputs of a uniform filter bank whose bandwidths corresponding to each filter are the same as the main lobe width of the window.

Implementation of Filter Banks (cont'd)



Implementation of Filter Banks (cont'd)

- Notice that by combining various filter bank channels we can create non-uniform filterbanks in frequency.

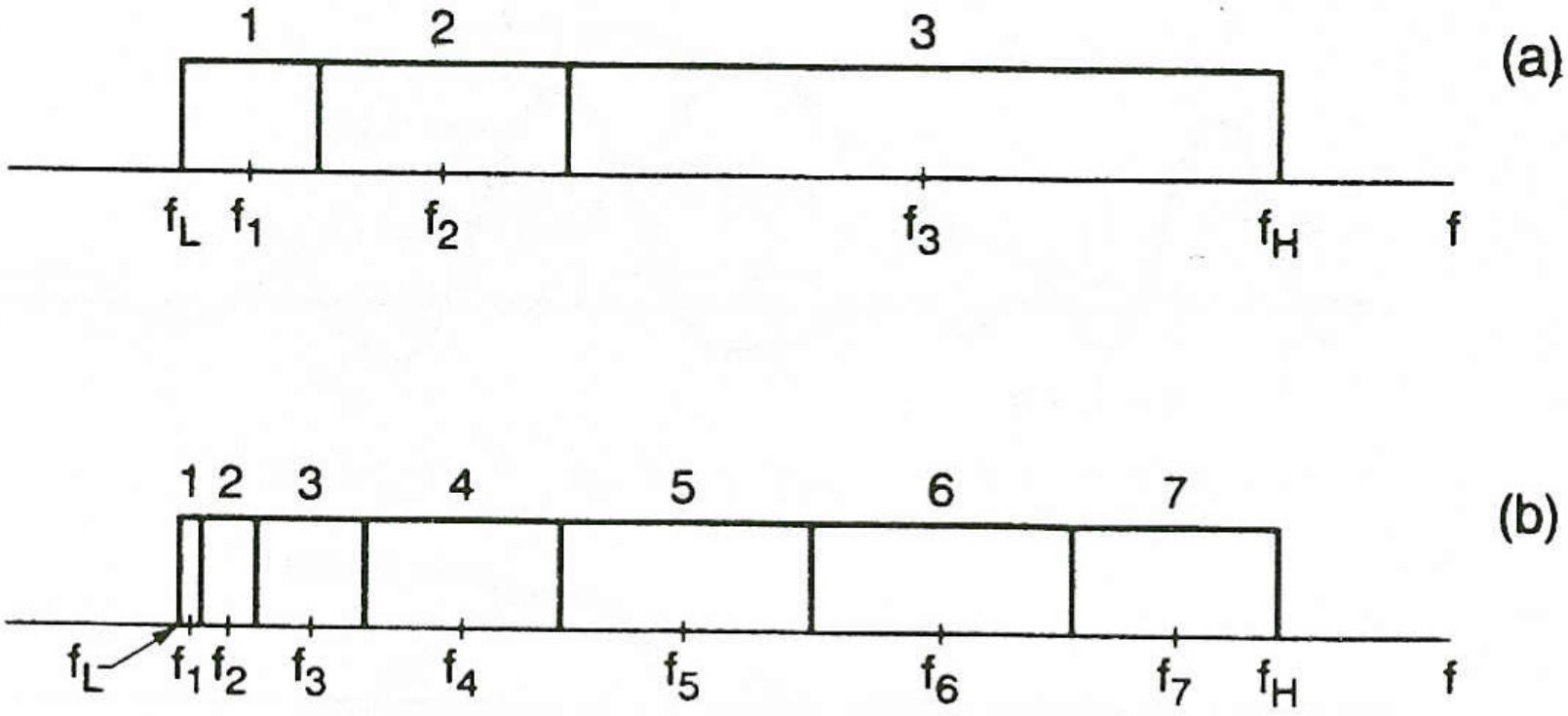


Figure 3.18 Two arbitrary nonuniform filter-bank ideal filter specifications consisting of either 3 bands (part a) or 7 bands (part b).

Are We Awake?

- Why is the current energy at a frequency a more robust feature than instantaneous air pressure?
- Why does everyone in ASR use Fourier analysis?

Recap

- Overview of Course
- Overview of Speech Recognition
- Speech Production and Perception
- Enough Signal Processing so you know how to make a Filter Bank.
- Next week: Signal Processing in Speech Recognition Systems and the Beginnings of how we model the time varying nature of speech.

Course Feedback

1. Was this lecture mostly clear or unclear? What was the muddiest topic?
2. Other feedback (pace, content, atmosphere)?