FONDAMENTI DI AUDIO DIGITALE *

Francesca Ortolani

1 Segnali

Nelle telecomunicazioni e in elettronica tipicamente si intende per segnale una funzione che trasporta informazione. L'informazione può considerarsi il risultato di un fenomeno o di un comportamento di un sistema. Ad esempio, un segnale elettrico è prodotto dallo spostamento di elettroni causato dalla generazione di un campo elettrico. Un segnale acustico è prodotto dalla variazione di pressione in un mezzo elastico. Ogniqualvolta ci troviamo di fronte ad una grandezza variabile, possiamo osservare un segnale. Si fa notare che non tutte le grandezze variabili trasportano informazioni *utili*. Il rumore è un esempio di questo tipo.

Possiamo distinguere i segnali per categorie:

CLASSIFICAZIONE SECONDO IL DOMINIO DELLA FUNZIONE

- **Segnali a tempo continuo:** il dominio[1] della funzione ha la cardinalità dell'insieme dei numeri reali (ovvero la variabile indipendente t può assumere con continuità tutti i valori compresi entro un certo intervallo, eventualmente illimitato).
- **Segnali a tempo discreto:** il dominio della funzione ha la cardinalità dell'insieme (discreto) dei numeri interi (ovvero la variabile indipendente n può assumere solo valori discreti compresi entro un certo intervallo, eventualmente illimitato).

CLASSIFICAZIONE SECONDO IL CODOMINIO DELLA FUNZIONE

- **Segnali ad ampiezza continua:** possono assumere con continuità tutti i valori di un intervallo, eventualmente illimitato.
- **Segnali ad ampiezza discreta:** possono assumere solo un insieme discreto di valori in un intervallo, eventualmente illimitato.

Le tipologie di segnali sono riassunte in Tabella 1.

*Estratto da "Sintesi del Suono - Appunti e approfondimenti per i corsi di Music Technology" - Volume 1 - 1st Edition. Rev. 1. e altri appunti

[1]Data una funzione matematica $f : X \to Y$, l'insieme X è il dominio di f; l'insieme Y è il codominio di f. In altre parole, definita la funzione $f(x)$, il suo argomento x appartiene al dominio (input), mentre il suo valore $f(x)$ appartiene al codominio (output).

	TEMPO CONTINUO	TEMPO DISCRETO
AMPIEZZA CONTINUA	SEGNALI ANALOGICI	SEQUENZE
AMPIEZZA DISCRETA	SEGNALI QUANTIZZATI	SEGNALI DIGITALI

Table 1: Tipi di segnali, tabella riassuntiva. [3].

In definitiva,

SEGNALI ANALOGICI $\longleftrightarrow$ Tempo continuo e ampiezza continua
SEGNALI DIGITALI $\longleftrightarrow$ Tempo discreto e ampiezza discreta

2 Conversione A/D - Teorema del Campionamento e Quantizzazione

Nell'era dell'audio digitale è bene dare qualche cenno sulla catena di conversione da segnale analogico a segnale digitale. Questo argomento è importante sia nel campo della sintesi del suono (in particolare nella Sintesi per Campionamento) sia nel campo più generale della registrazione digitale.

Figure 1: Percorso del segnale A/D $\rightarrow$ D/A

La **REGISTRAZIONE ANALOGICA** prevede che il segnale venga immagazzinato e riprodotto in modo continuo sia nel tempo che in ampiezza, ovvero il segnale può assumere tutti i possibili valori in ampiezza (non inoltriamoci nel discorso dei limiti di dinamica dei dispositivi elettronici) e questo avviene in modo continuo nel tempo.

La **REGISTRAZIONE DIGITALE** invece opera una discretizzazione sia sull'asse dei tempi che sulle ampiezze istantanee assunte dal segnale. La conversione da analogico a digitale avviene in più passi. La Fig. 2 mostra gli stadi fondamentali della conversione. In figura $v(t)$ è il segnale analogico in ingresso, $v[k]$ è la sequenza (segnale tempo discreto ad ampiezza continua) in uscita al campionatore (SAMPLER) e $b[k]$ è la sequenza di bit in uscita (segnale digitale).

Dal segnale analogico in ingresso vengono prelevati dei campioni (il segnale viene campionato) in istanti di tempo fissati, determinati dalla **FREQUENZA**

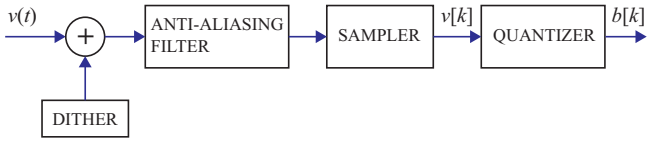

Figure 2: Schema a blocchi semplificato del convertitore A/D.

DI CAMPIONAMENTO f_c, definita come numero di campioni prelevati al secondo (si ricordi che la frequenza di campionamento è l'inverso del periodo di campionamento T_c).

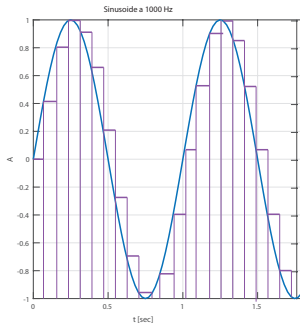

Figure 3: Campionamento, sample & hold

Il campionatore è costituito da un SAMPLE & HOLD attivato da un *trigger* che lavora alla frequenza di campionamento desiderata. Il trigger attiva il campionamento e il valore del segnale campionato viene "trattenuto" (*hold*) per un tempo sufficiente affinché la conversione possa essere eseguita.

La frequenza di campionamento non può essere qualsiasi, ma deve rispettare il **Teorema del Campionamento** (o di *Nyquist*).

TEOREMA DEL CAMPIONAMENTO

Per segnali a banda rigorosamente limitata, la frequenza di campionamento deve essere maggiore del doppio della massima frequenza che può assumere il segnale da campionare, affinché questo possa essere ricostruito correttamente a partire dai suoi campioni.

Ovvero:

$$f_c > 2f_{MAX} \tag{1}$$

Se il teorema è rispettato si può evitare il fenomeno chiamato **ALIASING**, ovvero la generazione di segnali spurii in un insieme di frequenze ottenuto ribaltando rispetto alla frequenza di Nyquist ($f_c/2$) la banda eccedente cioè:

$$f_{ALIAS} = \frac{f_c}{2} - \left(f_{MAX} - \frac{f_c}{2} \right) \tag{2}$$

Per non produrre aliasing il teorema del campionamento richiede che il segnale in ingresso sia limitato in banda. Il FILTRO ANTI-ALIASING ha proprio

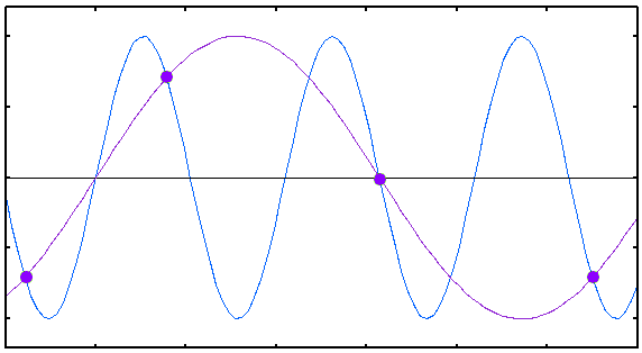

Figure 4: Segnale viola = aliasing.

questo scopo. A maggior ragione dato che f_{MAX} potrebbe essere superiore ai 20 kHz (limite superiore della banda udibile dall'essere umano), se non filtriamo, cioè se non limitiamo in banda il segnale in ingresso al campionatore, quasi sicuramente produrremo aliasing.

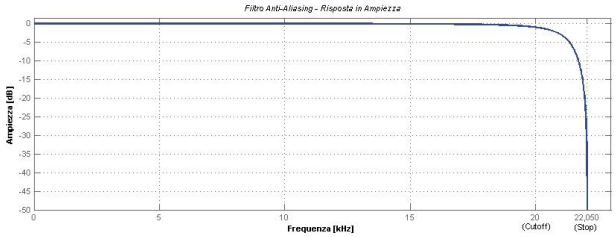

Figure 5: Risposta in ampiezza di un filtri anti-aliasing

Dimostreremo di seguito che la disuguaglianza stretta in (1) è necessaria. Si consideri la seguente famiglia di sinusoidi generate per diversi valori di θ:

$$x(t) = \frac{\cos(2\pi Bt + \theta)}{\cos(\theta)} = \cos(2\pi Bt) - \sin(2\pi Bt)\tan(\theta) \tag{3}$$

con $-\pi/2 < \theta < +\pi/2$. Campioniamo il segnale $x(t)$ negli instanti nT_s avendo scelto proprio $f_s = \frac{1}{T_s} = 2B$:

$$x(nT_s) = \cos(\pi n) - \sin(\pi n)\tan(\theta) = (-1)^n. \tag{4}$$

Si osserva che $\sin(\pi n) = 0, \forall n = 0, 1, 2 \ldots$, e di conseguenza $x(nT_s)$ può assumere solo i valori ± 1, venendosi a creare quindi un'ambiguità.

Le *frequenze di campionamento tipiche* che si possono trovare nei sistemi per processamento del segnale audio sono 44100 Hz, 88200 Hz, 48000 Hz, 96000 Hz, 192000 Hz. Da cosa derivano questi numeri?

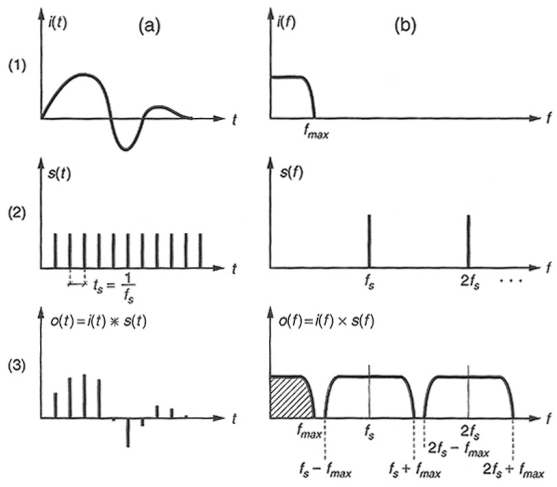

Figure 6: Campionamento nel dominio del tempo (a) e della frequenza (b). (1) segnale analogico da campionare (2) Treno di impulsi alla frequenza di campionamento (3) Segnale campionato [4].

$f = 20000$ Hz si chiama FREQUENZA DI TAGLIO
$f = 22050$ Hz si chiama FREQUENZA DI STOP.

Es. 44100
Abbiamo detto che il range di frequenze udibili dall'uomo arriva circa a 20000 Hz, dopo di che è molto probabile che lo spettro non si esaurisca a quella frequenza. Occorre quindi stabilire una f_{MAX} nota la quale possiamo poi scegliere la minima frequenza di campionamento per non produrre aliasing. I filtri in realtà non sono mai ripidi e squadrati (filtri ideali non realizzabili fisicamente) per cui con una frequenza di taglio a 20000 Hz non riusciremmo a limitare la banda del segnale proprio a quella frequenza, ma il filtro sarà più morbido. Supponendo di riuscire ad ottenere una frequenza massima a 22050 Hz con un filtro che comincia a tagliare a 20000 Hz, possiamo allora prendere una frequenza di campionamento di 44100 Hz (il doppio di 22050 Hz).

Il motivo per cui ricorrono sempre le stesse frequenze nei campionatori, nei software, etc, dipende dai componenti in commercio nel mondo dell'elettronica. Si tratta quindi di un problema di tecnologia in uso e di standard industriali. È meglio scegliere una frequenza di campionamento più alta o più bassa? Visto con gli occhi dell'ingegnere progettista, il campionamento produce in frequenza un numero di repliche infinite del segnale campionato. Nel momento in cui si vuole riscostruire il segnale di partenza, si deve isolare solo la replica in banda base (centrata attorno alla frequenza zero). Aumentando la frequenza di campi-

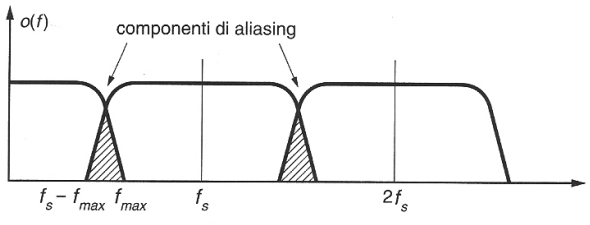

Figure 7: Presenza di aliasing nello spettro del segnale campionato [4].

onamento le repliche si distanziano tra loro maggiormente e diventa quindi più semplice realizzare il filtro che isolerà la replica da estrarre (semplice da realizzare = il filtro sarà meno ripido). Il problema interessa solo in parte all'utente finale, il quale potrà avere maggior sicurezza del buon esito del filtraggio nel momento in cui decida di lavorare con una frequenza di campionamento maggiore di 44100 Hz. Una conseguenza immediata del fatto che le righe spettrali si allontanano riguarda il rumore. Questo si distribuisce attorno alle righe dello spettro del segnale e siccome aumentando la frequenza di campionamento diminuisce la densità spettrale media del rumore, dato che questo si deve distribuire su tutti i valori nello spettro, esso scende di livello (la somma, ovvero la potenza totale del rumore, deve rimanere costante).

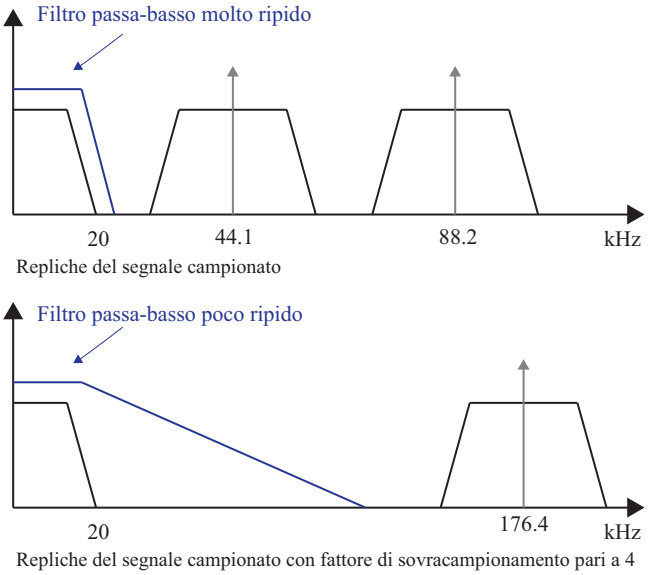

Figure 8: Filtro di ricostruzione del segnale.

Sovracampionare ha importanza soprattutto se si lavora con strumenti o plu-

gin che per il loro comportamento aggiungono armoniche (per qualsiasi motivo), ma non prevedono di filtrare la banda a 20 kHz. Questo fatto, per volontà o errore del progettista hardware o software, può re-introdurre aliasing durante le fasi di editing, di mix o di mastering di un brano. Detto questo, in uscita al campionatore il segnale sarà stato discretizzato nel tempo ma ancora può assumere tutti i possibili valori in ampiezza.

Il passo successivo è quindi quello di associare al numero infinito di valori assunti dal segnale elettrico un numero finito di valori di ampiezze. È questo il compito della QUANTIZZAZIONE, che associa ad ogni valore analogico (determinazione di una variabile aleatoria X con densità di probabilità $p_x(x)$ nota e supposta simmetrica attorno all'origine) un valore quantizzato q_i pari al valore che più si avvicina a x, scelto tra un insieme discreto di valori $Q = \{q_0, q_1, ...q_{M-1}\}$.

Si rappresenteranno allora, a seguito di questa operazione, i campioni prelevati dal segnale come *numeri interi binari* con un numero finito di *bit*. Maggiore è il numero di bit utilizzati, maggiore sarà la risoluzione con cui viene convertito il segnale. Questo numero di bit è quello che spesso è indicato come BIT DEPTH, ovvero numero di bit per campione. In un quantizzatore, se utilizziamo ad esempio 24 bit, possiamo discretizzare 2^{24} livelli di ampiezza diversi.

Nella Fig. 9 è rappresentata la caratteristica ingresso-uscita del quantizzatore uniforme. Si definisce ERRORE DI QUANTIZZAZIONE la differenza tra l'uscita quantizzata q e l'ingresso x:

$$e = q_i - x. \tag{5}$$

Indichiamo con Δ il PASSO DI QUANTIZZAZIONE, ovvero l'intervallo tra due livelli di quantizzazione:

$$\Delta = q_i - q_{i-1}. \tag{6}$$

e si supponga che esso sia costante (QUANTIZZATORE UNIFORME), si ha quindi che $-\frac{\Delta}{2} \leqslant e \leqslant \frac{\Delta}{2}$.
Calcoliamo dunque il RAPPORTO SEGNALE-RUMORE (SNR) di quantizzazione [1].

Si definisce $SNR_Q = \frac{P_s}{P_e}$ dove P_s è la potenza del segnale utile e P_e è la potenza del rumore (errore) di quantizzazione. Valutiamo dunque la potenza dell'errore di quantizzazione. Si nota che questa coincide con la sua varianza[2] σ_e^2 (essendo il rumore un fenomeno aleatorio). Il rapporto segnale rumore di quantizzazione risulta quindi:

$$SNR_Q = \frac{\sigma_x^2}{\sigma_e^2} \tag{7}$$

Si suppone che il rumore di quantizzazione si distribuisca in modo uniforme,

[2](simboli: VARIANZA = σ^2 , DEVIAZIONE STANDARD = σ. La varianza è la misura di quanto si discostano i valori della variabile aleatoria X dal valor medio.)

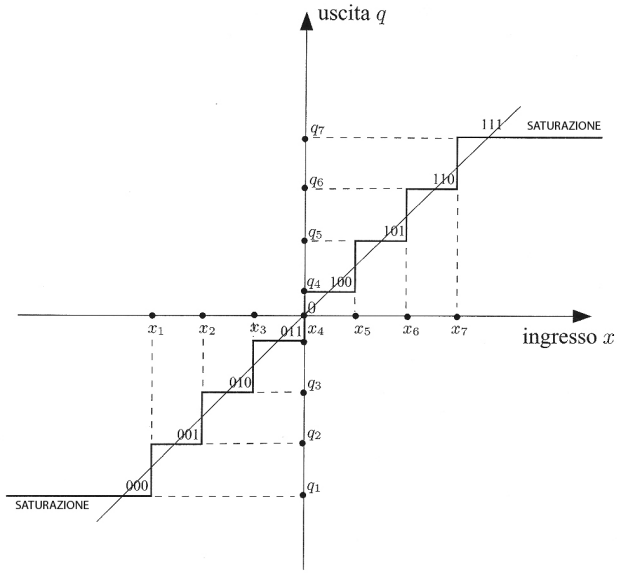

Figure 9: Caratteristica ingresso-uscita del quantizzatore uniforme [1].

ovvero la sua densità di probabilità è pari a:

$$p_e(x) = \begin{cases} \dfrac{1}{\Delta} \text{ per } |x| \leqslant \dfrac{\Delta}{2} \\ 0 \text{ altrove} \end{cases} \tag{8}$$

data la quale posso ricavare immediatamente la varianza del rumore di quantizzazione:

$$\sigma_e^2 = \frac{1}{\Delta} \int_{-\Delta/2}^{+\Delta/2} x^2 dx = \frac{\Delta^2}{12} \tag{9}$$

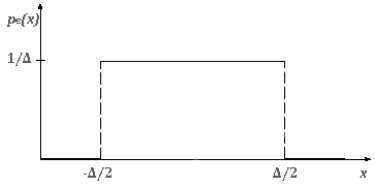

Figure 10: Distribuzione uniforme.

(la x^2 sotto il segno di integrale significa che la varianza è legata alla potenza, la quale è proporzionale al quadrato del segnale). Supponendo $-X_{MAX} \leqslant x \leqslant X_{MAX}$, rappresentando ciascun livello di quantizzazione con un numero di bit pari a B, ovvero si hanno un NUMERO DI LIVELLI pari a $M = 2^B$, si ha:

$$\Delta = \frac{2X_{MAX}}{2^B} \tag{10}$$

che sostituendo in (9) fornisce[3]:

$$\sigma_e^2 = \frac{\left(\frac{2X_{MAX}}{2^B}\right)^2}{12} = \frac{X_{MAX}^2}{3 \cdot 2^{2B}} \rightarrow SNR_Q = \frac{3 \cdot 2^{2B}\sigma_x^2}{X_{MAX}^2} \tag{11}$$

Supponiamo ora che il segnale in ingresso sia una sinusoide di ampiezza A e potenza $P_x = \frac{A^2}{2}$. Detto R il LIVELLO DI SATURAZIONE DEL QUANTIZZATORE, abbiamo:

$$\sigma_e^2 = \left(\frac{2R}{2^B}\right)\frac{1}{12} = \frac{R^2}{2^{2B}}\frac{1}{3} \tag{12}$$

$$SNR_Q = \frac{\frac{A^2}{2}}{\frac{R^2}{2^{2B}}\frac{1}{3}} = \frac{3}{2}\frac{A^2}{R^2}2^{2B} \tag{13}$$

$$SNR_{Q||dB} = 1,76 + 10\log_{10}\frac{A^2}{R^2} + 6,02B \tag{14}$$

La Fig. 11 mostra l'andamento di $SNR_{Q|dB}$ in funzione del rapporto $\left[\frac{A^2}{R^2}\right]_{dB}$. Si ha un andamento lineare per $A < R$ e il rapporto segnale-rumore è massimo per $A = R$ e decresce rapidamente quando $A > R$.

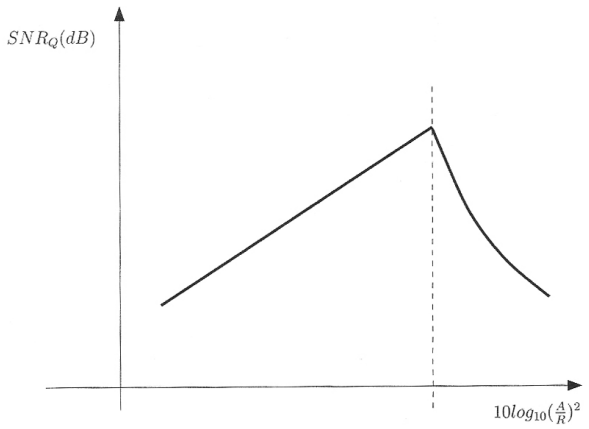

Figure 11: Andamento dell'SNR di quantizzazione [1]

Nota: una pratica molto comune è quella di utilizzare per il rapporto segnale-rumore la seguente rappresentazione conservativa:

$$SNR_Q \approx 2^{2B} \tag{15}$$

[3]la σ_x^2 rappresenta come detto la varianza di un segnale aleatorio; nel momento in cui il segnale fosse certo, allora andrebbe sostituita con la potenza P_x

$$SNR_{Q|dB} \approx 6B. \tag{16}$$

Inoltre ci sarebbe da dire che il segnale vocale, non essendo una sinusoide pura, necessita di una rappresentazione in termini di densità di probabilità più precisa. Si è sperimentato che la distribuzione di Laplace approssima in modo soddisfacente il segnale vocale:

$$p_X(x) = \frac{1}{\sqrt{2}\sigma_x} e^{-\frac{\sqrt{2}|x|}{\sigma_x}} \tag{17}$$

Non ci soffermiamo su ulteriori accorgimenti e miglioramenti progettuali.

Attenzione: il numero di bit e la frequenza di campionamento vanno pesati adeguatamente in quanto influiscono direttamente sulla quantità di dati memorizzati sull'hard disk, ovvero da questi dipendono le dimensioni dei file audio. Nota bene che 1 minuto di musica e 1 minuto di completo silenzio pesano allo stesso modo!!

Concludiamo il discorso parlando brevemente del DITHER. Il dither è rumore distribuito opportunamente aggiunto al segnale che si vuole quantizzare (o ri-quantizzare quando si diminuisce la risoluzione in bit) con lo scopo di diminuire l'errore di quantizzazione. Ciò che avviene operando una quantizzazione su un segnale che stiamo registrando o diminuendo la risoluzione in bit di un segnale già acquisito è mostrato chiaramente in Fig. 12.

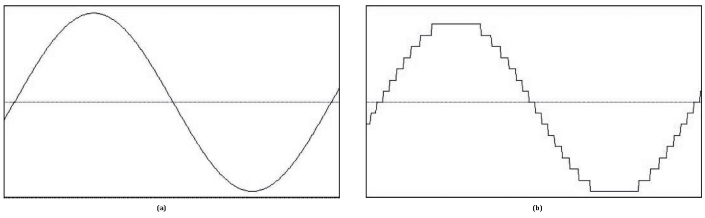

Figure 12: In (b) si osserva l'effetto della diminuzione della risoluzione in bit - in verticale - rispetto alla figura (a).

Dalla Fig. 12 si vede che ciascun gradino produrrà un artefatto udibile, simile alla presenza delle armoniche di un'onda quadra.

Ipotizziamo di avere un campione di ampiezza compresa tra due livelli di quantizzazione. A quale livello verrà assegnato il campione? Se fosse proprio a metà strada avremmo il 50% di probabilità di assegnare il campione ad uno o all'altro livello. Se ci trovassimo più vicini al livello più alto e operassimo un troncamento, questo porterebbe ad errore, perché il campione verrebbe assegnato al livello inferiore (cioè se ad esempio avessimo un campione di ampiezza 1.8, vorrei assegnarlo a 2 piuttosto che a 1). Se aggiungo al segnale del rumore di ampiezza $< \frac{\Delta}{2}$ permetterei al campione di superare la soglia di quantizzazione corretta e quindi troncando potrei ottenere il valore corretto.

Si possono utilizzare diversi tipi di dither che differiscono per la loro densità di probabilità: rettangolare, triangolare, gaussiana, dither colorato (spesso si

modella il dither in modo che esso abbia maggiore energia alle frequenze più alte, essendo questa la parte di spettro audio meno udibile dal sistema uditivo umano). La tipologia di dither più usata in applicazioni audio è quella *triangolare*.

3 Trasduzione del segnale acustico

Figure 13: Trasduzione e conversione del segnale acustico.

Abbiamo detto che alla sorgente il segnale acustico è un segnale di pressione acustica, ovvero è generato dalla variazione della pressione in un mezzo. Come possiamo portare il segnale acustico in una forma che può essere elaborata elettronicamente? Il segnale di pressione acustica deve essere convertito in un segnale elettrico attraverso un *trasduttore*. L'operazione di TRASDUZIONE è la trasformazione di una forma di energia in un'altra. In questo caso, il trasduttore di cui abbiamo bisogno è il *microfono*. Esistono diversi tipi di microfoni. I microfoni possono essere classificati secondo il loro principio di funzionamento, secondo il loro specifico utilizzo, secondo il loro pattern polare, secondo le loro caratteristiche elettriche di uscita (es. impedenza di uscita) etc. In questo momento, però, non ci interessa entrare nel dettaglio.

In uscita al sistema di Fig. 13 vogliamo effettuare l'operazione inversa ovvero convertire un segnale elettrico in segnale acustico. Anche in questo caso ci servirà un trasduttore e si tratta di un altoparlante. Come è stato detto per i microfoni anche gli altoparlanti esistono di vari tipi.

4 Formati dei file audio

Una volta che il suono è stato convertito da analogico a digitale, i dati audio grezzi vengono "impacchettati" in uno specifico formato. Un formato audio contiene quindi DATI di AUDIO GREZZO e ALTRI dati che definiscono e descrivono il particolare formato. I dati di audio digitale vengono CODIFICATI/DECODIFICATI da un CODEC[4] a seconda dello specifico formato di uscita. Un formato può suppotare uno o più CODEC.

Si possono distinguere i formati audio in due categorie [2]:

- FORMATI LINEARI
- FORMATI COMPRESSI (LOSSY, LOSSLESS)

[4]Un CODEC (Coder-Decoder) è un programma per computer oppure integrato in hardware per codificare e codificare dati grezzi audio (o video) in un particolare formato audio.

4.1 FORMATI LINEARI

I formati audio lineari sono *non* compressi, tipicamente codificati in PCM lineare. Il PCM (*Pulse Code Modulation*) è la rappresentazione digitale di un segnale analogico dove l'ampiezza del segnale è campionata ad intervalli regolari e quantizzata in un serie di simboli in codice numerico (solitamente binario). Il Linear PCM (LPCM) codifica i valori in ampiezza in modo proporzionale all'ampiezza del segnale in ingresso piuttosto che logaritmico o secondo un'altra relazione. In altre parole nel LPCM i livelli di quantizzazioni sono linearmente uniformi.

I file nei formati lineari possono essere caratterizzati dal numero di canali audio che essi trasportano. Si possono trovare file:

- MONO (1 canale)
- STEREO SPLIT (2 file mono per i canali LEFT e RIGHT)
- STEREO INTERLEAVED (un file stereo, 2 canali)
- (SURROUND) MULTICHANNEL SPLIT (tanti file quanti sono i canali trasportati)
- MULTICHANNEL INTERLEAVED (un unico file multicanale)

Caratteristiche principali dei formati lineari
Le caratteristiche principali dei formati audio lineari sono:

- FREQUENZA DI CAMPIONAMENTO: numero di campioni prelevati in un secondo (espressa formalmente in [Hz]).
- BIT DEPTH: numero di bit utilizzati per sample (risoluzione), ovvero numero di bit per rappresentare un campione (espressa formalmente in [bits]).

È importante valutare sempre quanto spazio sull'hard disk occuperà un file audio che stiamo registrando. Possiamo quindi fare alcuni calcoli: quanti MB contiene 1 minuto di audio, su CD? Le caratteristiche del CD audio sono $f_c = 44100$ [Hz], 16 [bits], stereo (2 canali).

$$\begin{aligned} 44100 \left[\frac{\text{samples}}{\text{sec}}\right] \times 16 \left[\frac{\text{bits}}{\text{sample}}\right] \times 60\,[\text{secs}] \times 2 &= 84672000\,[\text{bits}] \\ \rightarrow \frac{84672000}{8} &= 10584000\,[\text{Bytes}] \\ &\simeq 10\,[\text{MB}] \end{aligned}$$

Attenzione quindi a quanto spazio si ha a disposizione sull'hard disk.

Nel campo dell'*audio entertainment* si considerano tipicamente risoluzioni di $16 - 24$ bit e frequenze di campionamento maggiori di 44100 Hz in relazione ad applicazioni professionali. In realtà questa classificazione non ha senso pratico,

in quanto i parametri si impostano a seconda delle esigenze. Possiamo però dire che l'audio che si ascolta nei dischi o alla radio/TV è stata proabilmente registrata con non meno di 16 bit e 44100 Hz.

I formati audio non compressi sono tutti abbastanza simili tra loro e sono *flessibili*, nel senso che permettono più o meno tutte le combinazioni di sample rate e bit depth. Altro vantaggio nell'utilizzo di questi formati risiede nel fatto che *senza compressione* non si aggiunge ulteriore latenza all'intero processo di codifica, memorizzazione e riproduzione.

I formati audio lineari non compressi più conosciuti sono:

- WAVE (.wav): formato standard Microsoft/IBM. Basato sul formato RIFF che immagazzina i dati in *chunks* (pezzettini), quindi molto simile all'AIFF (Audio Interchange File Format) di Apple, basato sullo stesso metodo di immagazzinamento. Entrambi i formati hanno la codifica dei dati secondo PCM con una differenza: AIFF è *big endian*, mentre WAVE è *little endian*. Entrambi. Entrambi WAVE e AIFF supportano anche la compressione (AIFF-C, Windows ACM).

- SD2 (Sound Designer II): altro formato professionale, sviluppato negli anni '60 per Mac (Digidesign), usato da Pro Tools e Digital Performer. Identico a Wave/Aiff. Contiene anche informazioni relative al timecode.

- Acid Loops: si presentano come file wave. Sono stati creati per essere usati inizialmente con Acid (Sonicfoundry 1998). Contengono informazioni relative al metro, bpm, numero di battute con cui sono stati registrati. Vengono importati in un progetto per essere *stretchati* a seconda del bpm del progetto agganciandosi alla griglia della digital audio workstation su cui si sta lavorando.

La comodità di avere i chunks è che alcuni software che non sanno interpretare certi tipi di chunks semplicemente li ignorano e procedono ai chunks successivi. Nel chunk HEADER del Wave standard troviamo 32 bit che stabiliscono quale sia la dimensiona massima di un file wave: 2^{32}. Il risultato è circa 4 GB, che non è poco. Tuttavia si è pensato di abbattere questo limite ed è stato quindi introdotto il Wave 64 (.w64) (Sonicfoundry Sony). Il Wave 64 inoltre contiene dei metadata, come un altro derivato del Wave, ovvero il Broadcast Wave File, relativi all'autore, al titolo, al timecode/battute, altri dati.

Si fa notare che non si può creare un CD Audio direttamente da file Wave. I file Wave sono spogliati dell'header e sul CD vengono memorizzati solo i suoi dati audio grezzi.

4.2 FORMATI COMPRESSI

Possiamo distinguere i formati compressi in due categorie:

- FORMATI LOSSY (con perdita)

- FORMATI LOSSLESS (senza perdita)

La compressione audio è una forma di compressione dei dati che serve a ridurre la dimensione di un file audio per diversi scopi (distribuzione su internet, poco spazio a disposizione, etc.). La compressione avviene per mezzo di un CODEC.

In breve, una compressione LOSSY riduce i dati in modo che questi non si riescano a recuperare, mentre la compressione LOSSLESS invece è reversibile e comprime i dati in modo che possano essere recuperati. In entrambi i casi ciò che si cerca di fare è ridurre le informazioni ridondanti usando vari metodi di codifica, predizione lineare e altre tecniche. Si deve cercare un compromesso tra dimensioni del file e qualità dell'audio.

I formati compressi sono caratterizzati dal loro BITRATE, ovvero il numero di bit che sono immagazzinati, trasmessi o processati al secondo. È proprio la scelta del bitrate che maggiormente influisce sulla dimensione del file compresso:

$$\text{BITRATE} = f_c \times n_{bits} \times N_{CANALI} \tag{18}$$

Ad esempio, nel CD audio stereo si ha: $44100 \times 16 = 1411 \left[\frac{kbits}{sec}\right]$

Scegliendo nella compressione l'opzione CBR (Constant Bit Rate), l'algoritmo codificatore comprimerà mantenendo costante il bitrate al valore di quello selezionato, indipendentemente dal contenuto del file. Questo permette di predire il peso del file compresso. In altenativa si può optare per il VBR (Variable Bit Rate), in cui non si può predire la dimensione del file compresso con certezza. Usando il VBR tipicamente si può ottenere un file di qualità superiore rispetto al CBR, in quanto questo metodo utilizza algoritmi più sofisticati che *risparmiano* bit dove lo spettro del segnale è meno complesso e impiegano più bit nelle zone più ricche.

4.2.1 Formati Lossy

In psicoacustica si è visto che le frequenze dello spettro audio ($20-20000$ Hz) non vengono percepite tutte dall'uomo con la stessa sensibilità. Gli studi hanno rivelato che l'apparato uditivo umano è meno sensibile alle frequenze estreme dello spettro udibile. Le curve isofoniche di Fletcher-Munson (Fig. 14) descrivono proprio questo comportamento.
La compressione lossy solitamente riduce i dati relativi alle alte frequenze condificandole con meno accuratezza e o non includendole affatto. Anche i suoni che vengono mascherati da suoni più forti vengono "scartati". Ridurre il numero di bit utilizzati nella codifica introduce rumore e quindi si preferisce operare dove l'orecchio è meno sensibile.

- MP3: Bitrate da $32 \left[\frac{kbits}{sec}\right]$ a $320 \left[\frac{kbits}{sec}\right]$. È ancora ad oggi il formato compresso più conosciuto, ormai tra i peggiori in termini di qualità del suono. Il nome si riferisce all'MPEG-1 Audio Layer III dell'inizio degli anni '90. La qualità del CODEC può influire notevolmente sulla qualità finale del file risultante. CODEC per mp3: Fraunhofer, Lame. Questi CODEC offrono diverse combinazioni di bitrate, sample rate, bit depth, inoltre hanno la modalità MS Stereo (Somma e differenza dei 2 canali),

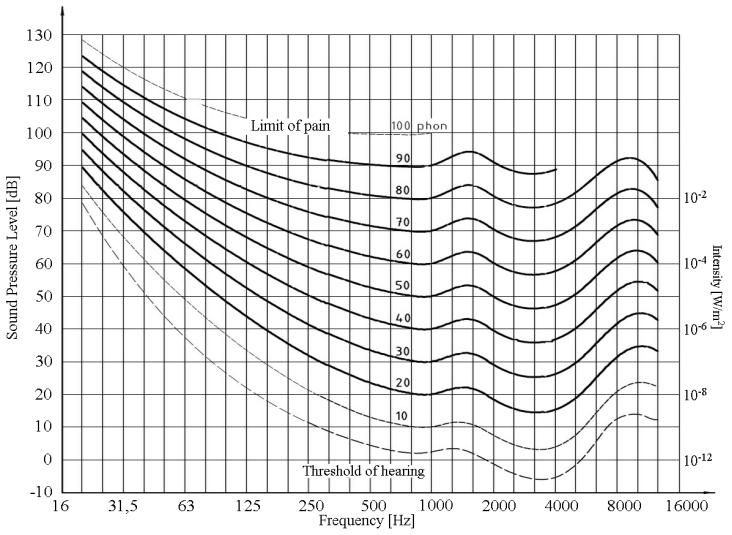

Figure 14: Curve di Fletcher-Munson

Intensity Stereo (mix in mono). L'mp3 può includere dei *tag* in formato ID3 contenenti nome dell'artista, titolo del brano, album e altro.

- REAL AUDIO (.ra, .rm, .rmvb): Sviluppato da Real Networks, usa molti CODEC. Si utilizza soprattutto con le internet radio come formato *streaming*[5]. Accanto ai file real audio si trovano dei file di testo (.ram) contenenti dei link ai file real audio.

- OGG: Formato Open Source che supporta molti CODEC, il più famoso è Vorbis. In realtà OGG è un *contenitore* per Vorbis, Theora (video), FLAC, Speex e altri. È possibile aggiungere dei tag in formato Vorbis. Il bitrate va da 96 $\left[\frac{kbits}{sec}\right]$ a 350 $\left[\frac{kbits}{sec}\right]$ (average, medio). A parità di dimensioni a confronto con un MP3 la qualità sonora dell'OGG è migliore e il taglio delle alte frequenze è meno drastico.

- WMA (Windows Medi Audio): Formato della Microsoft. Bitrate da 32 $\left[\frac{kbits}{sec}\right]$ a 320 $\left[\frac{kbits}{sec}\right]$. Il WMA esiste anche in versione lossless. Esistono 4 CODEC: WMA (classico), WMAPro (supporta multichannel e audio ad alta risoluzione), WMA Lossless CD (formato lossless), WMA Voice (mono, $f_c \leq 22050$ Hz, CBR fino a 20 $\left[\frac{kbits}{sec}\right]$, usato per segnali vocali)

- AAC: L'AAC è stato standardizzato dall'ISO come parte delle specifiche MPEG-2 e 4. Utilizzato in Itunes, sugli iPad, i Phones, Playstation 3 e

[5] In telecomunicazioni si intende per streaming un flusso di dati (es. audio/video) trasmesso a uno o più destinatari e riprodotto dal destinatario man mano che i dati arrivano a destinazione e vengono dunque scaricati.

Portable, Nokia, Nintendo Wii. Designato come il successore dell'MP3 e di qualità superiore rispetto a questo. Supporta fino a 48 canali, sample rates da 8 kHz a 96 kHz, bitrates arbitrari e lunghezza di frame variabile, joint stereo. Esiste sia in versione coperta da copyright (.mp4, Itunes Store) che non protetta da copyright (.m4a).

4.2.2 Formati Lossless

Oltre ai classici algoritmi di riduzione delle ridondanze, riconoscimento dei pattern e delle ripetizioni, i CODEC lossless applicano un *filtro sbiancante* che scorrela e appiattisce lo spettro. Il decoder ripristina il segnale originale con un'operazione inversa. Come si può capire, la compressione lossless è molto meno pesante rispetto a quella lossy e il rapporto di compressione è quindi molto più basso rispetto a quello lossy[6].

- FLAC (Free Lossless Audio Codec): Formato Open Source. Comprime i file audio originali fino al 50%. Supporta i tag.
- ALAC (Apple Lossless Audio Codec): i dati sono memorizzati in un contenitore m4a. Compressione dal 40% fino al 60%.
- WMA Lossless: vedi sopra WMA. Supporta anche il surround.

References

[1] M.G. Di Benedetto. *Comunicazioni Elettriche - Fondamenti*. Pearson Prentice Hall.

[2] P. Guaccero, L. Proietti, and L. Zaccheo. *Lezioni di Music Technology*. Appunti presi a lezione.

[3] M. Luise and G. M. Vitetta. *Teoria dei Segnali*. McGraw-Hill.

[4] J. Maes and M. Vercammen. *Digital Audio Technology*. Taylor & Francis Ltd Focal Press.

[6] Il rapporto di compressione si può definire come $RC = \frac{\text{size(File}_{\text{OUTPUT}})}{\text{size(File}_{\text{INPUT}})}$.