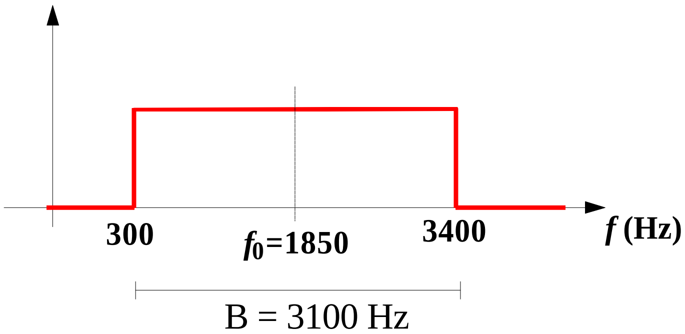
Appunti di
Elaborazione numerica dei segnali
Capitolo 10 (parte I) - **Segnale telefonico**

Codifica PCM

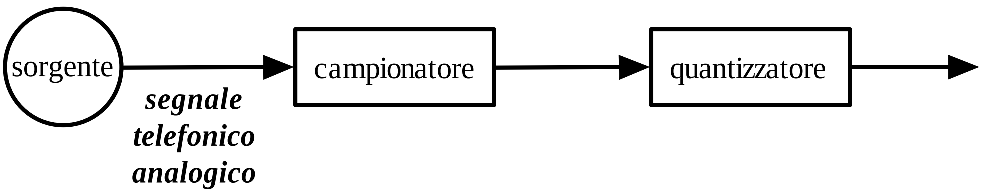
INTRODUZIONE: LO STANDARD PCM

Sappiamo bene che il **segnale telefonico** è una versione filtrata del **segnale vocale**; in particolare, esso è un segnale con componenti di frequenza comprese da 300 Hz e 3400 Hz e questa banda è tale da conservare l’intelligibilità del messaggio ed il riconoscimento del parlatore:



Intervallo di frequenze su cui è definito il segnale telefonico (analogico): le componenti di frequenza sono comprese da 300 Hz e 3400 Hz, per cui la frequenza centrale è a 1850 Hz e l’ampiezza di banda (netta) è di 3100 Hz. A fronte di questa ampiezza di banda netta, si considera sempre una banda lorda di 4 kHz

Ci concentriamo, in questo capitolo, sulla trasmissione numerica del segnale telefonico, la quale, come è noto, avviene secondo uno schema generale del tipo seguente:



Schema generico di un trasmettitore numerico per il segnale telefonico: la sorgente può intendersi come tutto ciò che serve a generare il segnale vocale di qualità telefonica, cioè un segnale con componenti spettrali che vanno da 300 Hz a 340 Hz; il successivo gruppo campionatore®quantizzatore converte la forma d’onda analogica in sequenza di bit. Tale sequenza di bit, prima di essere trasmessa, andrà verosimilmente in un codificatore di canale che aumenterà il numero di bit introducendo i codici di controllo dell’errore (CRC). C’è anche da considerare l’inevitabile filtro anti-aliasing da porre a monte del campionamento, allo scopo di imporre che sia il segnale sia l’immancabile rumore in ingresso al campionatore abbiano effettivamente la banda prevista

Il primo passo, per una trasmissione numerica di un segnale prettamente analogico, è notoriamente quello del **campionamento**: assumendo una banda lorda **B=4kHz** per il segnale telefonico analogico, il campionamento viene eseguito a frequenza **f_C=8kHz**, cui corrisponde un periodo di campionamento **T=125μsec**.

Il passo successivo è quello della **quantizzazione**, che permette di associare sequenze di bit a ciascuna delle misure (campioni) ottenute dal campionatore. Come già sappiamo, la **legge di**

quantizzazione (cioè il criterio con cui scegliere l'associazione tra le suddette misure e le corrispondenti sequenze di bit) deve essere pensata tenendo conto della statistica del segnale da quantizzare (se essa è disponibile). Nel caso del segnale telefonico (in generale del segnale vocale), ciascun parlatore ha *timbro* e *livello* di voce diversi dagli altri, per cui la legge di quantizzazione deve essere tale da soddisfare tutti i parlatori.

Le possibilità sono due: quantizzazione uniforme, che significa suddividere il range dei campioni in intervalli di quantizzazione di ampiezza costante, oppure quantizzazione non uniforme, nella quale invece la suddivisione prevede intervalli di quantizzazione di ampiezza diversa.

La scelta più semplice da implementare è sicuramente la **quantizzazione uniforme**: se si fa questa scelta, per il segnale telefonico è risultato che, per garantire le prestazioni anche nel caso peggiore (quello cioè dei parlatori più deboli), servono **12 bit** per campione, che corrispondono a $\mathbf{2^{12}}$ intervalli di quantizzazione (tutti di uguale ampiezza). La corrispondente *velocità di trasmissione* (**bit rate**) è allora

$$8\cdot 10^3\left(\frac{\text{campioni}}{\text{secondo}}\right)\cdot 12\left(\frac{\text{bit}}{\text{campione}}\right)=96\left(\frac{\text{kbit}}{\text{sec}}\right)$$

Il singolo canale telefonico, quantizzato uniformemente, richiede dunque una velocità di trasmissione di **96 kbit/sec**, che è un po' troppo elevata, in termini ovviamente di banda occupata, in quanto sappiamo bene che la banda cresce proporzionalmente alla velocità di trasmissione che si intende ottenere. Non solo, ma si tratta di una quantità troppo elevata anche per un altro motivo: essa è stata stimata per soddisfare anche le utenze peggiori, cioè i parlatori più deboli, il che rappresenta una sorta di sovradimensionamento, in quanto, nella maggior parte dei casi, non è un parlatore debole che conversa, ma una parlatore normale.

Dobbiamo allora capire in quale modo possiamo ridurre questa velocità di trasmissione.

Osserviamo che, *per effettuare una trasmissione numerica di un segnale qualsiasi, sono necessarie due successive discretizzazioni:*

- la prima è quella dovuta al campionamento (discretizzazione nel tempo), ed è reversibile, a patto ovviamente che si rispetti il teorema del campionamento;
- la seconda è invece quella dovuta alla quantizzazione (discretizzazione dei valori), che non è reversibile: sappiamo infatti che la quantizzazione introduce intrinsecamente (cioè per il meccanismo stesso con cui funziona) un rumore sovrapposto al segnale (**rumore di quantizzazione**), dal quale non si può prescindere. Tutt'al più, questo rumore può essere ridotto aumentando il numero di bit per campione, mentre invece il nostro scopo è proprio l'opposto, in quanto vogliamo trovare il modo di ridurre al minimo il numero di bit per campione.

D'altra parte, queste discretizzazione ci danno problemi fino ad un certo punto: infatti, in applicazioni come la trasmissione numerica del segnale telefonico, non è essenziale ricostruire esattamente la forma d'onda di partenza[1], ma è sufficiente ricostruirla nel modo più approssimato possibile, compatibilmente con il soddisfacimento dell'utenza.

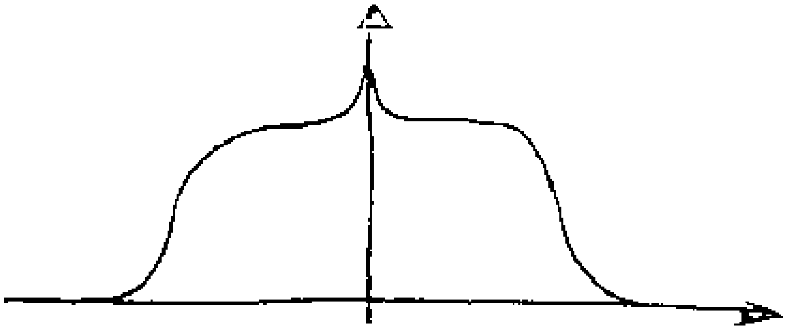
Questo discorso viene talvolta portato anche alle estreme conseguenze: si sceglie infatti di ricostruire, in ricezione, una forma d'onda completamente diversa da quella iniziale, purché però essa fornisca all'utenza la stessa sensazione che darebbe il segnale originale.

Ad ogni modo, noi restiamo nell'ambito della cosiddetta **codifica di forme d'onda**, il che significa che *siamo interessati, in ricezione, ad ottenere una*

[1] Tale forma d'onda di partenza non è altro che il segnale telefonico analogico ottenuto filtrando il segnale (sempre analogico) emesso dall'utente quando parla nella cornetta del telefono

ricostruzione della forma d’onda originale quanto migliore possibile.

La scelta della quantizzazione uniforme è l’unica possibile quanto non si conosce alcuna statistica attendibile del segnale in questione, per cui non si ha alcuno strumento per ottimizzare il sistema dal punto di vista della codifica binaria e quindi della velocità di trasmissione. Al contrario, *per il segnale telefonico sono disponibili sia statistiche della potenza del singolo parlatore sia statistiche della potenza media dei vari parlatori*[2]*.* Mettendo insieme queste statistiche, si è trovato che il segnale telefonico (inteso ovviamente come segnale aleatorio), ha una densità di probabilità delle ampiezze del tipo illustrato nella figura seguente:



Andamento schematico della densità di probabilità delle ampiezze del segnale telefonico: in ascisse sono riportati i valori delle ampiezze e in ordinate i corrispondenti valori della densità di probabilità

Come si nota, la caratteristica fondamentale di questa funzione è quella di avere un picco intorno ai valori molto piccoli (fluttuazioni molto lente): *i valori più probabili del segnale telefonico sono cioè quelli di ampiezza più piccola, mentre sono via via meno probabili i valori più grandi.*

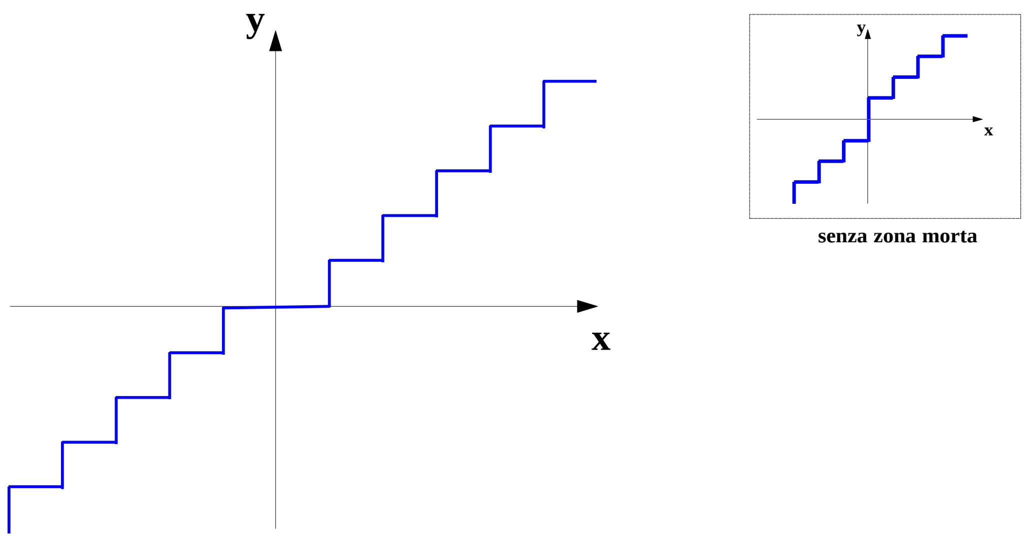
A questo punto, se è nota la statistica del segnale, è opportuno adattare ad essa la legge di quantizzazione, nella speranza di ridurre il *bit rate*. Come effettuiamo questo adattamento?

Nel caso del segnale telefonico, è sensato porsi lo scopo di minimizzare il cosiddetto **errore quadratico medio** in uscita dal quantizzatore: se indichiamo con x l’ingresso al quantizzatore e con y l’uscita, l’*errore quadratico medio* tra le due quantità è definito banalmente come il valor medio del quadrato della differenza tra uscita ed ingresso, vale a dire

$$e_m^2 = E\left[(y-x)^2\right]$$

Questa definizione è una misura della discretizzazione compiuta dal quantizzatore, in quanto dobbiamo tener conto che la legge di quantizzazione è del tipo seguente:

[2] Si può vedere, a tal proposito, il paragrafo 7.5, a pagina 69, del testo “*Comunicazioni Elettriche*” del prof. F. Carassa



Curva di quantizzazione uniforme; si notano essenzialmente due caratteristiche: la simmetria pari (non c'è motivo per cui non debba essere così) e la cosiddetta **zona morta** *(midtread), ossia l'intervallo di valori di x cui corrisponde un valore di y identicamente nullo. L'altra possibilità sarebbe una curva di quantizzazione* **senza zona morta** *(midrised), che è la duale della precedente ed è riportata nel riquadro in alto a destra*

Questo diagramma mostra il legame ingresso-uscita del quantizzatore: infatti, per ciascun valore x dell'ingresso, il quantizzatore associa un determinato valore y dell'uscita, cui corrisponde una determinata sequenza di bit; in ricezione, il ricevitore prende la sequenza di bit arrivata e vi associa il corrispondente valore numerico, commettendo perciò inevitabilmente un errore in tutti quei casi in cui $y \neq x$.

Si può ipotizzare che, all'interno del singolo intervallino sull'asse x, la densità di probabilità, condizionata al fatto che il segnale in ingresso cada in quell'intervallino, sia praticamente uniforme (questa approssimazione è tanto più valida quanto più piccoli sono gli intervallini). Sotto questa ipotesi, sappiamo che la varianza (cioè la potenza) del rumore di quantizzazione (cioè ciò che definiamo rigorosamente **errore di quantizzazione**) è pari al noto valore $\Delta^2/12$, dove Δ è l'ampiezza (costante) degli intervalli di quantizzazione. Abbiamo dunque che l'errore di quantizzazione è proporzionale a Δ^2.

A questo punto, se l'errore di quantizzazione sul generico campione, che cioè cade nel generico intervallino, è proporzionale al quadrato dell'ampiezza di tale intervallino, allora minimizzare l'errore quadratico medio (cioè l'errore pesato statisticamente tramite la probabilità che il segnale cada nei vari intervallini) significa imporre valori dell'errore maggiori per i valori del segnale che si presentano meno frequentemente e valori dell'errore minori per i valori del segnale che si presentano più frequentemente.

In altre parole, nel caso di una densità di probabilità come quella riportata prima, per minimizzare l'errore di quantizzazione basta rendere più piccolo il rumore di quantizzazione in corrispondenza di piccoli valori dell'ingresso e lasciarlo più grande per valori maggiori dell'ingresso. In termini ancora più pratici, *dovremo addensare gli intervalli di quantizzazione (rendendoli quindi via via più piccoli) in corrispondenza dei valori più probabili del segnale, mentre potremo farli più radi (e quindi più larghi) in corrispondenza dei valori meno probabili del segnale.* In questo modo, l'approssimazione con cui codifichiamo ciascun valore x dell'ingresso è migliore se x è piccolo (e quindi più probabile), mentre invece è peggiore se x è grande (e quindi meno probabile). In questo modo, mediamente otteniamo una minimizzazione dello scarto quadratico tra la sequenza ricostruita in ricezione e quella originaria di partenza.

Facciamo osservare che *questa procedura è sensata solo fino a quando è sensato usare l’errore quadratico medio come misura dell’errore.* Per un segnale telefonico abbiamo prima osservato che la procedura è valida, dato che l’orecchio è sensibile alla potenza del segnale ed alla sua distribuzione in frequenza (e quindi al valore quadratico medio, ipotizzando l’ergodicità del processo). Nel caso di un segnale televisivo, invece, quanto appena detto non è più applicabile, dato che l’occhio è una specie di *misuratore di forme d’onda*.

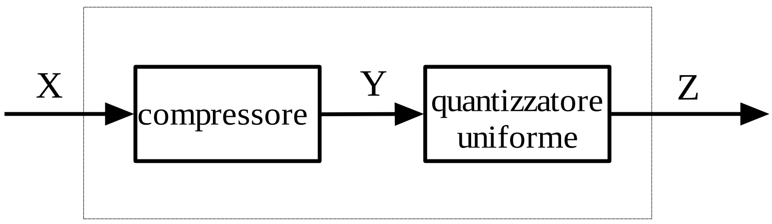
Tornando dunque alla codifica del segnale telefonico, l’utilizzo di una quantizzazione non uniforme costruita secondo i criteri prima esposti consente di ridurre il numero di bit per campione da 12 a **8**: in questo caso, la velocità di trasmissione richiesta è

$$8\cdot 10^3\left(\frac{\text{campioni}}{\text{secondo}}\right)\cdot 8\left(\frac{\text{bit}}{\text{campione}}\right)=64\left(\frac{\text{kbit}}{\text{sec}}\right)$$

Quindi, riassumendo, gli standard inclusi nella tecnica **PCM** (*Pulse Code Modulation*) prevedono una frequenza di campionamento di **8 kHz** ed una quantizzazione non uniforme ad **8 bit** per campione.

Curve di compressione

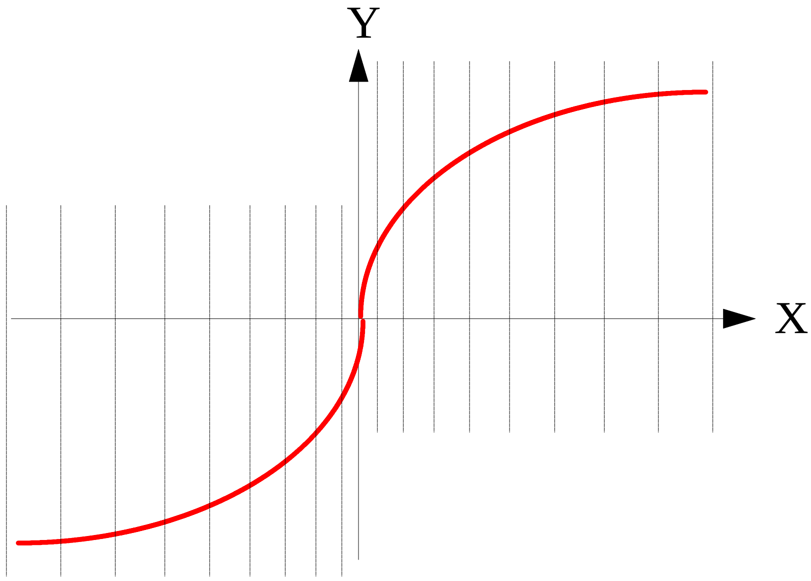
Segnaliamo che la quantizzazione non uniforme non è ottenuta direttamente tramite un quantizzatore non uniforme. Ciò che si fa, invece, è usare ancora un quantizzatore uniforme, a monte del quale, però, viene disposto un organo detto **compressore**:



La cascata dei due dispositivi realizza un quantizzatore uniforme: infatti, il compressore ha il compito di convertire la densità di probabilità del segnale X in ingresso da non uniforme, come è quella del segnale telefonico, in uniforme, in modo che poi la quantizzazione possa essere a sua volta uniforme. Vediamo di spiegarci un po' meglio.

Il segnale X che arriva in ingresso al compressore è l’insieme dei campioni prodotti dal campionatore[3]. La distribuzione dei campioni è quella descritta in precedenza: il segnale telefonico è un segnale aleatorio a media nulla (la sua densità di probabilità è simmetrica rispetto al valore 0) per il quale i valori più probabili sono quelli in prossimità dello zero, mentre quelli lontani dallo zero siano meno probabili. Di conseguenza, bisogna prevedere una legge di quantizzazione che preveda intervalli più fitti nei pressi dello 0 ed intervalli via via più larghi man mano che ci si allontana dallo zero. A questo serve proprio il **compressore**, per il quale il legame (istantaneo) tra ingresso e uscita può essere schematizzato dal grafico seguente:

[3] Si tratta, perciò, di un segnale analogico di natura impulsiva, dove i vari impulsi sono equispaziati di T ed hanno valore pari o proporzionale al valore assunto, nei corrispondenti istanti, dal segnale analogico di partenza



Legame istantaneo tra ingresso ed uscita del compressore: è la cosiddetta caratteristica (o curva) di compressione

L'effetto di questo compressore è il seguente: operando una suddivisione del range dinamico di X tale che ci sia un maggiore infittimento degli intervallini in prossimità del valore 0, *il compressore fornisce una suddivisione praticamente uniforme del segnale in uscita Y.*

Tale segnale Y risulta dunque avere una distribuzione praticamente uniforme nel proprio range, per cui esso viene successivamente quantizzato, questa volta in maniera uniforme.

La curva che lega i segnali X ed Y in ingresso ed in uscita dal compressore prende il nome di **caratteristica di compressione**; essa può essere di due tipi (cui corrispondono due distinte rappresentazioni analitiche), a seconda di quale sia lo standard utilizzato per la comunicazione telefonica:

- per il sistema telefonico numerico usato negli STATI UNITI, la legge matematica che rappresenta la caratteristica di compressione prende il nome di **μ-LAW** in base al fatto che in essa compare un parametro μ; si tratta della legge seguente:

$$Y = Y_{MAX} \frac{\ln\left[1+\mu\left(\frac{|X|}{X_{MAX}}\right)\right]}{\ln(1+\mu)} \operatorname{sgn}(X)$$

- per il sistema telefonico numerico usato invece in EUROPA, la legge matematica prende il nome di **A-LAW** (in base al fatto che in essa compare un parametro A) e si tratta della legge seguente:

$$Y = \begin{cases} Y_{MAX} \dfrac{A\left(\dfrac{|X|}{X_{MAX}}\right)}{1+\ln A} \operatorname{sgn}(X) & 0 \le \dfrac{|X|}{X_{MAX}} \le \dfrac{1}{A} \\[2ex] Y_{MAX} \dfrac{1+\ln\left[A\left(\dfrac{|X|}{X_{MAX}}\right)\right]}{1+\ln A} \operatorname{sgn}(X) & \dfrac{1}{A} \le \dfrac{|X|}{X_{MAX}} \le 1 \end{cases}$$

Per quanto riguarda nel dettaglio il funzionamento del quantizzatore uniforme, abbiamo già osservato che esso opera la quantizzazione servendosi di **8 bit** per campione, ossia quindi dividendo il range dinamico del segnale Y in 2^8=256 intervallini, tutti di uguale ampiezza.

In tal modo, è stato stimato quanto vale il rapporto segnale-rumore, sia per la μ-LAW sia per la A-LAW, in uscita dal quantizzatore: tale rapporto assume in entrambi i casi il valore di **38 dB**, a patto che μ=255 e che A=87.56.

Per concludere sull'aspetto della compressione, è bene ricordare che, *se in trasmissione effettuiamo una compressione, in ricezione dovremo eseguire l'operazione duale (**decompressione**), che riporti la densità di probabilità del segnale ai valori originari (quelli cioè disposti non uniformemente sul range considerato).* In caso contrario, se cioè ricostruissimo il segnale analogico direttamente dai campioni ricevuti, otterremmo un segnale distorto.

Osservazione sul rumore termico in ricezione

E' interessante osservare che, nel discorso appena fatto, non sono subentrate minimamente considerazioni riguardanti un altro tipo di rumore, anch'esso sempre presente, che è il **rumore termico**: sappiamo infatti che, dopo la trasmissione sul mezzo trasmissivo, il segnale si sarà portato dietro non solo il rumore di quantizzazione, ma anche l'immancabile rumore termico. In effetti, nel corso di *Comunicazioni Elettriche* abbiamo anche visto come si esegue il dimensionamento degli apparati per ridurre, nel modo desiderato, l'influenza del rumore termico sul segnale; in particolare, abbiamo verificato che, nei sistemi di trasmissione numerica, il rumore termico ha molta meno influenza rispetto agli analoghi sistemi analogici: infatti, abbiamo visto che basta alzare di appena 1 dB la potenza trasmessa per ottenere una diminuzione di circa 1 ordine di grandezza della probabilità di errore. Questo risultato dice in pratica che è possibile sovradimensionare i sistemi (così come si fa nella realtà) per rendere la trasmissione praticamente insensibile al rumore termico. Da qui scaturisce immediatamente che, *in un sistema di trasmissione numerica, la qualità del segnale ricostruito dipende quasi interamente dalla qualità con cui, in trasmissione, è stata effettuata la conversione analogico®digitale.* Di qui, quindi, l'importanza di scegliere nel modo ottimale gli apparati che passano dal segnale telefonico analogico di partenza alla corrispondente sequenza di bit da trasmettere.

Cenni alle linee ISDN

La quantità di **64 kbit/sec**, proprio per i discorsi dei paragrafi precedenti, è di cruciale importanza nei sistemi di trasmissione numerici in cui anche il segnale vocale transita sui mezzi trasmissivi. Per questo motivo, essa è diventata una specie di unità di misura della capacità delle linee di trasmissione. Ad esempio, nella moderna **rete ISDN** (*Integrated Service Data Network*), si dice che un utente, stipulando un apposito *contratto* con il gestore telefonico (la Telecom Italia per intenderci), ha a disposizione (nel contratto base) una **linea ISDN** che corrisponde a due linee da 64 kbit/sec più una terza linea a 16 kbit/sec, per una capacità totale, quindi, pari a

$$64\left(\frac{\text{kbit}}{\text{sec}}\right)+64\left(\frac{\text{kbit}}{\text{sec}}\right)+16\left(\frac{\text{kbit}}{\text{sec}}\right)=144\left(\frac{\text{kbit}}{\text{sec}}\right)$$

Questo significa che l'utente può effettuare un collegamento dati (ad esempio ad Internet) a 64 kbit/sec e, contemporaneamente, avere una linea libera, sempre da 64 kbit/sec, per parlare al telefono. La linea da 16 kbit/sec, invece, detta **linea di segnalazione**, non serve all'utente, ma alla rete telefonica per far funzionare il tutto (vi transitano tutte le informazioni di controllo e sincronizzazione degli apparati).

CENNI ALLA GENERAZIONE DEL SEGNALE VOCALE

Se volessimo ridurre ulteriormente la velocità di trasmissione, al di sotto dei 64 kbit/sec, dovremmo necessariamente utilizzare delle informazioni a priori sul sistema in considerazione. Queste informazioni a priori riguardano, evidentemente, il modo con cui nasce il *segnale vocale* dal quale poi, tramite filtraggio, si ottiene il segnale telefonico da convertire in digitale.

Il **segnale vocale** viene formato nel cosiddetto **tratto vocale**, che va dalle *corde vocali* (ingresso) alle *labbra* (uscita). E' questo un sistema riverberante, disadattato all'uscita (le labbra) ed abbastanza ben adattato in ingresso (le corde vocali).

Il segnale di ingresso è generato dalla vibrazione delle corde vocali ed è di due tipi:

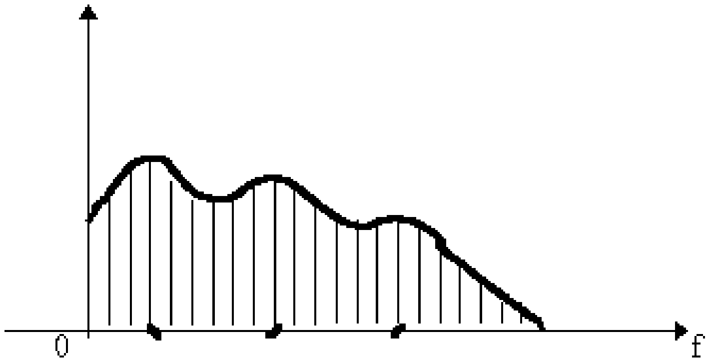
- **vocalizzato**, quando le corde vocali generano una sequenza di impulsi con cadenza regolare;
- **non vocalizzato**, quando le corde vocali generano un fruscio praticamente gaussiano, a spettro abbastanza uniforme.

Cominciamo dal segnale vocalizzato. La sequenza di impulsi generata durante la formazione di segnale vocalizzato ha una frequenza di ripetizione che va da circa 100 Hz (voce roca, toni bassi, tipicamente maschile) fino a 250-300 Hz (voce melodiosa, toni alti, tipicamente femminile o di bambini). Ogni impulso ha una durata di qualche millisecondo e quindi uno spettro che si estende fino ad 8-10 kHz (reciproco della durata). Si chiama **timbro** (in inglese **pitch**) la distanza temporale tra gli impulsi; il timbro è caratteristico di ciascun parlatore; anzi, per ogni parlatore non è nemmeno rigorosamente costante, ma varia a seconda dell'intonazione del parlato: ad esempio, quando il tono è *interrogativo* , il timbro è crescente col tempo, mentre invece quando il tono è *dichiarativo* , il timbro è decrescente nel tempo.

Essendo un segnale vocalizzato una sequenza di impulsi praticamente periodica, il suo spettro è praticamente uno spettro a righe; la distanza tra le righe è detta **frequenza di pitch**. Essa è l'inverso del pitch (che infatti viene rigorosamente chiamato *periodo di pitch*).

Gli impulsi attraversano il tratto vocale e vengono così sagomati spettralmente dalla funzione di trasferimento del tratto vocale. Tale funzione di trasferimento è stata stimata essere a soli poli (**modello AR**). Trattandosi di un sistema distribuito e non di una cascata di tronchi di linea uniformi, non è possibile determinare il numero dei poli equivalente al sistema senza fare delle approssimazioni: generalmente, si sceglie di utilizzare un modello costituito da 10 celle, ciascuna con un solo polo (per un totale quindi di 10 poli).

Osservando lo spettro di diversi segnali vocalizzati (finestrati nel tempo), si è visto che presentano caratteristiche comuni, vale a dire andamento decrescente in frequenza, frequenza massima abbastanza piccola (non oltre gli 8 -10 kHz), ed un certo numero di massimi (cioè frequenze di risonanza):



Tipico andamento dello spettro di un segnale ottenuto effettuando un troncamento (nel tempo), tramite una opportuna funzione finestra, di un suono vocalizzato

Si è stimato che, al massimo, ci sono 3-4 frequenze di risonanza per ogni vocale (cioè 3-4 frequenze in corrispondenza delle quali, dati i fenomeni di risonanza, si concentra la maggior parte dell’energia); queste frequenze di risonanza sono dette **formanti** e la loro posizione serve a caratterizzare abbastanza le **vocali** stesse. Anche queste frequenze sono caratteristiche del parlatore.

Si tenga presente che la distinzione delle vocali non è sufficiente alla comprensione del parlato; al contrario, l’intelligibilità del parlato è soprattutto legata alle **consonanti**, che sono sostanzialmente dei transitori tra una vocale (che può essere mantenuta per lungo tempo) ed un’altra.

Per quanto riguarda, invece, i **suoni non vocalizzati**, essi sono generati dal fruscio emesso dalle corde vocali quando non generano impulsi (si pensi, ad esempio, a quando si parla sottovoce): in questo caso, le corde vocali non vibrano, ma si aprono, lasciando passare l’aria; il suono viene generato da strozzature che possono trovarsi nella parte posteriore della gola (**suoni gutturali**) o nella cavità orale (**suoni sibilanti**).

Ci sono poi i cosiddetti **toni nasali**, che sono generati da interferenze tra emissioni acustiche attraverso il tratto vocale ed emissioni acustiche attraverso il **setto nasale**. Quest’ultimo, infatti, costituisce un cammino parallelo a quello del tratto vocale, per cui si ha la somma di due uscite, dovute al fatto che uno stesso ingresso (quello generato dalle corde vocali) passa attraverso due funzioni di trasferimento distinte (appunto quella del tratto vocale e quella della cavità nasale). Tipico è il caso delle **vocali nasali**: in questo caso, la cavità nasale ha una funzione di trasferimento che presenta non più solo poli, ma anche degli zeri (si tratta del parallelo tra due funzioni di trasferimento a soli poli), per cui l’effetto non è più descrivibile in termini di modello a soli poli (modello AR), ma serve un modello poli-zeri (**modello ARMA**).

Tecniche a riduzione di ridondanza

INTRODUZIONE

Abbiamo detto che il nostro nuovo obbiettivo è quello di ridurre la velocità di trasmissione, per il generico canale telefonico, al di sotto dei **64 kbit/sec** previsti dalla codifica PCM classica. Se rimaniamo nell’ambito della codifica PCM, ci accorgiamo che l’unico punto su cui poter “lavorare” riguarda proprio le caratteristiche del segnale codificato, cioè il segnale telefonico; in particolare, *la caratteristica che nella tecnica PCM viene ignorata è la correlazione tra i campioni prodotti dal campionatore:* abbiamo detto che la tecnica PCM prevede una frequenza di campionamento di 8 kHz, cui corrisponde un periodo di campionamento di 125 μsec; quindi, il campionatore produce un campione ogni 125 μsec; è allora plausibile ritenere che, nel cavo orale, non si verifichino grossi cambiamenti in questo intervallo di tempo[4], il che si traduce in una certa regolarità temporale del segnale, ossia quindi nell'esistenza di una certa correlazione tra ciascun campione ed il successivo.

In altre parole, per quelle che sono le caratteristiche fisiche del sistema di generazione (il tratto vocale) del segnale vocale, è lecito ritenere che tale segnale (e quindi anche la sua versione filtrata corrispondente al segnale telefonico) non sia bianco (cioè incorrelato), ma abbia uno spettro di potenza più complesso, rappresentativo appunto della correlazione tra i campioni. La tecnica PCM, così come è stata descritta prima, non tiene conto di questa correlazione ed in questo senso è una **tecnica a ridondanza**: significa che tale tecnica trasmette, in effetti, più informazioni di quante effettivamente ne servono, in quanto parte di tali informazioni sono correlate tra di loro.

Per comprendere a pieno questi concetti, basti pensare a quanto si è detto nel capitolo sulla *predizione lineare* (sulla quale torneremo più avanti): data una sequenza di bit, è lecito pensare che ciascun campione porti con se una certa quantità di informazione relativa ad una certo numero di campioni successivi; si può allora tentare di sfruttare queste informazioni (laddove si riesca ad individuarle) per stimare i campioni successivi.

Su questi concetti si basano le **tecniche di trasmissione a riduzione di ridondanza**: si sfrutta il più possibile l'informazione che ciascun campione contiene riguardo altri campioni, al fine di ridurre al minimo il numero di bit da trasmettere. Detto in altre parole, si cerca di eliminare dal segnale, prima della trasmissione, tutto ciò che in ricezione non sarà più necessario.

Possiamo scendere in maggiori dettagli, riprendendo proprio i concetti della predizione lineare.

TECNICA DPCM

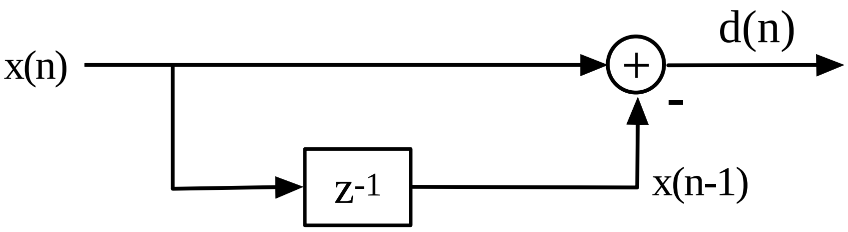
Osservando che, per il meccanismo della propria generazione, il segnale vocale ha delle fluttuazioni lente nel tempo, *invece di trasmettere ogni campione come se fosse indipendente dagli altri, si cerca di sfruttare il legame esistente tra i vari campioni.* Nel caso più semplice, si sfrutta in particolare il legame tra ciascun campione ed il precedente.

Il concetto di fondo è quello per cui, anziché trasmettere sul canale il valore di un certo campione, si trasmette un altro valore, molto più piccolo, che corrisponde alla differenza tra il campione stesso ed il campione precedente:

$$\boxed{d(n) = x(n) - x(n-1)}$$

Si mette dunque in piedi la cosiddetta **codifica differenziale**, nota con **DPCM** (che sta appunto per *Differential PCM*).

Lo schema a blocchi del trasmettitore è in questo caso fatto nel modo seguente:



Semplice schema a blocchi di un trasmettitore DPCM

[4] Le variazioni di assetto della cavità orale avvengono sempre con una certa lentezza

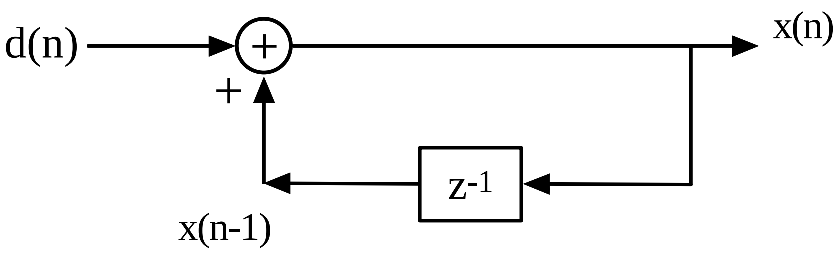
Il vantaggio è ovvio: dovendo trasmettere una differenza tra due campioni, è ragionevole pensare che ci voglia un numero inferiore di bit rispetto a quello necessario per i singoli campioni, con conseguente riduzione del *bit rate*.

In effetti, questo vantaggio è reale se tra ogni coppia di campioni successivi esiste una certa somiglianza, la quale, in termini di statistica del secondo ordine, si misura tramite la correlazione esistente tra campioni a distanza 1. Al contrario, se i campioni successivi sono molto diversi tra loro, il risparmio può venir meno.

In generale, diciamo *che la codifica DPCM è un modo semplice di tener conto del modello del sistema, ossia sostanzialmente della regolarità temporale del segnale.*

A livello teorico, i discorsi appena fatti possono funzionare. A livello pratico, invece, la cosa risulta più complessa. Vediamo perché.

Il primo problema riguarda gli errori di trasmissione. Abbiamo detto che, a partire da un certo campione x(n) della sequenza in poi, non vengono più trasmessi i campioni in se, ma la differenza d(n) tra tali campioni e i rispettivi campioni precedenti; in ricezione, bisogna risalire ai valori dei campioni, facendo l'operazione duale. Ciò significa che il ricevitore sarà fatto nel modo seguente:



Semplice schema a blocchi di un ricevitore DPCM

Supponiamo che il trasmettitore abbia cominciato la trasmissione delle differenze a partire dal campione x(n); ciò significa che il trasmettitore ha trasmesso la sequenza

$$x(1),\dots,x(n-1),d(n),d(n+1),d(n+2),\dots\dots$$

Il ricevitore si comporta nel modo seguente:

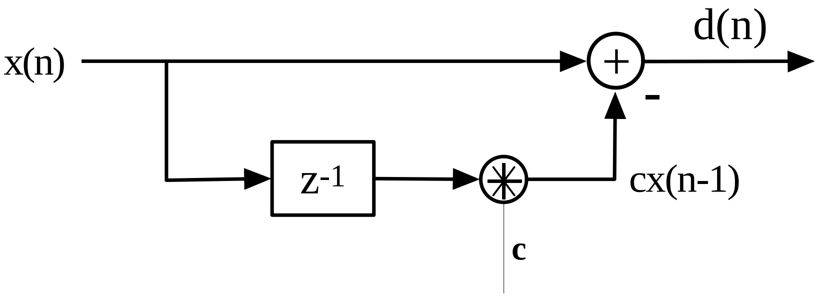
$$\begin{aligned}
&x(n)=d(n)+x(n-1)\\
&x(n+1)=d(n+1)+x(n)\\
&x(n+2)=d(n+2)+x(n+1)\\
&x(n+3)=d(n+3)+x(n+2)\\
&\dots
\end{aligned}$$

In assenza di errori di trasmissione, il meccanismo funziona. In presenza di errori, invece, subentra una inevitabile complicazione: <u>in presenza anche di un solo errore, esso si propaga su tutti i campioni successivi</u>. Ad esempio, se il ricevitore riceve errato il valore di d(n+1), tutti i campioni, a partire da x(n+1), risulteranno sbagliati, anche se le trasmissioni da d(n+2) in poi sono state corrette:

$$
\begin{aligned}
&x(n) = d(n) + x(n-1)\\
&\tilde{x}(n+1) = \tilde{d}(n+1) + x(n)\\
&\tilde{x}(n+2) = d(n+2) + \tilde{x}(n+1)\\
&\tilde{x}(n+3) = d(n+3) + \tilde{x}(n+2)\\
&....
\end{aligned}
$$

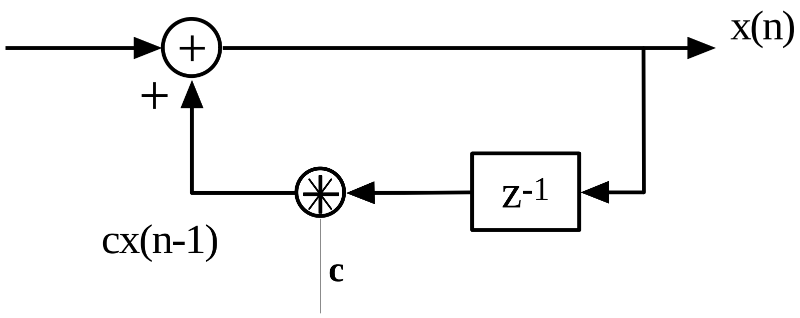
In altre parole, col meccanismo appena descritto, se c'è un errore di trasmissione, il ricevitore non è in grado di recuperarlo, per cui continua a ricostruire una forma d'onda errata da quel momento in poi, anche se la trasmissione torna ad essere esatta.

L'unica soluzione possibile a questo tipo di inconveniente è <u>fare in modo che il ricevitore non abbia una memoria infinita</u>, in modo che un dato errore non si propaghi all'infinito, ma solo su un certo numero (inevitabile) di campioni successivi. Per ottenere questo risultato, è sufficiente introdurre una moltiplicazione per una costante c<1, in modo che l'effetto dell'eventuale errore pesi sempre meno al passare del tempo. Lo schema a blocchi del trasmettitore diventa allora il seguente:



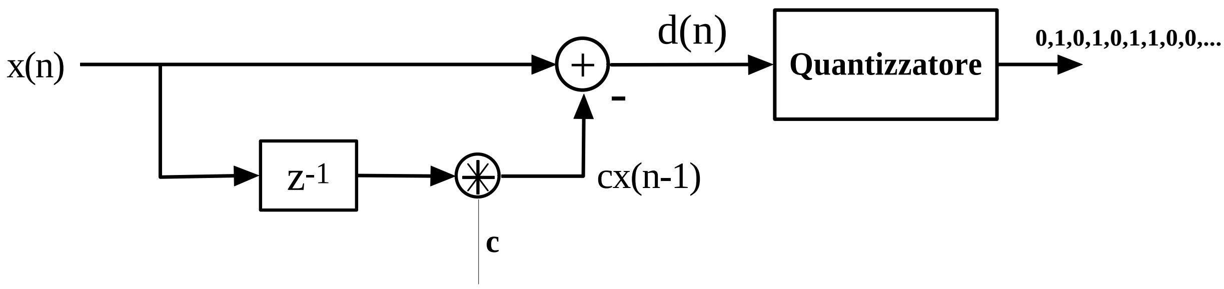
Schema a blocchi semplificato di un trasmettitore DPCM con memoria finita, garantita dalla costante moltiplicativa c<1

Il ricevitore dovrà ovviamente avere uno schema duale di quello appena disegnato:



Schema a blocchi di un ricevitore DPCM con memoria finita

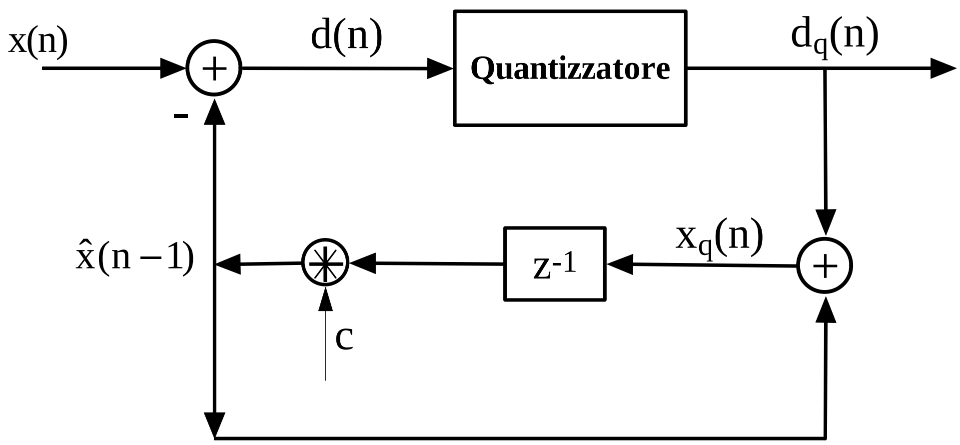
C'è un'altra considerazione da fare. Se torniamo allo schema a blocchi del trasmettitore (penultima figura) esso manca di un dispositivo fondamentale che è il quantizzatore, il quale si occupa di associare, ai campioni d(n) da trasmettere, le corrispondenti sequenze di bit:



Schema a blocchi di un trasmettitore DPCM con memoria finita (c<1), con dettaglio sul quantizzatore, il quale genera la sequenza binaria che realmente viene trasmessa

A questo punto, si nota una sostanziale differenza tra quello che accade in trasmissione e quello che accade in ricezione: infatti, mentre il trasmettitore ha a disposizione il valore vero di x(n-1), il ricevitore ha a disposizione solo un valore approssimato di x(n-1), dove l'approssimazione deriva dal fatto che il ricevitore riceve i valori quantizzati di d(n), che in generale differiscono dai valori veri a causa dell'inevitabile rumore di quantizzazione. Potremmo in un certo senso affermare che il ricevitore ha solo una *stima* di d(n), per cui è in grado di calcolare solo una *stima* di x(n-1), a patto però di tener presente che il concetto di *stima*, in questo caso, non è legato al fatto che venga calcolata una predizione, ma appunto al rumore di quantizzazione.

Questo fatto rappresenta un problema, in quanto abbiamo capito prima che il meccanismo DPCM funziona solo se il ricevitore esegue le operazioni esattamente duali del trasmettitore. Non avendo gradi di libertà, in questo contesto, sul ricevitore, dobbiamo allora fare in modo che il trasmettitore "operi" esattamente come fa il ricevitore, ossia basandosi sulla sequenza in uscita dal quantizzatore. Lo schema del trasmettitore deve allora diventare il seguente:



Schema completo di un trasmettitore DPCM. Si nota che il campione precedente quello attuale viene calcolato sulla base dei valori quantizzati della differenza d(n)

Con questo schema, il trasmettitore esegue le operazioni esattamente come il ricevitore (il cui schema a blocchi è sempre quello di prima), ossia sostanzialmente calcola $\hat{x}(n-1)$ così come fa il ricevitore:

$$\hat{x}(n-1) = cx_q(n-1)$$

CODIFICHE PREDITTIVE

Possiamo inquadrare la codifica DPCM vista nel precedente paragrafo come una applicazione dei concetti, considerati in capitoli precedenti, sulla *predizione lineare*. Vediamo di spiegarci meglio.

Supponiamo che i campioni prodotti dal campionatore, a passo T, siano i seguenti:

$$x(1), x(2), \ldots, x(n-N), x(n-N+1), \ldots, x(n-1), x(n), x(n+1), \ldots$$

Supponiamo di aver trasmesso i campioni da x(1) a x(n-1) e di dover dunque trasmettere adesso x(n). Allora, prima di effettuare tale trasmissione, calcoliamo una "stima" di x(n), ossia determiniamo x(n) sulla base dei valori assunti da una parte dei campioni precedenti, per esempio quelli a partire da x(n-N). Per semplicità, usiamo una **stima lineare**, ossia valutiamo la stima di x(n), che indichiamo con $\hat{x}(n)$, secondo una semplice combinazione lineare del tipo seguente:

$$\hat{x}(n) = a_1 x(n-1) + a_2 x(n-2) + \ldots + a_N x(n-N)$$

Definiamo adesso la quantità d(n) in modo diverso rispetto al paragrafo precedente:

$$d(n) = x(n) - \hat{x}(n)$$

In questo caso, non si tratta più della differenza tra il campione attuale ed il precedente, bensì dell'**errore** che eventualmente commettiamo tra il valore esatto del campione e la stima da noi effettuata.

Allora, il sistema utilizzato è il seguente: nel momento in cui è disponibile, in trasmissione, il campione x(n), anziché trasmettere tale campione, si valuta la sua stima $\hat{x}(n)$, si valuta poi d(n) e si trasmette sul canale la versione quantizzata di d(n); in uscita dal canale, viene nuovamente calcolata la stima $\hat{x}(n)$ e quindi, sommando tale stima al valore ricevuto $d_q(n)$, si ottiene il campione desiderato (a meno sempre dell'errore dovuto al rumore di quantizzazione).

Analogamente a questo visto nella codifica DPCM, il vantaggio è nel fatto che d(n) ha un valore molto piccolo, tanto più piccolo quanto più è accurata la stima $\hat{x}(n)$, per cui sarà certamente necessario un numero inferiore di bit rispetto a quello necessario per trasmettere direttamente x(n). E' ovvio, inoltre, che sia il trasmettitore sia il ricevitore devono usare lo stesso meccanismo per effettuare le stime, in modo che il valore di $\hat{x}(n)$ sia lo stesso in trasmissione ed in ricezione.

Quindi, la trasmissione procede nel modo seguente:

- vengono prima trasmessi, così come sono, un certo numero di campioni iniziali del segnale;
- a partire da uno di essi (solitamente intorno al decimo), si passa a trasmettere d(n) quantizzato, con conseguente riduzione del numero di bit da inviare e quindi con conseguente aumento della velocità di trasmissione.

E' ovvio, come detto poco fa, che questo metodo ha successo quando d(n) è effettivamente più piccolo di x(n), ossia, in definitiva, quando la stima $\hat{x}(n)$ è molto accurata. L'accuratezza della stima dipende evidentemente dai coefficienti di $\hat{x}(n)$, per cui tutto il problema si riduce a determinare il valore di tali coefficienti tali che venga minimizzato il valore di d(n).

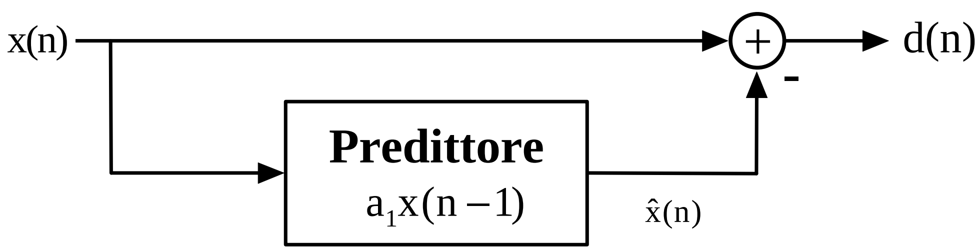
Il caso più semplice è chiaramente quello in cui la stima viene effettuata sulla base solo del campione precedente a quello attuale. Questo significa utilizzare una legge del tipo seguente:

$$\boxed{\hat{x}(n) = a_1 x(n-1)}$$

Come ben sappiamo, si parla in questo caso di **stima lineare del primo ordine**, proprio perché si usa solo il campione precedente quello da stimare.

Ci accorgiamo immediatamente che, se prendiamo $a_1=1$, questo modo di procedere coincide con la tecnica DPCM. Anzi, se facciamo riferimento alle considerazioni che hanno portato, nel DPCM, all'introduzione della costante moltiplicativa *c*, ci accorgiamo che la coincidenza tra i due modi di procedere diventa generale. Di conseguenza, non solo potremo adesso ripetere considerazioni analoghe a quelle fatte per il DPCM, ma potremo fare anche nuove considerazioni che a loro volta varranno anche per il DPCM.

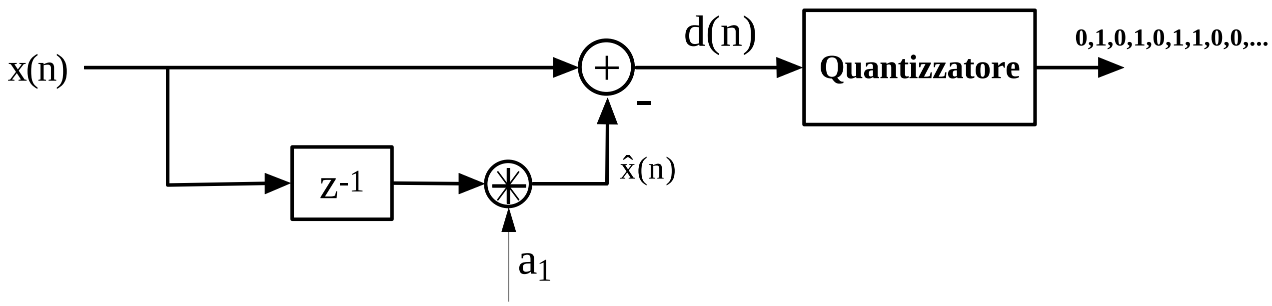
Lo schema a blocchi utilizzato in trasmissione dalle tecniche predittive del primo ordine è del tipo seguente:



Schema di una trasmissione con predizione lineare del primo ordine: noti i campioni x(n) (campione attuale) e x(n-1) (campione precedente), si usa quest'ultimo per stimare x(n) e trasmettere la differenza d(n) tra la stima ed il valore reale

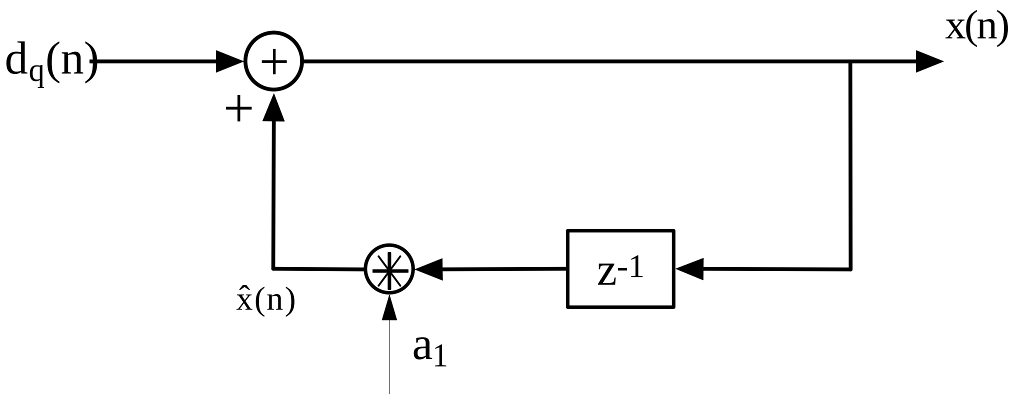
Bisogna così determinare il coefficiente a_1 in modo da ridurre al minimo d(n). Questo aspetto sarà ripreso più avanti.

Scendendo nei dettagli del predittore lineare, nel caso di una predizione del primo ordine è facile riportarne lo schema a blocchi, per cui possiamo perfezionare la figura precedente, introducendo tra l'altro anche il quantizzatore, il quale, ricevendo in ingresso d(n), genera la corrispondente sequenza di bit da trasmettere:



Schema di una trasmissione con tecnica predittiva, con dettaglio del predittore lineare del primo ordine e con l'indicazione del quantizzatore

Per quanto riguarda, invece, il comportamento del ricevitore, lo schema a blocchi è il duale del precedente:



Schema di un ricevitore con predizione lineare del primo ordine. Il segnale in ingresso è d(n) quantizzato, ossia codificato come sequenza di bit

Ancora una volta, abbiamo il problema degli errori di trasmissione. Supponiamo, come fatto prima, che il trasmettitore abbia cominciato la trasmissione delle differenze a partire dal campione x(n), il che significa che ha trasmesso la sequenza

$$x(1),\dots,x(n-1),d(n),d(n+1),d(n+2),\dots\dots$$

Il ricevitore si comporta nel modo seguente:

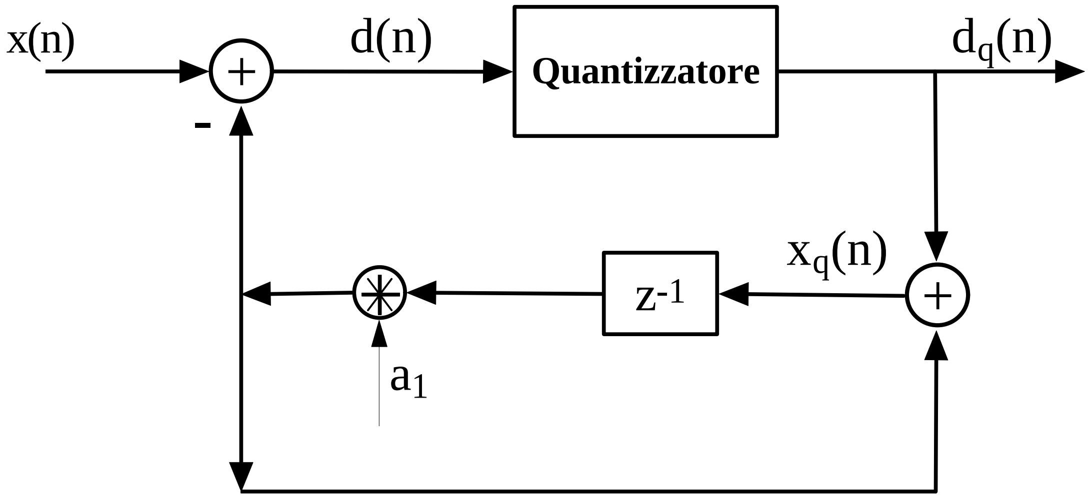
$$x(n)=d(n)+\hat{x}(n)=d(n)+a_1x(n-1)$$
$$x(n+1)=d(n+1)+\hat{x}(n+1)=d(n+1)+a_1x(n)$$
$$x(n+2)=d(n+2)+\hat{x}(n+2)=d(n+1)+a_1x(n+1)$$
$$\dots$$

In assenza di errori di trasmissione, il meccanismo funziona. In presenza di errori, invece, abbiamo problemi: in presenza anche di un solo errore, esso si propaga su tutti i campioni successivi. Ad esempio, se il ricevitore riceve errato il valore di d(n+1), tutti i campioni, a partire da x(n+1), risulteranno sbagliati, anche se le trasmissioni da d(n+2) in poi sono state corrette.

Quindi, ancora una volta, se c'è un errore di trasmissione, il ricevitore non è in grado di recuperarlo. Per evitare che il singolo errore si propaghi all'infinito, il sistema dovrà possedere una memoria finita, in modo che possa sempre riagganciarsi al segnale trasmesso dopo un certo tempo, più o meno lungo.

Sempre in analogia a quanto visto per la tecnica DPCM, abbiamo anche qui problemi legati alla posizione del quantizzatore. Infatti, per come sono stati disegnati gli schemi a blocchi del trasmettitore e del ricevitore, quest'ultimo esegue le proprie stime in modo diverso dal trasmettitore: infatti, mentre il trasmettitore ha a disposizione d(n) ottenuto come differenza tra x(n) ed $\hat{x}(n)$, il ricevitore ha a disposizione solo la sequenza di bit che il quantizzatore ha associato a d(n) e il valore numerico corrispondente a questa sequenza non necessariamente corrisponde al valore vero di d(n), dato l'inevitabile errore di quantizzazione.

Per dualizzare perfettamente le operazioni fatte in trasmissione e ricezione, dobbiamo dunque considerare un trasmettitore fatto nel modo seguente:



Con questo schema, il trasmettitore esegue le stime esattamente come il ricevitore, il cui schema a blocchi è esattamente uguale a prima: sostanzialmente, *il funzionamento è quello per cui la differenza d(n) viene quantizzata opportunamente, dopo di che viene sia trasmessa sia usata per aggiornare il segnale predetto e, eventualmente, la base dati per le future predizioni.*

Osservazione: guadagno di predizione

(richiami di *Teoria dei Segnali*) Facciamo qualche calcolo per caratterizzare maggiormente le tecniche appena esposte, che cioè sfruttano la predizione di un campione in funzione di quello precedente:

$$\hat{x}(n) = a_1 x(n-1)$$

Il nostro obbiettivo è dunque quello di determinare a_1 in modo da ridurre quanto più è possibile la variabilità di d(n).

Dovendo considerare il segnale telefonico come un processo stocastico, ciascuno dei campioni da esso prelevati costituirà una variabile aleatoria, per cui anche d(n) sarà una variabile aleatoria; di conseguenza, ridurre la variabilità di d(n) significa ridurre la sua varianza.

Dobbiamo dunque trovare una espressione della varianza di d(n); per fare questo, facciamo due ipotesi semplificative:

- la prima è che il segnale telefonico sia un processo stocastico stazionario, il che significa che esso conserva invariate nel tempo le sue caratteristiche statistiche[5];
- la seconda, senz'altro più realistica della prima, è invece che il segnale telefonico sia un processo a media nulla; questo, in base anche alla stazionarietà, implica che

$$E[x(n)] = E[x(n-1)] = 0$$

e anche che

$$\mathrm{Var}\big(x(n)\big) = E\big[x^2(n)\big]$$

[5] E' bene ricordare che questa ipotesi non è molto lecita nel caso del segnale telefonico, ma per i nostri ragionamenti ci fa comunque comodo

E' facile ricavare che anche d(n) risulta essere a media nulla: infatti, dato che $d(n) = x(n) - \hat{x}(n)$, dato che la media è lineare, dato che $E[x(n)] = E[x(n-1)] = 0$, possiamo scrivere che

$$E[d(n)] = E[x(n)] - E[\hat{x}(n)] = -E[\hat{x}(n)] = -E[a_1 x(n-1)] = -a_1 E[x(n-1)] = 0$$

Da qui scaturisce che la varianza di d(n) è

$$\mathrm{Var}(d(n)) = E[d^2(n)] - E^2[d(n)] = E[d^2(n)]$$

Sostituendo l'espressione di d(n) e sfruttando la linearità della media, abbiamo che

$$\begin{aligned}\mathrm{Var}(d(n)) &= E[x^2(n) + \hat{x}^2(n) - 2x(n)\hat{x}(n)] = E[x^2(n)] + E[\hat{x}^2(n)] - 2E[x(n)\hat{x}(n)] = \\ &= E[x^2(n)] + a_1^2 E[x^2(n-1)] - 2E[x(n) \cdot a_1 x(n-1)]\end{aligned}$$

Data l'ipotesi di stazionarietà del segnale telefonico, è chiaro che $E[x^2(n)] = E[x^2(n-1)]$, per cui

$$\mathrm{Var}(d(n)) = (1 + a_1^2)E[x^2(n)] - 2a_1 E[x(n)x(n-1)]$$

La quantità $E[x(n)x(n-1)]$ non è altro che la funzione di autocorrelazione di x(n), calcolata per ritardo unitario; analogamente, $E[x^2(n)]$ è la stessa funzione di autocorrelazione, calcolata però per ritardo nullo: deduciamo che

$$\begin{cases} R_X(1) = E[x(n)x(n-1)] \\ R_X(0) = E[x^2(n)] = E[x(n) \cdot x(n)] \end{cases} \longrightarrow \boxed{\mathrm{Var}(d(n)) = (1 + a_1^2)R_X(0) - 2a_1 R_X(1)}$$

Possiamo a questo punto trovare il valore di a_1 che minimizza questa quantità: basta derivare rispetto ad a_1 e imporre che la derivata sia uguale a zero; otteniamo quanto segue:

$$\frac{d}{da_1}\mathrm{Var}(d(n)) = 2a_1 R_X(0) - 2R_X(1) \longrightarrow \boxed{a_1^* = \frac{R_X(1)}{R_X(0)}}$$

Quindi, il valore del coefficiente a_1 che minimizza la variabilità di d(n) (e quindi anche il numero di bit da usare per la corrispondente codifica) è $R_X(1)/R_X(0)$.

Sostituendo questo valore nell'espressione di $\mathrm{Var}(d(n))$, otteniamo anche una misura della suddetta variabilità:

$$\begin{aligned}\mathrm{Var}(d(n)) &= \left(1 + \frac{R_X^2(1)}{R_X^2(0)}\right)R_X(0) - 2\frac{R_X(1)}{R_X(0)}R_X(1) = \left(R_X(0) + \frac{R_X^2(1)}{R_X(0)}\right) - 2\frac{R_X^2(1)}{R_X(0)} = \\ &= R_X(0) - \frac{R_X^2(1)}{R_X(0)}\end{aligned}$$

Scritto in forma migliore, possiamo affermare che il valore minimo della varianza di d(n) è

$$\boxed{[\mathrm{Var}(d(n))]_{MIN} = R_X(0)\left(1 - \frac{R_X^2(1)}{R_X^2(0)}\right)}$$

Ricordiamo inoltre che il rapporto $R_X(1)/R_X(0)$ ha un ben preciso significato, che è quello di **coefficiente di autocorrelazione** di x(n), calcolato ancora per un ritardo unitario:

$$\rho(1)=\frac{E[x(n)x(n-1)]-E[x(n)]E[x(n-1)]}{\sqrt{Var(x(n))Var(x(n-1))}}=\frac{E[x(n)x(n-1)]}{\sqrt{E[x^2(n)]\cdot E[x^2(n-1)]}}=\frac{R_X(1)}{E[x^2(n)]}=\frac{R_X(1)}{R_X(0)}$$

Tornando dunque alla varianza minima di d(n), scriviamo che

$$[Var(d(n))]_{MIN}=R_X(0)\left(1-\rho^2(1)\right)$$

Questa espressione ci aiuta a capire se e quanto abbiamo guadagnato.

Intanto, il coefficiente di correlazione varia nell'intervallo [-1,1], per cui il suo quadrato varia tra 0 ed 1; in particolare, il valore esatto dipende dal segnale e dal grado di correlazione in esso esistente. Questo ci dice che, una volta preso per a_1 proprio il valore di ρ(1), siamo certi di essere nella situazione migliore possibile, ma ancora dipendiamo dal valore assunto da ρ(1).

Il caso migliore che ci può capitare è ovviamente che $\rho^2(1)=1$, in quanto avremmo $[Var(d(n))]_{MIN}=0$. Tuttavia, questo è un caso ideale assolutamente non realizzabile[6]. Dovremo sempre accontentarci di $\rho^2(1)<1$.

In generale, possiamo valutare la convenienza della tecnica DPCM basandoci sul concetto del **guadagno di predizione**, definito come rapporto tra la funzione di autocorrelazione di x(n) e la funzione di autocorrelazione di d(n), entrambe calcolate in 0:

$$\boxed{G=\frac{R_X(0)}{R_d(0)}}$$

Il motivo per cui questo parametro G quantifica la bontà del sistema messo in piedi è evidente: se G è grande, significa che $R_d(0)$ è minore di $R_X(0)$, il che significa che il segnale d(n) trasmesso presenta correlazione minore rispetto ad x(n), ossia che abbiamo ottimizzato la trasmissione[7]. Vista la cosa da un altro punto di vista, possiamo anche scrivere (ricordando che sia x(n) sia d(n) si suppongono processi stazionari a media nulla), che

$$\begin{cases}R_X(0)=E[x^2(n)]=\text{varianza di } x(n)\\ R_d(0)=E[d^2(n)]=\text{varianza di } d(n)\end{cases}\longrightarrow G=\frac{Var[x(n)]}{Var[d(n)]}$$

Con questa nuova espressione, il valore di G ci dice quanto la varianza del segnale x(n) è più grande o più piccola di quella del segnale trasmesso: se G>1, significa che il segnale trasmesso ha meno varianza di x(n) e quindi potrà essere codificato con meno bit.

Deduciamo che è possibile parlare rigorosamente di "*guadagno*" solo nel caso in cui G>1.

Vediamo ad esempio quanto vale il guadagno di predizione quando prendiamo come valore del coefficiente a_1 proprio il valore migliore possibile, ossia $R_X(1)/R_X(0)$: in questo caso, abbiamo trovato prima che la varianza di d(n) assume il suo valore minimo, per cui

$$a_1^*=\frac{R_X(1)}{R_X(0)}\longrightarrow[Var(d(n))]_{MIN}=R_X(0)\left(1-\rho^2(1)\right)\longrightarrow G=\frac{R_X(0)}{R_d(0)}=\frac{R_X(0)}{R_X(0)\left(1-\rho^2(1)\right)}=\frac{1}{1-\rho^2(1)}$$

[6] D'altra parte, dire che la varianza di una variabile aleatoria è nulla significa dire che tale variabile assume sempre lo stesso valore (pari al valore medio), per cui sarebbe del tutto inutile trasmettere d(n), visto che E[d(n)]=0

[7] A questo proposito, possiamo ricordare quanto visto nel capitolo sulla predizione lineare: mentre il segnale x(n) di partenza presenta una certa correlazione tra i campioni (cioè ha uno spettro di potenza colorato), l'errore di predizione d(n) tende invece ad essere incorrelato (spettro di potenza bianco), il che significa che l'informazione trasmessa è quella essenziale, senza ridondanza.

Dato che $\rho^2(1)$ varia nell'intervallo $[0,+1]$, G può essere maggiore o uguale di 1 a seconda di quale sia il valore di $\rho(1)$. Il caso peggiore che può capitare è che ρ=0, nel quale caso G=1, ossia non otteniamo né un guadagno né una perdita: d(n) e x(n) hanno uguale varianza, per cui vanno codificate con lo stesso numero di bit. Questo significa che prendendo il valore $\underline{a_1^*}$, male che vada non perdiamo niente. In generale, quanto più ρ cresce, tanto maggiore è il nostro guadagno.

Un caso particolare è quello in cui prendiamo a_1=1, ossia stimiamo il campione attuale x(n) tramite il valore del campione precedente (è quello che si fa nella codifica DPCM, dove trasmettiamo la differenza tra il campione attuale e quello precedente). E' facile ricavare che il guadagno di predizione vale in questo caso

$$G = \frac{1}{2(1-\rho(1))}$$

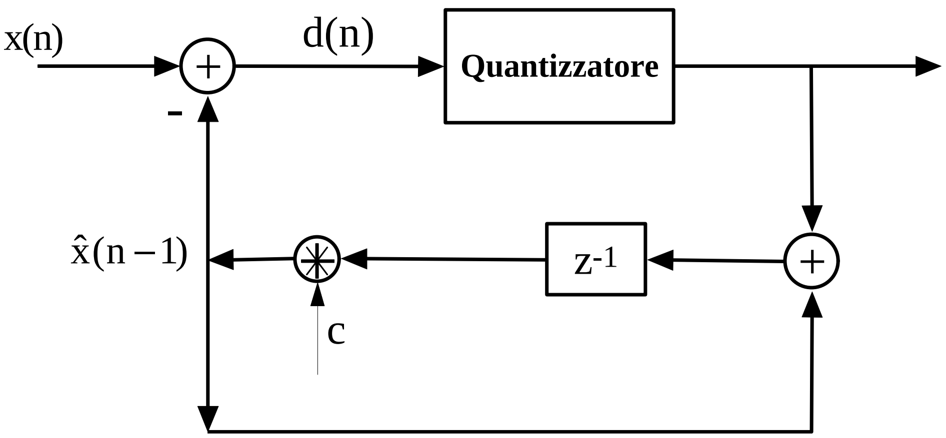
Da quest'ultima relazione si deduce che

$$\frac{1}{2(1-\rho(1))} \geq 1 \quad \text{se e solo se} \quad \rho(1) > 0.5$$

Richiedere un valore di $\rho(1)$ maggiore di 0.5 significa richiedere una correlazione particolarmente elevata tra x(n) e x(n-1) ed è dunque l'unica condizione sotto la quale noi possiamo avere un guadagno quando a_1=1.

QUANTIZZAZIONE NELLA CODIFICA DPCM: MODULAZIONE DELTA

Supponiamo di aver accettato di usare una **codifica DPCM**, secondo i criteri esposti nei precedenti paragrafi. Ciò significa che il trasmettitore può essere modellato secondo il seguente schema a blocchi, peraltro già proposto in precedenza:



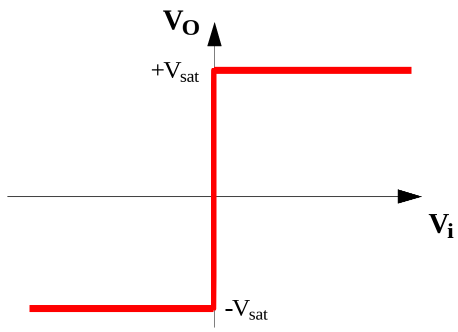
Concentriamoci sul funzionamento del quantizzatore.

Se vogliamo ottenere il massimo risparmio possibile, in termini di numero di bit da trasmettere, la cosa più semplice da fare è effettuare la quantizzazione ad 1 solo bit: questo significa che trasmettiamo solo il segno della differenza tra il campione attuale e quello precedente, in modo da indicare se il campione attuale è maggiore o minore del precedente.

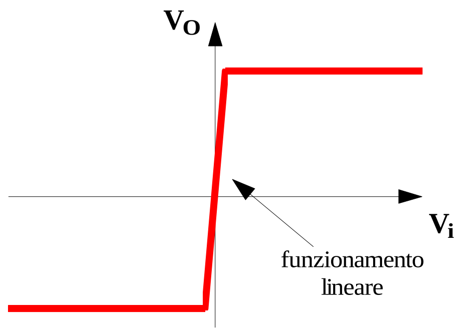
La velocità di trasmissione che otteniamo in questo caso è estremamente bassa (anche se non è la minima possibile, come si vedrà più avanti):

$$8\cdot 10^3\left(\frac{\text{campioni}}{\text{secondo}}\right)\cdot 1\left(\frac{\text{bit}}{\text{campione}}\right)=8\left(\frac{\text{kbit}}{\text{sec}}\right)$$

A livello implementativo, è facile realizzare un simile quantizzatore, in quanto dovrà trattarsi di un banale **comparatore** (ad esempio un amplificatore operazionale in configurazione non invertente):

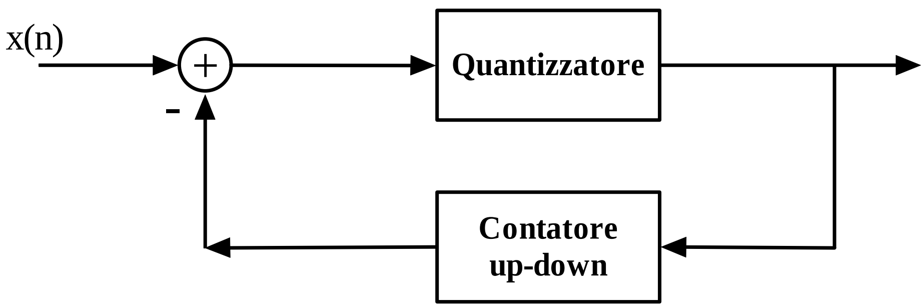


Caratteristica ingresso-uscita di un *comparatore ideale*



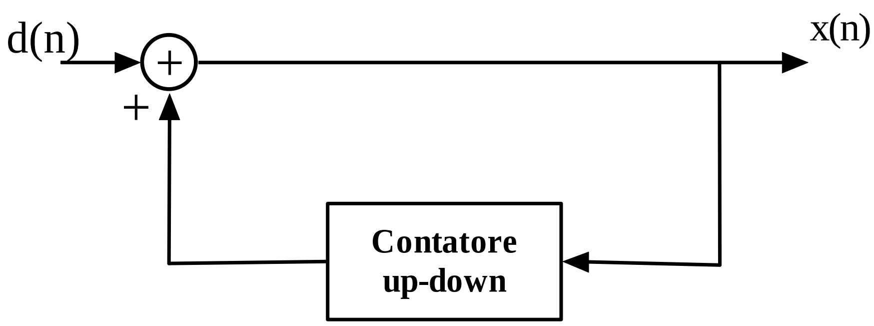
Caratteristica ingresso-uscita di un *comparatore reale*

Altrettanto facile è implementare il ramo di retroazione del trasmettitore, che potrà essere un semplice **contatore**, che viene continuamente aggiornato per permettere il successivo confronto:



Schema semplificato del trasmettitore per la modulazione delta

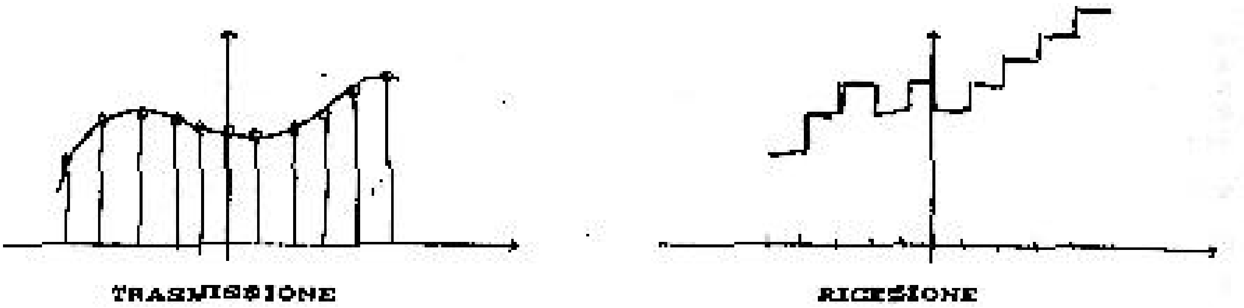
Per gli stessi motivi, anche il ricevitore potrà essere formato da un semplice contatore:



Schema semplificato del ricevitore per la modulazione delta

Concentriamoci sul funzionamento proprio del ricevitore.

Se dunque decidiamo di trasmettere solo il segno della differenza d(n), in ricezione ci troveremo a ricostruire una **forma d'onda a gradini**, ciascuno di ampiezza costante e definita a priori:

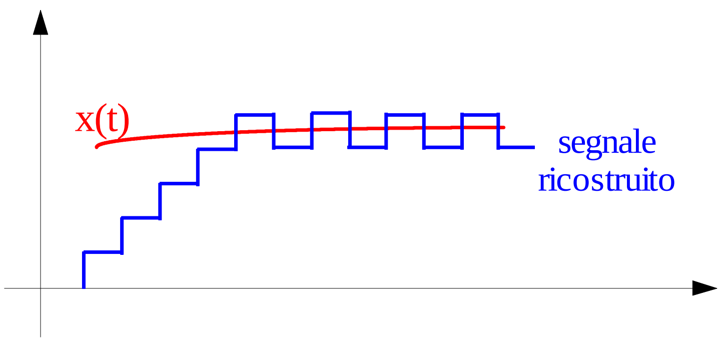


Ogni gradino va verso l'alto se il segno è positivo (bit ricevuto 0, campione attuale maggiore del precedente), mentre va verso il basso in caso contrario (bit ricevuto 1, campione attuale minore del precedente): infatti, il ricevitore, avendo memoria del campione precedente, decide, in base al segno che è arrivato, se incrementare o decrementare il livello, ad esempio, di tensione in uscita. L'incremento e/o il decremento sono entrambi pari ad un δ prefissato, da cui il nome di **modulazione delta** attribuito a questa tecnica.

Appare ovvio che non si ottenga una forma d'onda identica a quella di partenza, ma ciò è insito nel meccanismo stesso della quantizzazione.

Questo modo di procedere si presta a due tipi di errori: il rumore granulare in zone piatte e la saturazione di pendenza.

Il **rumore granulare in zone piatte** è un fenomeno ben evidenziato nell'ultima figura riportata: se l'andamento del segnale analogico di partenza è molto regolare (fluttuazioni dolci tra un campione ed il successivo), il segno dell'errore è alternativamente positivo e negativo, dato che non è possibile mantenere un valore costante per d(n); la conseguente forma d'onda a gradini presenterà allora delle oscillazioni (pezzi di onda quadra), anche laddove queste non erano inizialmente presenti:



Queste oscillazioni rappresentano appunto il rumore granulare nelle zone piatte.

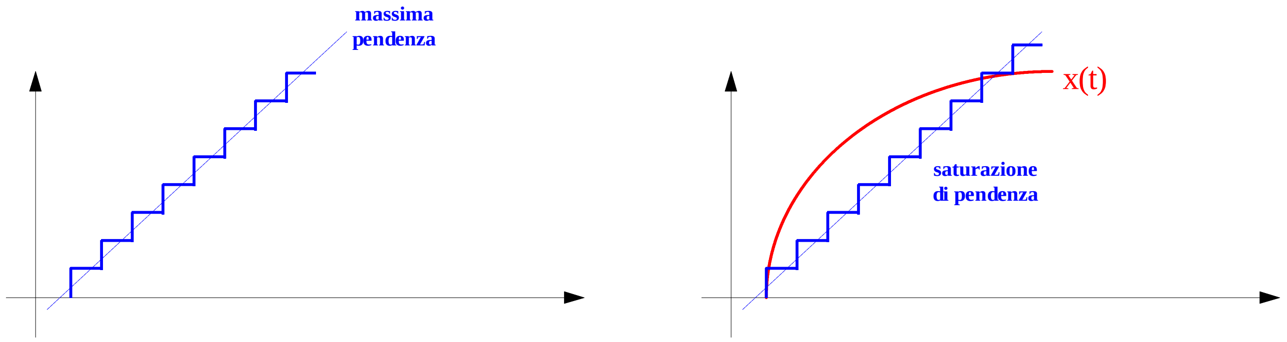
Appare evidente che un modo di ridurre questo fenomeno è ridurre l'altezza δ degli scalini.

La **saturazione di pendenza** si crea in un contesto in qualche modo opposto a quello del rumore granulare, nel senso che si tratta di un fenomeno presente quando il segnale di partenza ha un andamento molto ripido.

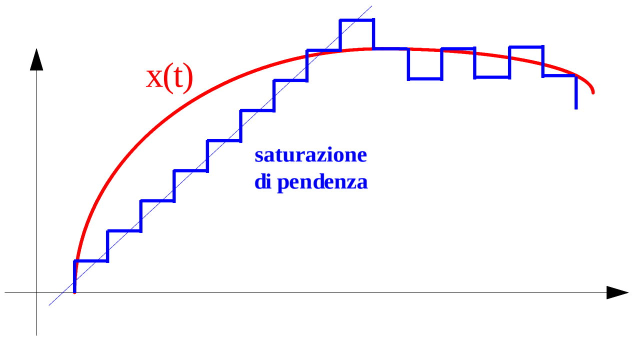
Abbiamo infatti osservato che, scegliendo di trasmettere solo il segno della differenza d(n), la forma d'onda ricostruita è a gradini, dove ogni gradino ha altezza e durata fisse: l'altezza è δ (positiva o negativa), mentre la durata è ovviamente pari al passo di campionamento T. Questo fatto ha come immediata conseguenza che la forma d'onda ricostruita possa salire (gradini sempre positivi) con una pendenza massima pari all'altezza degli scalini diviso la larghezza:

$$\text{pendenza massima} = \frac{\delta}{T}$$

Per capirci meglio, consideriamo la figura seguente:



Nella figura a sinistra, è indicata la massima pendenza che la forma d'onda ricostruita può assumere, cioè appunto δ/T. Nella figura a destra è invece rappresentato il caso in cui la forma d'onda di partenza sale con una pendenza maggiore di quella massima ottenibile al ricevitore: in una situazione come questa, dato appunto che la forma d'onda ricostruita non può andare oltre la massima pendenza, si crea un inevitabile differenza tra quanto ricostruito e la forma d'onda di partenza. Il sistema, quindi, si sgancia dalla forma d'onda originale, per riagganciarsi solo quando questa scende oppure quando cresce più lentamente:



Questo effetto (detto di **slope overload**, cioè appunto *sovraccarico di pendenza*) è chiaramente non lineare, in quanto viene praticamente tagliato un pezzo del segnale. E' ovvio, inoltre, che il discorso vale ugualmente se il segnale originale scende con pendenza elevata: in questo caso, il modulatore delta non può far altro che scendere con la velocità che il sistema gli consente, tagliando nuovamente la forma d'onda fin quando questa non rallenta la discesa oppure riprende a salire.

Anche per questo effetto esiste un rimedio (sia pure parziale), che è quello di fare i gradini più alti, in modo da poter seguire pendenze maggiori (a parità di periodo di campionamento). Si tratta però del rimedio opposto a quello necessario per ridurre l'effetto del rumore granulare prima descritto. Si capisce perciò che, *in fase di dimensionamento, bisognerà fissare un valore d per l'altezza dei gradini che sia di compromesso tra le due esigenze.* Al contrario, l'unico modo di venire incontro ad entrambe le esigenze è quello di infittire il passo di campionamento, cioè aumentare la frequenza di campionamento: con questa soluzione, a parità di altezza degli scalini, aumenta la massima pendenza che si può seguire.

Un altro notevole vantaggio dell'aumento della frequenza di campionamento riguarda il rumore granulare in zone piatte: infatti, avendo detto che tale rumore corrisponde a delle oscillazioni in corrispondenza degli intervalli di tempo in cui il segnale originale ha andamento uniforme, è evidente che una riduzione di T equivale ad un maggiore ritmo delle suddette oscillazioni, il che significa che il rumore diventa di più alta frequenza; questo è un vantaggio in quanto potremo poi filtrare (in modo analogico) il segnale per smussare le oscillazioni ed ottenere una descrizione sufficientemente accurata della forma d'onda originale.

Ancora, se riduciamo T, cioè se avviciniamo gli istanti in cui preleviamo i campioni del segnale di partenza, è intuitivo aspettarsi che la correlazione tra campioni successivi aumenti, il che consente di ottenere maggiori prestazioni dalla codifica differenziale.

In generale, la tecnica di sovracampionare il segnale (oltre gli 8 kHz previsti dallo standard PCM) prima di quantizzarlo è quella implementata nei cosiddetti **convertitori A/D a sovracampionamento**.

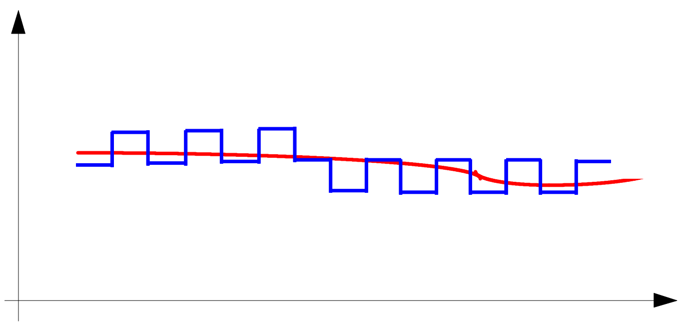
Modulazione delta adattativo (ADPCM)

Esiste un altro rimedio per il problema della saturazione di pendenza e consiste nell'usare un modello per il segnale leggermente più "evoluto" di quello visto nel paragrafo precedente. Ancora una volta, si tratta di sfruttare l'ipotesi di regolarità temporale del segnale originale e, inoltre, la non stazionarietà del segnale stesso.

Abbiamo osservato che il sistema non riesce a seguire il segnale originale quando questo sale (o scende) troppo rapidamente e per un certo periodo di tempo. Allora, si può pensare di mettere in piedi un sistema di **codifica delta-adattativo** (brevemente **ADPCM**, ossia *Adaptative Differential PCM*) con il quale, in situazioni particolari di crescita (decrescita) persistente, l'altezza degli scalini venga via via aumentata. In altri termini, *se ci si accorge che è necessario mandare vari gradini in salita (o in discesa), la struttura deve aumentare via via l'altezza degli scalini.* Per esempio, si può pensare di aumentare δ di un fattore P se due bit successivi hanno lo stesso segno (situazione di inseguimento) e di ridurlo di un fattore Q (non necessariamente uguale a P) se invece hanno segni opposti (situazione di inattività).

E' ovvio che, per ottenere un simile risultato, è necessario introdurre maggiore memoria nel sistema, in quanto esso deve sempre conoscere un pezzo della storia passata del segnale.

L'altra situazione in cui questo procedimento funziona è quando il segnale originale è quasi piatto, come indicato nella figura seguente:



Sappiamo che in questo caso gli scalini tendono alternativamente a salire e scendere, per cui si pone il problema che abbiamo inquadrato come *rumore granulare*, ossia come oscillazioni indesiderate sulla forma d'onda ricostruita. Se il sistema ha memoria, si "accorge" che la forma d'onda si mantiene approssimativamente costante per un periodo di tempo abbastanza lungo, per cui provvedere a ridurre l'altezza degli scalini (ad esempio del fattore Q citato prima), in modo da ridurre l'ampiezza delle oscillazioni.

Come già ricordato più volte in precedenza, *meccanismi di questo tipo, nei quali cioè si usi il passato per predire il futuro, funzionano solo se il* *futuro cambia molto lentamente;* se invece ci sono variazioni brusche del segnale, si crea un transitorio nel quale il sistema perde inevitabilmente l'aggancio, dando luogo a distorsioni sulla forma d'onda ricostruita rispetto a quella originale; esaurito il transitorio, il sistema si adatta nuovamente al segnale e la ricostruire riprende più o meno fedelmente.

Tecniche DPCM con quantizzazione a più bit

I discorsi del paragrafo precedente valgono nel caso in cui si adotti una codifica PCM differenziale con una quantizzazione ad 1 solo bit: si tratta cioè di trasmettere solo il segno della differenza d(n) tra il valore reale del campione x(n) e il valore della sua stima. E' possibile però pensare di usare quantizzatori a più bit, nel qual caso si può riprendere il discorso sull'errore quadratico medio: l'ottimizzazione della curva di quantizzazione può essere fatta al fine di minimizzare l'errore quadratico medio tra uscita ed ingresso del quantizzatore.

Se si sceglie di usare una quantizzazione non uniforme, in modo da dare errori più piccoli per i valori del segnale più probabili, bisogna rendere gli intervalli di quantizzazione tanto più piccoli quanto più i corrispondenti valori del segnale sono probabili: nel caso del segnale telefonico, si può ad esempio far crescere esponenzialmente l'ampiezza degli intervalli man mano che ci si allontana dallo zero.

Con questo tipo di quantizzazione, il risultato è sostanzialmente quello di determinare *eventi* più o meno equiprobabili, dove per *evento* intendiamo il fatto che un campione cada in un determinato intervallino. Di conseguenza, all'uscita del quantizzatore abbiamo sequenze di bit che sono sostanzialmente equiprobabili, per cui la loro lunghezza potrà essere costante e la velocità di emissione di bit (**bit rate**) sarà anch'essa costante nel tempo.

Se, invece, si sceglie una quantizzazione uniforme, gli *eventi*, nel senso definito poco fa, non sono più equiprobabili: gli eventi relativi a piccoli valori del segnale saranno più probabili degli eventi relativi ad alti valori del segnale. Allora, anziché usare una lunghezza costante per le sequenze di bit (nel qual caso si otterrebbe un bit rate troppo alto, ad esempio i già citati 96 kbit/sec per il segnale telefonico), si può pensare di usare una lunghezza variabile: ai valori più piccoli del segnale (eventi più probabili) andranno assegnate configurazioni di bit più corte, mentre invece ai valori più alti (eventi meno probabili) saranno assegnate configurazione più lunghe. In questo caso, quindi, il *bit rate* non sarà sempre uniforme.

La possibilità che il bit rate non sia uniforme crea dei problemi: infatti, ogni canale di trasmissione è ottimizzato per trasmettere a velocità sostanzialmente fissa, ad esempio a 64 kbit/sec; di conseguenza, *è opportuno cercare di regolarizzare il flusso delle informazioni generate, possibilmente imponendo che venga raggiunta la massima velocità di trasmissione consentita dal canale stesso.* Questo implica sostanzialmente due cose sul funzionamento del quantizzatore:

- quando il quantizzatore prende a generare bit a velocità superiore a quella consentita dalla trasmissione, bisogna forzarlo a ridurre la velocità, riducendo inevitabilmente anche la qualità;
- quando, invece, il quantizzatore genera bit a velocità inferiore a quella massima, allora è opportuno introdurre dei bit di ridondanza.

In generale, quindi, *in funzione dei bit che sono disponibili o dei bit che mancano, bisogna forzare dei modi di funzionamento a qualità più elevata o a qualità più bassa, in modo che il flusso medio di generazione sia rigorosamente pari al flusso medio di trasmissione.*

Possiamo fare una ulteriore osservazione, sempre riguardo il caso di un bit rate non uniforme: perché una trasmissione numerica con messaggi di lunghezza diversa possa funzionare, è chiaramente necessario che ogni messaggio contenga, oltre ai dati veri e propri, anche le informazioni circa la propria lunghezza[8]; solo così, infatti, il ricevitore è in grado di distinguere un messaggio da un altro; allora, se si dovesse verificare qualche errore su un dato messaggio, potrebbero esserci conseguenze anche molto gravi: infatti, se l'errore avviene proprio sui bit che indicano la lunghezza del messaggio, il ricevitore, che non ha modo di accorgersi dell'errore, interpreta in modo sbagliato la lunghezza stessa, per cui, da quel punto in poi, non è più capace di distinguere una parola di codice dalla successiva. In realtà, l'errore non si propaga all'infinito, in quanto, come detto in precedenza, si fa in modo che il ricevitore abbia una memoria finita, per cui, dopo un certo tempo, esso riuscirà comunque a riagganciarsi. Ad esempio, per ottenere dei periodici riassestamenti del sistema si può pensare di trasmettere alcuni campioni non in modo differenziale.

Riepilogo sulle tecniche PCM

Riassumendo, *la codifica PCM differenziale (DPCM), la modulazione delta (DPCM con quantizzazione ad 1 solo bit), la modulazione delta-adattativa (ADPCM con quantizzazione ad 1 solo bit), sono tutti sistemi che tentano di seguire la forma d'onda da trasmettere.* Il meno costoso di essi, cioè la modulazione delta (eventualmente adattativa), non può scendere al di sotto del singolo bit per campione, che corrisponde quindi ad un bit rate di **8 kbit/sec** per una frequenza di campionamento di 8 kHz.

Ci sono sistemi ancora più sofisticati, dove la riduzione di ridondanza è portata all'estremo, che consentono di ottenere un segnale ancora intelligibile con bit rate di soli **800 bit/sec**.

Se poi si vuole rinunciare a individuare la personalità del parlatore, si può pensare ad un sistema fatto nel modo seguente: tramite un apposito sistema di **riconoscimento del segnale vocale**, la voce del parlatore viene convertita in un testo alfanumerico, contenente appunto quello che è stato detto; si trasmette poi solo questo testo (il che richiede notoriamente una velocità di trasmissione bassissima); in ricezione, si usa un sistema di **sintesi vocale del testo**, tramite il quale la voce viene appunto sintetizzata artificialmente sulla base del testo ricevuto. Un sistema di questo tipo richiede una velocità di trasmissione di **40-50 bit/sec**: si ottiene perciò il massimo risparmio, a prezzo, però, della impossibilità di riconoscere la personalità e lo stato d'animo del parlatore[9].

Osserviamo, inoltre, come traspare dalle considerazioni dei paragrafi precedenti, che le tecniche a riduzione di ridondanza sono particolarmente sensibili nei riguardi degli errori di trasmissione, tanto più quanto minore è il bit rate che si vuole ottenere. Per questo motivo, *le tecniche a riduzione di ridondanza, dopo aver ottenuto un abbassamento del bit rate in base ai discorsi dei paragrafi precedenti, provvedono ad alzare leggermente tale bit rate introducendo i codici per il controllo degli errori.* In questo modo, si ottiene un valido compromesso tra velocità di trasmissione richiesta e qualità del servizio: esistono ad esempio codificatori ADPCM normalizzati ed utilizzati con bit rate di **32 kbit/sec**, con un risparmio quindi di un fattore 2 rispetto ai codificatori PCM (64 kbit/sec).

[8] Come si può ricordare dal corso di Teoria dei Segnali, non è necessario che il generico gruppo di bit contenga proprio il valore della propria lunghezza; al contrario, per distinguere una parola di codice da un'altra, di lunghezza diversa, basta imporre che nessuna parola di codice costituisca parte iniziale di un'altra (**codifica di Huffman**).

[9] Sostanzialmente, il sistema appena descritto è l'opposto di un sistema di codifica cosiddetto trasparente, in cui la forma d'onda è trasmessa fedelmente al ricevitore: ciò significa che il segnale vocale, di banda approssimativamente 15 kHz, viene campionato, ad esempio, a 40 kHz e quantizzato con 16 bit per campione. La qualità del servizio è ottima, a spese del bit rate, che in questo caso vale 40*16=**640 kbit/sec**. Tale bit rate è paragonabile a quello necessario per la trasmissione del segnale musicale.

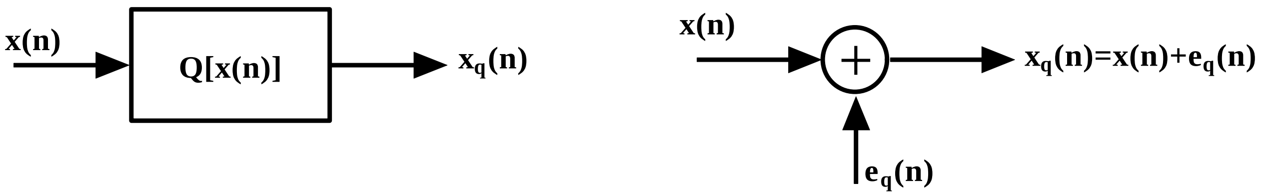
A questo proposito, in tutti i sistemi descritti la qualità del servizio è misurata in termini di scostamento tra forma d'onda trasmessa e forma d'onda originale: si fa in modo che questo scostamento rimanga sempre al di sotto del limite di accettabilità dell'utente.

Complementi sulla codifica DPCM

ANALISI DEGLI ERRORI DI QUANTIZZAZIONE

Per determinare gli effetti di quantizzazione sulle prestazioni di un convertitore A/D, dobbiamo necessariamente adottare un approccio di tipo statistico: infatti, la dipendenza dell'errore di quantizzazione dalle caratteristiche del segnale di ingresso e la natura non lineare del quantizzatore stesso rendono inattuabile una analisi di tipo deterministico, tranne in alcun casi particolarmente semplici.

Nell'adottare un approccio statistico, assumiamo che l'errore di quantizzazione sia casuale e lo modelliamo tramite un rumore che risulta sommato al segnale originale (cioè non quantizzato), dando il segnale finale quantizzato. La figura seguente illustra questo modello:



Dispositivo fisico ***Modello matematico***

Questa figura mostra che il segnale $x_q(n)$ in uscita dal quantizzatore è rappresentabile come somma del segnale utile x(n) (uguale a quello in ingresso) e del rumore di quantizzazione $e_q(n)$ (che rende l'uscita inevitabilmente diversa dall'ingresso).

Possiamo subito fare una considerazione: se il segnale analogico di ingresso è all'interno del range di ingresso del quantizzatore, l'errore di quantizzazione è necessariamente limitato in ampiezza: per essere precisi, se indichiamo con $e_q(n)$ tale errore, se supponiamo di aver adottato una quantizzazione uniforme e se indichiamo con Δ l'ampiezza (costante) degli intervalli di quantizzazione, possiamo evidentemente scrivere che

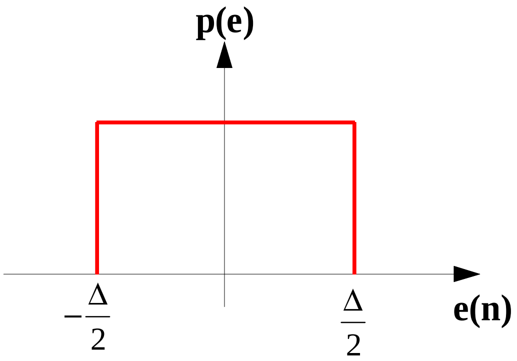
$$|e_q(n)| \le \frac{\Delta}{2} \qquad \forall n$$

In questa ipotesi, l'errore risultante è detto **rumore granulare**.

L'altra possibile situazione è invece quella in cui l'ingresso del quantizzatore presenta valori anche al di fuori della dinamica di ingresso: in questo caso, l'errore $e_q(n)$ non sarà più limitato in ampiezza e si parla allora di **overload noise** (ossia *rumore da sovraccarico*). Questo tipo di rumore può determinare varie distorsioni sul segnale: l'unico rimedio possibile è quello di scalare opportunamente il segnale prima di mandarlo al quantizzatore, in modo che il suo range dinamico si avvicini il più possibile a quello di ingresso del quantizzatore o, ancora meglio, sia contenuto in esso (nel qual caso avremo solo rumore granulare, come detto). Nell'analisi che seguirà, faremo l'ipotesi che non ci sia *overload noise*.

Per semplificare la nostra analisi, facciamo le seguenti ipotesi circa le proprietà statistiche del segnale $e_q(n)$:

- in primo luogo, oltre a supporre che ci sia solo rumore granulare, per cui $|e_q(n)| \le \Delta/2$, supponiamo anche che $e_q(n)$ sia uniformemente distribuito nell'intervallo $[-\Delta/2,\Delta/2]$; ciò significa che, in tale intervallo, risulta uniformemente distribuita la funzione *densità di probabilità dell'errore di quantizzazione*, come indicato nella figura seguente:



- in secondo luogo, supponiamo che la sequenza $e_q(n)$ sia una sequenza di rumore bianco stazionario: questo significa, in altre parole, che ciascun campione è incorrelato con tutti gli altri, ossia anche che la funzione di autocorrelazione corrisponde ad un impulso piazzato nell'origine;
- supponiamo inoltre che $e_q(n)$ sia anche incorrelata con la sequenza di ingresso x(n);
- infine, supponiamo che il segnale x(n) sia a media nulla e sia stazionario (così come $e_q(n)$).

Queste ipotesi non sempre sono verificate. In generale, possiamo dire che lo sono tanto più quanto più il passo di quantizzazione Δ è piccolo e quanto più il segnale x(n) attraversa vari livelli di quantizzazione tra due campioni successivi (ossia sostanzialmente quanto più brusche sono le variazioni temporali del segnale).

Sotto queste ipotesi, l'effetto del rumore $e_q(n)$ che si somma al segnale può essere notoriamente quantificato tramite il rapporto segnale/rumore, eventualmente espresso in unità logaritmiche:

$$\mathrm{SQNR} = \left.\frac{S}{N}\right|_{dB} = 10\log_{10}\frac{S}{N} = 10\log_{10}\frac{P_X}{P_N}$$

In questa relazione, **SQNR** sta per *Signal-to-Quantization Noise Ratio* e corrisponde appunto al rapporto tra la potenza P_X del segnale e la potenza P_N del rumore di quantizzazione. Avendo a che fare con segnali aleatori, sappiamo che tali potenze vanno misurate in termini di valore quadratico medio, ossia di varianza:

$$\mathrm{SQNR} = 10\log_{10}\frac{\sigma_X^2}{\sigma_N^2} = 10\log_{10}\frac{E[x^2(n)]}{E[e_q^2(n)]}$$

A questo punto, mentre la potenza del segnale utile x(n) deve essere valutata caso per caso, possiamo dire qualcosa in più a proposito del rumore: abbiamo infatti supposto che la sua densità di probabilità sia uniformemente distribuita sull’intervallo [-Δ/2,Δ/2]; applicando allora la definizione di valore quadratico medio e considerando che si suppone anche che il processo di rumore sia a media nulla, possiamo scrivere che

$$P_N = \sigma_N^2 = \int_{-\Delta/2}^{\Delta/2} e^2 p(e)de = \frac{1}{\Delta}\int_{-\Delta/2}^{\Delta/2} e^2 de = \frac{\Delta^2}{12}$$

Abbiamo dunque trovato il noto valore **$\Delta^2/12$** della potenza del rumore di quantizzazione, valido per una quantizzazione uniforme e per una p(e) uniformemente distribuita.

Sostituendo questa espressione in quella del rapporto S/N, otteniamo

$$SQNR = 10\log_{10}\frac{\sigma_X^2}{\sigma_N^2} = 20\log_{10}\frac{\sigma_X}{\sigma_N} = 20\log_{10}\frac{\sigma_X}{\Delta/\sqrt{12}} = 20\log_{10}\frac{\sigma_X}{\Delta/2\sqrt{3}}$$

Questa espressione può essere combinata con quella che quantifica Δ: infatti, sappiamo che, se R è l’ampiezza del range di ingresso del quantizzatore e si divide tale range in 2^{b+1} intervalli di quantizzazione (cioè si quantizza con b+1 bit per campione), allora si può chiaramente scrivere che $\Delta = R/2^{b+1}$. Sostituendo in SQNR, otteniamo

$$SQNR = 20\log_{10}\frac{\sigma_X}{\frac{R}{2^{b+1}}\cdot\frac{1}{2\sqrt{3}}} = 20\log_{10}\frac{\sigma_X}{R\cdot\frac{1}{2^{b+2}\sqrt{3}}} = 20\log_{10}\left(2^{b+2}\sqrt{3}\right) - 20\log_{10}\frac{R}{\sigma_X}$$

Separando, nel primo logaritmo, il termine dipendente da *b*, si conclude che

$$\boxed{SQNR = 6.02\cdot b + 16.81 - 20\log_{10}\frac{R}{\sigma_X}}$$

Questa formula mostra chiaramente quanto le prestazioni del quantizzatore dipendano, a parità di deviazione standard s_X del segnale, dal range di ingresso R del quantizzatore e dal numero di bit usati per la quantizzazione.

Per avere una idea concreta, supponiamo che il segnale di ingresso x(n) abbia una distribuzione gaussiana a media nulla e deviazione standard σ_X; supponiamo inoltre di utilizzare un quantizzatore il cui range di ingresso valga 6 volte la varianza di x(n), il che significa che tale range è $[-3\sigma_X,+3\sigma_X]$, per cui la sua ampiezza è $R=6\sigma_X$. Sostituendo, otteniamo

$$SQNR[dB] = 6.02\cdot b + 16.81 - 20\log_{10}\frac{6\sigma_X}{\sigma_X} = 6.02\cdot b + 1.25 \cong 6.02\cdot b$$

Questa formula è spesso utilizzata per specificare la precisione richiesta dal quantizzatore: *essa dice semplicemente che ogni incremento di 1 bit per la quantizzazione corrisponde ad un incremento del rapporto S/N di circa **6 dB**.*

E’ ovvio che questo è un risultato non proprio generale, date le numerose ipotesi fatte. A causa delle limitazioni fisiche dei quantizzatori reali, le prestazioni reali sono peggiori di quelle teoriche fornite da una simile formula: in generale, possiamo dire che il numero di bit effettivi da usare per quantificare la precisione del dispositivo è minore del numero nominale. Per esempio, un quantizzatore nominalmente a 16 bit fornisce una precisione reale corrispondente a 14 bit di quantizzazione.

CONVERTITORI A/D A SOVRACAMPIONAMENTO

*Il concetto di fondo dei convertitori A/D cosiddetti **a sovracampionamento** è quello di aumentare la frequenza di campionamento del segnale fino al punto in cui anche una bassa risoluzione del quantizzatore risulta sufficiente per ottenere la qualità desiderata.* Infatti, se sovracampioniamo un segnale, di fatto riduciamo la variazione del segnale stesso tra ogni coppia di campioni successivi e questo ovviamente abbassa i requisiti di risoluzione richiesti al successivo quantizzatore. Ce ne accorgiamo facilmente con la formula, trovata nel paragrafo precedente, circa il rapporto S/N all'uscita del quantizzatore:

$$\mathrm{SQNR} = 10\log_{10}\frac{\sigma_X^2}{\sigma_N^2} = 20\log_{10}\frac{\sigma_X}{\Delta/\sqrt{12}} = \ldots = 6.02\cdot b + 16.81 - 20\log_{10}\frac{R}{\sigma_X}$$

dove ricordiamo che $\Delta = R/2^{b+1}$.

Se il range dinamico del segnale x(n), che è ovviamente proporzionale alla deviazione standard σ_X del segnale stesso, approssima il range R del quantizzatore, segue che Δ è proporzionale a σ_X. Allora, fissato il numero di bit di quantizzazione, la potenza σ_N^2 del rumore di quantizzazione è proporzionale alla varianza del segnale x(n); di conseguenza, per un fissato valore di SQNR, una riduzione della varianza ci permette anche di ridurre il numero di bit del quantizzatore; oppure, ciò che è lo stesso, una riduzione della varianza, a parità di numero di bit di quantizzazione, dà un più alto valore di SQNR, ossia una migliore qualità della conversione.

CODIFICA DIFFERENZIALE E MODULAZIONE DELTA

Proprio nell'ottica di ridurre al minimo il range dinamico del segnale in ingresso al quantizzatore si innesta il discorso della **codifica differenziale**. Infatti, se consideriamo il segnale d(n) ottenuto come differenza tra un campione x(n) ed il precedente, otteniamo un segnale con varianza estremamente ridotta:

$$\begin{aligned}
&d(n) = x(n) - x(n-1)\\
&\sigma_d^2 = E\left[d^2(n)\right] = E\left[(x(n)-x(n-1))^2\right] = E\left[x^2(n)\right] + E\left[x^2(n-1)\right] - 2E\left[x(n)x(n-1)\right] =\\
&\quad = 2R_{XX}(0) - 2R_{XX}(1) = 2\sigma_X^2 - 2R_{XX}(1) = 2\sigma_X^2\left[1 - \frac{R_{XX}(1)}{\sigma_X^2}\right] = 2\sigma_X^2\left[1-\rho_{XX}(1)\right]
\end{aligned}$$

dove $\rho_{XX}(1)$ è il coefficiente di autocorrelazione di x(n) calcolato per ritardo unitario.

In base a questa espressione, se $\rho_{XX}(1)$ è superiore a 0.5, otteniamo che σ_d^2 è inferiore a σ_X^2. Di conseguenza, possiamo pensare di quantizzare d(n) anziché x(n), a patto ovviamente di poter risalire successivamente ad x(n) tramite i valori quantizzati $d_q(n)$.

Richiedere la condizione $\rho_{XX}(1) > 0.5$ *significa richiedere una elevata correlazione tra campioni successivi del segnale e questo lo si può ottenere, a parità di segnale, solo aumentando la frequenza di* *campionamento, cioè ravvicinando gli istanti in cui tali campioni vengono prelevati.*

Un approccio ancora migliore consiste nell'introdurre una costante di proporzionalità *c* che pesi il campione x(n-1) precedente quello attuale x(n): in questo caso, abbiamo che

$$\begin{aligned}
&d(n) = x(n) - c \cdot x(n-1) \\
&\sigma_d^2 = E\left[d^2(n)\right] = E\left[\left(x(n) - c \cdot x(n-1)\right)^2\right] = E\left[x^2(n)\right] + c^2 E\left[x^2(n-1)\right] - 2cE\left[x(n)x(n-1)\right] = \\
&\quad = R_{XX}(0) + c^2 R_{XX}(0) - 2cR_{XX}(1) = \left(1+c^2\right)\sigma_X^2 - 2cR_{XX}(1) = \sigma_X^2\left[\left(1+c^2\right) - 2c\frac{R_{XX}(1)}{\sigma_X^2}\right] = \\
&\quad = \sigma_X^2\left[\left(1+c^2\right) - 2c\rho_{XX}(1)\right]
\end{aligned}$$

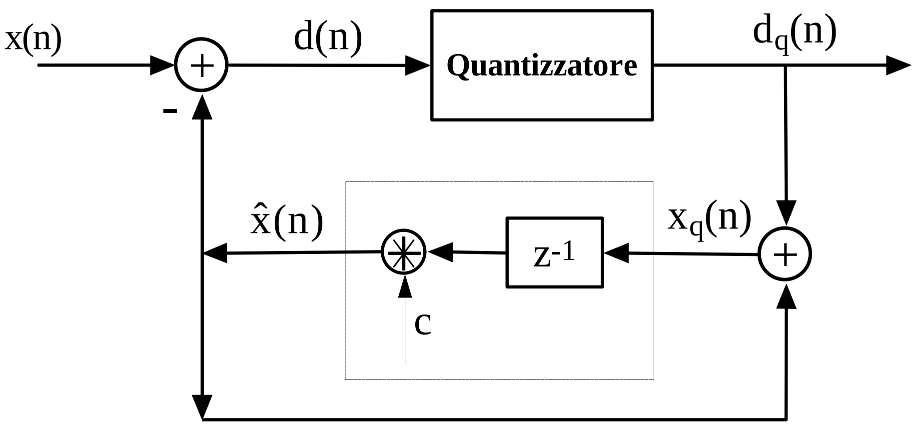
In questo caso, il valore ottimale della costante di peso, cioè quello che minimizza la varianza del segnale d(n), risulta evidentemente $c = \frac{\rho_{XX}(1)}{\sigma_X^2}$.

Come si può notare, il fatto di prendere $d(n) = x(n) - c \cdot x(n-1)$ significa in pratica calcolare la differenza tra il valore vero del campione e una sua predizione lineare ottenuta basandosi solo sul campione precedente: possiamo perciò porre

$$d(n) = x(n) - \hat{x}(n) = x(n) - c \cdot x(n-1)$$

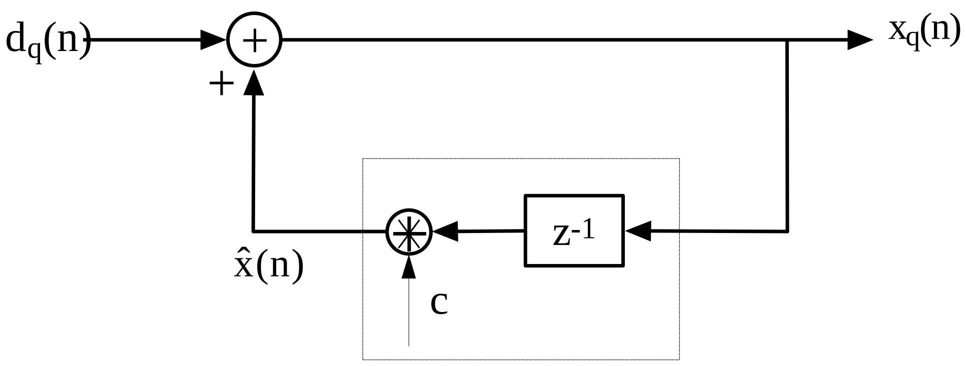
Quanto più la stima è accurata, tanto più piccola sarà la varianza di d(n) e quindi tanto migliori potranno essere le prestazioni ottenute dal quantizzatore.

Lo schema logico di un codificatore che attua questo tipo di codifica è il seguente:



Trasmettitore DPCM

Il ricevitore (o decodificatore) dovrà comportarsi in maniera sostanzialmente duale, per cui sarà fatto nel modo seguente:



Ricevitore DPCM

E' ovvio che questi due schemi possono essere generalizzati sostituendo un generico **predittore lineare di ordine N** al blocco costituito dalla cascata del ritardatore (z^{-1}) e del moltiplicatore, blocco che costituisce un semplice predittore lineare del primo ordine.

La cosa più importante da notare nei due schemi appena riportati (e, in particolare, in quello del trasmettitore) è che la predizione lineare non viene calcolata sulla base di x(n-1), ma sulla base della sua versione quantizzata $x_q(n-1)$:

$$\hat{x}(n) = c \cdot x_q(n-1)$$

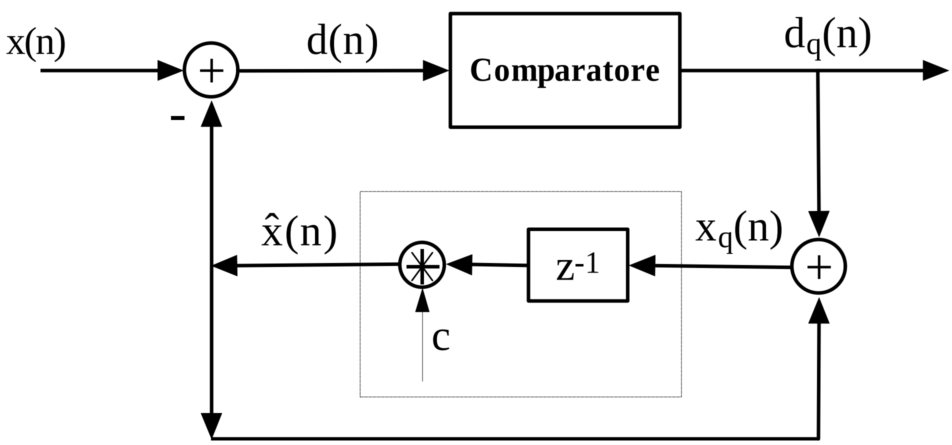
Questo è necessario per garantire che trasmettitore e ricevitore eseguano le predizione nello stesso identico modo e basandosi sugli stessi dati (a meno, ovviamente, di errori di trasmissione).

Facciamo inoltre osservare che l'uso dell'anello di retroazione nel trasmettitore è necessario per evitare che si accumulino gli errori di quantizzazione. In questa configurazione, infatti, l'errore di quantizzazione è

$$e(n) = d(n) - d_q(n) = x(n) - \hat{x}(n) - d_q(n) = x(n) - \left[\hat{x}(n) + d_q(n)\right] = x(n) - x_q(n)$$

Si nota dunque che, *con lo schema proposto, l'errore di quantizzazione commesso sul segnale è esattamente pari a quello commesso sulla differenza d(n).*

La forma più semplice di quantizzazione per d(n) è la cosiddetta **modulazione delta**, che consiste nell'individuare solo il segno della differenza d(n).



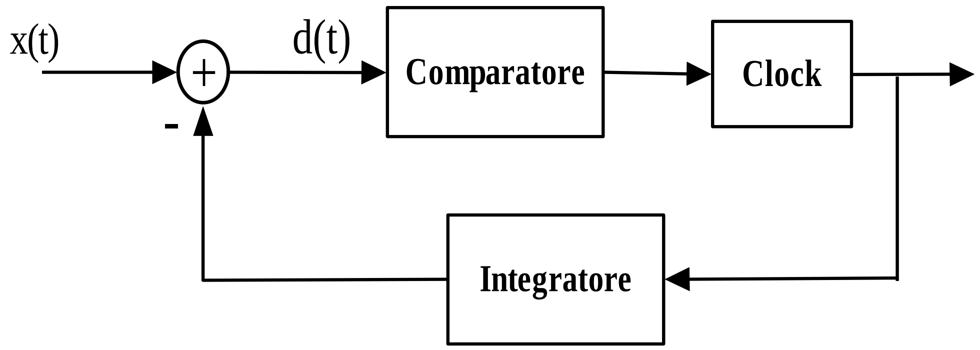
Trasmettitore DPCM per la modulazione delta: al posto del quantizzatore generico, è stato inserito un comparatore, cioè un quantizzatore ad 1 bit

Si osserva immediatamente che

$$x_q(n) = d_q(n) + x(n) = d_q(n) + c \cdot x_q(n-1)$$

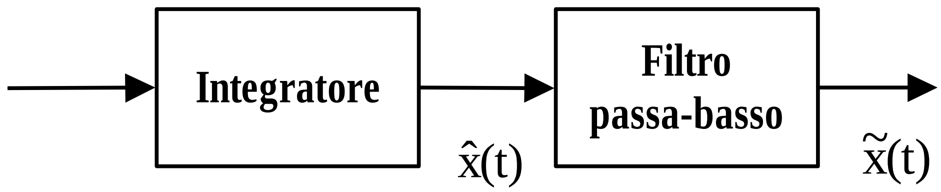
Questa relazione non è altro che l’equivalente tempo-discreto di un integratore analogico. Nel caso particolare di c=1, abbiamo un **integratore ideale**.

Possiamo allora disegnare lo schema teorico di un dispositivo reale che implementi il trasmettitore numerico prima riportato:



Schema semplificato di un dispositivo reale che implementi il trasmettitore per la modulazione delta

Il ricevitore sarà invece fatto nel modo seguente:



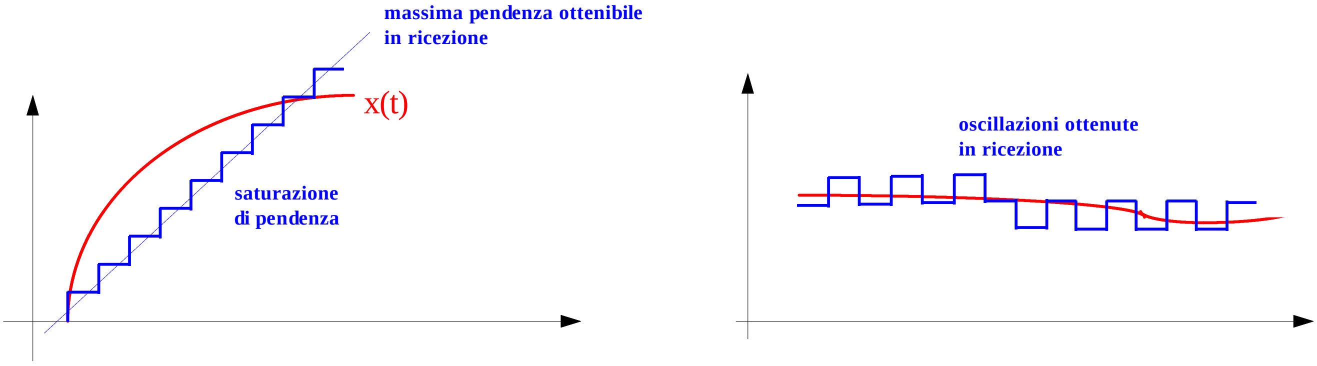
Schema semplificato di un dispositivo reale che implementi il ricevitore per la modulazione delta

Il filtro passa-basso serve per eliminare le componenti fuori banda, ossia comprese nell’intervallo di frequenza tra B (banda di x(t)) e $f_C/2$, dato che f_C>>B a causa del sovracampionamento.

MODULAZIONE SIGMA-DELTA

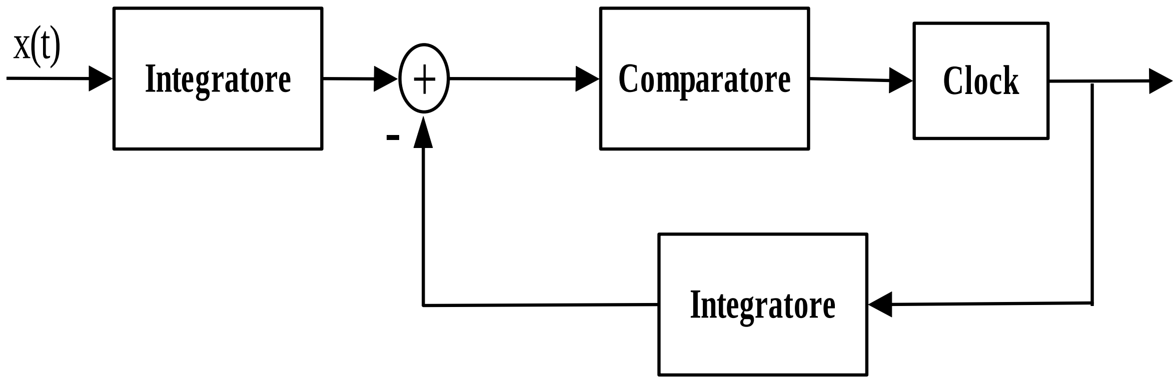
I problemi della *modulazione delta* sono sostanzialmente due, come è noto:

- la distorsione dovuta alla **saturazione di pendenza** (*slope overload*): si ha quando il segnale x(t) di partenza cresce più rapidamente di quanto non possa fare la forma d’onda a scalini che, con la modulazione delta, si ottiene in ricezione;
- la distorsione dovuta al **rumore granulare in zone piatte**: si ha quando il segnale x(t) di partenza rimane sostanzialmente costante, mentre invece la forma d’onda a scalini ottenuta in ricezione presenta delle oscillazioni (pezzi di onda quadra).



La figura a sinistra mostra il fenomeno della saturazione di pendenza, mentre quella a destra mostra il fenomeno del rumore granulare in zone piatte. In entrambi i casi, abbiamo una forma d'onda ricostruita distorta

Un possibile modo di ridurre questi due tipi di distorsioni consiste nel modificare lo schema del trasmettitore nel modo seguente:

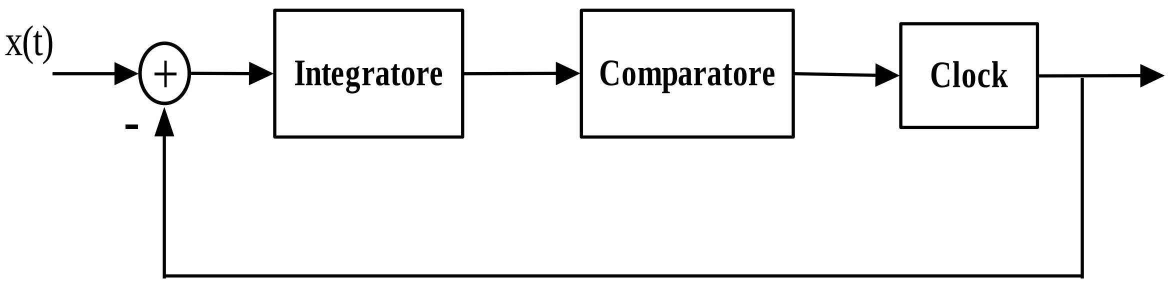


Abbiamo cioè semplicemente spostato, a monte del dispositivo, l'integratore che prima era a valle. Questo comporta sostanzialmente due effetti:

- il primo è quello per cui l'integratore dà maggiore peso alle componenti di bassa frequenza[10] di x(t), il che corrisponde ad aumentare la correlazione del segnale, visto che sono favorite le fluttuazioni più lente (bassa frequenza) rispetto a quelle più veloci (alta frequenza);
- in secondo luogo, risulta anche semplificato lo schema del ricevitore: infatti, se in trasmissione usiamo il segnale integrato come ingresso al codificatore, in ricezione dovremo fare l'operazione duale, ossia una derivazione; ma abbiamo visto prima che il ricevitore della *modulazione delta* presenta un integratore a monte del filtro passa-basso; allora, se aggiungiamo ancora più a monte un derivatore, esso si cancella perfettamente con l'integratore, per cui il ricevitore si riduce banalmente ad un filtro passa-basso.

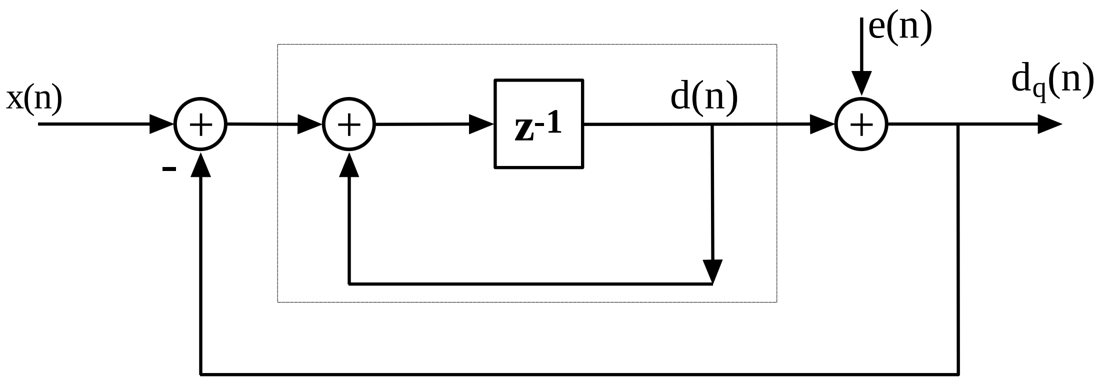
D'altra parte, lo schema del trasmettitore può essere ulteriormente semplificato considerando che i due integratori possono diventare uno solo, purché posto dopo il nodo comparatore:

[10] Ricordiamo che un integratore ideale ha una funzione di trasferimento $1/j\omega$, per cui ha un modulo che vale teoricamente infinito a frequenza 0 e poi va via via diminuendo



Schema implementativo di un ***modulatore sigma-delta***

Per questo dispositivo, possiamo disegnare il seguente modello tempo-discreto equivalente:



Schema tempo-discreto di un modulatore sigma-delta del tipo rappresentato nella figura precedente

Questo dispositivo implementa la cosiddetta **modulazione sigma-delta**, ovviamente in trasmissione (mentre invece in ricezione abbiamo detto che è sufficiente un filtro passa-basso).

Il modello tempo-discreto che abbiamo costruito si basa su due operazioni essenziali:

- in primo luogo, abbiamo modellato il comparatore (cioè il quantizzatore ad 1 bit) semplicemente tramite un rumore additivo e(n), che supponiamo sia bianco con varianza $\sigma_N^2 = \Delta^2/12$; questo in accordo all’analisi precedentemente fatta sul rumore di quantizzazione;
- in secondo luogo, l’integratore che compare prima del comparatore è stato modellato tramite un sistema tempo-discreto (racchiuso nel rettangolo a bordo tratteggiato) avente la seguente funzione di sistema (trasformata zeta della risposta all’impulso):

$$H(z) = \frac{z^{-1}}{1 - z^{-1}}$$

Fatte queste premesse, facciamo una analisi del trasmettitore.

Possiamo ad esempio facilmente calcolare la trasformata zeta della sequenza $d_q(n)$ che viene realmente trasmessa:

$$D_q(z) = \frac{H(z)}{1 + H(z)} X(z) + \frac{1}{1 + H(z)} E(z)$$

Questa formula ci consente di distinguere due funzioni di trasferimento, una per il rumore di quantizzazione e l'altra per il segnale:

$$X(z) \longrightarrow H_S(z) = \frac{H(z)}{1+H(z)} \xrightarrow{z=e^{j\omega T}} H_S(\omega) = \frac{H(\omega)}{1+H(\omega)}$$
$$E(z) \longrightarrow H_N(z) = \frac{1}{1+H(z)} \xrightarrow{z=e^{j\omega T}} H_N(\omega) = \frac{1}{1+H(\omega)}$$

Perché il sistema garantisca una buona qualità di funzionamento, la funzione di trasferimento del segnale $H_S(\omega)$ dovrà essere il più possibile piatta nella banda di interesse, ossia [0,B]. Viceversa, la funzione di trasferimento $H_N(\omega)$ del rumore dovrà garantire la massima attenuazione possibile nella stessa banda, mentre dovrà dare minima attenuazione nella banda $[B,f_C/2]$.

Per un **modulatore sigma-delta del primo ordine**, l'integratore è dato da $H(z) = \frac{z^{-1}}{1-z^{-1}}$, per cui possiamo esplicitare le due funzioni di trasferimento di nostro interesse:

$$\text{segnale} \longrightarrow H_S(z) = \frac{H(z)}{1+H(z)} = z^{-1} \xrightarrow{z=e^{j\omega T}} H_S(\omega) = e^{-j\omega T}$$
$$\text{rumore} \longrightarrow H_N(z) = \frac{1}{1+H(z)} = 1 - z^{-1} \xrightarrow{z=e^{j\omega T}} H_N(\omega) = 1 - e^{-j\omega T}$$

Si nota dunque che $H_S(\omega) = e^{-j\omega T}$ non distorce minimamente il segnale, in quanto ha modulo unitario e fase puramente rettilinea. Di conseguenza, le prestazioni del sistema sono tutte determinate dalla funzione di trasferimento del rumore, $H_N(\omega) = 1 - e^{-j\omega T}$: a tal proposito, possiamo scrivere che

$$H_N(\omega) = 1 - e^{-j\omega T} = e^{-j\omega\frac{T}{2}}\left[e^{j\omega\frac{T}{2}} - e^{-j\omega\frac{T}{2}}\right] = 2je^{-j\omega\frac{T}{2}}\left[\frac{e^{j\omega\frac{T}{2}} - e^{-j\omega\frac{T}{2}}}{2j}\right] = 2je^{-j\omega\frac{T}{2}}\sin\left(\omega\frac{T}{2}\right) = 2je^{-j\omega\frac{T}{2}}\sin\left(\pi\frac{f}{f_C}\right)$$

Questa espressione consente di individuare immediatamente il modulo di questa funzione di trasferimento:

$$|H_N(\omega)| = 2\left|\sin\left(\pi\frac{f}{f_C}\right)\right|$$

Questo modulo ci consente a sua volta di calcolare la varianza del rumore di quantizzazione, calcolata con riferimento alla banda [-B,B] di nostro interesse: infatti, se indichiamo con $S_N(\omega) = \sigma_N^2 / f_C = \Delta^2 / 12 f_C$ la densità spettrale di potenza (per ipotesi costante in frequenza) del rumore di quantizzazione e(n), sappiamo che la densità spettrale di potenza in uscita si ottiene moltiplicando $S_N(f)$ per il modulo quadro di $H_N(\omega)$: quindi

$$S_{N,out}(\omega) = |H_N(\omega)|^2 S_N(\omega) = 2\sin\left(\pi\frac{f}{f_C}\right)\cdot\frac{\Delta^2}{12 f_C} = \frac{\Delta^2}{6 f_C}\sin\left(\pi\frac{f}{f_C}\right)$$

Integrando questa quantità nella banda di interesse, otteniamo la varianza $\sigma^2_{N,out}$ del rumore che, in uscita dal trasmettitore, si trova sovrapposta al segnale: si può allora dimostrare che, per f_C>>B, tale varianza risulta proporzionale alla quantità $\sigma^2_N\left(\frac{B}{f_C}\right)^3$.

Questo risultato mostra, ad esempio, che un eventuale raddoppio della frequenza di campionamento comporta una riduzione della potenza di rumore di 9 dB: 3dB sono dovuti ad una riduzione di $S_N(\omega)=\Delta^2/12f_C$, mentre gli altri 6dB sono dovuti al filtraggio operato da $H_N(\omega)$. Ulteriore abbassamento della potenza di rumore si otterrebbe usando integratori più complessi.

In conclusione, possiamo affermare che, con la modulazione sigma-delta, la potenza di rumore $\sigma^2_{N,out}$ in uscita può essere ridotta incrementando la frequenza di campionamento, in quanto questo consente di distribuire la potenza del rumore di quantizzazione su una banda più larga e quindi di filtrarla maggiormente tramite un apposito filtro.

Autore: **SANDRO PETRIZZELLI**
e-mail: sandry@iol.it
sito personale: http://users.iol.it/sandry
succursale: http://digilander.iol.it/sandry1