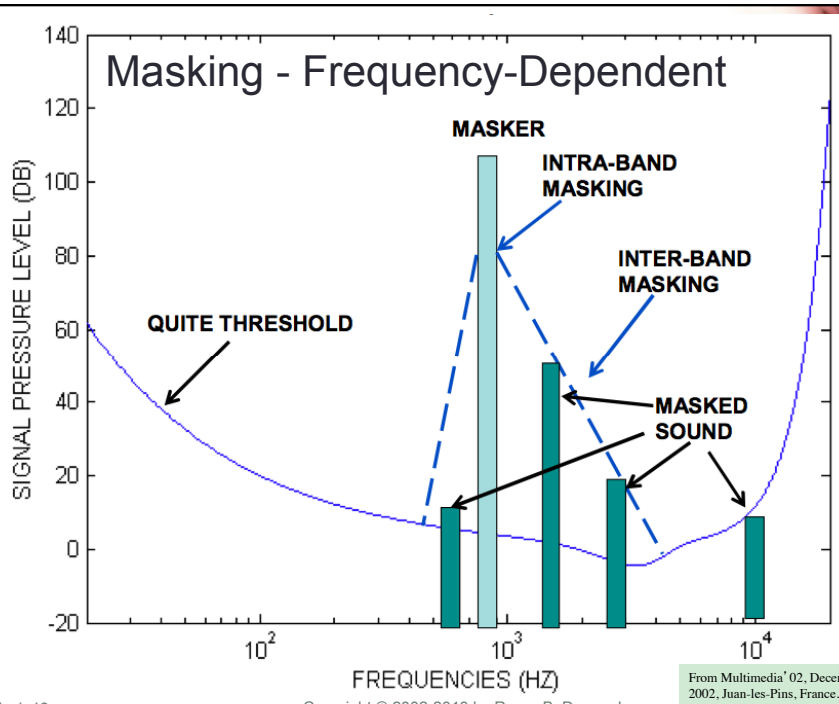
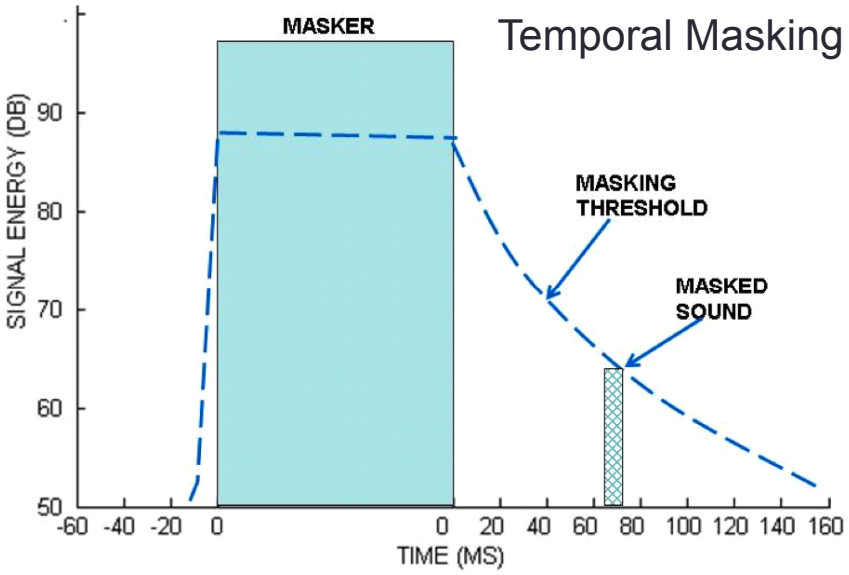
PSYCHO-PERCEPTUAL CODING AND MP3

Masking - Frequency-Dependent



From Multimedia '02, December 1-6, 2002, Juan-les-Pins, France. Copyright 2002 ACM

Temporal Masking



From Multimedia' 02, December 1-6, 2002, Juan-les-Pins, France. Copyright 2002 ACM

Masking, Perceptual Coding

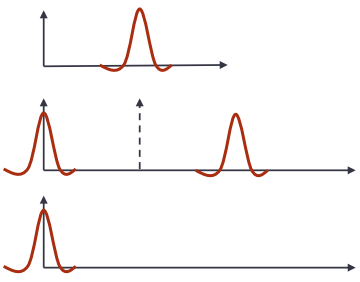
- Ear as filter bank
- Loud sound in a channel masks soft sounds in adjacent channels
- Quantize as much as possible – using masking to hide quantization effects
- Frequency shift bands and sample at minimum sample rate

Some Insight on Frequency Domain and Coding

- PCM is time domain – not so much sample-to-sample correlation. Changing all the time.
- Spectral data tends to be more static. The spectrum at time *t* is a good predictor for spectrum at time *t*+1.
- In tonal music, spectrum is relatively sparse: non-zero only where there are sinusoids. This is easier to encode efficiently.

A Note on Banded Processing

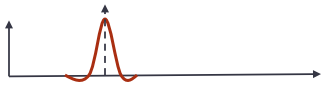
- You might think when a signal is separated into N bands, you'd get N x Original Rate
- Spectral view of a band:
- Recall that multiplication by a sinusoid will create shifted copies of original spectrum:
- Now, low-pass filter to get just the lower band:
- In theory, you can encode the signal as N bands at 1/N the sample rate, so the total information is the same.



Banded Processing, Continued

- To recover original band, start with encoded signal

- Multiply by sine to frequency shift



MP3 – MPEG Audio Layer 3

- Part of broader video compression standard
- A standard for consistent decoding
- Encoding algorithm details not specified
- 6-to-1 encoding of 48kHz audio gives no perceptual difference in extensive tests
- Fraunhofer claims 12-to-1

Step 1: Filter Bank

- Signal is filtered into 32 bands of equal width.
- Each band is sub-sampled by factor of 32.
- Some simplifications:
  - Bands have overlap
  - Bands are wider than sample-rate/32, so subsampling causes aliasing
  - Filter and inverse are slightly lossy

Step 2: Psychoacoustic Model

- How much can each band be quantized?
- Take 1024 point FFT of audio, group spectrum into *critical bands*
- Identify *tonal components* (sinusoids) and assign tonal index to each critical band
  (Note: noise masking and tonal masking differ)
- From spectrum, compute masking threshold for each of 32 subbands.
- Compute signal-to-mask ratio for each of 32 subbands.

Coding

- Each subband is transformed with modified discrete cosine transform (MDCT) of length 18 or 6 subband samples.
- Essentially, this is a short-term frequency representation of the subbands.
- This allows for more efficient coding, e.g. if only one sinusoid is in the subband, one MDCT coefficient should be high and others low.
- Quantize MDCT coefficients according to signal-to-masking ratio.

Coding (2)

- There are 576 coefficients per frame (18 MDCT coefficients x 32 subbands)
- Order them by increasing frequency.
- Highest frequencies tend to be zeros (no bits expended here)
- Next is a run of -1, 0, and 1, encoded 4 at a time into alphabet of 81 symbols.
- Remaining values coded in pairs.

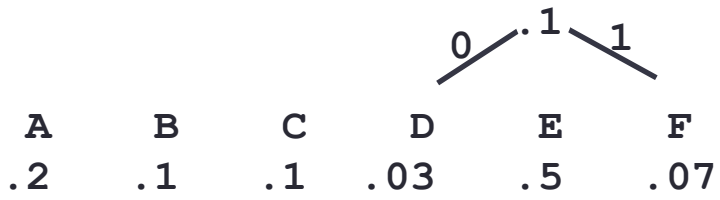
Huffman Coding

- Most popular technique for removing coding redundancy
- Gives smallest number of code symbols per source symbol
- Symbols are coded one at a time.
- Codes are variable length;
  chosen to minimize expected bit length

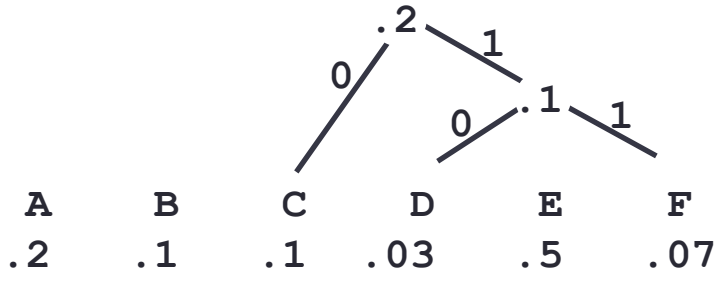
Huffman Code Example

A	B	C	D	E	F
.2	.1	.1	.03	.5	.07

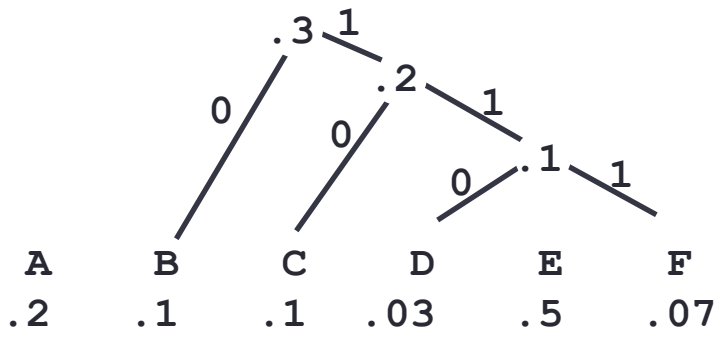
Huffman Code Example



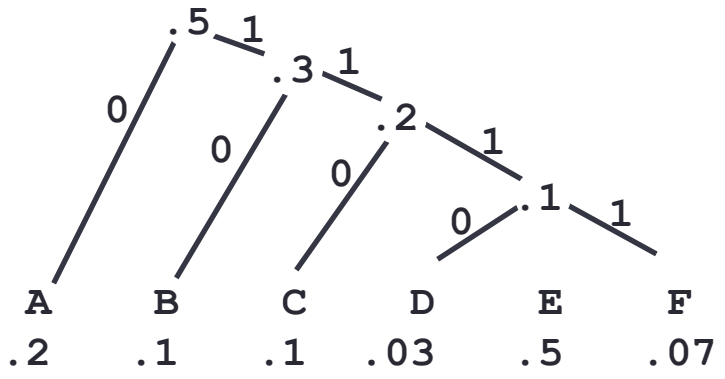
Huffman Code Example



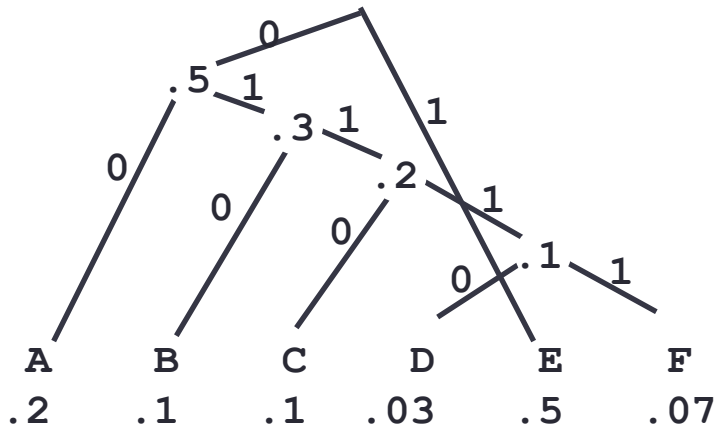
Huffman Code Example



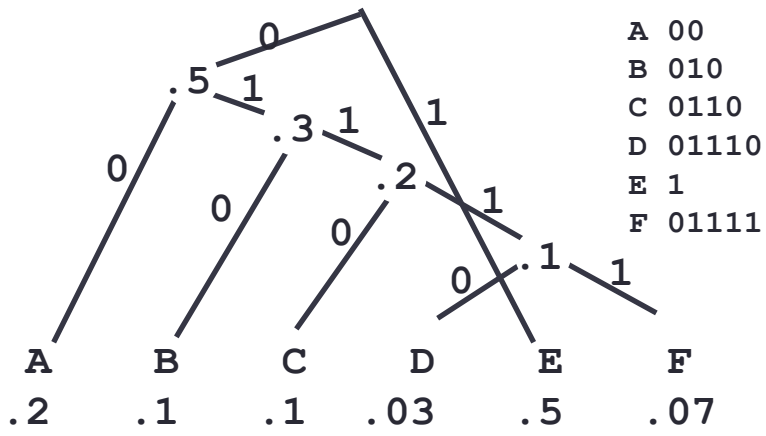
Huffman Code Example



Huffman Code Example



Huffman Code Example



Huffman Code Example

```
Average Length =

   2(.2)+3(.1)+4(.1)+
   5(.03)+1(.5)+5(.07)

  = 2.1 bits/symbol
```

A	B	C	D	E	F
.2	.1	.1	.03	.5	.07

Tricky Concept

- Symbols (A, B, C, …) can represent numbers or vectors, e.g.
  A = [-1, -1, -1, -1]
  B = [-1, -1, -1, 0]
  C = [-1, -1, -1, 1]
  D = [-1, -1, 0, -1]
  E = [-1, -1, 0, 0]
  F = [-1, -1, 0, 1]
  etc.

Bit Reservoir

- If a frame is encoded with fewer bits than the compressed data rate, e.g. 64kbps, allows, bits are “donated” to “bit reservoir.”
- Later, encoder can use bits from “bit reservoir”, temporarily exceeding maximum data rate, to do the best job of encoding audio.

Bits Allocation

- Bits are allocated to bands where quantization noise exceeds masking threshold.
- Iterative allocation of bits followed by recalculation of noise.
- This makes encoding slow.

Summary

- Masking reduces our ability to hear “everything”
- Including quantization noise
- MP3 descriptions are highly quantized (but these are frequency domain descriptions)
- After quantization, use run-length coding and Huffman coding to encode descriptions efficiently