



Università degli Studi di Palermo  
*Dipartimento di Ingegneria Informatica*



# Elaborazione di Immagini e Suoni / Riconoscimento e Visioni Artificiali 12 c.f.u.

Anno Accademico 2008/2009

Docente: ing. Salvatore Sorce

## La compressione audio

sulla base di materiale didattico originale del prof. Vincenzo Lombardo – MultiD@MS Torino

Facoltà di Lettere e Filosofia



## La compressione audio

L'arte di minimizzare le risorse per i dati audio

- ridurre la memoria occupata
- ridurre i costi di trasmissione
  
- Risultati attuali: buona qualità in confronto all'audio non compresso (stesso tasso di trasferimento dati)



## Obiettivo: *riproduzione trasparente*

- Audio che anche “orecchie” sensibili non riescono a distinguere dall'originale
- Numero minimo di bit mantenendo una riproduzione trasparente del segnale



## L'audio digitale

- Vantaggi
  - alta fedeltà delle copie del segnale
  - robustezza
  - gamma dinamica estesa
- Svantaggio: alto tasso di trasferimento dati
  - campionamento: 44.1 kHz (CD), 48 kHz (DAT)
  - quantizzazione: PCM lineare a 16 bit



## Perché nasce la compressione

L'audio digitale non compresso occupa ...

- molto spazio sull'hard-disk per la memorizzazione
- una banda ampia sul canale di trasmissione



## Esempio: brano di 1 min con qualità CD

- Parametri
  - campionamento = 44,100 campioni/sec
  - quantizzazione = 16 bit/campione = 2 byte/campione
  - canali = 2
- Memoria =  $44.100 \text{ c/s} * 2 \text{ can} * 2 \text{ byte/c} * 60 \text{ s}$   
~ 10 Mbyte per ogni minuto di musica stereo.
- Tempo necessario per il trasferimento di un minuto di musica stereo su un collegamento a 200 kbps =  
$$\frac{10 \text{ Mbyte} * 8 \text{ bit/byte}}{(200 \text{ kbps} * 60 \text{ s})} = \frac{10 * 1024 * 1024 * 8}{200 * 1024 * 60} \sim 7 \text{ minuti}$$

... per scaricare un solo minuto di musica stereo!  
(sarebbe impossibile qualsiasi applicazione *live*, come la web-radio)

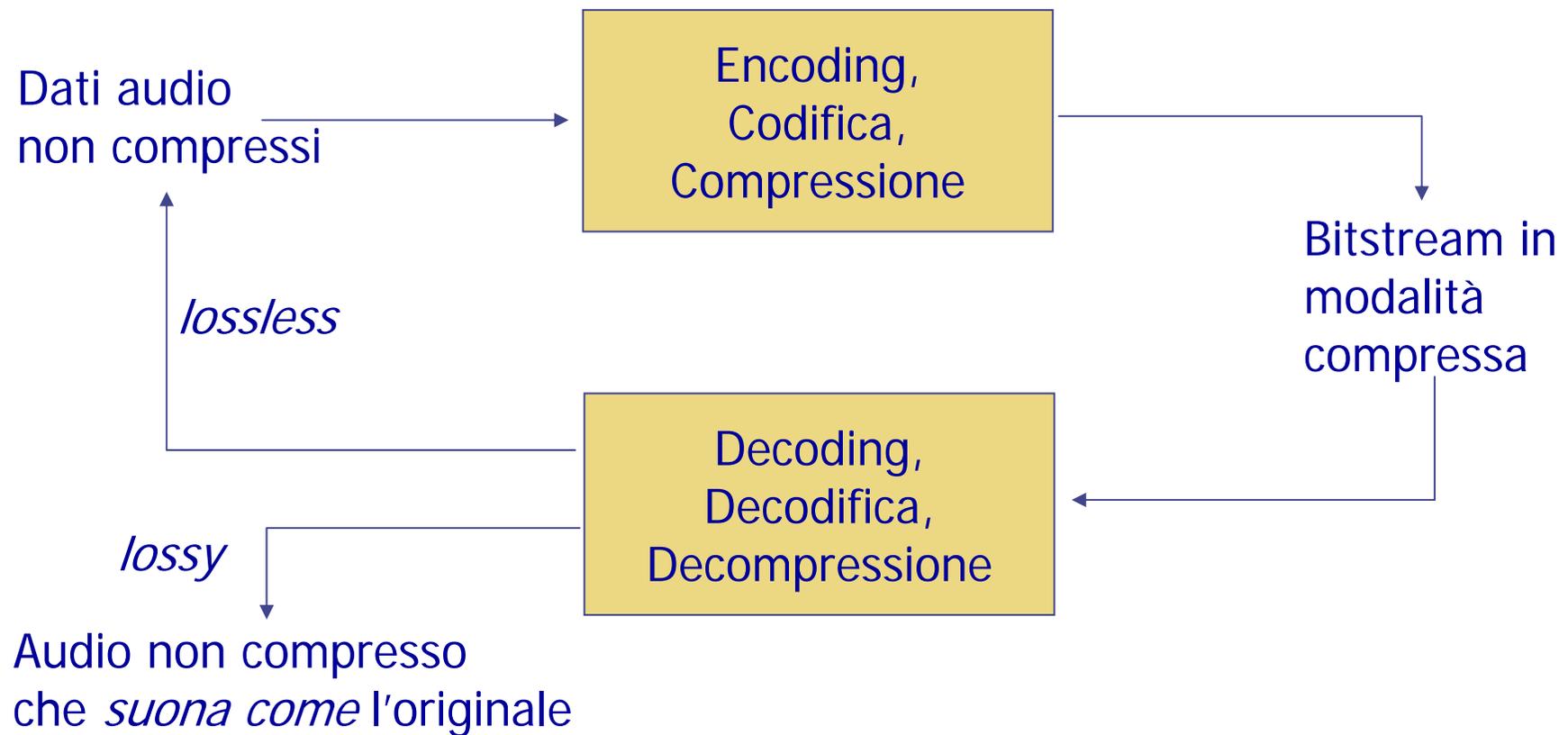


## Due livelli di applicazioni

- Bassa qualità
  - telefonia
- Alta qualità
  - colonne sonore per giochi su CD-ROM
  - memorie allo stato solido (flash) per i suoni
  - audio su Internet
  - broadcast di audio digitale (Radio e TV)

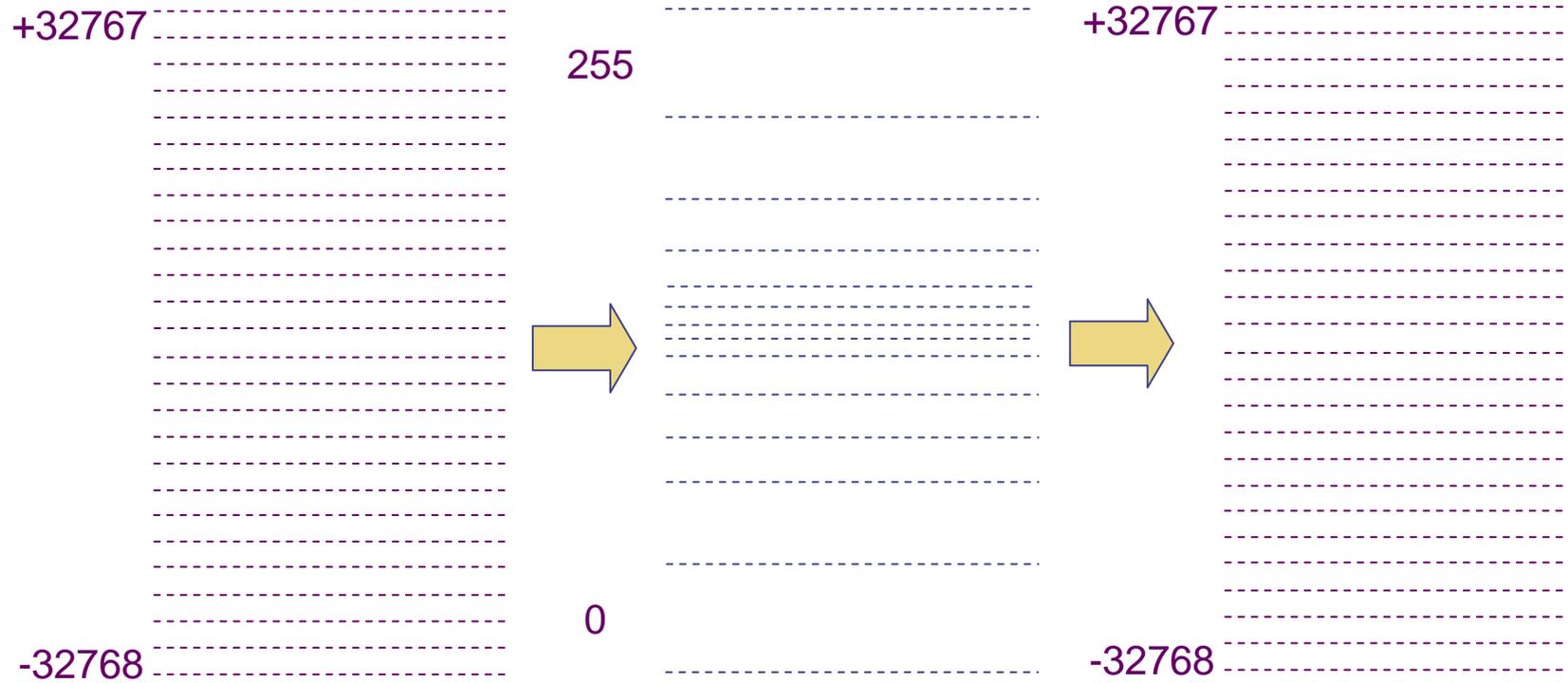


## Compressione e decompressione

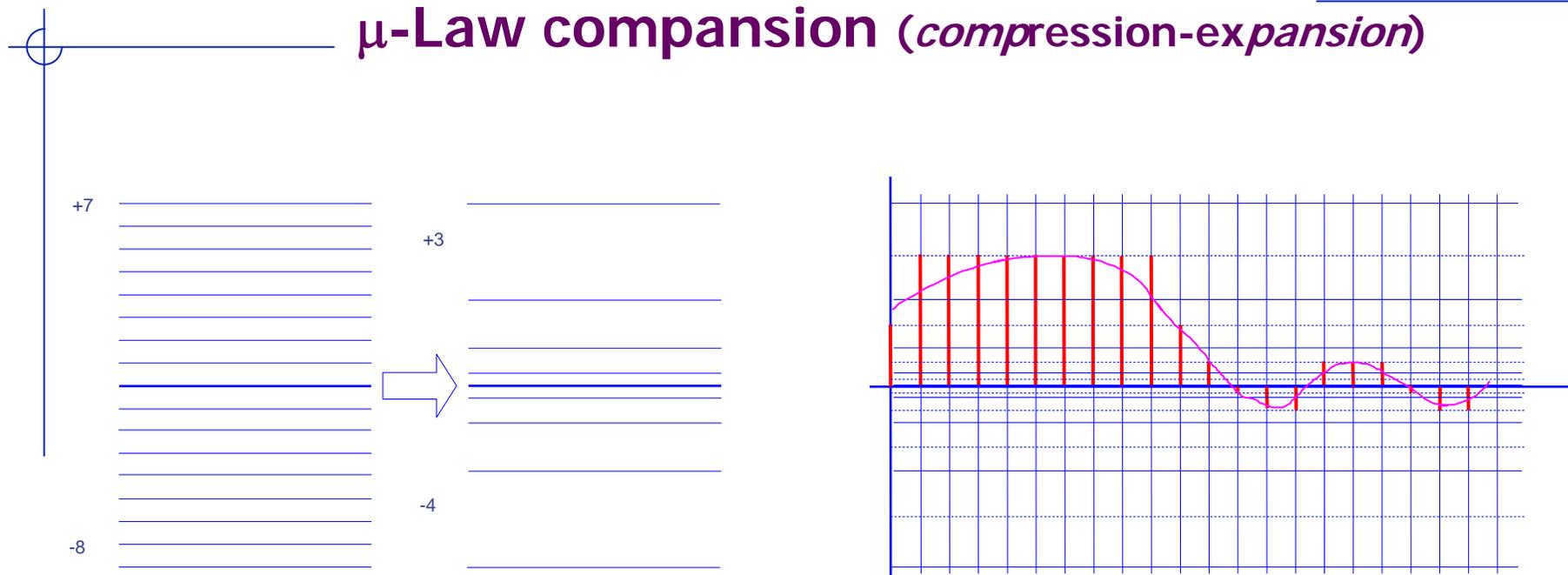




# ITU-T G.711, $\mu$ -law e A-law



## $\mu$ -Law compansion (*compression-expansion*)



- Quantizzazione logaritmica
- Gamma dinamica di 14 bit con una codifica a 8-bit
- Usata per i servizi voce ISDN in Nord America e Giappone
- Semplice il computo della codifica

## Codifica $\mu$ -law

Codifica PCM lineare  
(a 16 bit)

$$y = \begin{cases} 128 + \frac{127}{\ln(1+\mu)} \times \ln(1+\mu|x|) & x \geq 0 \\ 127 - \frac{127}{\ln(1+\mu)} \times \ln(1+\mu|x|) & x < 0 \end{cases}$$

codifica  
 $\mu$ -Law  
a 8-bit

Decodifica PCM lineare  
(a 16 bit)

$$x = \begin{cases} \frac{\exp\left(\frac{128+y}{127} \times \ln(1+\mu)\right) - 1}{\mu} & y \geq 128 \\ -\frac{\exp\left(\frac{127-y}{127} \times \ln(1+\mu)\right) - 1}{\mu} & y < 128 \end{cases}$$

## Esempi di codifica $\mu$ -law

Campione originale	x in (-1,+1)	Nuovo campione
-32768	-1	0
...	...	...
-32100	-0,9796142	0
-32000	-0,9765625	1
...	...	...
-200	-0,00610351	106
-100	-0,0030517	113
...	...	...
0	0	128
...	...	...
100	0,0030517	141
200	0,00610351	149
...	...	...
32000	0,9765625	254
32100	0,9796142	255
...	...	...
32767	1	255

## Lossy $\mu$ -law

- 200  
(a 16 bit)

$$y = \begin{cases} 128 + \frac{127}{\ln(1+\mu)} \times \ln(1+\mu|x|) & x \geq 0 \\ 127 - \frac{127}{\ln(1+\mu)} \times \ln(1+\mu|x|) & x < 0 \end{cases}$$

106  
a 8-bit

- 193  
(a 16 bit)

$$x = \begin{cases} \frac{\exp\left(\frac{128+y}{127} \times \ln(1+\mu)\right) - 1}{\mu} & y \geq 128 \\ -\frac{\exp\left(\frac{127-y}{127} \times \ln(1+\mu)\right) - 1}{\mu} & y < 128 \end{cases}$$

$\mu = 255$   
 $0 \leq x \leq 1$

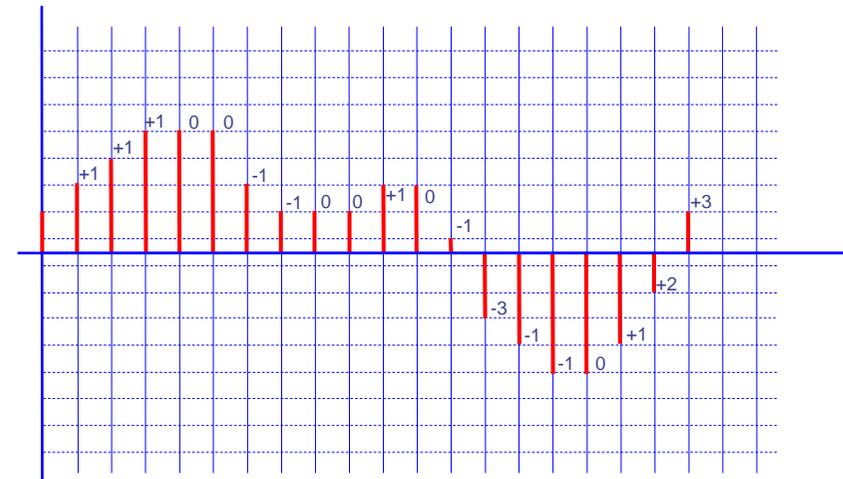
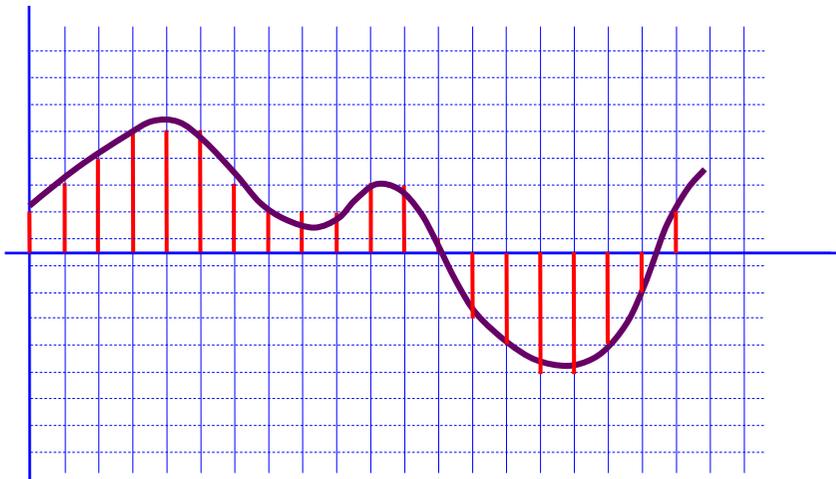


## ADPCM (CCITT G.721, G.723, e ITU-T G.726)

- Standard per la compressione di dati vocali
  - CCITT G.721 (ADPCM a 32 Kbps)
  - CCITT G.723 (ADPCM a 24 e 40 Kbps)
- ADPCM = Adaptive Differential PCM
  - metodo comune di compressione
  - buon compromesso tra velocità di elaborazione, tasso di compressione e decodifica di qualità



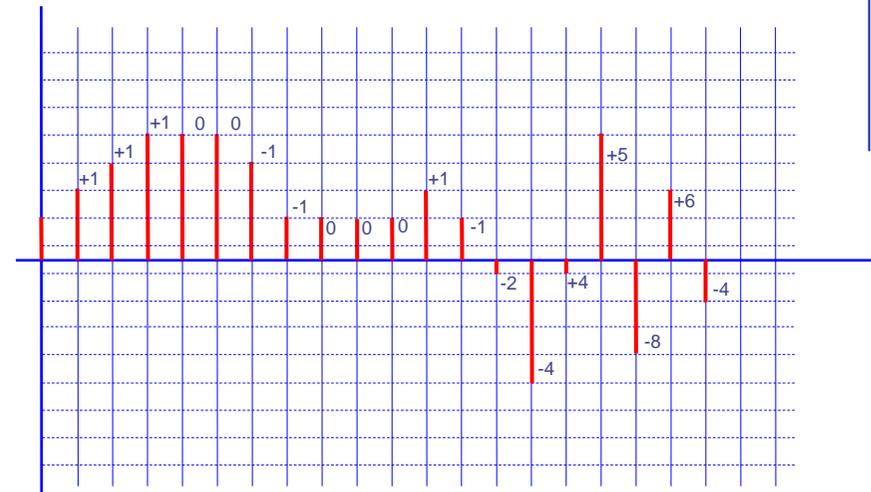
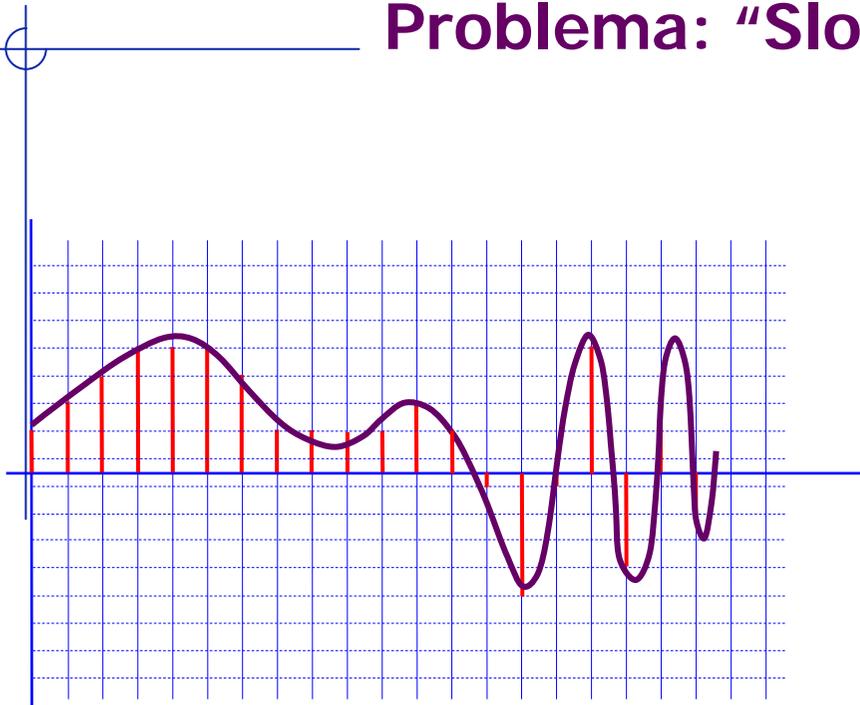
## Codifica della differenza (DPCM)



- Ridondanze temporali tra campioni (è molto probabile che campioni “vicini” nel tempo siano abbastanza simili, cioè differiscano di poco)
- La differenza tra due campioni a  $n$  bit si può rappresentare con meno bit di  $n$
- Si memorizza la differenza rispetto al campione precedente



## Problema: "Slope Overload"



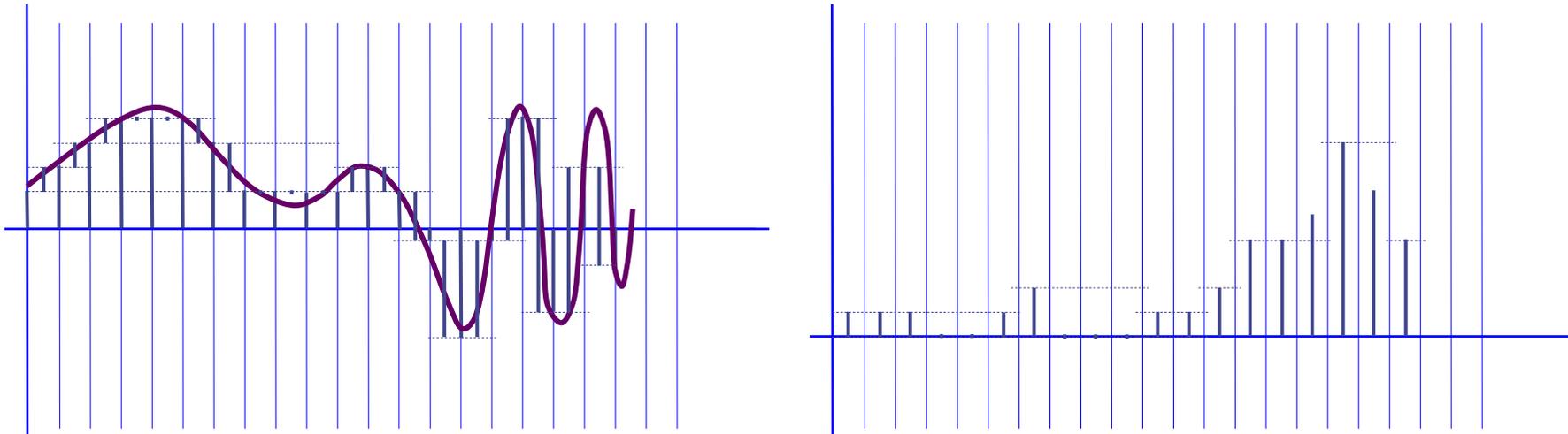
In questa zona bastano  
2 bit per codificare le  
differenze

In questa  
zona servono  
4 bit

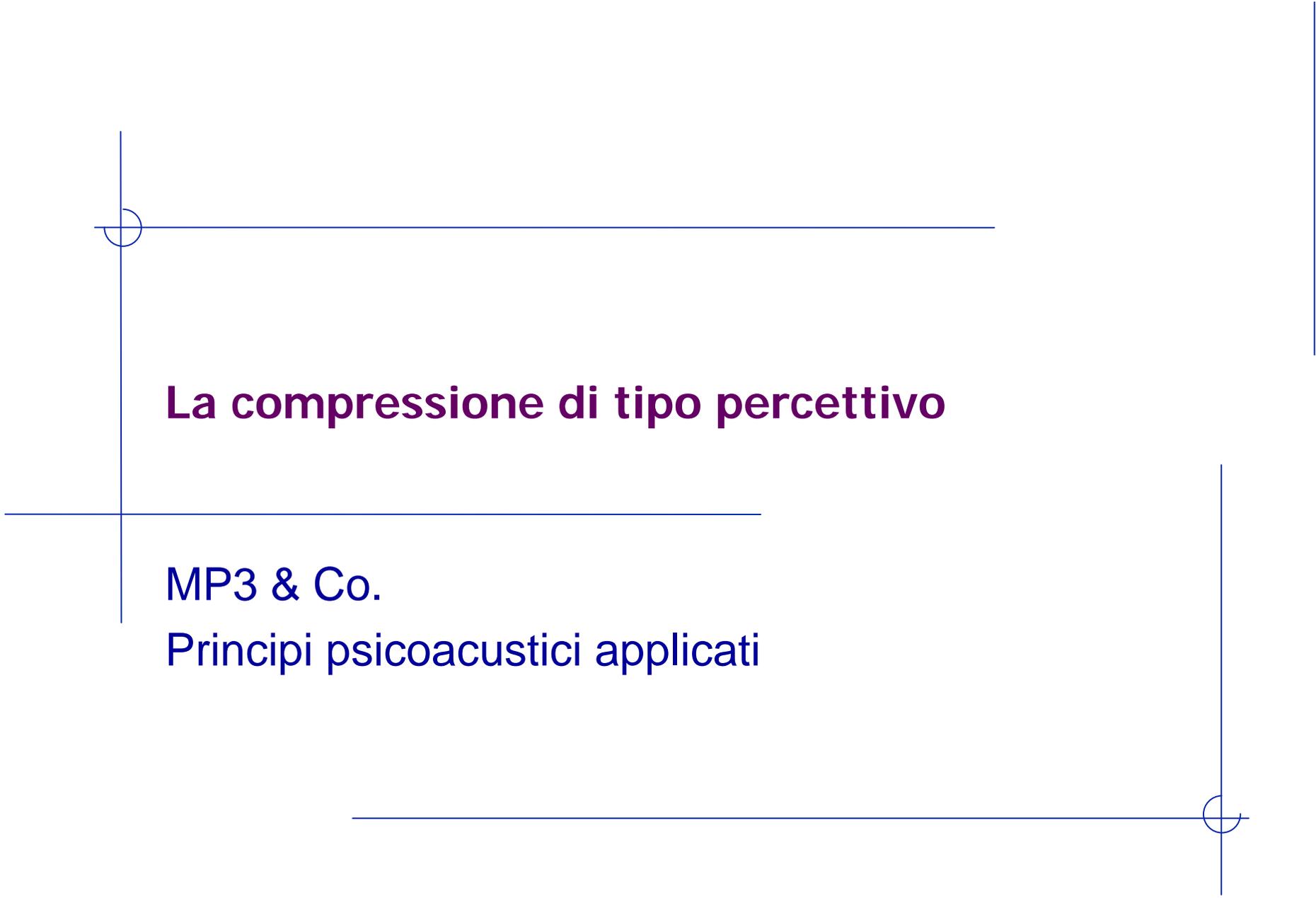
- Differenze elevate (alte frequenze) non si possono rappresentare con un numero piccolo di bit
- Gli errori introdotti porterebbero a distorsioni sulle alte frequenze



## Adaptive DPCM (ADPCM)



- Passi di quantizzazione più grandi per differenze più grandi, e viceversa
- Si usano i campioni precedenti per stimare i cambiamenti futuri

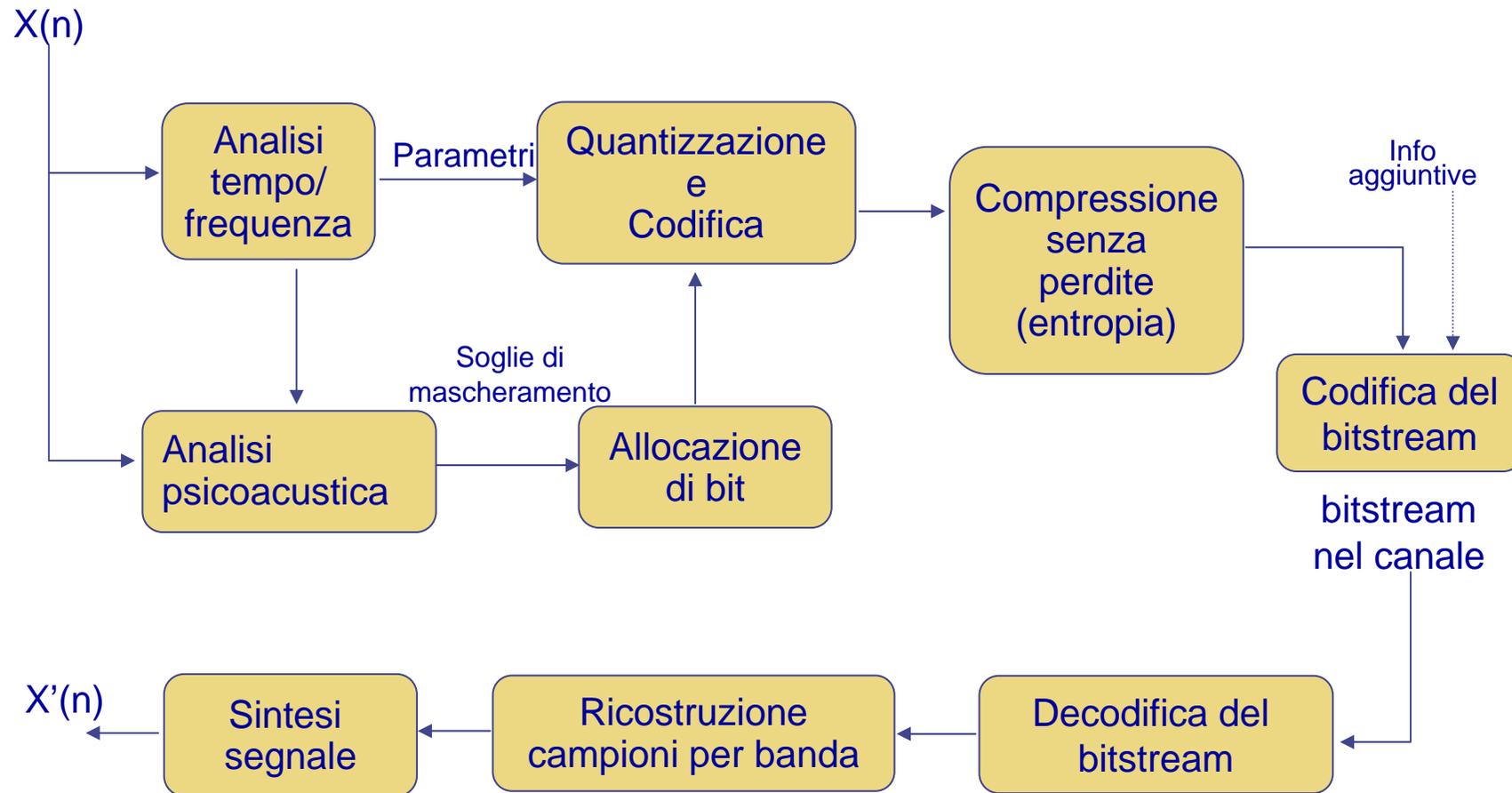


## **La compressione di tipo percettivo**

MP3 & Co.

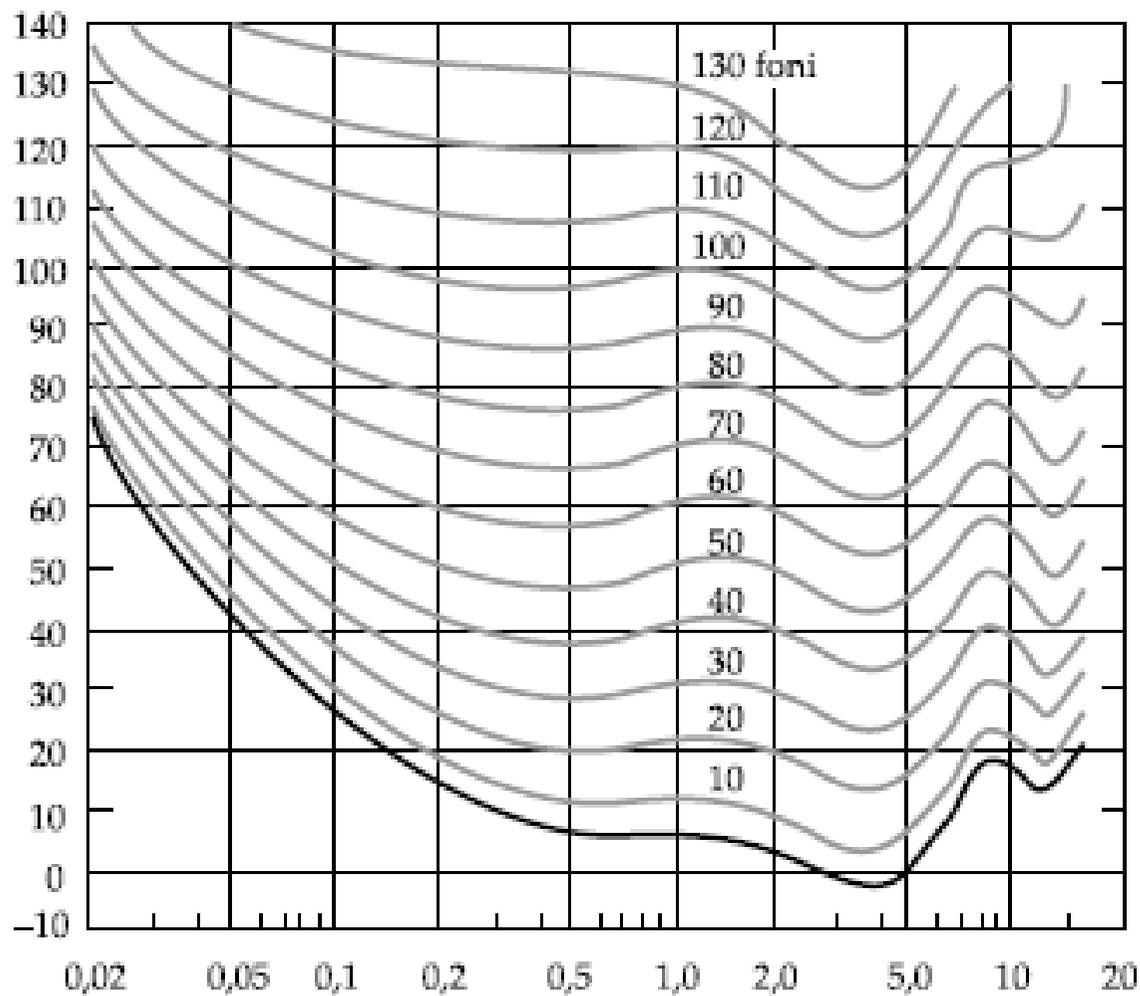
Principi psicoacustici applicati

## Schema generale



$X'(n) \neq X(n)$ , ma nei limiti della larghezza di banda disponibile, deve "suonare come"  $X(n)$

## Soglia assoluta dell'udito





## Le bande di frequenza critiche

Suono 1: nota pura a frequenza costante  $f_0 = 220$  Hz

Suono 2: glissando (frequenza variabile con continuità) tra 220 Hz e 290 Hz

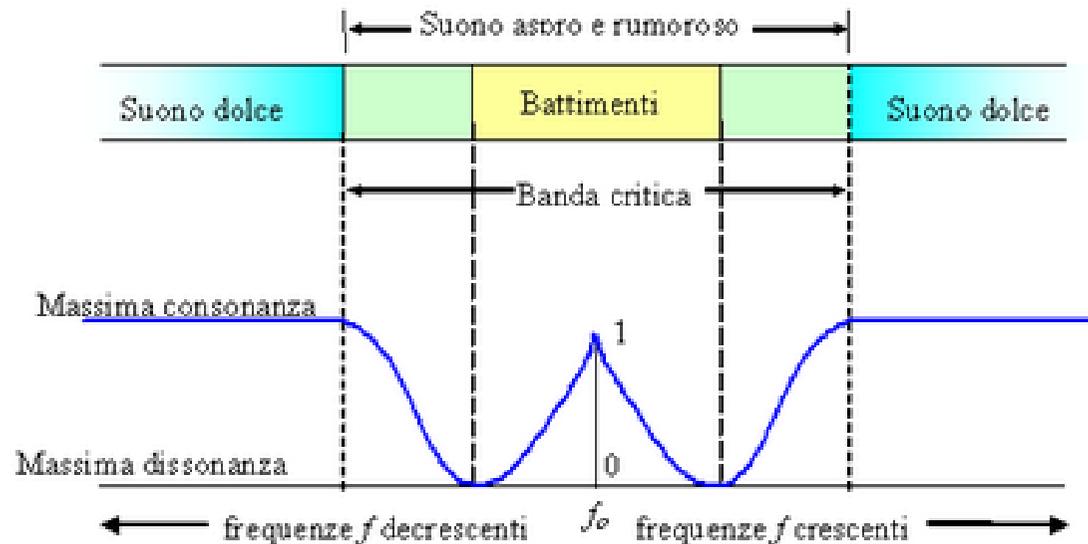
Suono 3: sovrapposizione dei suoni 1 e 2

Cosa si avverte?

- All'inizio le due frequenze sono uguali e viene percepita solo la nota iniziale;
- Non appena la frequenza variabile inizia a discostarsi dal valore  $f_0$  percepiamo nell'ordine:
  - un suono che subisce lente oscillazioni dell'intensità (è questo il fenomeno dei *battimenti*);
  - un suono nel quale le oscillazioni dell'intensità aumentano di frequenza fino a tradursi in un suono aspro, estremamente dissonante e fastidioso per l'orecchio;
  - un suono che torna ad essere gradevole ma nel quale si percepiscono chiaramente le due note che lo costituiscono come entità separate (è quello che i musicisti chiamano un bicordo).



## Le bande di frequenza critiche



Le note cominciano ad essere percepite come distinte dall'istante in cui il tono più alto ha una frequenza di circa 250 Hz.

Le due note in questo istante differiscono di 30 Hz e cioè ad una distanza molto maggiore del potere risolutivo in frequenza per suoni non simultanei.

Verifica:

Suono 1: nota pura a 245 Hz

Suono 2: nota pura a 250 Hz



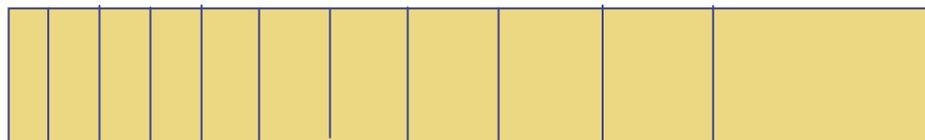
## Le bande di frequenza critiche

Suoni molto ravvicinati in frequenza eccitano posizioni molto vicine della membrana basilare, compromettendo la facoltà di percepire i due suoni come entità distinte.

Due suoni vengono percepiti come distinti solo se coinvolgono terminazioni nervose sufficientemente distanti, cioè se mettono in oscillazione regioni differenti della membrana basilare, ovvero se cadono in due bande critiche distinte.

Una **banda critica** è quindi un intervallo di frequenze entro alla quale *due toni puri simultanei* non possono essere percepiti come distinti.

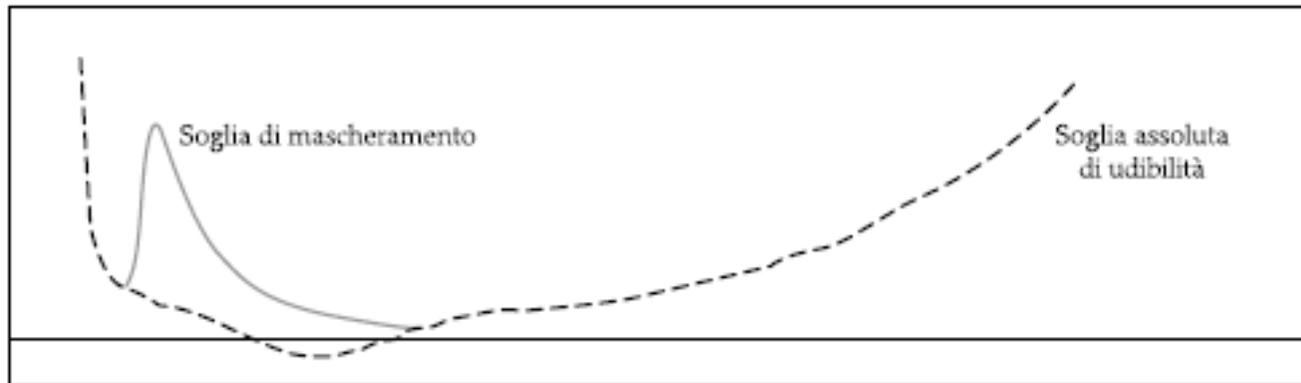
Bande larghe 100 Hz fino a circa 500 Hz, poi l'ampiezza cresce con la frequenza di centro banda



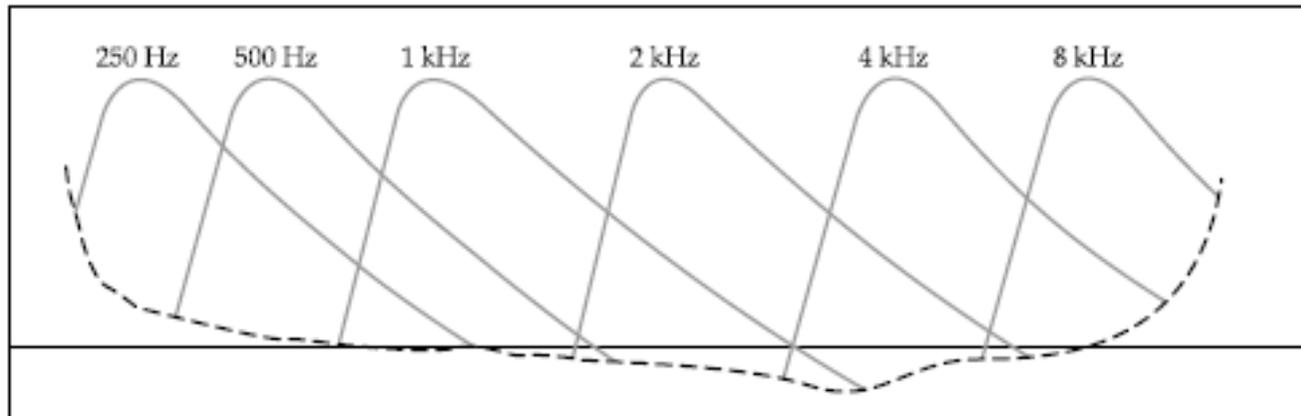
<i>Banda</i>	<i>Centro (Hz)</i>	<i>Estremi (Hz)</i>
<b>1</b>	50	-100
<b>2</b>	150	100-200
<b>3</b>	250	200-300
...		
<b>7</b>	700	630-770
...		
<b>11</b>	1370	1270-1480
...		
<b>15</b>	2500	2320-2700
...		
<b>19</b>	4800	4400-5300
<b>20</b>	5800	5300-6400
...		
<b>25</b>	19500	15500-



## Soglie di mascheramento globale

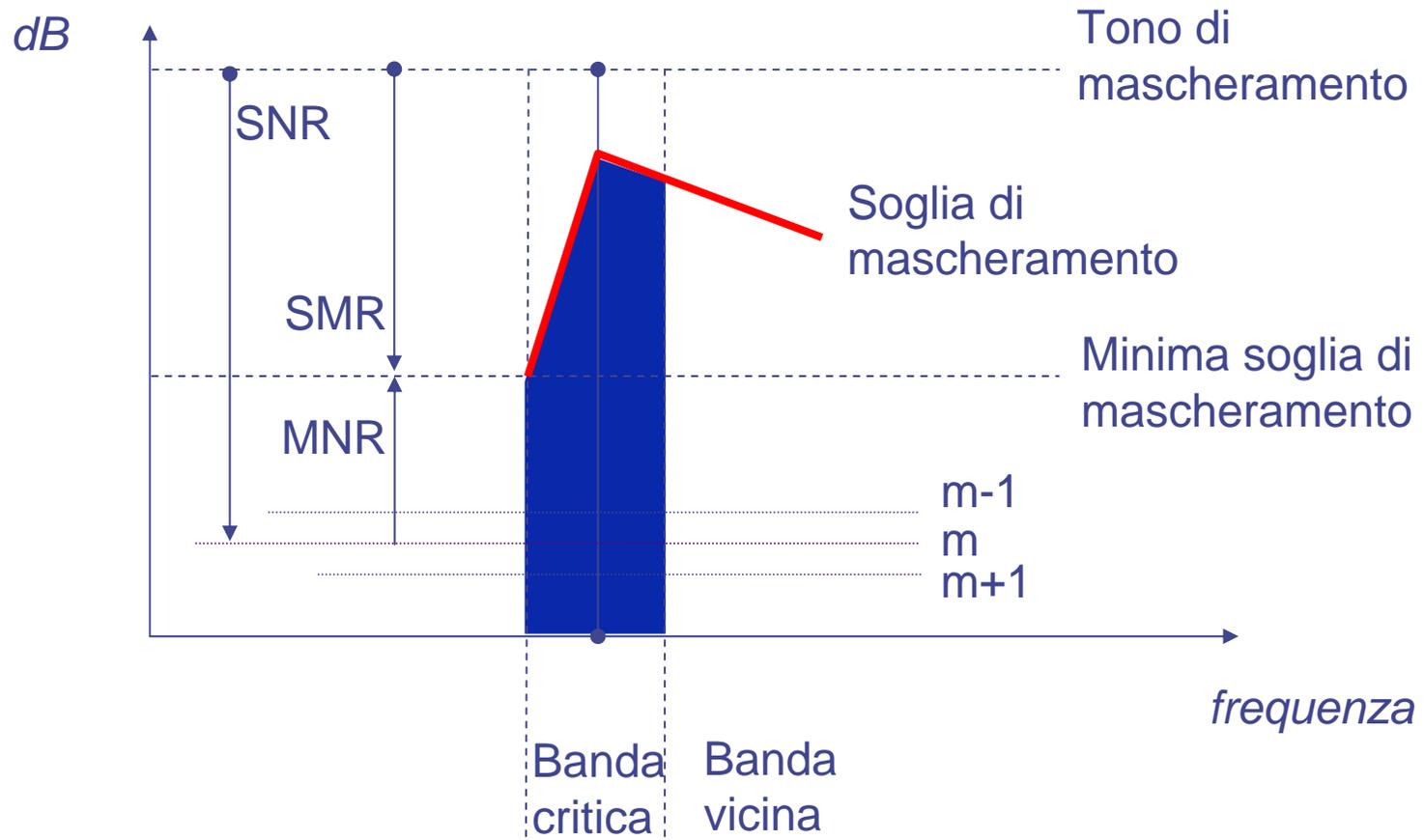


a)



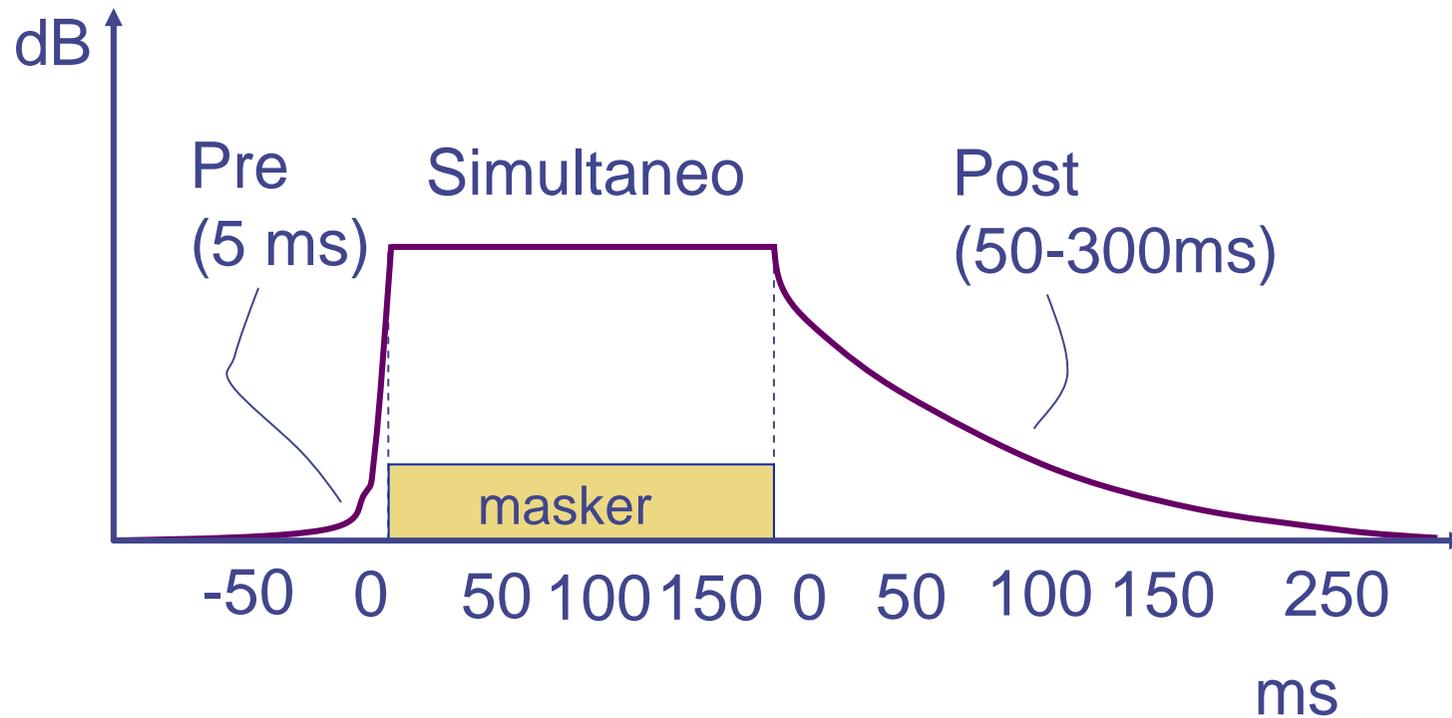
b)

# Mascheramento simultaneo





## Mascheramento temporale





## MPEG (Motion Picture Experts Group)

Gruppo di lavoro di ISO/IEC per lo sviluppo di standard internazionali per:

- compressione, decompressione, elaborazione, e rappresentazione codificata di
  - Video
  - Audio
  - combinazioni A/V



## Standard MPEG

- MPEG-1: memorizzazione/recupero video/audio (11/92)
- MPEG-2, standard per la TV digitale (11/94)
- MPEG-4
  - v. 1, standard applicazioni multimediali (10/98)
  - v. 2, standard audio/video HDTV (12/99)
- MPEG-7: standard per la rappresentazione dei contenuti
  - ricerca, filtraggio, gestione di info multimediale
  - rilasciato a luglio 2001



## MPEG-1/2 Layer III (MP3)

- 1987 progetto Eureka (Digital Audio Broadcasting)
- lavoro su audio percettivo in collaborazione tra Fraunhofer IIS e l'Università di Erlangen
- risultato: algoritmo molto potente ISO-MPEG Audio Layer-III
  
- <http://it.wikipedia.org/wiki/Mp3>



## Lo schema di compressione MPEG-1

- Struttura di base comune su più layer
  - calcolo delle sottobande
  - calcolo del modello psicoacustico
- 3 Layer
  - Layer I: semplice, poco efficace
  - Layer III: complesso, efficace
- Scelta del layer - rapporto qualità/compressione



## Modalità di compressione

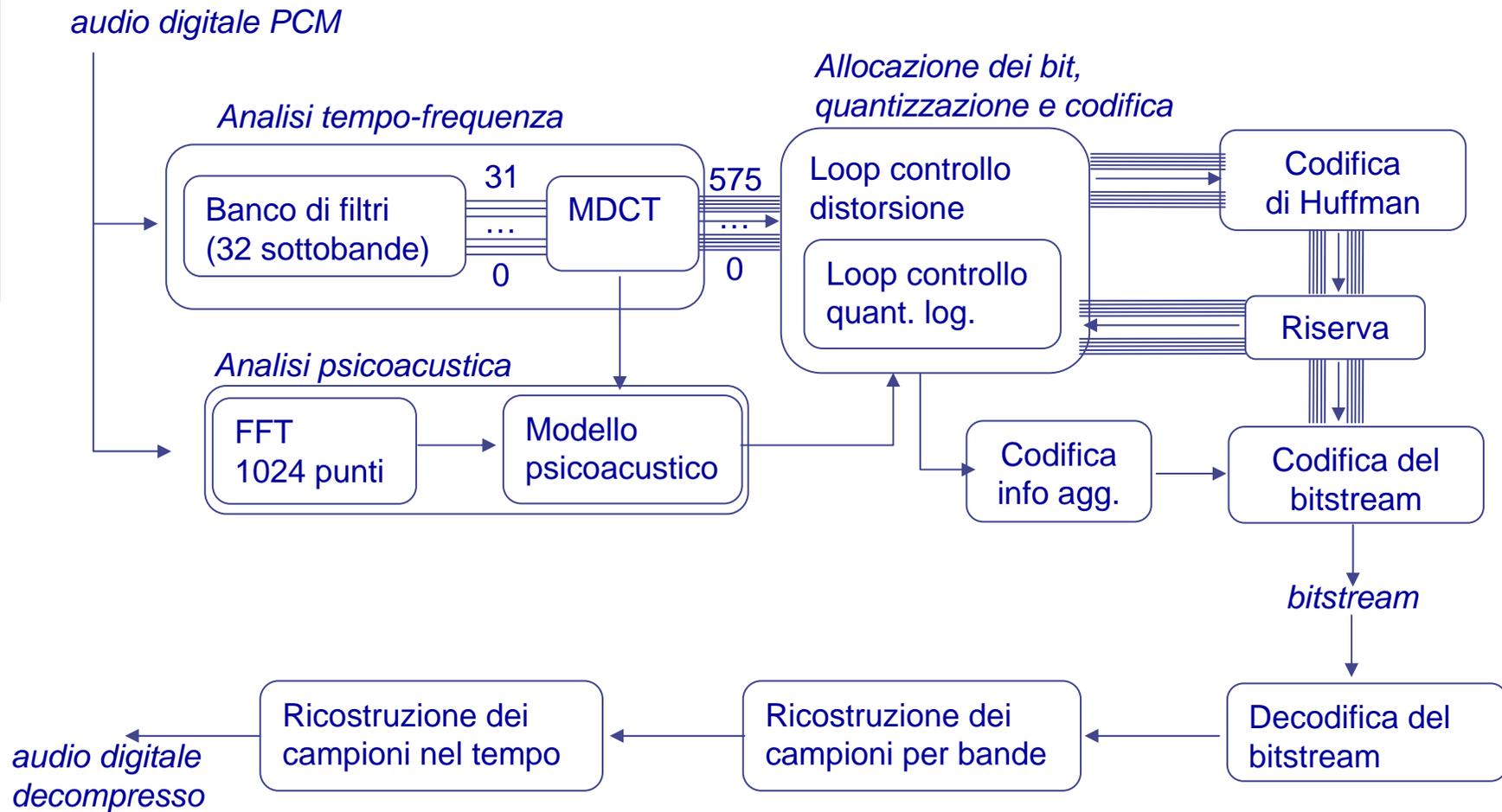
- Più frequenze di campionamento (32/44.1/48)
- Bitstream compresso supporta mono, dual mono, stereo, joint stereo
- Bit rate da 32 a 224 kbps (compressione da 2,7 a 24 volte) – tassi fissi e variabili
- Supporta il controllo e la correzione degli errori
- Informazioni supplementari



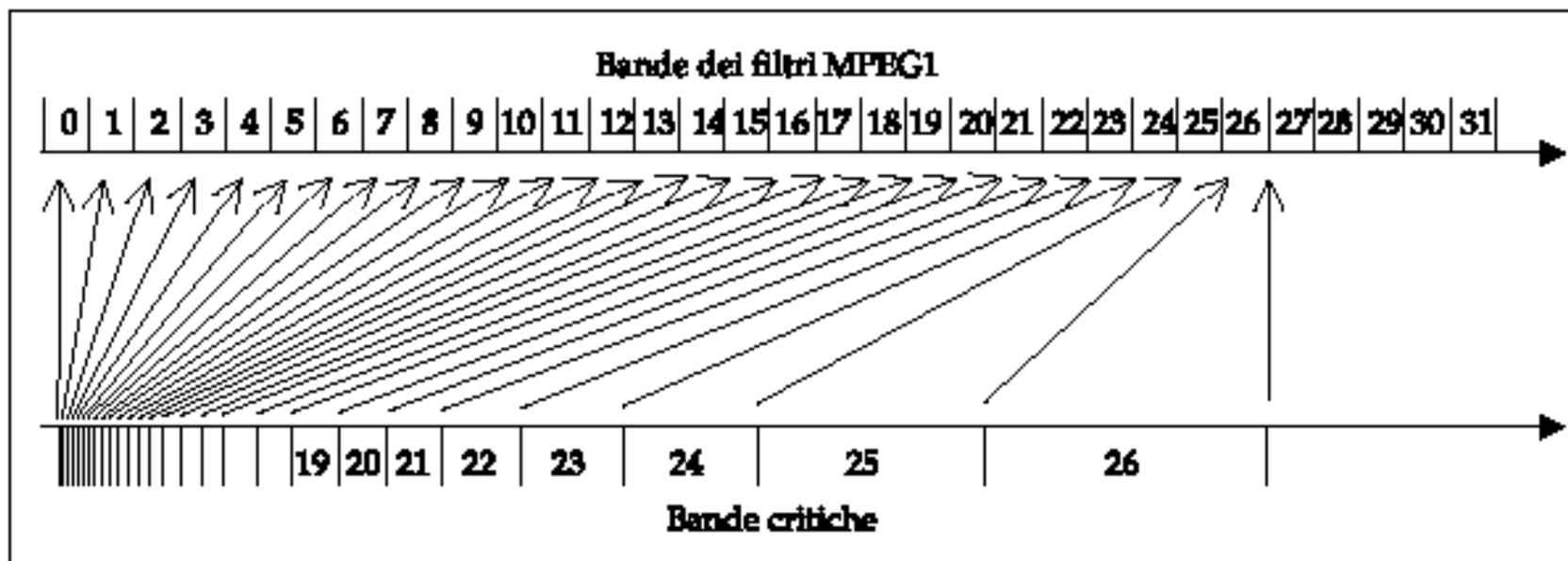
## I tre livelli di compressione

- Layer I
  - il più semplice (bitrate oltre 128 kbps a canale)
  - DCC di Philips usa la compressione di Layer I a 192
- Layer II
  - complessità media (bitrate circa 128 kbps a canale)
  - applicazioni in DAB
- Layer III
  - il più complesso, migliore qualità
  - 64 kbps adatto per trasmissione audio su ISDN

# MPEG 1 - Layer III



## Corrispondenza bande critiche e MPEG

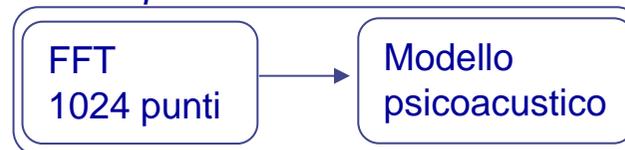




## Modello psicoacustico

- Calcolo soglie mascheramento globale
- Necessario per stabilire il livello max di quantizzazione
- Non si deve udire il rumore di quantizzazione

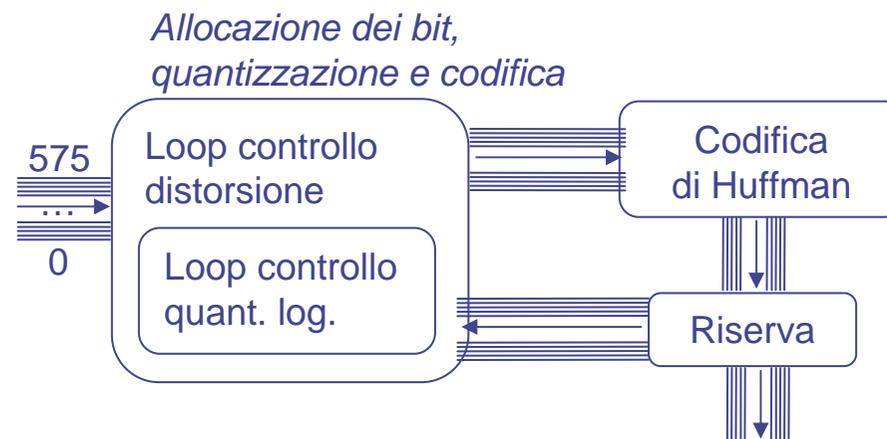
*Analisi psicoacustica*





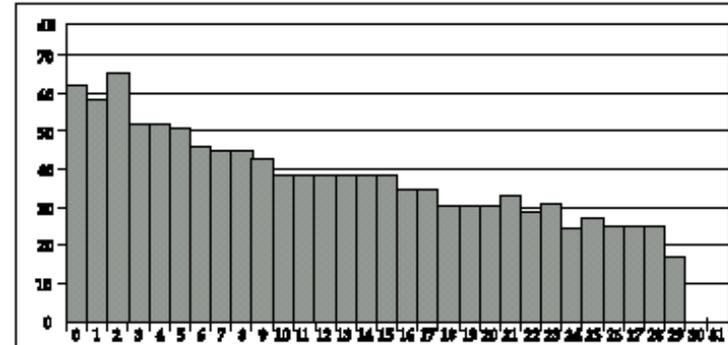
## Compressione dei dati audio

- Più cicli di quantizzazione e codifica
- Modello psicoacustico + bitrate in uscita

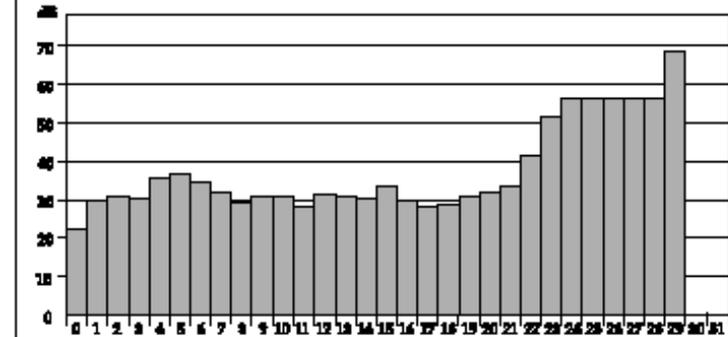


## Allocazione dei bit

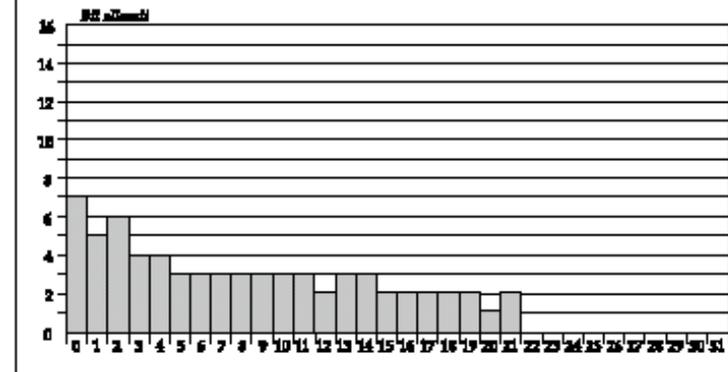
Spettro di ampiezza per 32 sottobande



Soglia globale di mascheramento



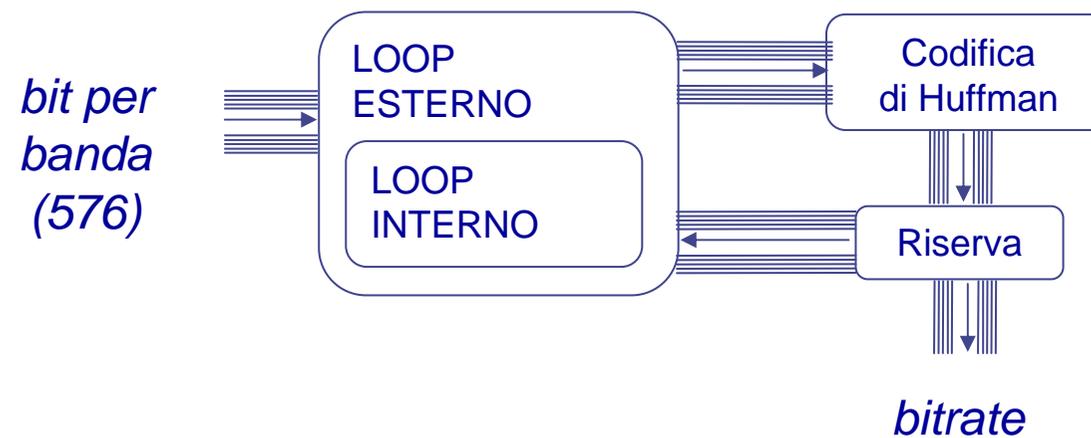
Allocazione di bit per banda





## Due cicli di lavoro

- Loop interno sui limiti del bitrate (RATE LOOP)
- Loop esterno sul controllo del rumore per banda





## Tipiche riduzioni di MPEG

Riferimento standard (CD):

$$16 \text{ bit} * 2 \text{ canali} * 44100 \text{ sr} = 1.411 \text{ kbps}$$

➤ 1:4 con Layer I (PASC)

■ corrisponde a 384 kbps per un segnale stereo

➤ 1:6...1:8 con Layer II

■ corrisponde a 256..192 kbps per un segnale stereo

➤ 1:10...1:12 con Layer III (MP3)

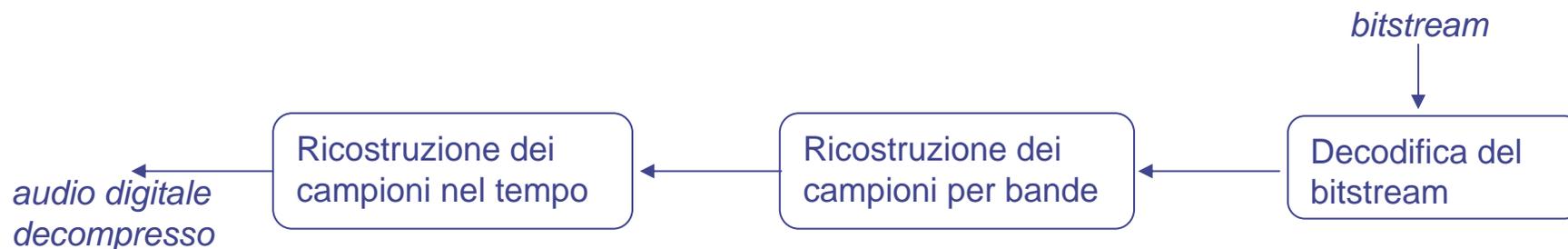
■ corrisponde a 128..112 kbps per un segnale stereo

sempre mantenendo *percettivamente* la qualità audio CD



## Decodifica

- Sintetizza un segnale a partire dalle componenti spettrali codificate
- Non si ha più lo stesso segnale!!!
- Tutto dipende dal bit-rate





## Esempi di bitrate

- *bit-rate* = numero medio di bit consumati da un secondo di dati audio (kbps)
- bit-rate per il CD = 1411.2 kbps
- bit-rate per MP3 per qualità CD = 128 kbps



## Le performance di MP3

- qualità telefonica: 96:1 (2.5 kHz / mono / 8 kbps)
- meglio di AM radio: 24:1 (7.5 kHz / mono / 32 kbps)
- simile a FM radio: 26...24:1 (11 kHz / stereo / 56...64 kbps)
- quasi-CD: 16:1 (15 kHz / stereo / 96 kbps)
- CD: 14..12:1 (>15 kHz / stereo / 112..128kbps)
  - prende approx. 1Mb/minute di spazio hard-disk
- Oltre: 8...4:1 per la musica acustica

## MP3 Codec

- programma del tipo di una libreria di sistema (collezione di funzioni)
- vengono lanciate dai programmi di frontend
- codec in distribuzione con frontend MP3



## L'encoder migliore

- Non ha senso chiedersi quale sia l'encoder migliore: la risposta dipende dalle esigenze
  - encoder veloci/lenti ( $>$  velocità,  $<$  fedeltà audio)
  - confrontare mp3 ottenuti da encoder diversi a parità di bit-rate
- Consiglio pratico: creare MP3 con basso (ma adeguato) bitrate con encoder 'lenti'



## I Frontend MP3

- interfaccia ai codec
- Alcuni frontend implementano funzioni per “normalizzare” il volume, o realizzano ID TAGS
- IDTAG è informazione (testuale, in genere) nel file di layer III (autore, titolo, etc ...)
- Esempio: MP3Editor



## Player MP3

- suona mentre decomprime
- i dati audio (campioni) vengono inviati alla scheda per la conversione