Audio-coding standards

The goal is to provide CD-quality audio over telecommunications networks.

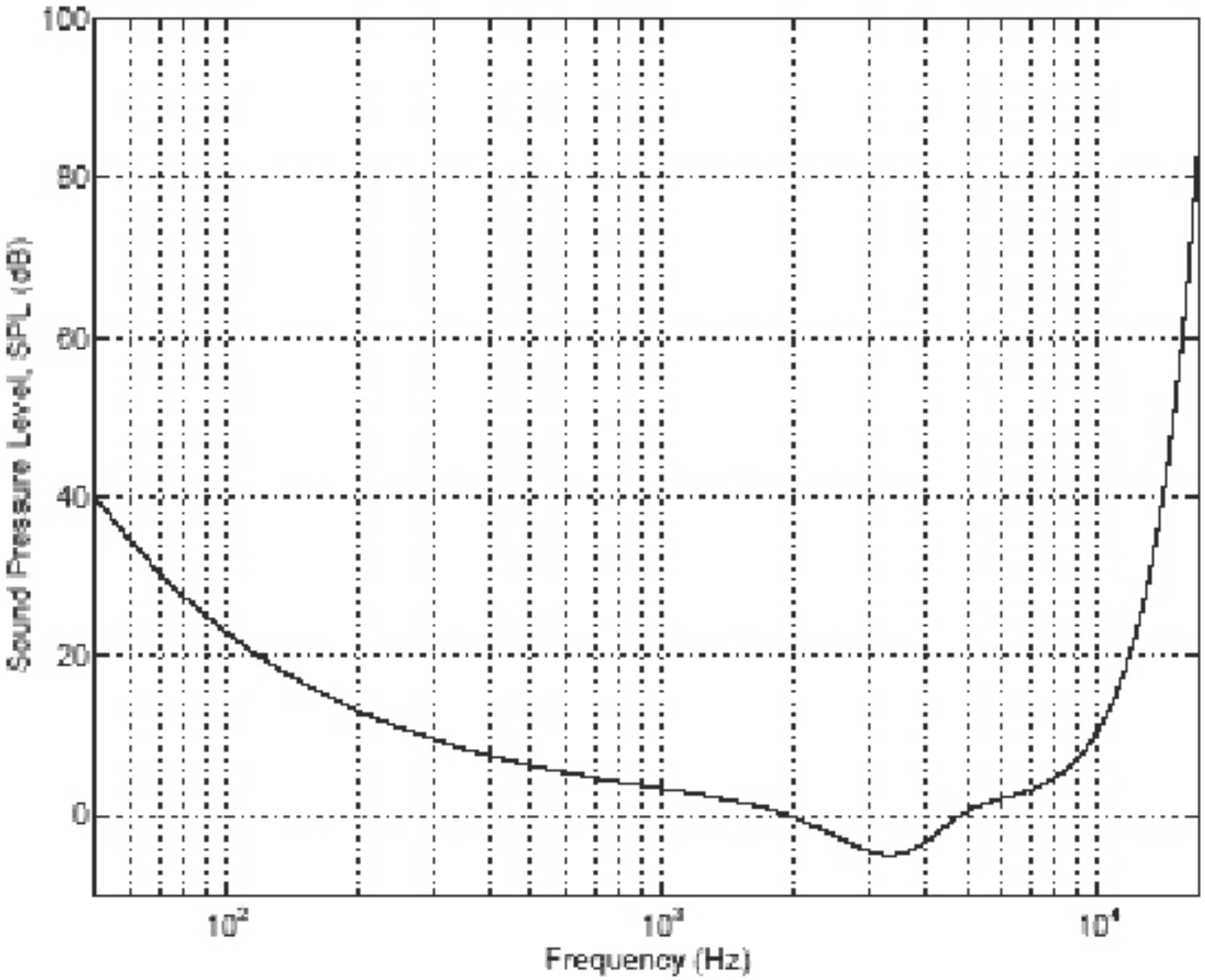
Almost all CD audio coders are based on the so-called psychoacoustic model of the human auditory system. This model allows to remove the parts of the signal that the human cannot perceive. Moreover, the amount of quantization noise that is inaudible can be calculated.

Audio coders work in the range from 20Hz to 20 kHz.

The human auditory system has remarkable detection capability with a range (called dynamic range) of over 120 dB from very quiet to very loud sounds. The absolute threshold T_q of hearing

characterize the amount of energy in a pure tone such that it can be detected by listener in noiseless environment.

Audio coding standards



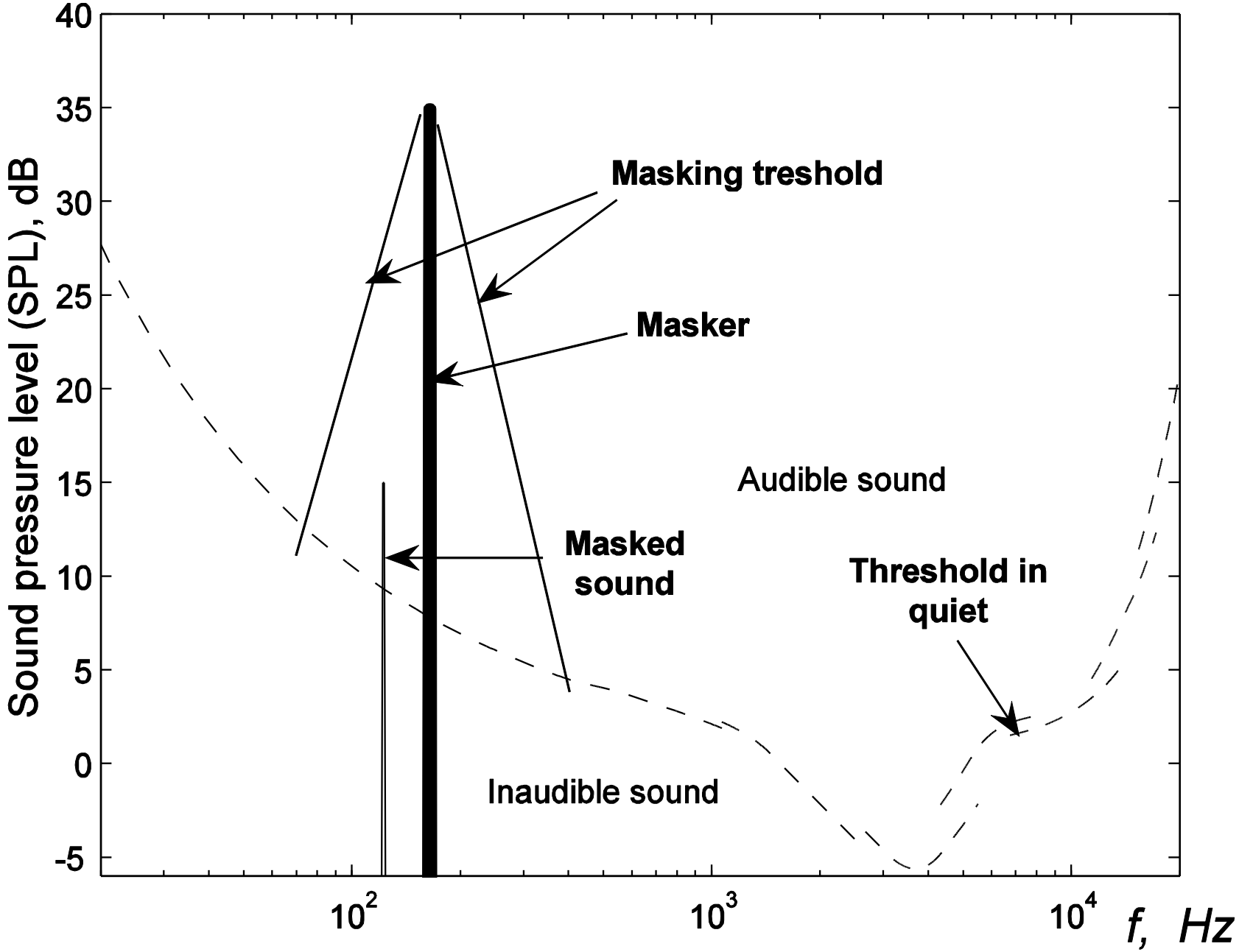
Audio coding standards

However, louder sounds mask or hide weaker ones.

The psychoacoustic model takes into account this masking property that occurs whenever the presence of strong audio signal makes a temporal or spectral neighborhood of weaker audio signals imperceptible.

The noise detection threshold is a modified version of the absolute threshold with its shape determined by the input signal at any given time.

In order to estimate this threshold we have first to understand how the ear performs spectral analysis.


40
35
30
25
20
15
10
5
0
-5
Sound pressure level (SPL), dB
Masking treshold
Masker
Audible sound
Masked
sound
Threshold in
quiet
Inaudible sound
10^2
10^3
10^4
f, Hz

Psychoacoustic model

It is based on the following observations:

- The cochlea can be viewed as a bank of highly overlapping bandpass filters. Moreover, the cochlear filters passbands are of non-uniform bandwidth, and the bandwidths increase with increasing frequency.

Dependence of resolution on frequency for human auditory system can be expressed in terms of critical-bandwidths. A critical band is a range of frequencies over which the masking SNR remains more or less constant.

Critical bandwidths are less than 100 Hz for the lowest audible frequencies and more than 4 kHz at the highest.

- Noise and tone have different masking properties.

Psychoacoustic model

If B is the critical-band number then:

Tone masking noise: $E_N = E_T - (14.5 + B)$ dB (14.1)

Noise masking tone : $E_T = E_N - K$ dB, (14.2)

where E_N, E_T is noise and tone energy, respectively, K is in the range of 3-6 dB.

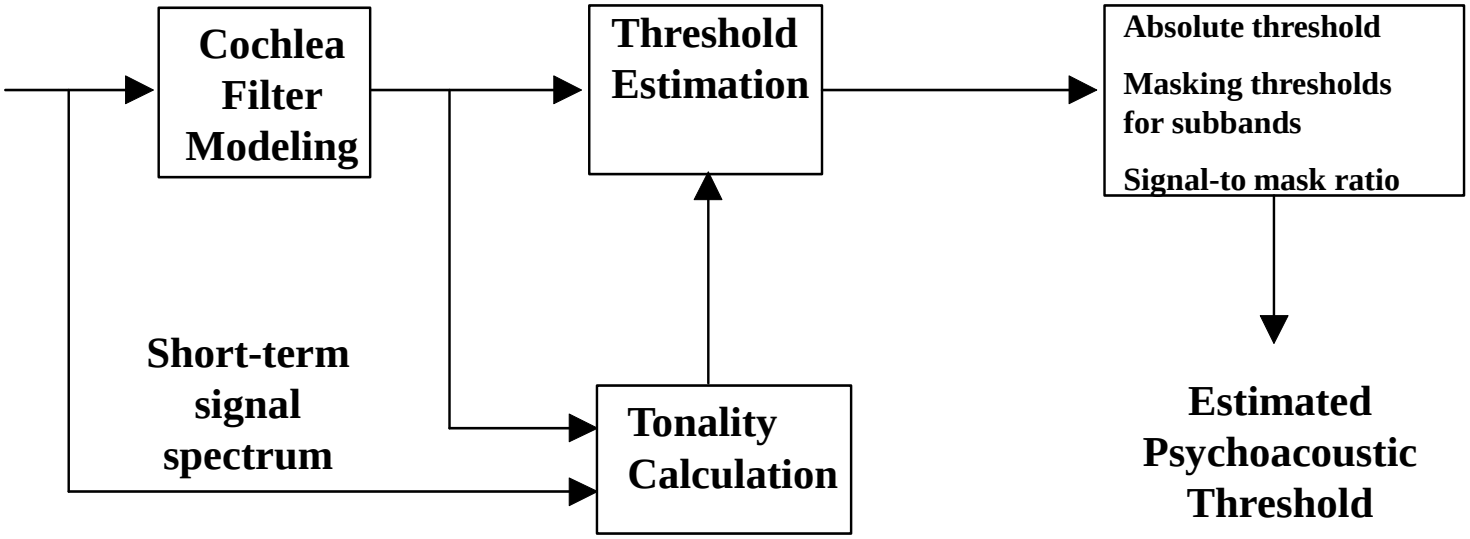
•Speech and audio signals are neither pure tones nor pure noise but rather mixture of both.

•(14.1),(14.2) capture only the contributions of individual tone-like or noise-like maskers. Typically each frame contains a collection of both masker types. These individual masking thresholds are combined to form a global masking threshold.

Psychoacoustic model

Inter-band masking means that masker centered within one critical band has some predictable effect on detection thresholds in other critical subbands. This effect is known as spread of masking and is modeled as a spreading function.

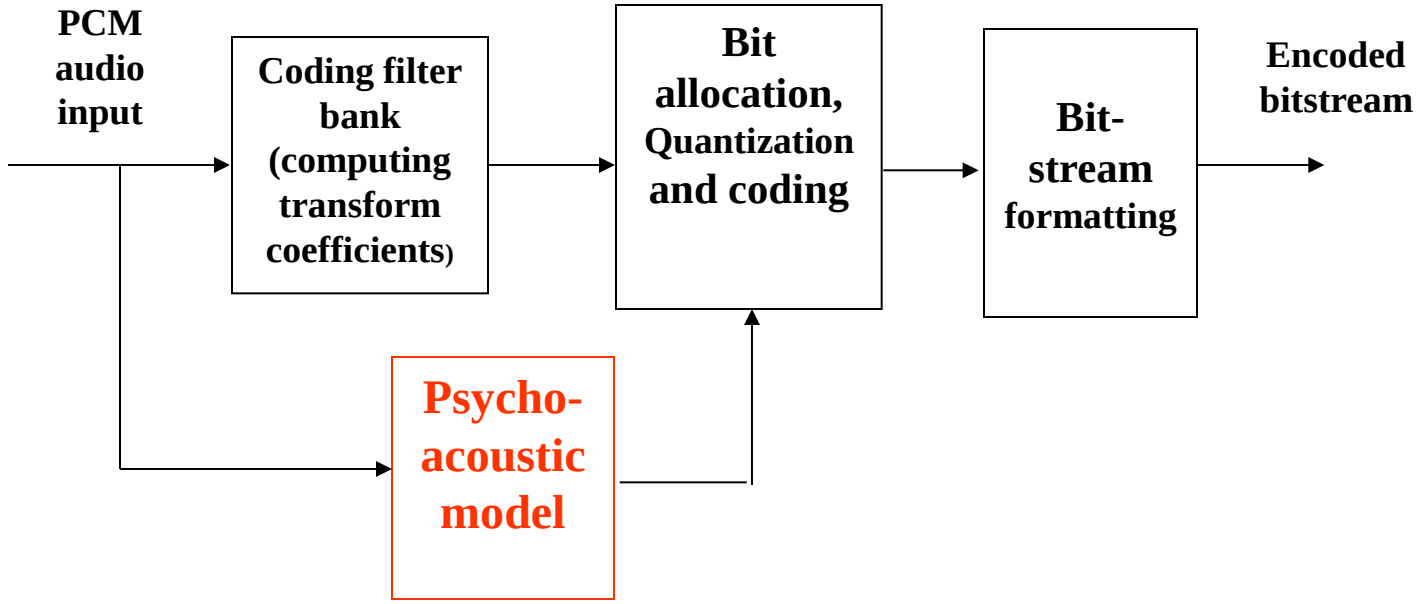
Psychoacoustic model

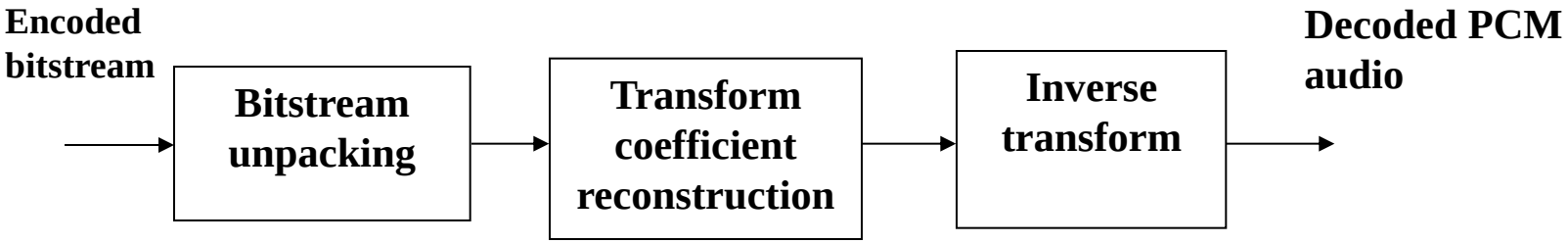


Psychoacoustic model

- Using short-term spectrum we classify frequency components of each subband as tone or noise
- For tone-like and noise-like components individual masking thresholds are computed
- The individual masking thresholds and T_q are combined to form a global threshold and minimum over subband components is computed
- Signal-to-mask ratio for subband is computed
- Mask-to-noise ratio for subband is computed

Generic perceptual audio coder and decoder





Generic perceptual audio coder

In audio standards the audio stream passes through a filter bank that divides the input into multiple subbands of frequency. This type of transform coding is called subband coding. This transform is overlapped.

The input audio stream simultaneously passes through a psychoacoustic model that determines the ratio of the signal energy to the masking threshold for each subband.

The quantization and coding block uses the signal-to-mask ratios to decide how to apportion the total number of code bits available for the quantization of the subband signal to minimize the audibility of the quantization noise.

Usually scalar uniform quantization of transform coefficients is used. Quantized coefficients are coded by the Huffman code.

Overview of audio standards

ISO Motion Pictures Experts Group (ISO-MPEG/audio) is one part of three part compression standard that includes video and systems.

•The original MPEG (sometimes it is referred as MPEG-1) was created for mono sound systems and had three layers, each providing greater compression ratio.

•MPEG-2 was created to provide stereo and multichannel audio capability.

•MPEG advanced audio coder (MPEG-AAC) has the same quality as MPEG-2 but at half the bit rate.

•MPEG-4 audio is a complete toolbox to do everything from low bit rate speech coding to high quality audio coding or music synthesis. Contains (high efficiency) HE AAC and HE AAC v2 modes.

MPEG-1

MPEG-1 has 3 layers.

Layer I, the simplest provides bit rates above 128 kb/s per channel (compression ratios 3-4).

Layer II, has intermediate complexity and provides bit rates around 128 kb/s per channel (compression ratios 5-6). Main applications: storage video on CD-ROM and transmitting of audio information over ISDN channels.

Layer III is the most complex and provides bit rates around 64 kb/s per channel (compression ratios 10-12). It is used for low-rate compression systems.

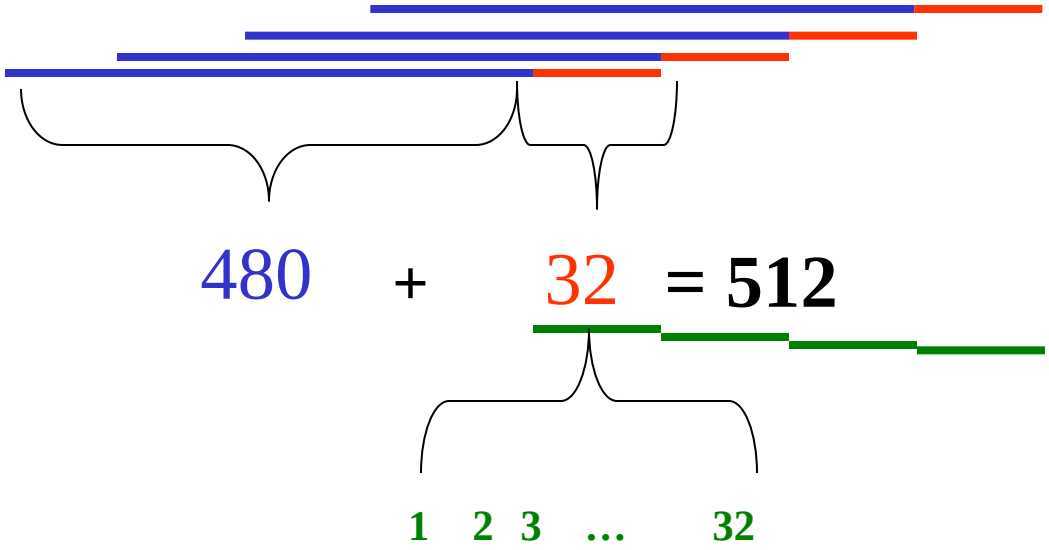
Transform

The polyphase filter bank is common to Layers I and II. Layer III uses hybrid transform which is based on the modified DCT and filter bank coding.

The polyphase filter bank splits the audio signal into 32 equal-width frequency subbands. Each subband is 750 Hz wide.

- The equal widths of the subbands do not accurately reflect the human auditory system behavior.
- The filter bank and its inverse are not lossless transforms.
- Adjacent filter bands have a major frequency overlap.

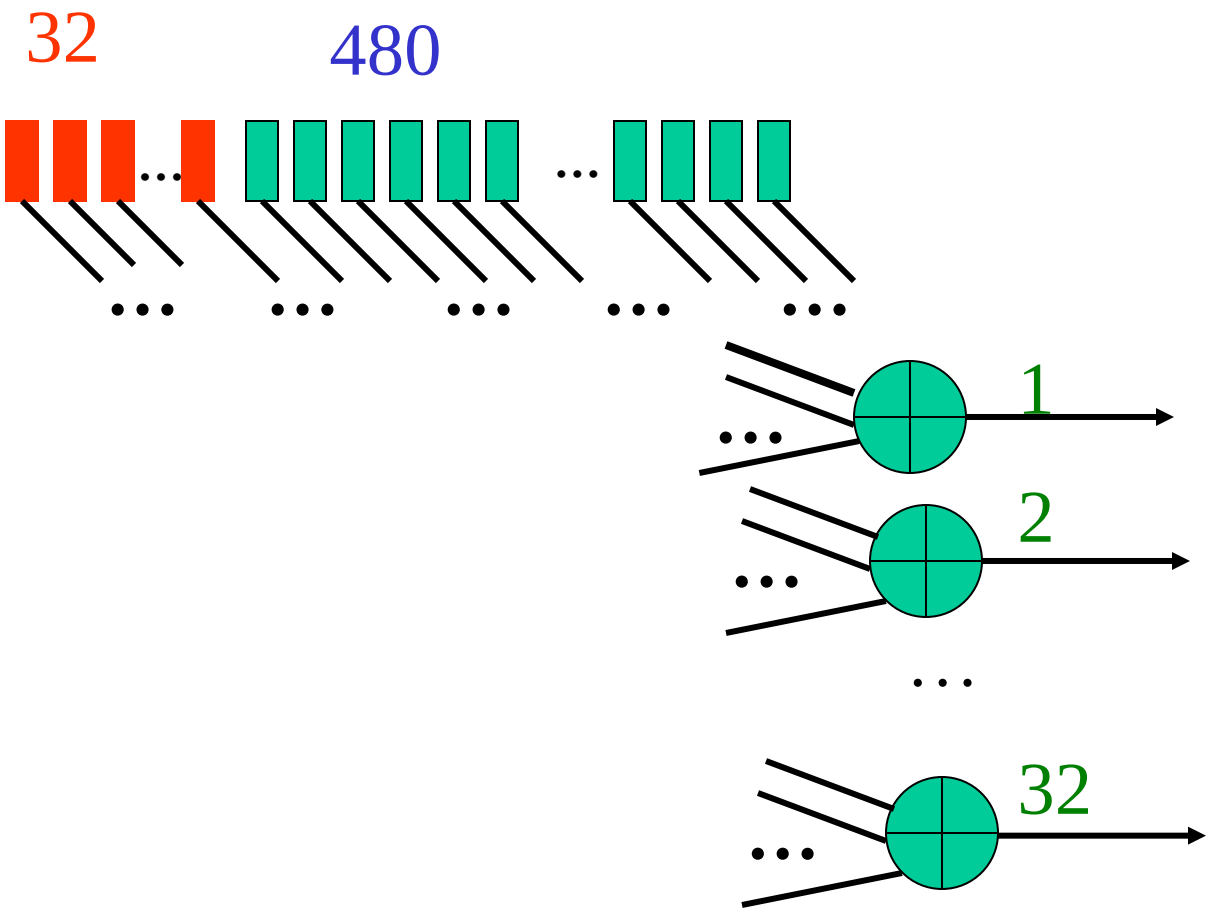
Polyphase filter bank



Input frames

Transform coefficients

Polyphase filter bank



Polyphase filter bank

The transform coefficients at the output of the i th filter represent the filter bank outputs

$$s_i(n) = \sum_{m=0}^{511} x(n-m)H_i(m)$$

decimated by a factor of 32, that is

$$y_i(n) = s_i(32n) = \sum_{m=0}^{511} x(32n-m)H_i(m),$$

where $x(n)$ is the input audio stream and

$$H_i(m) = h(m)\cos\left(\frac{(i+1/2)(m-16)}{32}\pi\right)$$

is the pulse response of the i th filter.

Polyphase filter bank

All filters are obtained from the one prototype low-pass filter with pulse response $h(m)$

$$h(m) = \begin{cases} -C(m), \text{if } \lfloor m/64 \rfloor \text{is odd} \\ C(m), \text{otherwise} \end{cases},$$

$C(m)$ is a transform window.

A direct implementation of the above equations requires

$32 \times 512 = 16384$ multiplications and $32 \times 511 = 16352$ additions to compute 32 filter outputs.

Polyphase filter bank

Taking into account the periodicity of the cosine function we can obtain the equivalent but computationally more efficient representation

$$s_i(n) = \sum_{k=0}^{63} M(i,k) \sum_{j=0}^{7} C(k+64j)\, x(n-(k+64j)), \qquad (14.3)$$

where $C(k)$ is one of 512 coefficients of the transform window,

$$M(i,k) = \cos\left(\frac{(i+1/2)(k-16)\pi}{32} \right),$$

$i = 0,...,31, \;\; k = 0,...,63.$

Period of $M(i,k)$ is 128 and $h(k)$ is equal to - $C(k)$ with period 64.

Polyphase filter bank

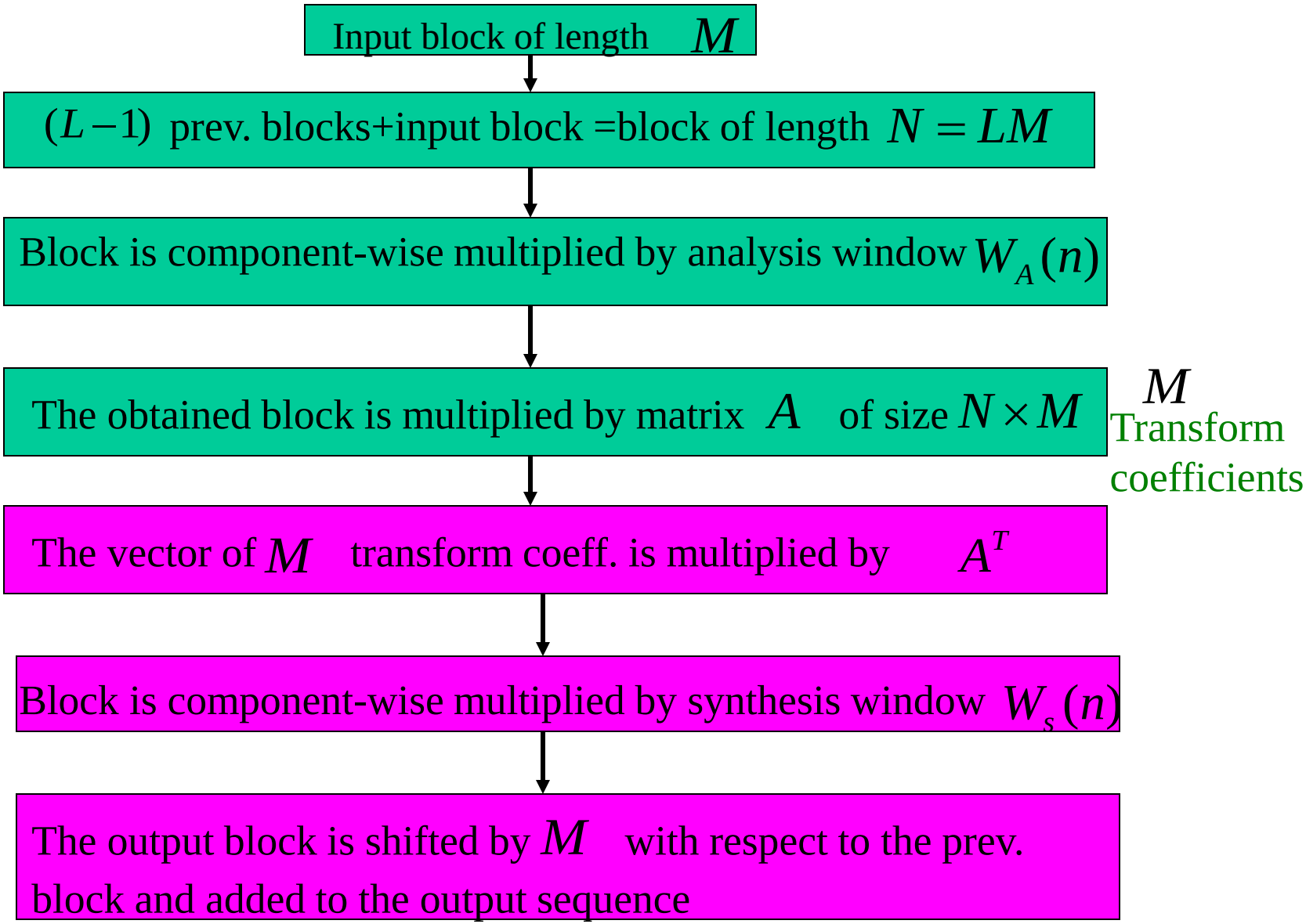
Using this formula we perform only

$512+32 \times 64=2560$ multiplications and $64 \times 7+32 \times 63=2464$ additions.

More efficient implementation of filtering can be obtained by reducing (14.3) to the modified cosine transform, which can be implemented using the fast Fourier transform.

The Layer III of MPEG-1 besides polyphase filter bank uses the modified cosine transform (or two MDCTs).

Modified DCT



MDCT

The transform matrix A of size $N \times M$ can be written as

$$A = Y \cdot T,$$

where T of size $M \times M$ is the matrix of DCT-IV transform with entries

$$t_{kn} = \sqrt{\frac{2}{M}} \cos\left(\left(k + \frac{1}{2}\right)\left(n + \frac{1}{2}\right)\frac{\pi}{M}\right), \; k, n = 0, \ldots, M-1,$$

The matrix Y describes the preprocessing step

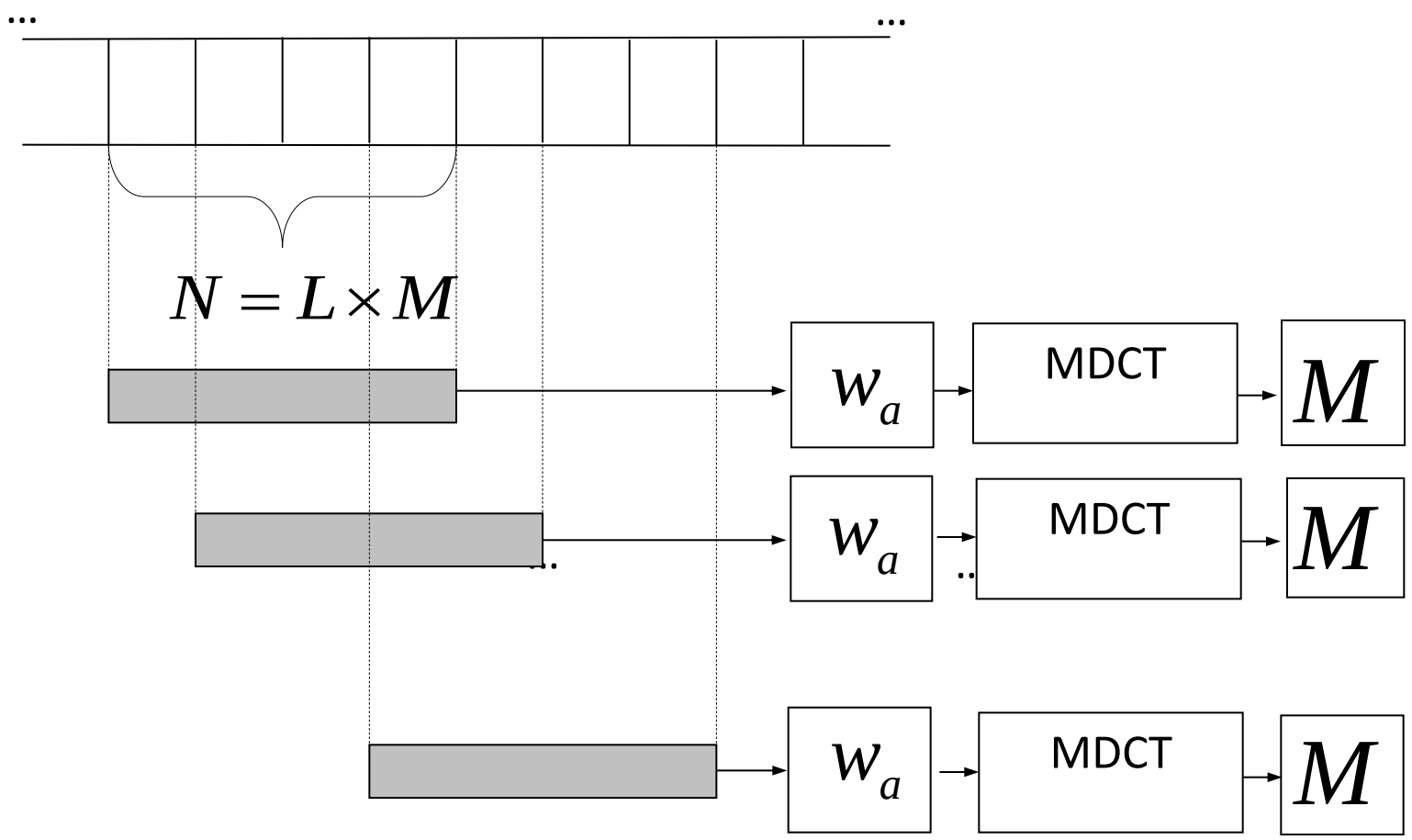
$$Y = \left[Y_0 \middle| Y_1 \middle| -Y_0 \middle| -Y_1 \middle| Y_0 \middle| Y_1 \ldots\right]^T,$$

Y_0, Y_1 of size $M \times M$ have the form

$$Y_0 = \begin{pmatrix} 0 & 0 \\ I & -J \end{pmatrix}, \; Y_1 = \begin{pmatrix} -J & -I \\ 0 & 0 \end{pmatrix},$$

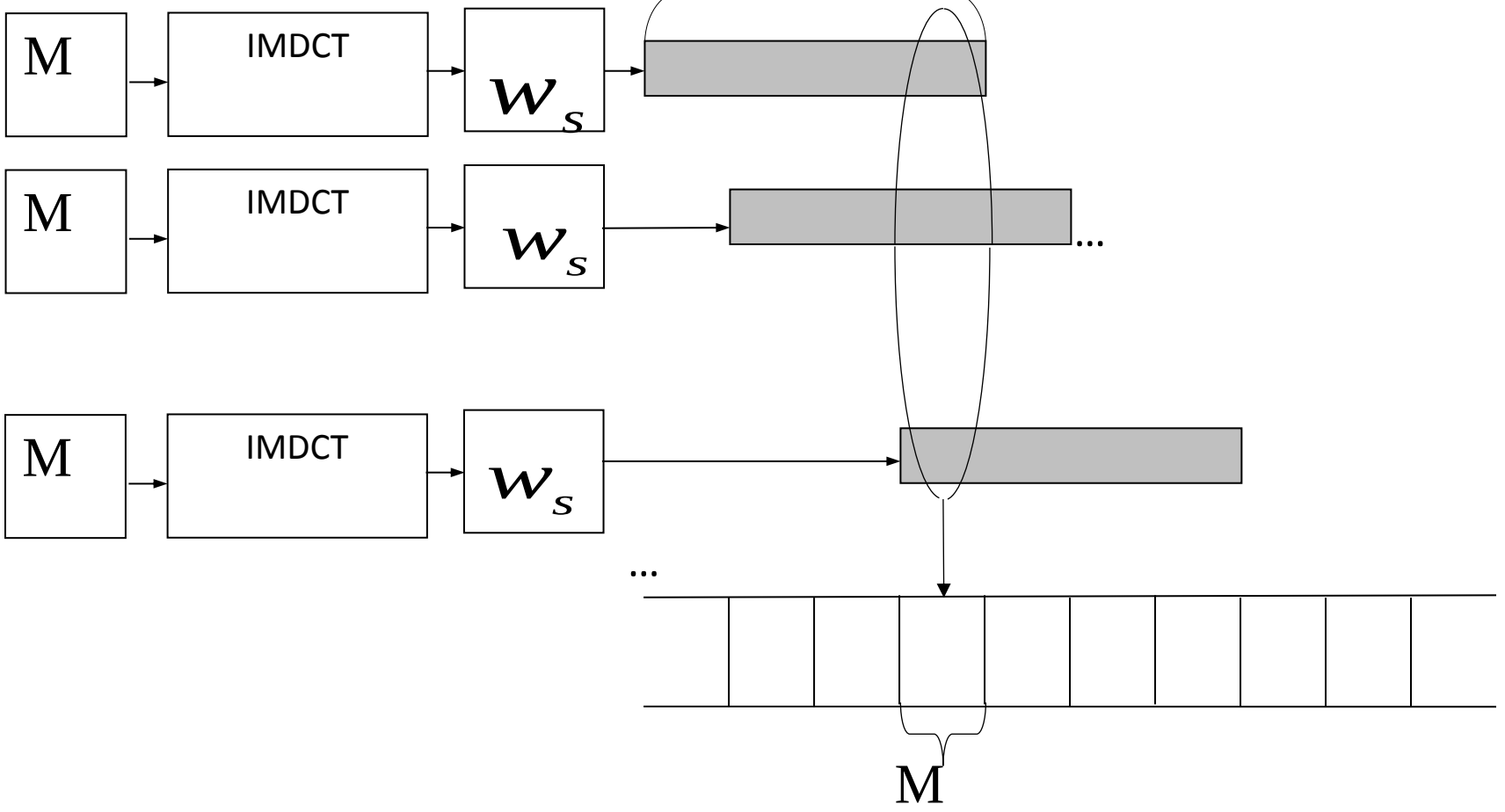
I is the identity matrix of size $M/2 \times M/2$ and J is the contra-identity matrix.

MDCT



MDCT

$$N = M \times L$$



PAM-I and PAM-II

PAM uses a separate, independent, time-to-frequency transform because it needs finer frequency resolution for accurate calculation of masking thresholds. For PAM-I it is 512-point FFT and for PAM-II it is 1024-point FFT.

Main steps of computing signal-to-mask ratio:

1. Calculate the sound pressure level for each subband
2. Calculate the absolute threshold $T_q(i)$ for i th spectral component of each subband. This threshold is the lower bound on the audibility of sound.
3. Separate spectral components into tonal and noise-like components.

PAM-I and PAM-II

PAM-I identifies tonal components based on local peaks of the audio power spectrum. Other components PAM-I sums into single nontonal component with frequency index equal to the geometric mean of the enclosing critical subband.

PAM-II computes a tonality index as a function of frequency.

The tonality index is based on a measure of predictability. This index is used to interpolate between pure tone-masking-noise or noise-masking-tone values.

4. Apply a spreading function.

5. Compute the global masking threshold for each spectral component i

$$T_g(i) = f(T_q(i), T_t(i,j), T_n(i,k)), \quad j = 1,...,m, \quad k = 1,...,n$$

$T_t(i,j), T_n(i,k)$ are thresholds for tone and noise components.

PAM-I and PAM-II

6. Find the minimum masking threshold for subband.

$$T_{\min}(n) = \min_i T_g(i) \quad \text{dB},$$

where $n = 1,...,32$ and minimum is taken over all spectral components of the n th subband.

7. Calculate the signal-to-mask ratio for each subband

$$SMR(n) = L_{sb} - T_{\min}(n).$$

For each subband the mask-to-noise ratio is computed

$$MNR(n) = SNR(n) - SMR(n),$$

where $SNR(n)$ is determined by the quantizer and tabulated.

Stereo coding

1. Intensity stereo (scaled sum of left and right channels)
2. Middle-side (MS) stereo (sum and difference of left and right channels)
3. Parametric stereo (stereo is synthesized from mono signal by using parameters : inter-channel phase difference, inter-channel correlation and so on.

HE-AAC combines Spectral Band Replication (high-frequency part is properly scaled low-frequency part shifted to the high-frequency region) and Parametric stereo

Demo

Original 1.44 Mb/s 

64 kb/s 

32 kb/s 

8 kb/s 

Original 1.44 Mb/s 

64 kb/s 

32 kb/s 

8 kb/s 

Shannon-Fano-Elias Coding

Let $x \in X = \{1, \ldots, M\}$, $p(x) > 0$,
$p(1) \geq p(2) \geq \cdots \geq p(M)$.
The cumulative sum is associated with the symbol x

$$Q(x) = \sum_{a \prec x} p(a),$$

that is,
$Q(1) = 0, Q(2) = p(1), \ldots, Q(M) = \sum_{i=1}^{M-1} p(i)$.

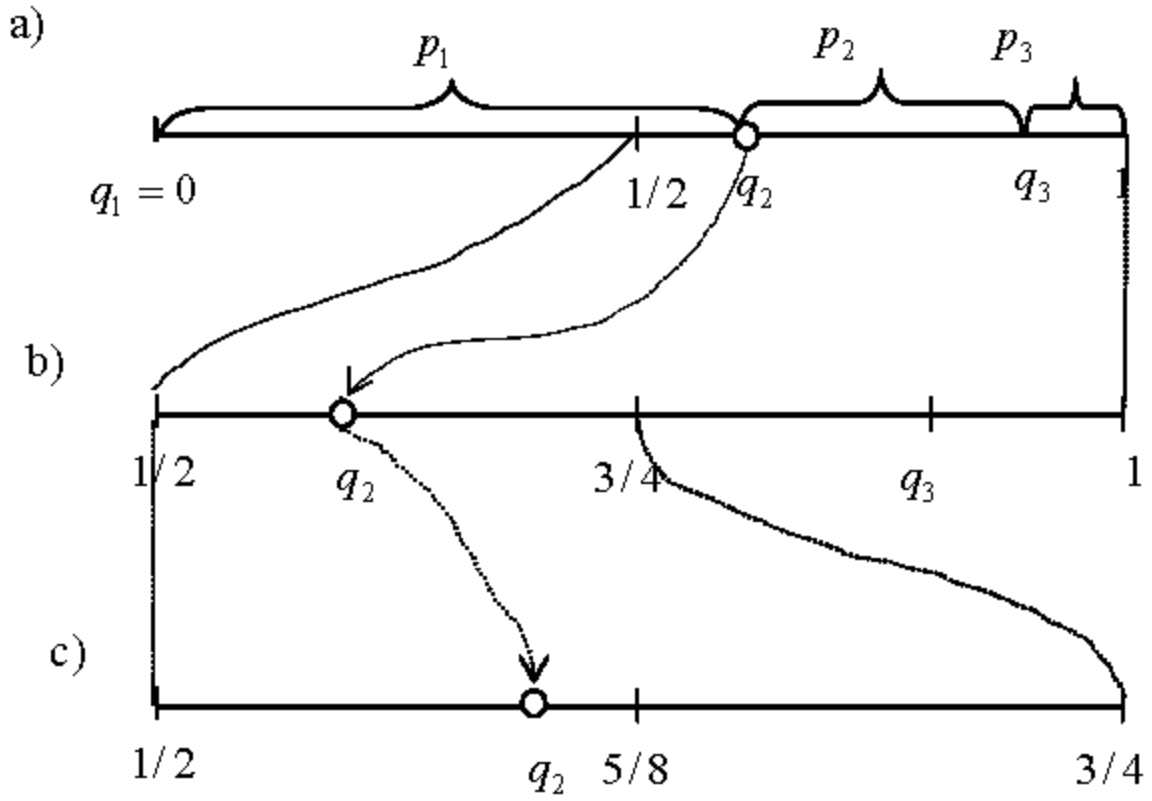
Then $\lfloor Q(m) \rfloor_{l_m}$ is a codeword for m,
where $l_m = -\lceil \log_2 p(m) \rceil$

Shannon-Fano-Elias Coding

x	$p(x)$	Q	Q in binary	$l(x)$	codeword
1	0.6	0	0.0	1	0
2	0.3	0.6	0.1001...	2	10
3	0.1	0.9	0.1110...	4	1110

$L = 1.6$ bits $H(X) = 1.3$ bits

Graphical Interpretation of Shannon-Fano-Elias coding



Since $q_2 > 1/2$ then code symbol = 1

Since $q_2 < 3/4$ then code symbol = 0

Segment length =1/4 and symbol probability $p_2 = 0{,}3 > 1/4$.

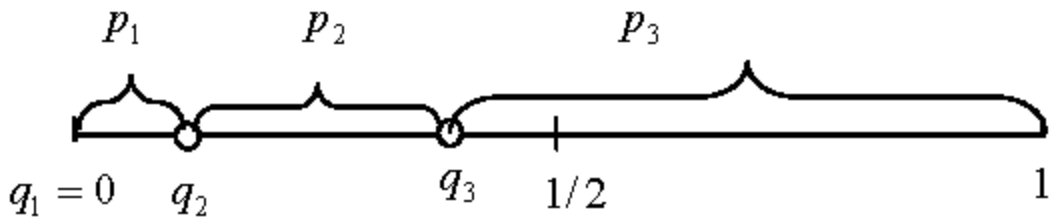
Shannon-Fano-Elias Coding

Choosing length l_m we used only right segment with respect to the point $Q(m)$. This segment is always shorter than the corresponding left segment since symbol probabilities are ordered in descending order.

$$H(X) \leq L < H(X) + 1.$$

Shannon-Fano-Elias coding

a)



Since $q_2 < 1/2$ then
code symbol = 0

Gilbert-Moore Coding

Let $x \in X = \{1, \ldots, M\}$, $p(x) > 0$.

The cumulative sum is associated with the symbol x

$$Q(x) = \sum_{a \prec x} p(a),$$

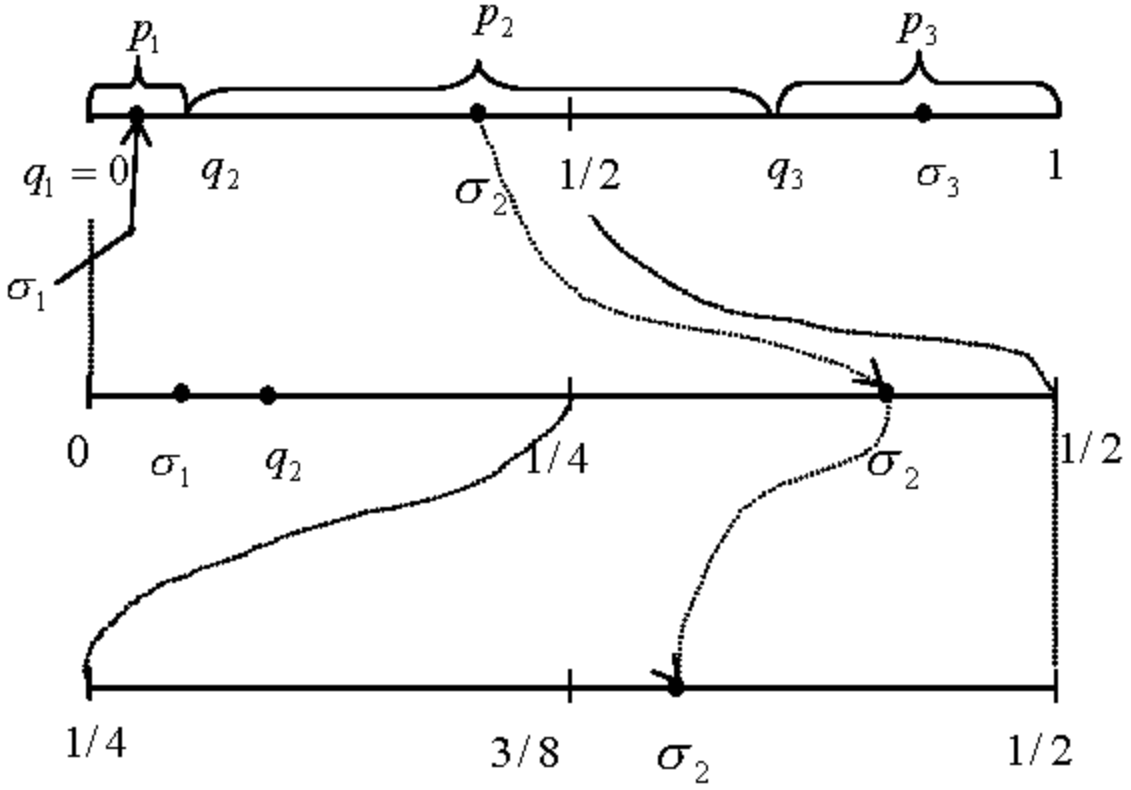
that is,

$Q(1) = 0$, $Q(2) = p(1), \ldots, Q(M) = \sum_{i=1}^{M-1} p(i)$.

Introduce $\sigma(x) = Q(x) + \frac{p(x)}{2}$

Then $\hat{\sigma}(m) = \lfloor \sigma(m) \rfloor_{l_m}$ is a codeword for m, where $l_m = -\lceil \log_2(p(m)/2) \rceil$.

Graphical Interpretation of Gilbert-Moore code



Since $\sigma_2 < 1/2$ then
code symbol $= 0$

Since $\sigma_2 > 1/4$ then
code symbol $= 1$

Segment

length $= 1/4 < \frac{p_2}{2} = 0{,}3$.

Gilbert-Moore Coding

We put point $\sigma(m)$ to the center of the segment $Q(m) + p(m)/2$ and choose length of codeword in such a manner that if l_m binary symbols have been transmitted the length of the interval of uncertainty is less than or equal to $p(m)/2$.

Gilbert-Moore Coding

x	$p(x)$	Q	σ	l	GM	ShFE
1	0.1	0.0	0.00001...	5	00001	0000
2	0.6	0.0001..	0.01100...	2	01	0
3	0.3	0.10110...	0.11011...	3	110	10

$L = 2.6$ bits $H(X) = 1.3$ bits

$$H(X) + 1 \leq L < H(X) + 2$$

Arithmetic coding

Arithmetic coding is a direct extension of the Gilbert-Moore coding scheme.

Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ be an M-ary sequence of length n. We construct the modified cumulative distribution function

$$\sigma(\mathbf{x}) = \sum_{\mathbf{a} \prec \mathbf{x}} p(\mathbf{a}) + \frac{p(\mathbf{x})}{2} = Q(\mathbf{x}) + \frac{p(\mathbf{x})}{2},$$

where $\mathbf{a} \prec \mathbf{x}$ means that $\mathbf{a}$ is lexicographically less than $\mathbf{x}$, $l(\mathbf{x}) = -\lceil \log_2(p(\mathbf{x})/2) \rceil$.

The code rate R is equal to

$$\frac{1}{n} \sum_{\mathbf{x}} p(\mathbf{x}) l(\mathbf{x}) = \frac{1}{n} \sum_{\mathbf{x}} p(\mathbf{x}) (\lceil \log_2 \frac{1}{p(\mathbf{x})} \rceil + 1)$$

$$< \frac{H(X^n) + 2}{n}$$

Implementation of arithmetic coding

$$Q(\mathbf{x}_{[1,n]}) = \sum_{\mathbf{a} \prec \mathbf{x}} p(\mathbf{a}) =$$

$$\sum_{\mathbf{a}:\mathbf{a}_{[1,n-1]} \prec \mathbf{x}_{[1,n-1]}, a_n} p(\mathbf{a}) +$$

$$\sum_{\mathbf{a}:\mathbf{a}_{[1,n-1]} = \mathbf{x}_{[1,n-1]}, a_n \prec x_n} p(\mathbf{a}),$$

where $\mathbf{x}_{[1,i]} = x_1, x_2, \ldots, x_i$. It is easy to see that

$$Q(\mathbf{x}_{[1,n]}) = Q(\mathbf{x}_{[1,n-1]}) + \sum_{\mathbf{a}:\mathbf{a}_{[1,n-1]} = \mathbf{x}_{[1,n-1]}, a_n \prec x_n} p(\mathbf{a})$$

$$= Q(\mathbf{x}_{[1,n-1]}) + p(\mathbf{a}_{[1,n-1]}) \sum_{a_n \prec x_n} p(a_n / \mathbf{a}_{[1,n-1]}).$$

Arithmetic coding

If the source generates symbols independently

$$p(\mathbf{a}_{[1,n-1]}) = \prod_{i=1}^{n-1} p(a_i),$$

$$\sum_{a_n \prec x_n} p(a_n/\mathbf{a}_{[1,n-1]}) = \sum_{a_n \prec x_n} p(a_n) = Q(x_n),$$

where $Q(x_i)$ denotes the cumulative probability for x_i.

$$Q(\mathbf{x}_{[1,n]}) = Q(\mathbf{x}_{[1,n-1]}) + p(\mathbf{x}_{[1,n-1]})Q(x_n),$$

$$p(\mathbf{x}_{[1,n-1]}) = p(\mathbf{x}_{[1,n-2]})p(x_{n-1}).$$

Implementation of arithmetic coding

$$F = 0; G = 1;\ Q(1) = 0;$$

for $j = 2 : M$

$$Q(j) = Q(j-1) + p(j-1);$$

end;

for $i = 1 : n$

$$F \leftarrow F + Q(x_i) \times G;$$

$$G \leftarrow G \times p(x_i);$$

end;

$$F = F + G/2; l = -\lceil \log_2 G/2 \rceil; \hat{F} \leftarrow \lfloor F * 2^l \rfloor;$$

Implementation of arithmetic coding

$X = \{a, b, c\}$,
$p(a) = 0.1,\ p(b) = 0.6,\ p(c) = 0.3$

$\mathbf{x} = (bcbab),\ n = 5$

i	x_i	$p(x_i)$	$Q(x_i)$	F	G
0	-	-	-	0.0000	1.0000
1	b	0.6	0.1	0.1000	0.6000
2	c	0.3	0.7	0.5200	0.1800
3	b	0.6	0.1	0.5380	0.1080
4	a	0.1	0.0	0.5380	0.0108
5	b	0.6	0.1	0.5391	0.0065

Codeword length $-\lceil \log_2 G \rceil + 1 = 9$

Implementation of arithmetic coding

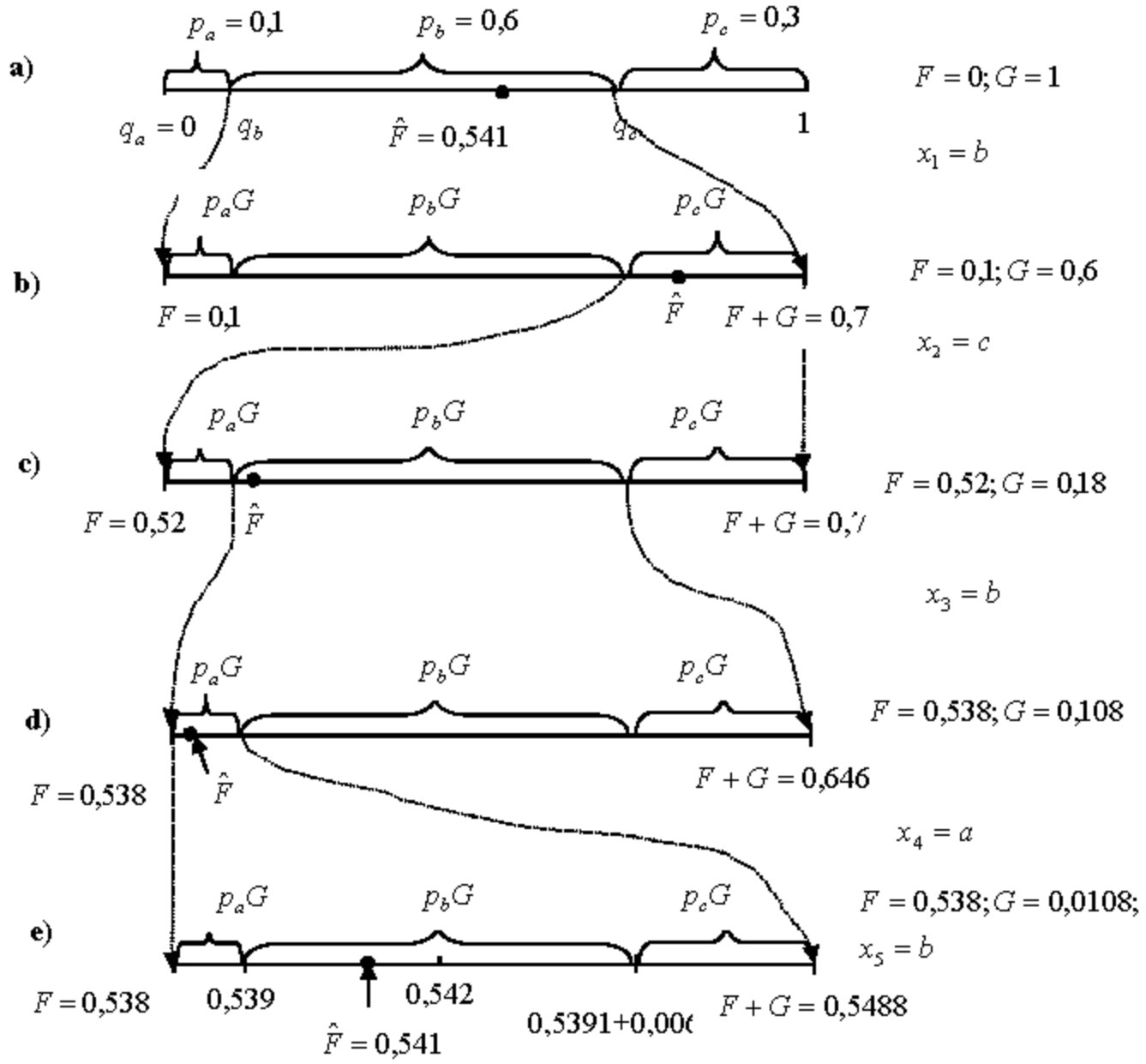
$F + G/2 = 0.5423...$ and
codeword $\hat{F} = \lfloor F + G/2 \rfloor_l = 100010101$

$H(X) = 1.3$ bits $R = 1.8$ bit/symbol

Let $p(1), \ldots, p(M)$ be numbers with binary representation of length d. Then at the first step F and G will be numbers with binary representaton of length $2d$. Next steps will require length of binary representation $3d, \ldots, nd$.

The complexity of coding procedure can be estimated as

$$d + 2d + \cdots + nd = \frac{n(n+1)d}{2}$$


a)
$p_a = 0{,}1$
$p_b = 0{,}6$
$p_c = 0{,}3$
$q_a = 0$
q_b
$\hat{F} = 0{,}541$
q_c
1
$F = 0; G = 1$
$x_1 = b$
b)
p_aG
p_bG
p_cG
$F = 0{,}1$
$\hat{F}$
$F + G = 0{,}7$
$F = 0{,}1; G = 0{,}6$
$x_2 = c$
c)
p_aG
p_bG
p_cG
$F = 0{,}52$
$\hat{F}$
$F + G = 0{,}7$
$F = 0{,}52; G = 0{,}18$
$x_3 = b$
d)
p_aG
p_bG
p_cG
$F = 0{,}538$
$\hat{F}$
$F + G = 0{,}646$
$F = 0{,}538; G = 0{,}108$
$x_4 = a$
e)
p_aG
p_bG
p_cG
$F = 0{,}538$
0,539
0,542
$\hat{F} = 0{,}541$
0,5391+0,00(
$F + G = 0{,}5488$
$F = 0{,}538; G = 0{,}0108;$
$x_5 = b$

Decoding of arithmetic code

$$\hat{F} \leftarrow \hat{F}/2^l; S = 0; G = 1;$$

for $i = 1 : n$

$$j = 1;$$

while $S + Q(j+1) \times G < \hat{F}$ **and** $j \leq M$

$$j \leftarrow j + 1$$

end;

$$S \leftarrow S + Q(j) \times G;$$

$$G \leftarrow G \times Q(j);$$

$$x_i = j;$$

end;

a, b, c $p(a) = 0.1$, $p(b) = 0.6$, $p(c) = 0.3$

Codeword 0100010101 $\hat{F} = 0.541$ $\hat{F} = 0.0100010101$

S	G	Hyp.	Q	$S + QG$	x_i	p
0.0000	1.000	a	0.0	$0.0000 < \hat{F}$		
		b	0.1	$\mathbf{0.1000 < \hat{F}}$	b	0.6
		c	0.7	$0.7000 > \hat{F}$		
0.1000	0.6000	a	0.0	$0.1000 < \hat{F}$		
		b	0.1	$0.1600 < \hat{F}$	c	0.3
		c	0.7	$\mathbf{0.5200 < \hat{F}}$		
0.5200	0.1800	a	0.0	$0.5200 < \hat{F}$		
		b	0.1	$\mathbf{0.5380 < \hat{F}}$	b	0.6
		c	0.7	$0.6460 > \hat{F}$		
0.5380	0.1080	a	0.0	$\mathbf{0.5380 < \hat{F}}$		
		b	0.1	$0.5488 > \hat{F}$	a	0.1
0.5380	0.0108	a	0.0	$0.5380 < \hat{F}$		
		b	0.1	$\mathbf{0.5391 < \hat{F}}$	b	0.6
		c	0.7	$0.5456 > \hat{F}$		

Implementation of arithmetic coding

Problems

- Algorithm requires high computational accuracy (theoretically infinite)
- Computational delay = length of the sequence to be encoded