

# Campionamento quantizzazione ed elaborazione numerica

SONO esposti i metodi che consentono di rappresentare un segnale per mezzo dei suoi *campioni*<sup>1</sup> temporali, permettendone la codifica in forma numerica: infatti i campioni sono una sequenza di *numeri*<sup>2</sup> e quindi (dopo la *quantizzazione*) *bit*. Le operazioni descritte ai precedenti capitoli e che operano sul segnale  $x(t)$ , possono pertanto essere eseguite direttamente sulla sua rappresentazione numerica, dando luogo alle tecniche indicate come *elaborazione numerica* dei segnali, di cui anche si fornisce qualche accenno, relativo alle rappresentazione frequenziale di sequenze, ed al suo impiego nelle operazioni di filtraggio.

## 4.1 Teorema del campionamento

Esprime la possibilità di ricostruire un segnale *limitato in banda* a partire dai suoi campioni:

*Un segnale con spettro nullo a frequenze maggiori di  $W$  è completamente descritto dai suoi valori prelevati ad intervalli temporali regolari  $t_n = nT_c$ , con  $n$  intero e periodo di campionamento  $T_c \leq \frac{1}{2W}$ ; da questi è possibile risalire ai suoi valori per qualunque altro istante*

La frequenza  $f_{c_{min}} = \frac{1}{T_{c_{Max}}} = 2W$ , chiamata *velocità di Nyquist*<sup>3</sup>, corrisponde alla *minima* velocità con cui occorre campionare un segnale  $x(t)$  limitato in banda, ed è pari al doppio della massima frequenza  $W$  presente nel segnale. Se questa condizione è

<sup>1</sup>Al termine *campione* è associato il *valore* di un segnale ad un determinato istante, e può essere considerato come sinonimo di esemplare, o esempio, ovvero *sample* in inglese; da non confondere con *champion*, o primatista!

<sup>2</sup>*Digits* in inglese, che a sua volta deriva dal latino *digitus*, da cui il termine *digitale* come sinonimo di *numerico*. In effetti il *dito* era una unità di misura utilizzata prima che nell'impero Romano, in Grecia, Egitto e Mesopotamia.

<sup>3</sup>Questo teorema è stato derivato indipendentemente e in tempi diversi da *Borel*, *Whittaker*, *Kotelnikov* e *Shannon*. Il contributo di Nyquist è in realtà relativo al problema di determinare la massima velocità di segnalazione  $f_s$  su di un canale limitato in banda, vedi § 15.2.2.2.

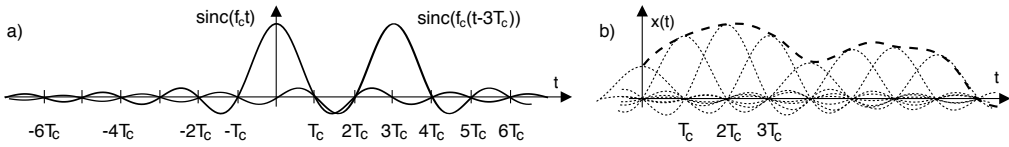


Figura 4.1: a) - Funzione sinc  $(f_c t)$  centrata in  $t = 0$  e traslata in  $t = 3T_c$ ; b) - ricostruzione del segnale limitato in banda mediante la formula (4.1)

rispettata, il segnale originario può essere (ad esempio<sup>4</sup>) ricostruito ricorrendo ad una formula di *interpolazione*<sup>5</sup> (detta *cardinale*) che utilizza i campioni di segnale  $x(nT_c)$  nell'espressione

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT_c) \cdot \text{sinc}(f_c(t - nT_c)) \quad (4.1)$$

che si basa sulla ripetizione ritmica del segnale  $\text{sinc}(f_c t) = \frac{\sin \pi f_c t}{\pi f_c t}$ , detto per questo motivo *seno cardinale* (pag. 1.5.3). Come richiamato in fig. 4.1-a),  $\text{sinc}(f_c t)$  passa da zero per gli istanti  $t = n/f_c = nT_c$ , e dunque sommando i termini  $\text{sinc}(f_c(t - nT_c))$  centrati a multipli di  $T_c$  e con ampiezza  $x(nT_c)$  si ottiene il risultato di 4.1-b), ovvero un segnale che per  $t = nT_c$  vale *esattamente*  $x(nT_c)$ , mentre negli istanti intermedi il valore si forma come *somma* di tutte le “code” dei sinc adiacenti.

Osserviamo ora che la (4.1) può essere realizzata mediante lo schema simbolico mostrato in fig. 4.2, ovvero moltiplicando il segnale  $x(t)$  per un treno di impulsi  $\pi_{T_c}(t)$  con periodo  $T_c \leq 1/2W$ , ed il risultato  $x^\bullet(t)$  fatto passare attraverso un filtro con risposta impulsiva  $h(t) = \text{sinc}(f_c t)$ . Il segnale  $x^\bullet(t) = x(t) \cdot \pi_{T_c}(t)$  è dunque costituito (eq. (3.13)) da impulsi con area pari ai campioni di segnale, ossia

$$x^\bullet(t) = \sum_{n=-\infty}^{\infty} x(nT_c) \delta(t - nT_c)$$

e per ogni impulso  $x(nT_c) \delta(t - nT_c)$  presente in ingresso al filtro si otterrà in uscita una replica della risposta impulsiva centrata sulla posizione dell'impulso, ovvero  $x(nT_c) \cdot \text{sinc}(f_c(t - nT_c))$ , e cioè

$$\begin{aligned} y(t) &= x^\bullet(t) * h(t) = \left[ \sum_n x(nT_c) \delta(t - nT_c) \right] * \text{sinc}(f_c t) = \\ &= \sum_n x(nT_c) \text{sinc}(f_c(t - nT_c)) \end{aligned}$$

che corrisponde alla (4.1). Per dimostrare che il segnale  $y(t)$  così ottenuto eguaglia il segnale originario  $x(t)$ , deriviamo l'espressione di  $X^\bullet(f) = \mathcal{F}\{x^\bullet(t)\}$ . Ricordando il risultato a pag. 81 per la trasformata di un treno di impulsi, possiamo scrivere

<sup>4</sup>Al § 4.2.2 troveremo che in realtà la formula (4.1) non è l'unica possibile.

<sup>5</sup>L'*interpolazione* individua un insieme di metodi per ottenere un segnale che passi per  $N$  punti (istante, valore) prefissati, vedi <https://it.wikipedia.org/wiki/Interpolazione>.

Qualora i punti siano *prelevati* a frequenza  $f_c \geq f_{c_{min}}$  da un segnale limitato in banda, la (4.1) fornisce i suoi valori *esatti* anche per istanti  $t \neq nT_c$ , vedi

[https://it.wikipedia.org/wiki/Formula\\_di\\_interpolazione\\_di\\_Whittaker-Shannon](https://it.wikipedia.org/wiki/Formula_di_interpolazione_di_Whittaker-Shannon).

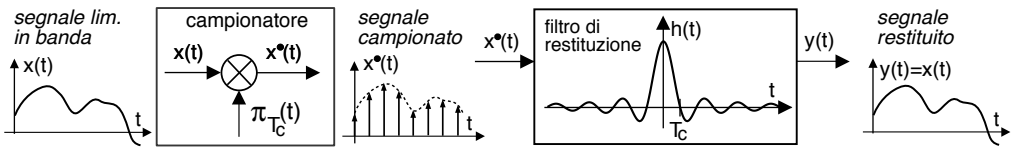
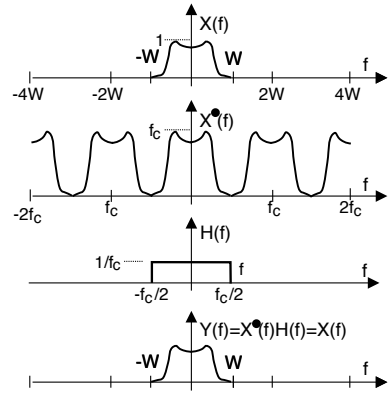


Figura 4.2: Campionamento del segnale  $x(t)$  e sua ricostruzione mediante filtraggio del segnale campionato  $x^*(t)$

$$\begin{aligned} X^*(f) &= \mathcal{F}\{x(t) \cdot \pi_{T_c}(t)\} = X(f) * \frac{1}{T_c} \Pi_{\frac{1}{T_c}}(f) = X(f) * \frac{1}{T_c} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T_c}\right) = \\ &= f_c \cdot \sum_{n=-\infty}^{\infty} X(f) * \delta(f - nf_c) = f_c \cdot \sum_{n=-\infty}^{\infty} X(f - nf_c) \end{aligned} \quad (4.2)$$

dove il penultimo passaggio scambia l'integrale (di convoluzione) di una somma con una somma di integrali, e l'ultimo passaggio tiene conto della proprietà di convoluzione con un impulso (§ 3.4.4).

Lo spettro di  $x^*(t)$  è dunque un segnale *periodico in frequenza* costituito da infinite repliche di  $X(f)$ , centrate a multipli della frequenza di campionamento  $f_c$ , e che *nel caso in cui* si sia scelto  $f_c = 2W$  ovvero pari al suo valore *minimo*, appare come mostrato a lato<sup>6</sup>.



A questo punto osserviamo che al filtro con  $h(t) = \text{sinc}(f_c t)$  corrisponde una risposta in frequenza

$$H(f) = \frac{1}{f_c} \text{rect}_{f_c}(f) \quad (4.3)$$

ovvero quella di un *passa basso ideale*, che permette l'attraversamento delle sole frequenze nell'intervallo  $(-f_c/2, f_c/2)$ , e dunque dell'unica replica spettrale di  $X^*(f)$  centrata in  $f = 0$ . In uscita è quindi presente un segnale  $y(t)$  con spettro di ampiezza

$$Y(f) = H(f) X^*(f) = \frac{1}{f_c} f_c X(f) = X(f)$$

che è perfettamente equivalente al segnale originario, ma ricostruito sulla base dei suoi soli campioni  $x(nT_c)$ . Per questo motivo il filtro  $H(f)$  è anche noto come *filtro di restituzione*.

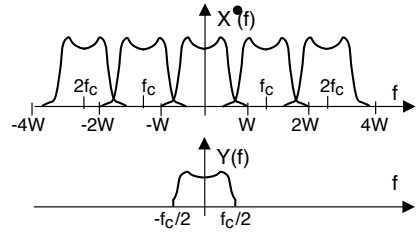
Abbiamo così verificato la correttezza della (4.1) che esprime il teorema del campionamento nella sua forma *cardinale*, ossia quando  $f_c$  è *esattamente* pari a  $2W$ , cioè pari al suo valore minimo. Analizziamo ora cosa accade se la condizione  $f_c \geq 2W$  non è rispettata.

<sup>6</sup>Il risultato ottenuto replica in frequenza quello della trasformata di segnali periodici nel tempo: ad un segnale periodico *in frequenza* con periodo  $f_c$  corrisponde una *antitrasformata* di Fourier costituita da impulsi *nel tempo* distanziati dall'inverso  $T_c = 1/f_c$  del periodo  $f_c$ .

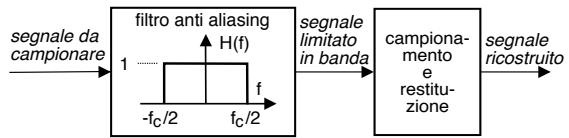
### 4.1.1 Aliasing

Questo termine ha origine dalla parola inglese<sup>7</sup> *alias* (copia, clone) e sta ad indicare il fenomeno che si produce nell'applicare il teorema del campionamento quando i requisiti non sono soddisfatti, e cioè quando la frequenza di campionamento è inferiore alla velocità di Nyquist, ossia  $f_c = \frac{1}{T_c} < 2W$  (ovvero  $T_c > \frac{1}{2W}$ ). In questo caso la (4.2) indica come le repliche spettrali che compongono  $X^*(f)$  siano più ravvicinate, e si sovrappongono, come rappresentato dalla figura a lato: l'aliasing è infatti indicato anche come FOLD-OVER, o *ripiegamento*.

Quando questo avviene il filtro passa-basso di restituzione non è più in grado di estrarre la replica centrata in  $f = 0$ , e dunque il segnale  $y(t)$  alla sua uscita si differenzia da  $x(t)$  in particolare modo per i contenuti energetici nella regione delle frequenze più elevate<sup>8</sup>.



**Filtro anti-aliasing** Il fenomeno dell'aliasing può insorgere, oltre che nel caso in cui si commetta il banale errore di adottare  $f_c < 2W$ , anche a causa di una imperfetta limitazione in banda del segnale da campionare, che infatti viene sempre preventivamente filtrato, in modo da assicurarsi che non contenga componenti a frequenze maggiori della metà di quella di campionamento.



### 4.1.2 Ortogonalità delle funzioni sinc

Si può dimostrare<sup>9</sup> che le funzioni *sinc* costituiscono una base di rappresentazione ortogonale per segnali limitati in banda, in quanto

$$\int_{-\infty}^{\infty} \text{sinc}(f_c(t - kT_c)) \text{sinc}(f_c(t - hT_c)) dt = T_c \cdot \delta(h, k) = \begin{cases} 0 & \text{se } h \neq k \\ T_c & \text{se } h = k \end{cases} \quad (4.4)$$

Pertanto sussiste un *equivalente* del teorema di Parseval (2.3), ed il valore dell'energia di un segnale limitato in banda è calcolabile a partire dai suoi campioni, e vale

$$\begin{aligned} \mathcal{E}_x &= \int_{-\infty}^{\infty} x(t) x^*(t) dt = \sum_k \sum_h x_k x_h^* \int_{-\infty}^{\infty} \text{sinc}(f_c(t - kT_c)) \text{sinc}(f_c(t - hT_c)) dt \\ &= \sum_k \sum_h x_k x_h^* T_c \delta(h, k) = T_c \sum_k |x_k|^2 \end{aligned}$$

<sup>7</sup>In realtà *alias* è di origine latina !!!

<sup>8</sup>In un segnale audio, ad esempio, ci si accorge che c'è aliasing quando è udibile una distorsione (rumore) congiuntamente ai passaggi con maggior contenuto di alte frequenze, vedi ad es.

<https://dspillustrations.com/pages/posts/misc/aliasing-and-anti-aliasing-filter.html>

<sup>9</sup>Applicando il teorema di Parseval (§ 3.2) e la proprietà di traslazione temporale, la (4.4) diviene

$$\int_{-\infty}^{\infty} T_c \text{rect}_{f_c}(f) e^{-j2\pi f k T_c} T_c \text{rect}_{f_c}(f) e^{+j2\pi f h T_c} df = (T_c)^2 \int_{-f_c/2}^{f_c/2} e^{-j2\pi f \frac{k-h}{f_c}} df$$

in cui l'esponenziale complesso sotto integrale compie un numero *intero* di oscillazioni a media nulla per  $f \in [-f_c/2, f_c/2]$  se  $k \neq h$ , e dunque in tal caso l'integrale è nullo; al contrario, l'esponenziale vale 1 se  $k = h$ , ed il suo integrale definito vale  $f_c$ , determinando il risultato mostrato, in cui  $\delta(h, k)$  è il simbolo di *Kronecker*, che vale uno quando  $h = k$  e zero altrimenti.

## 4.2 Aspetti realizzativi del campionamento

Mentre lo schema discusso al § 4.1 illustra gli aspetti teorico-matematici del teorema del campionamento e ne fornisce uno schema operativo di riferimento, nelle implementazioni reali si attuano una serie di stratagemmi volti ad ovviare agli aspetti che in pratica non è possibile realizzare.

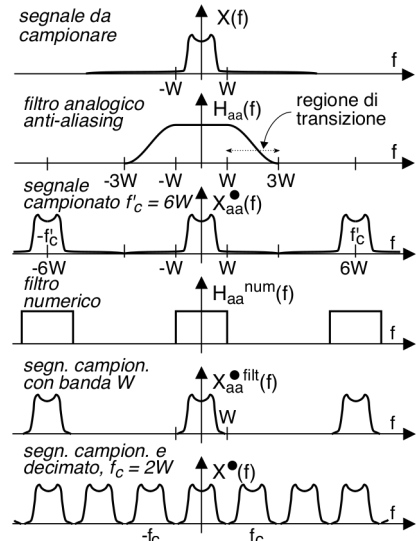
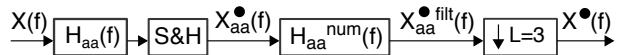
Un primo fattore di criticità sono i due filtri *analogici* previsti, quello *anti-aliasing* e l'altro *di restituzione*. Anticipando parte di ciò che sarà discusso al cap. 5, qualsiasi specifica particolarmente stringente si traduce per essi in un aumento del numero di componenti necessari, e quindi del suo costo, oltre che del suo volume. In particolare è pressoché impossibile realizzare una risposta in frequenza  $H(f)$  esattamente rettangolare, mentre ciò che si può fare è un filtro che presenti una *regione di transizione* (pag. 121) tra banda passante e banda soppressa di ampiezza non nulla, a cui corrisponde una complessità del filtro tanto minore quanto più la regione è estesa.

### 4.2.1 Sovracampionamento e decimazione numerica

Anziché tentare di realizzare un filtro anti-aliasing che lasci passare *intatte* le frequenze inferiori alla metà della  $f_c$  desiderata e blocchi quelle strettamente superiori, se ne adotta uno con una regione di transizione *più graduale* (seconda riga della fig. a lato), dunque più facilmente realizzabile, e si adotta una freq. di campionamento pari al doppio della massima frequenza passante ( $3W$  nell'esempio, e dunque  $f'_c = 6W$ ), producendo un primo segnale campionato  $X_{aa}^*(f)$  privo di aliasing.

I campioni così ottenuti sono quindi elaborati mediante un filtro *numerico*, che come vedremo al § 5.2.1 presenta una  $H_{aa}^{num}(f)$  periodica in frequenza, può realizzare una regione di transizione assai *più ripida*, e soprattutto non altera lo spettro di fase se non per un contributo *lineare* (pag. 239). Il segnale campionato  $X_{aa}^*(f)$  che ne risulta ha ora l'occupazione di banda desiderata di  $W$  (5<sup>a</sup> riga di fig.), e la sua frequenza di campionamento può essere *ridotta* a quella prevista  $f_c = 2W$  operando esclusivamente nel dominio numerico grazie ad uno speciale elemento *decimatore* (§ 5.4.1), che nel nostro esempio provvede a rimuovere due campioni su tre.

La figura a lato riassume la sequenza delle operazioni descritte.



### 4.2.2 Generalizzazione del filtro di restituzione

Anche per questo secondo filtro *analogico* la teoria prevede l'adozione di un passa basso ideale, ed anche per questo caso valgono le obiezioni già discusse, e dunque la

soluzione *semplice* a questo problema è quella di adottare una  $f_c = 2W' > 2W$  di lavoro maggiore della minima, ovvero *sovracampionare* il segnale, in modo che le repliche spettrali siano più distanziate tra loro, e quindi anche un filtro *non ideale* con una regione di transizione non nulla possa ancora isolare l'unica replica di  $X^*(f)$  in banda base, come si osserva in figura 4.3.

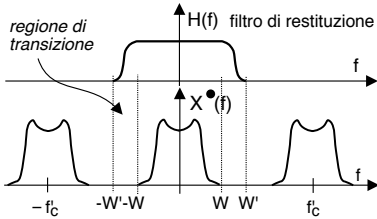
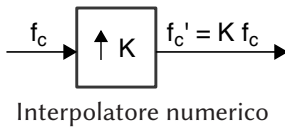


Figura 4.3: Distanziamento delle repliche spettrali mediante innalzamento della  $f_c$

La conseguenza diretta di questo modo di operare è che non essendo più  $H(f)$  rettangolare, la risposta impulsiva di questo nuovo filtro di restituzione non è più un  $\text{sinc}(f_c t)$ ! Ciò significa che in questo caso la formula di interpolazione non ha più l'espressione *cardinale* fornita dalla (4.1), nella quale evidentemente il  $\text{sinc}(f_c t)$  può essere sostituito da una qualunque  $h(t)$  purchè la relativa trasformata  $H(f)$  si annulli *prima di incontrare* una replica spettrale del segnale originario.

### 4.2.3 Interpolazione numerica

Anziché procedere come appena descritto, si è fatta strada una soluzione ancora differente, che evita di sovracampionare il segnale, dato che questo significherebbe appesantire tutte le operazioni da svolgere nel dominio numerico. La frequenza di campionamento  $f_c$  viene invece mantenuta al suo valore minimo consentito dalla selettività del filtro numerico  $H_{aa}^{num}(f)$  discusso al (§ 4.2.1), ma prima di effettuare la restituzione la freq. di campionamento  $f_c$  viene *innalzata* per un fattore intero  $K$ , ovvero  $f'_c = K \cdot f_c$ , ad opera di un apposito *filtro numerico interpolatore* (§ 5.4.2) che ogni due campioni



ne inserisce nel mezzo altri  $K - 1$  con valori *intermedi*. In tal modo il distanziamento mostrato in fig. 4.3 risulta essere ben maggiore, e le specifiche (pag. 121) per il filtro di restituzione divengono ancora meno stringenti.

### 4.2.4 Approssimazione degli impulsi

Un ultimo importante aspetto in cui la pratica differisce dalla teoria è che nella fase di restituzione *non è possibile* generare il segnale  $x^*(t)$  a causa dell'impossibilità di realizzare gli impulsi  $\delta(t)$ ; al suo posto viene prodotto un segnale  $x^o(t)$  (leggi *x cerchiato* anziché *pallino*) mediante l'uso di un circuito *Sample and Hold* (s&H, ovvero *campiona e mantiene*) il cui schema di principio è mostrato in fig. 4.4-a)<sup>10</sup>.

Quando  $s_1$  si chiude il valore di tensione di ingresso viene trasferito ai capi del condensatore, dopodiché  $s_1$  si apre e lo stesso valore viene mantenuto costante per un

<sup>10</sup>Non entriamo nei dettagli del funzionamento del *buffer* (vedi ad es. [https://it.wikipedia.org/wiki/Amplificatore\\_separatore](https://it.wikipedia.org/wiki/Amplificatore_separatore)) qui esemplificato dall'amplificatore operazionale a controreazione unitaria: è sufficiente dire che agisce come un *adattatore di impedenza*, consentendo al condensatore di caricarsi in modo pressoché istantaneo, e di non scaricarsi prima che  $s_2$  sia chiuso, in quanto il secondo amplificatore presenta una impedenza di ingresso pressoché infinita.

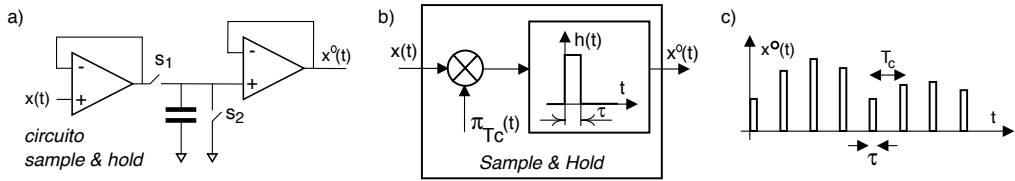


Figura 4.4: a) - realizzazione elettronica di un *sample & hold*; b) - schema simbolico relativo; c) - segnale di uscita

tempo  $\tau \leq T_c$  e reso disponibile in uscita; trascorso il tempo  $\tau$  si chiude  $s_2$ , il condensatore si scarica, e l'uscita si annulla. Alle operazioni appena descritte corrisponde lo schema simbolico illustrato in fig. 4.4-b), per il quale il segnale di uscita è descritto nella forma

$$x^\circ(t) = \sum_n x(nT_c) \cdot \text{rect}_\tau(t - nT_c)$$

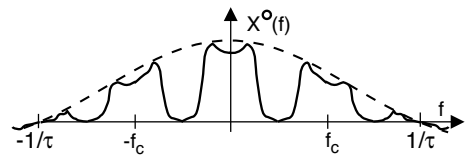
in cui al posto degli impulsi *matematici* viene adottato un treno di impulsi *rettangolari* modulati in ampiezza (vedi fig. 4.4-c)). Il filtro passa basso di restituzione  $H(f)$  viene ora alimentato da  $x^\circ(t)$  anziché da  $x^\bullet(t)$ , e per determinare quale sia la sua nuova uscita, riscriviamo<sup>11</sup>  $x^\circ(t)$  nella forma

$$\begin{aligned} x^\circ(t) &= \sum_n x(nT_c) \cdot \text{rect}_\tau(t) * \delta(t - nT_c) = \\ &= \text{rect}_\tau(t) * \sum_n x(nT_c) \cdot \delta(t - nT_c) = \text{rect}_\tau(t) * x^\bullet(t) \end{aligned} \quad (4.5)$$

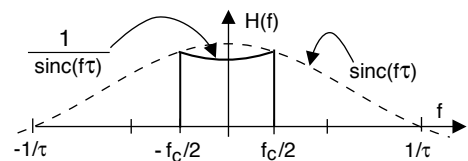
il cui spettro risulta pari a

$$X^\circ(f) = \tau \text{sinc}(f\tau) \cdot X^\bullet(f) \quad (4.6)$$

Osserviamo quindi che usare rettangoli di base  $\tau < T_c$  al posto degli impulsi equivale a moltiplicare  $X^\bullet(f)$  per un involuppo  $\tau \text{sinc}(f\tau)$  che, seppure con  $\tau \ll T_c$  non causa grossi inconvenienti (gli zeri posti ad  $\frac{1}{\tau}$  si allontanano dall'origine, e  $\text{sinc}(f\tau)$  nei pressi di  $f = 0$  è praticamente costante), per  $\tau$  prossimo a  $T_c$  produce una alterazione dell'ampiezza della replica in banda base, ovvero un fenomeno di *distorsione lineare di ampiezza* (§ 8.2).



**Contromisure** Una strada per compensare la distorsione lineare dovuta ai *rect* del s&H si basa sulla conoscenza di  $\tau$  per progettare un filtro di restituzione con una  $H(f)$  che presenta un andamento *reciproco* di quello del  $\text{sinc}(f\tau)$ , in modo che  $H(f) \cdot \tau \text{sinc}(f\tau) = \text{costante}$ . Tale accorgimento prende il nome di *sinc correction*<sup>12</sup>.



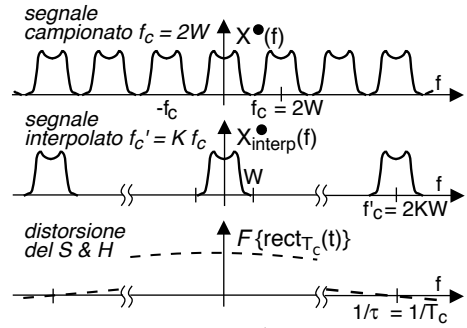
Un'alternativa per ridurre l'entità delle distorsione lineare introdotta dal s&H anche qualora si scelga la durata  $\tau$  dei *rect* pari al periodo di campionamento  $T_c$ <sup>13</sup> passa per

<sup>11</sup> Anche la fig. 4.4-b) conferma l'eq. (4.5)

<sup>12</sup> Vedi ad es. <https://www.maximintegrated.com/en/app-notes/index.mvp/id/3853>

<sup>13</sup> Scegliere  $\tau = T_c$  può semplificare la progettazione del s&H, riducendone i costi.

l'adozione dell'interpolazione numerica descritta al § 4.2.3 ed approfondita al § 5.4.2, attuata con un fattore di interpolazione  $K$  sufficientemente elevato da rendere tale distorsione praticamente trascurabile, come illustrato in figura



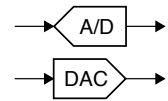
**4.2.5 Conversione A/D e D/A**

Dopo aver discusso gli aspetti teorici e realizzativi della conversione da segnale analogico  $s(t)$  a sequenza di campioni  $s^*(n)$  (vedi fig. 1.2 a pag. 3), per completare il processo di digitalizzazione non resta che occuparci del passaggio da  $s^*(n)$  a segnale quantizzato  $s_q$ , e da questo a sequenza numerica  $s_n$ . Infatti, perché il segnale possa essere memorizzato, trasmesso od elaborato in forma numerica, occorre rappresentare i valori dei campioni  $x(nT_c)$ , che sono grandezze a precisione infinita, mediante un numero finito di  $M$  bit per campione. La fig. 4.5 inquadra questo ultimo passaggio noto come *conversione analogico-digitale* (A/D) nel contesto di quanto discusso finora, rappresentando anche l'operazione inversa (DAC) che ricostruisce il valore analogico (sia pur quantizzato) a partire dalla sua codifica binaria. A lato, i simboli usati per rappresentare tutto l'insieme di operazioni in maniera concisa. Senza alcuna pretesa di completezza<sup>14</sup>, illustriamo ora una possibile implementazione dei due dispositivi citati.

**Quantizzatore A/D a rampa lineare**

Per fissare le idee, descriviamo il funzionamento di uno schema semplice, con l'ausilio della fig. 4.6.

Il valore  $x^\circ(nT_c)$  viene confrontato con il segnale  $s(t)$  prodotto dal generatore di rampa, e finché il secondo non supera il valore del primo, il comparatore emette un livello logico vero (il *gate*) e quindi il segnale di clock in ingresso all'AND si ripresenta in uscita, facendo avanzare il contatore binario ad  $M$  bit. Quando  $s(t)$  supera  $x^\circ(nT_c)$  il *gate* si annulla e l'AND viene interdetto, in modo che il conteggio si arresti. Prima di iniziare un nuovo periodo della rampa il valore presente nel contatore



<sup>14</sup>Per approfondimenti sulla conversione A/D e D/A vedi ad es. [http://sms.unipv.it/misure/ME/Conversione\\_A-D\\_Slides.pdf](http://sms.unipv.it/misure/ME/Conversione_A-D_Slides.pdf)

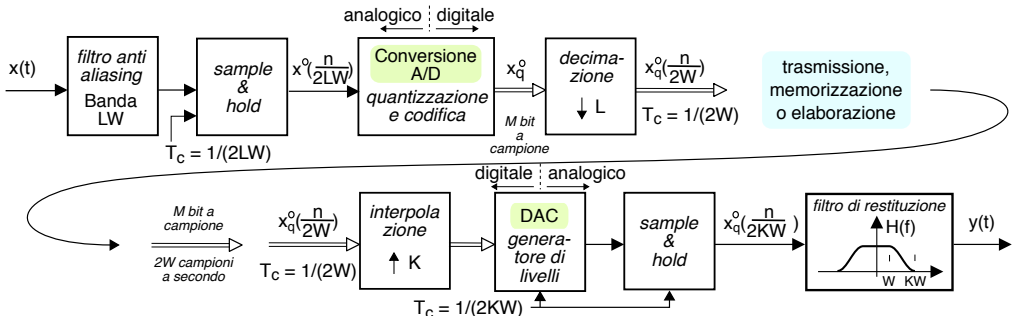


Figura 4.5: Campionamento, quantizzazione e restituzione nel mondo reale

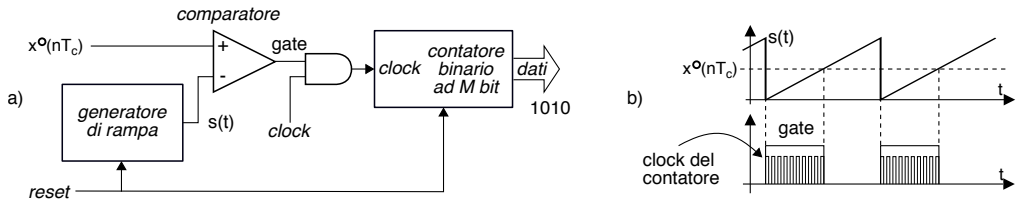


Figura 4.6: Quantizzatore a rampa lineare: a) - schema circuitale; b) - forme d'onda

(pari a *dieci* nell'esempio in figura) viene letto ed adottato come *codifica binaria*  $c(x_q)$  del valore  $x^o(nT_c)$  in ingresso, ed il processo può ripartire per il campione successivo.

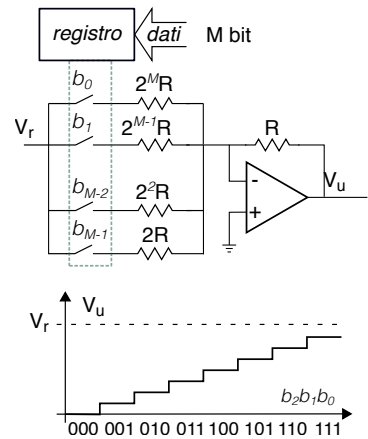
**Convertitore D/A a resistenze pesate** Il generatore di livelli DAC presente in fig. 4.5 consiste in un circuito elettronico in grado di produrre una tensione  $V_u$  proporzionale al valore  $V_b$  codificato da una *parola binaria* ad  $M$  bit

$$V_b = b_{M-1}2^{M-1} + \dots + b_32^3 + b_22^2 + b_12 + b_0$$

posta in ingresso al DAC. Una soluzione *semplice* fa uso di un amplificatore operazionale *sommatore* come in figura, con ciascuna delle resistenze di ingresso  $R_i$  di valore  $2^i R$  ( $i$  da 1 a  $M$ ) collegata o meno alla tensione di riferimento  $V_r$  a seconda se il bit  $b_{M-i}$  sia uno o zero. In uscita sarà presente una tensione<sup>15</sup>

$$V_u = -V_r \sum_{i=1}^M b_{M-i} 2^{-i}$$

in cui cioè il bit più significativo  $b_{M-1}$  pesa per  $1/2$ , quello successivo per  $1/4$ , e così via, dando luogo alla funzione ad  $L = 2^M$  livelli mostrata in figura.



### 4.3 Quantizzazione e codifica binaria

Al di là degli aspetti tecnologici, dal punto di vista dei segnali il processo di quantizzazione e conversione D/A va studiato in relazione all'effetto che la scelta del numero  $M$  di bit/campione produce sulla *qualità* del segnale ricostruito. Maggiore è  $M$ , e più elevata risulta la velocità binaria  $f_b = f_c \cdot M$  bit/sec, a cui corrisponde un maggior impegno di risorse trasmissive o di memoria. Ma riducendo il valore di  $M$  aumenta l'entità dell'errore di quantizzazione  $\varepsilon_q = x - x_q$ , di cui aumenta la potenza, e dunque peggiora il *rapporto segnale rumore*, come ora andiamo a studiare. Notiamo inoltre che essendo la quantizzazione un processo *non lineare* (§ 8.3) la densità spettrale di  $\varepsilon_q$

<sup>15</sup>Per effetto della massa virtuale dell'amplificatore operazionale, in ciascuno dei resistori per cui  $b_{M-i} = 1$  scorre una corrente  $I_i = \frac{V_r}{2^i R}$ , la cui somma  $I_T$  scorre anche nella  $R$  di controreazione, per cui

$$V_u = -R \cdot I_T = -R \sum_{i=1}^M b_{M-i} \frac{V_r}{2^i R} = -V_r \sum_{i=1}^M b_{M-i} 2^{-i}$$

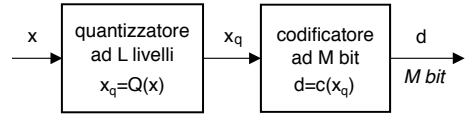
e moltiplicando e dividendo per  $2^M$  si ha  $V_r = -\frac{V_r}{2^M} \sum_{i=1}^M b_{M-i} 2^{M-i}$ , ossia compresa tra 0 e  $-V_r \frac{2^M - 1}{2^M}$ .

A parte il segno meno, ad esempio con  $M = 3$  bit e  $V_r = 10$  si ha un *quanto* di  $10/8 = 1.25$ , e valori 0.0, 1.25, 2.5, ..., 8.75, vedi anche

<http://www.elemania.altervista.org/adda/architetture/arc1.html>

presenta anche frequenze *assenti* nel segnale originario, rendendo *bianco* il processo di errore<sup>16</sup> (§ 7.2.4).

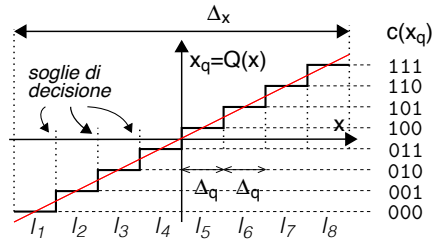
Ai fini della discussione che segue separiamo la funzione svolta da un dispositivo come il quantizzatore a rampa di fig. 4.6 nei due blocchi concettuali disegnati a fianco, in cui i valori dei campioni prima vengono *quantizzati* dal dispositivo  $x_q = Q(x)$ , e successivamente espressi (o *codificati*) mediante una sequenza di  $M$  bit  $d = c(x_q)$ . Il tipo di codifica  $c(x_q)$  realizzato dal quantizzatore a rampa prende il nome di...



### 4.3.1 Quantizzazione uniforme

In questo caso l'intervallo dei valori che l'ingresso  $x$  può assumere, indicato come *dinamica*  $\Delta_x = x_{max} - x_{min}$ , viene suddiviso in  $L = 2^M$  intervalli  $I_k$  di eguale ampiezza<sup>17</sup>  $\Delta_q = \frac{\Delta_x}{L}$ , separati da  $L - 1$  *soglie di decisione*, come riportato in figura per  $M = 3$  bit.

Dato che  $L$  è pari<sup>18</sup>, la curva di quantizzazione  $x_q = Q(x)$  che associa il valore  $x_q(k)$  a tutti gli  $x$  che ricadono in uno stesso intervallo  $I_k$  risulta *a scaletta* come in figura, e fornisce valori quantizzati<sup>19</sup>



$$x_q = \Delta_q \cdot \left( \left\lfloor \frac{x}{\Delta_q} \right\rfloor + \frac{1}{2} \right)$$

che rispetto alla *linea rossa* (che esprime la retta  $x_q = x$ ), sono *esatti* solo *a metà* dell'intervallo, mentre all'interno dello stesso l'errore  $\varepsilon_q = x - x_q$  si mantiene compreso tra  $-\Delta_q/2$  e  $\Delta_q/2$ . Dato poi che con  $M$  bit si possono descrivere le  $L = 2^M$  configurazioni binarie corrispondenti a numeri compresi da 0 a  $2^M - 1$ , sulla destra è riportata la corrispondenza dei codici  $d = c(x_q)$  che è possibile assegnare ai livelli quantizzati.

Per proseguire con l'analisi della distorsione, aggiungiamo ora l'ipotesi che il campione  $x$  da quantizzare sia stato prelevato da un membro di processo *stazionario ergodico*<sup>20</sup> con d.d.p. uniforme a media nulla, ovvero pari a  $p_X(x) = \frac{1}{\Delta_x} \text{rect}_{\Delta_x}(x)$ .

<sup>16</sup>In realtà per valori particolarmente bassi di  $M$  il segnale di errore  $\varepsilon_q$  tende a divenire fortemente correlato (§ 7.1.4) a quello del segnale originale, vedi ad es.

<https://dspillustrations.com/pages/posts/misc/how-does-quantization-noise-sound.html>

<sup>17</sup>Se invece gli intervalli hanno ampiezze differenti il quantizzatore è detto *non uniforme*, vedi il § 4.3.2.

<sup>18</sup>Il caso di  $L$  pari, diretta conseguenza dell'essere  $L = 2^M$  una potenza di due, è detto *mid-rise* in quanto il grafico  $x = Q(x)$  sale per  $x = 0$ , mentre ad  $L$  dispari (caso *mid-tread*) corrisponde una regola di quantizzazione basata sull'*arrotondamento* di  $x$ , ed esiste un valore quantizzato che esprime un *valore nullo*. Per approfondimenti, vedi ad es.

[https://www.tutorialspoint.com/digital\\_communication/digital\\_communication\\_quantization.htm](https://www.tutorialspoint.com/digital_communication/digital_communication_quantization.htm)

<sup>19</sup>La notazione  $\left\lfloor \frac{x}{\Delta_q} \right\rfloor$  individua un *troncamento*, ovvero il numero intero subito inferiore ad  $\frac{x}{\Delta_q}$ . Ad esempio, se  $-4 < x < 4$ , allora avremo  $x_q = -3.5, -2.5, -1.5, -0.5, 0.5, 1.5, 2.5, 3.5$ .

<sup>20</sup>Nel seguito della sezione sono usati i concetti definiti al capitolo 6, a cui si rimanda per le definizioni mancanti.

### 4.3.1.1 SNR di quantizzazione

La distorsione in cui si incorre a seguito del processo di quantizzazione è tanto minore quanto maggiore risulta il valore del *rapporto segnale-rumore*  $SNR_q = \mathcal{P}_x / \mathcal{P}_\epsilon$ , ovvero il rapporto tra la potenza del segnale da quantizzare  $\mathcal{P}_x$ , e quella  $\mathcal{P}_\epsilon$  dell'errore  $\epsilon_q = x - x_q$ . La fig. 4.7 mostra un possibile andamento temporale per  $x(t)$ , assieme alla sua versione quantizzata  $x_q^\circ(t)$ , ed al corrispondente errore  $\epsilon_q(t)$ , per il quale si fa l'ulteriore ipotesi che anch'esso sia un processo ergodico a media nulla, anch'esso descritto da una d.d.p. uniforme  $p(\epsilon) = \frac{1}{\Delta_q} \text{rect}_{\Delta_q}(\epsilon)$  ma con dinamica  $\Delta_q = \Delta_x/L$ , e che sia anche *statisticamente indipendente*<sup>21</sup> da  $x(t)$ .

In queste ipotesi, la potenza  $\mathcal{P}_x$  è pari alla varianza della v.a.  $x$ , e risulta<sup>22</sup>

$$\mathcal{P}_x = \sigma_x^2 = \frac{\Delta_x^2}{12}$$

mentre per quanto riguarda la potenza del segnale di errore  $\epsilon_q(t)$  il risultato è lo stesso, ma espresso nei termini di  $\Delta_q$ , ovvero

$$\mathcal{P}_\epsilon = \sigma_\epsilon^2 = \frac{\Delta_q^2}{12} = \frac{1}{12} \left( \frac{\Delta_x}{L} \right)^2$$

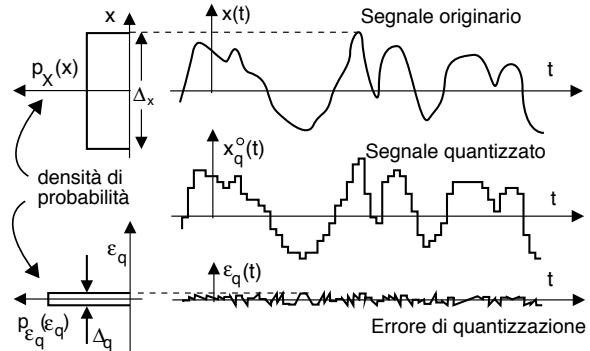


Figura 4.7: Processo di quantizzazione per segnali a distribuzione di ampiezza uniforme

Siamo dunque in grado di valutare l' $SNR$  di quantizzazione come

$$SNR_q = \frac{\mathcal{P}_x}{\mathcal{P}_\epsilon} = \frac{\Delta_x^2}{12} 12 \left( \frac{L}{\Delta_x} \right)^2 = L^2$$

e dunque  $SNR_q$  cresce in modo *quadratico* con l'aumentare dei livelli, ovvero se  $L$  raddoppia  $SNR_q$  quadruplica. Ricorrendo alla notazione in decibel<sup>23</sup> per l' $SNR$ , otteniamo il risultato  $SNR_q(L)|_{dB} = 10 \log_{10} L^2 = 20 \log_{10} L$  e, ricordando che  $L = 2^M$ , si ottiene

$$SNR_q(M)|_{dB} = M \cdot 20 \log_{10} 2 \simeq 6 \cdot M \quad \text{dB} \quad (4.7)$$

dato che  $\log_{10} 2 \simeq 0.3$ . Pertanto è possibile concludere che<sup>24</sup>

<sup>21</sup>Questa ipotesi, come anche quella delle v.a. uniformi, sono manifestamente *non vere* in generale, ma permettono di giungere ad un risultato abbastanza semplice, e che può essere molto utile nei progetti di dimensionamento.

<sup>22</sup>Assumendo che il processo sia ergodico, la potenza (media temporale) eguaglia (eq. (6.15)) la corrispondente media di insieme, ovvero il momento di secondo ordine  $m_x^{(2)}$ , che a sua volta è pari alla varianza  $\sigma_x^2$ , essendo  $m_x = 0$ . Vedi § 6.2.3 per il calcolo di  $\sigma_x^2 = \Delta_x^2/12$ .

<sup>23</sup>Una discussione relativa alla misura delle grandezze in decibel, è fornita al § 8.1. Qui ci limitiamo ad usare i dB come misura relativa di un rapporto, ossia

$$SNR_q(dB) = 10 \log_{10} \frac{\mathcal{P}_x}{\mathcal{P}_\epsilon} = 10 \log_{10} \mathcal{P}_x - 10 \log_{10} \mathcal{P}_\epsilon = \mathcal{P}_x [dBV^2] - \mathcal{P}_\epsilon [dBV^2]$$

in cui le grandezze espresse in  $dBV^2$  rappresentano potenze di segnale di tensione, in unità logaritmiche.

<sup>24</sup>In alcuni testi alla (4.7) viene aggiunto un termine costante di 1.76 dB, derivante dall'adozione di un segnale sinusoidale con dinamica  $\Delta$ , anziché un processo uniforme. Ma non ho mai afferrato il senso di

La qualità del quantizzatore uniforme espressa da  $SNR_q [dB]$  aumenta linearmente con il numero  $M$  di bit/campione, con un incremento di 6 dB per ogni bit utilizzato in più.

Consideriamo ora cosa accade qualora il segnale in ingresso  $x$  abbia una dinamica minore a quanto previsto: in tal caso  $\sigma_x^2$  si riduce, mentre  $\sigma_\epsilon^2 = \frac{1}{12} \left(\frac{\Delta_x}{L}\right)^2$  non cambia, e dunque  $SNR_q$  peggiora, come se avessimo ridotto il numero di livelli. Viceversa, la tecnica illustrata di seguito è in grado di mantenere un  $SNR_q$  accettabile anche con bassi livelli di segnale.

### 4.3.2 Quantizzazione non uniforme

La tecnica di cui al titolo nasce sia come risposta al problema appena evidenziato, ossia il peggioramento di  $SNR_q$  in presenza di bassi segnali, sia in considerazione del fatto che in realtà il processo  $x$  non ha distribuzione di ampiezza uniforme come fino ad ora ipotizzato, rendendo il risultato (4.7) di ridotta utilità pratica. Ma svolgiamo prima una breve digressione su come, conoscendo con esattezza la  $p_X(x)$  del processo da quantizzare, si possa giungere al miglior risultato.

**Minimizzazione dell'errore di quantizzazione** Si può mostrare<sup>25</sup> che per una  $p_X(x)$  generica il quantizzatore ottimo (che rende massimo  $SNR_q$ ) non suddivide la dinamica  $\Delta_x$  in  $L - 1$  intervalli di uguale estensione  $\Delta_q = \frac{\Delta_x}{L}$ , ma l'estensione  $\Delta_q(k)$  varia ed è diversa per ogni intervallo  $I_k$  con  $k = 1, 2, \dots, L$ , adottando in particolare intervalli di estensione ridotta nelle regioni in cui i valori  $x$  sono più probabili, ovvero più frequenti, e dunque per i quali la  $p_X(x)$  è più grande. In tal modo ai valori  $x$  più probabili corrisponde un errore di quantizzazione  $\sigma_q^2(k) = \Delta_q^2(k)/12$  ridotto; viceversa, le regioni corrispondenti a valori di  $x$  più rari adottano intervalli di estensione  $\Delta_q(k)$  più grande, e conseguentemente un  $\sigma_q^2(k)$  maggiore. Il risultato è quello di variare il contributo alla  $\mathcal{P}_\epsilon$  complessiva, riducendolo per i valori più frequenti, ed aumentandolo per quelli meno: infatti,  $\mathcal{P}_\epsilon$  può essere valutata anche come un valore atteso, pesando i contributi  $\sigma_q^2(k)$  con le rispettive probabilità

$$\mathcal{P}_\epsilon = E_k \left\{ \sigma_q^2(k) \right\} = \sum_{k=1}^L p_k \sigma_q^2(k)$$

in cui  $p_k = \int_{I_k} p_X(x) dx$  è la probabilità che  $x \in I_k$ . Il modo ottimo di disporre le soglie  $(\theta_{k-1}, \theta_k)$  che delimitano  $I_k$  in modo da rendere minima  $\mathcal{P}_\epsilon$ , è noto come algoritmo di LLOYD-MAX<sup>26</sup>, ed il risultato che si ottiene è legato alla teoria esposta al § 9.5.

un  $SNR$  positivo con  $M = 0$  bit/campione.

<sup>25</sup>Vedi ad es. [http://en.wikipedia.org/wiki/Quantization\\_\(signal\\_processing\)](http://en.wikipedia.org/wiki/Quantization_(signal_processing))

<sup>26</sup>Il metodo è iterativo, ed inizia suddividendo l'intervallo  $\Delta_x$  in modo uniforme. Per ogni iterazione:

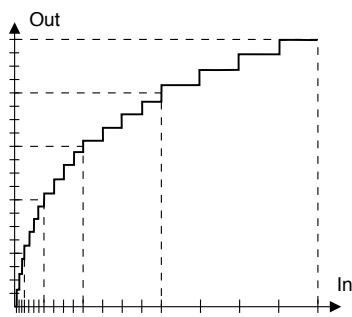
- si determinano i valori quantizzati  $x_k$  (detti centroidi) come  $x_k = E \{x \in I_k\} = \int_{I_k} x \cdot p_X(x/k) dx =$

$$\frac{\int_{I_k} x p_X(x) dx}{p_k}$$

in cui  $p_k = \int_{I_k} p_X(x) dx$ . In tal modo, i valori  $x_k$  si spostano (internamente a  $I_k$ ) verso la regione in cui  $p_X(x)$  ha un valore più elevato, ovvero dove la v.a. si addensa;

- si ri-calcolano i confini di decisione  $\theta_k$  come  $\theta_k = \frac{x_k + x_{k+1}}{2}$ , seguendo lo spostamento degli  $x_k$ .

Le iterazioni si arrestano quando non si riscontrano cambiamenti apprezzabili.



codifica lineare	PCM legge A
s000000wxyz	s000wxyz
s0000001wxyz	s001wxyz
s000001wxyza	s010wxyz
s00001wxyzab	s011wxyz
s0001wxyzabc	s100wxyz
s001wxyzabcd	s101wxyz
s01wxyzabcde	s110wxyz
s1wxyzabcdef	s111wxyz

Figura 4.8: Caratteristica i/o di un quantizzatore logaritmico e relativa tabella di codifica

**Codifica PCM** Nella quantizzazione del segnale vocale, anche se è arbitrario identificare con esattezza una  $p_X(x)$ , si verifica strumentalmente che quest'ultima è addensata nelle regioni con valori più piccoli. Per questo motivo la legge di quantizzazione che si è adottata per ottenere gli 8 bit a campione utilizzati nel PCM<sup>27</sup> segue un andamento logaritmico<sup>28</sup>, e dimezza progressivamente la pendenza della caratteristica di ingresso-uscita del quantizzatore all'aumentare dei valori in ingresso.

La fig. 4.8 mostra un esempio di tale realizzazione (per i soli valori positivi), in cui l'andamento logaritmico viene approssimato individuando (a partire dall'origine) *regioni* di valori della  $x$  la cui ampiezza di volta in volta raddoppia, e suddividendo ogni regione in un uguale numero di intervalli equispaziati. La caratteristica non lineare è realizzabile per via completamente numerica: per prima cosa si realizza un campionamento uniforme con  $M = 12$  bit a campione, a cui corrispondono  $L = 2^{12} = 4096$  livelli. Per ogni campione, il numero di bit pari a zero nella parte più significativa degli  $M$  individua la *regione* dei valori di ingresso, mentre i bit rimanenti sono *shiftati* a destra, per mantenere costante il numero di *intervalli* per regione, ottenendo in definitiva una rappresentazione in *virgola mobile* del valore del campione.

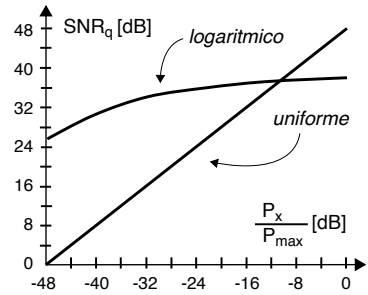
**Esempio** La tabella a destra di fig. 4.8 esemplifica il processo di conversione PCM *legge A*, in cui a partire dalla quantizzazione uniforme a 12 bit (colonna di sinistra) si ottengono gli 8 bit della codifica PCM (colonna di destra), ripartiti in un bit di segno (s), tre bit di esponente, e quattro di mantissa (wxyz). Il bit più significativo dei 12 rappresenta il segno, e resta inalterato. Il numero di zeri più significativi dei 12 individua in quale delle 8 regioni<sup>29</sup> cade il valore di ingresso, e tale regione è codificata mediante i 3 bit del codice PCM che seguono quello del segno. Delle restanti cifre della codifica uniforme, se ne conservano solo le 4 più significative. Ad esempio, il valore 10000001010 diviene 1-000-1010 (come risulta dalla prima riga della tabella), mentre 000110011010 diventa 0-101-1001, come riporta la sesta riga.

<sup>27</sup>La sigla PCM sta per *Pulse Code Modulation*, e trae origine dalla tecnica di quantizzazione di un segnale vocale di qualità telefonica (§ 11.1.2), anche se è stato poi adottato per indicare l'intera gerarchia di moltiplicazione plesiocrona (§ 24.3.1). Etimologicamente il termine deriva dall'*onda PAM* (§ 7.7.4) in cui degli impulsi sono *Modulati in Ampiezza*, mentre in questo caso le ampiezze degli impulsi sono *Codificate*.

<sup>28</sup>L'andamento esatto della curva segue uno di due standard, denominati legge  $\mu$  (per USA e Giappone) e legge A (per gli altri), lievemente diverse nella definizione, ma sostanzialmente equivalenti.

<sup>29</sup>Per motivi grafici, nella parte sinistra della figura sono mostrate solo 5 regioni, divise in 4 intervalli.

I risultati del *mapping* illustrato possono essere inseriti in una ROM come coppie di *ingresso-uscita*, utilizzando dunque 4096 posizioni di memoria da 8 bit ciascuna; al contrario presso il DAC di restituzione, sono sufficienti 256 posizioni, di 12 bit ciascuna. Nella figura che segue si mostra l'andamento di  $SNR_q$  al variare della potenza di segnale rispetto al massimo possibile, nei due casi del quantizzatore uniforme, oppure con compressione logaritmica. Come si può notare, l'effetto della quantizzazione logaritmica penalizza l' $SNR$  per i segnali a piena dinamica, ma peggiora molto più lentamente al diminuire della stessa. Al cap. 10 l'argomento della codifica di sorgente audio viene ripreso, e debitamente approfondito.



Le prossime sezioni non tengono più conto delle questioni relative alla quantizzazione dei campioni, anche se nella realizzazione di dispositivi reali può rivestire interesse di progetto.

#### 4.4 Trasformata di Fourier di sequenze

Avendo illustrato come sia sufficiente la conoscenza dei soli campioni temporali  $x_n = x(nT_c)$  per descrivere completamente un segnale tempo continuo e limitato in banda  $x(t)$ , e come alla sequenza tempo-discreta dei suoi campioni  $x_n$  corrisponda una *periodizzazione* in frequenza  $X^\bullet(f)$ , notiamo come ciò sia in qualche modo *speculare* alla proprietà dei segnali *periodici nel tempo*, che possiedono una rappresentazione equivalente nel dominio della frequenza, costituita dalla sequenza dei coefficienti  $X_n$  noti come serie di Fourier. L'analogia è più stringente di quanto non possa apparire, dato che è assolutamente lecito ed esatto usare l'espressione della serie di Fourier (2.7) nella direzione *opposta*<sup>30</sup>, ossia per ottenere lo spettro periodico di ampiezza  $X^\bullet(f)$  a partire dalla sequenza dei campioni temporali  $x_n$ :

$$X^\bullet(f) = \sum_{n=-\infty}^{\infty} x_n e^{-j2\pi f n T_c} \quad (4.8)$$

definendo così una *trasformata di Fourier a tempo discreto*<sup>31</sup> o **DTFT**, che produce una  $X^\bullet(f)$  periodica in frequenza di periodo  $f_c = \frac{1}{T_c}$ <sup>32</sup>, in cui  $T_c$  è il periodo di campiona-

<sup>30</sup>A prima vista può sembrare *ardito* accettare che i coefficienti di Fourier (4.9) siano pari ai campioni di segnale  $x_n$ , ma se proviamo a calcolare

$$\begin{aligned} X^\bullet(f) &= \mathcal{F}\{x^\bullet(t)\} = \int_{-\infty}^{\infty} \left( \sum_{n=-\infty}^{\infty} x_n \delta(t - nT_c) \right) e^{-j2\pi f t} dt = \\ &= \sum_{n=-\infty}^{\infty} x_n \int_{-\infty}^{\infty} e^{-j2\pi f t} \delta(t - nT_c) dt = \sum_{n=-\infty}^{\infty} x_n e^{-j2\pi f n T_c} \end{aligned}$$

otteniamo esattamente la (4.8).

<sup>31</sup>Condizione sufficiente per la convergenza della serie (4.8) è che risulti  $\sum_{n=-\infty}^{\infty} |x_n| < \infty$ , in quanto

$$|X^\bullet(f)| = \left| \sum_{n=-\infty}^{\infty} x_n e^{-j2\pi f n T_c} \right| \leq \sum_{n=-\infty}^{\infty} |x_n|$$

<sup>32</sup>Infatti se applichiamo la (4.8) per calcolare  $X^\bullet(f + f_c)$  si ottiene

$$\sum_{n=-\infty}^{\infty} x_n e^{-j2\pi(f+f_c)nT_c} = \sum_{n=-\infty}^{\infty} x_n e^{-j2\pi f n T_c} e^{-j2\pi f_c n T_c} = X^\bullet(f)$$

mento con cui sono prelevati i valori  $x_n$ <sup>33</sup>. Alla (4.8) è associata una *antitrasformata*, in grado di valutare i campioni temporali  $x_n$  a partire dalla conoscenza di un periodo di  $X^\bullet(f)$ , definita come

$$x_n = \frac{1}{f_c} \int_{-\frac{f_c}{2}}^{\frac{f_c}{2}} X^\bullet(f) e^{j2\pi f n T_c} df \quad (4.9)$$

e che è del tutto analoga (a parte il segno) alla (2.6) che calcola i coefficienti della serie di Fourier. Molte delle proprietà già note per la serie e la trasformata di Fourier sono valide anche in questo caso, come ad esempio

- simmetria coniugata per  $X^\bullet(f)$  nel caso di  $x_n$  reale;
- linearità:  $ax_n + by_n \iff aX^\bullet(f) + bY^\bullet(f)$ ;
- ritardo:  $x_{n-k} \iff X^\bullet(f) e^{-j2\pi f k T_c}$ ;
- modulazione:  $x_n e^{j2\pi f_0 n T_c} \iff X^\bullet(f - f_0)$ ;
- convoluzione:  $z_n = x_n * y_n = \sum_{k=-\infty}^{\infty} x_k y_{n-k} \iff Z^\bullet(f) = X^\bullet(f) \cdot Y^\bullet(f)$ ;
- prodotto:  $z_n = x_n \cdot y_n \iff Z^\bullet(f) = \frac{1}{f_c} \int_{-f_c/2}^{f_c/2} X^\bullet(\sigma) Y^\bullet(f - \sigma) d\sigma$ .

Ciò permette di svolgere operazioni sui segnali (come analisi spettrale e filtraggio) senza dover svolgere calcoli analitici (integrali e trasformate), bensì operando direttamente sui campioni di segnale mediante appositi algoritmi di calcolo numerico eseguiti su dispositivi ottimizzati a tale scopo<sup>34</sup>, e quindi effettuare il processo di conversione D/A per riottenere un risultato tempo-continuo, come mostrato in fig. 4.9.

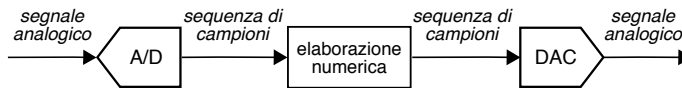


Figura 4.9: Elaborazione numerica di segnale analogico

Resta comunque il fatto che nelle (4.8) e (4.9) compaiono tuttora una somma di infiniti termini ed un integrale di funzione continua, mentre per effettuare le operazioni di elaborazione numerica possono essere usate solo sequenze di durata finita e somme. Per questo motivo affrontiamo la sezione successiva, che illustra come ciò possa essere risolto eseguendo il campionamento anche in frequenza, e limitando i segnali ad una durata limitata.

## 4.5 Trasformata discreta di Fourier

Disponendo di una sequenza di lunghezza *finita* composta da  $N$  valori  $x_n$ ,  $n = 0, 1, \dots, N-1$ , corrispondenti a campioni di un segnale  $x(t)$  prelevati con ritmo  $f_c = \frac{1}{T_c}$ , si indica

dato che, essendo  $f_c = \frac{1}{T_c}$ , risulta  $e^{-j2\pi f_c n T_c} = e^{-j2\pi n} = 1$  per qualsiasi  $n$ .

<sup>33</sup>Proprio come ai coefficienti della serie di Fourier per segnali periodici, intervallati di  $F$  Hz, corrisponde un segnale periodico nel tempo, di periodo  $T = \frac{1}{F}$ .

<sup>34</sup>I chip progettati appositamente per svolgere calcoli di elaborazione numerica del segnale sono detti DSP (*Digital Signal Processor*), che tipicamente eseguono somme di prodotti; Nel caso di dati multidimensionali, sono invece adottate le GPU nate per scopi di accelerazione grafica, vedi ad es. [https://en.wikipedia.org/wiki/Multidimensional\\_DSP\\_with\\_GPU\\_Acceleration](https://en.wikipedia.org/wiki/Multidimensional_DSP_with_GPU_Acceleration)

come DISCRETE FOURIER TRANSFORM (DFT) la nuova sequenza<sup>35</sup>

$$X_m = \sum_{n=0}^{N-1} x_n e^{-j2\pi \frac{m}{N} n} \quad (4.10)$$

univocamente definita per  $m = 0, 1, \dots, N - 1$ , e che costituisce una approssimazione<sup>36</sup> del campionamento in frequenza della trasformata  $X(f) = \mathcal{F}\{x(t)\}$ , calcolata per  $f = \frac{m}{N}f_c$ , e moltiplicata per  $f_c$ :

$$X_m \simeq f_c X\left(f = \frac{m}{N}f_c\right) \quad (4.11)$$

Notiamo subito che la (4.10) è valida per qualsiasi  $m$ , ed ha un andamento periodico con periodo  $N$ , a cui corrisponde la frequenza  $f_c = \frac{1}{T_c}$  (vedi fig. 4.10), in accordo con la periodicità in frequenza evidenziata per  $X^\bullet(f)$  (vedi (4.8) e (4.2)); per questo motivo, qualora il segnale originario  $x(t)$  contenga componenti a frequenze maggiori di  $\frac{f_c}{2}$ , gli  $X_m$  con indici prossimi ad  $\frac{N}{2}$  presenteranno errore di aliasing<sup>37</sup>. Notiamo inoltre come i valori  $X_m$  siano tutti relativi a frequenze  $\geq 0$ , ma nel caso di una sequenza  $x_n$  reale la proprietà di simmetria coniugata, associata alla periodicità in frequenza, rende il risultato interessante solo per indici  $m \leq \frac{N}{2}$ , dato che successivamente si trovano valori coniugati a quelli della prima metà.

<sup>35</sup>La (4.10) può essere fatta discendere dalla (4.8) vincolando  $f$  ad assumere i soli valori discreti  $f = \frac{m}{N} \frac{1}{T_c}$ , e limitando l'indice della sommatoria ad un insieme finito di campioni, vedi fig. 4.10.

<sup>36</sup>Una prima fonte di approssimazione deriva dall'operazione di finestrata legata all'uso di un numero finito di campioni, operando quindi su  $x_w(t) = x(t)w(t_c)$  anziché su  $x(t)$ . Per analizzare le altre fonti di approssimazione, iniziamo a scrivere l'espressione di  $X_w(f) = \mathcal{F}\{x_w(t)\}$  per  $f = \frac{m}{N}f_c$ :

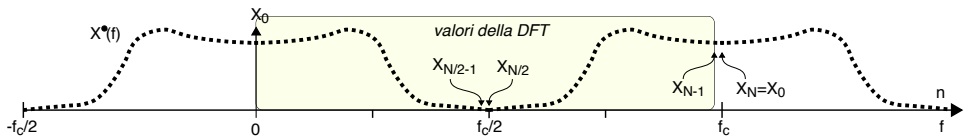
$$X_w\left(f = \frac{m}{N}f_c\right) = \int_0^{(N-1)T_c} x(t) e^{-j2\pi \frac{m}{N}f_c t} dt \simeq \sum_{n=0}^{N-1} x_n \cdot \int_0^{(N-1)T_c} \text{sinc}(f_c(t - nT_c)) e^{-j2\pi \frac{m}{N}f_c t} dt$$

in cui la seconda eguaglianza utilizza l'interpolazione cardinale  $x(t) = \sum_{n=-\infty}^{\infty} x_n \cdot \text{sinc}(f_c(t - nT_c))$  fornita dalla (4.1), ed introduce una seconda fonte di approssimazione legata all'intervallo *finito* di variazione per  $n$ : infatti, benché l'integrale abbia estensione limitata, i valori di  $x(t)$  che cadono entro tale estensione, dovrebbero dipendere da *tutti* i suoi campioni. L'ultimo integrale è a sua volta una approssimazione (a causa degli estremi di integrazione limitati, e peggiore per i sinc centrati in prossimità dei confini della finestra) della trasformata (calcolata in  $f = \frac{m}{N}f_c$ ) di  $\text{sinc}(f_c(t - nT_c))$ , pari quest'ultima a  $T_c \text{rect}_{f_c}(f) e^{-j2\pi f n T_c}$ , che quando valutata per  $f = \frac{m}{N}f_c$ , fornisce il risultato

$$X_w\left(f = \frac{m}{N}f_c\right) \simeq T_c \sum_{n=0}^{N-1} x_n e^{-j2\pi \frac{m}{N} n}$$

per valori  $|m| \leq \frac{N}{2}$ , a causa della estensione limitata (in frequenza) di  $\text{rect}_{f_c}(f)$ . E' però facile verificare che  $X_w\left(\frac{m}{N}f_c\right)$  è periodica in  $m$  con periodo  $N$ , cosicché i valori assunti per  $m = \frac{N}{2} + 1, \frac{N}{2} + 2, \dots$  sono uguali a quelli per  $m = -\frac{N}{2} + 1, -\frac{N}{2} + 2, \dots$

<sup>37</sup>Come osservato al § 4.1.1, lo spettro  $X^\bullet(f)$  di un segnale campionato a frequenza  $f_c$  è costituito dalle repliche del segnale originario, distanziate di multipli di  $f_c$ :  $X^\bullet(f) = \sum_{n=-\infty}^{\infty} X(f - nf_c)$ , e coincide con  $X(f)$  per  $-f_c/2 < f < f_c/2$ , se  $X(f)$  è limitata in banda tra  $\pm W$  ed  $f_c \geq 2W$ . Al contrario, se  $f_c < 2W$ , allora le repliche  $X(f - nf_c)$  si sovrappongono, e la (4.11) si riscrive come  $X_m \simeq f_c X^\bullet\left(f = \frac{m}{N}f_c\right)$ .

Figura 4.10: Valori della DFT e campioni di  $X^*(f)$  associati per  $f = \frac{n}{N}f_c$ 

**DFT come prodotto matriciale** Notiamo infine che la (4.10) può essere espressa in forma matriciale: ad esempio, per  $N = 4$  si ottiene

$$\begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & e^{-j\frac{\pi}{2}} & e^{-j\pi} & e^{-j\frac{3\pi}{2}} \\ 1 & e^{-j\pi} & e^{-j2\pi} & e^{-j3\pi} \\ 1 & e^{-j\frac{3\pi}{2}} & e^{-j3\pi} & e^{-j\frac{9\pi}{2}} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (4.12)$$

da cui notiamo la proprietà di *simmetria* per la matrice dei coefficienti.

**Esempio** Allo scopo di *concretizzare* le differenze tra la trasformata di Fourier ed i valori forniti dalla DFT, in fig. 4.11-a sono riportati i valori  $|X_m|$ , normalizzati in ampiezza, per la DFT di una sinusoidale a 10 Hz, adottando due diverse finestre di analisi (vedi § 3.8.4), prelevando alla medesima frequenza di campionamento (100 Hz) un numero variabile di campioni (mostrato in figura), e ponendo i rimanenti a zero, per calcolare in tutti i casi la medesima DFT a 256 punti<sup>38</sup>. Il risultato è quindi confrontato (fig. 4.11-b) con quello ottenibile per via analitica calcolando la  $\mathcal{F}$ -trasformata dello stesso segnale, adottando le medesime finestre temporali, di durata uguale al primo caso. Le curve ottenute nel caso di 80 msec (e 8 campioni!) dipendono da meno di un periodo di segnale, e perciò presentano una componente continua apprezzabile. Aumentando la durata della finestra, l'approssimazione di calcolare una  $\mathcal{F}\{\}$  mediante la DFT migliora, anche se persiste un ridotto potere di risoluzione spettrale.

**Osservazione** Qualche lettore potrebbe stupirsi di non trovare due linee spettrali, come da attendersi per una sinusoidale. Ma questo può accadere a patto che il numero di campioni  $N$  su cui si effettua la DFT sia un multiplo intero  $k$  del numero di campioni  $M = T/T_c$  che ricadono entro uno stesso periodo  $T$  della sinusoidale<sup>39</sup>, con  $k$  che esprime quanti periodi entrano in  $N$  campioni, ed  $M$  quanti campioni/periodo. In tal caso l'applicazione della IDFT (4.13) produce una sequenza ancora periodica. Nel nostro esempio, essendo  $T = 1/10 = 100$  msec e scegliendo  $N = 64$  punti e  $k = 6$  periodi, il vincolo  $N = kM = kTf_c$  permette di ottenere  $f_c = \frac{N}{kT} = \frac{64}{6 \cdot 10^{-1}} = 106, \bar{6}$  Hz, ovvero  $M = Tf_c = 10^{-1} \cdot 106, \bar{6} = 10, \bar{6}$  campioni/periodo. La fig. 4.12 mostra questo risultato, evidenziando come la riga spettrale si manifesti per  $m = 6$ , ossia alla frequenza  $f = \frac{m}{N}f_c = \frac{6}{64}106, \bar{6} = 10$  Hz, mentre la riga presente in  $m = 58$  è in realtà il ripiegamento periodico di quella a  $-10$  Hz.

<sup>38</sup>Il metodo esposto di porre a zero i campioni fino al raggiungimento di una potenza di due è detto ZERO PADDING. Il calcolo della DFT su di un numero di punti pari ai campioni di segnale disponibili non avrebbe dato luogo all'effetto finestra, ma avrebbe fornito in tutti i casi andamenti simili a quello osservabile per 256 punti. Infine, notiamo che nelle figure sono mostrati solo i primi 128 valori, essendo i rimanenti speculari.

<sup>39</sup>Con la ovvia condizione che sia  $M > 2$  per rispettare il vincolo  $f_c > 2/T$

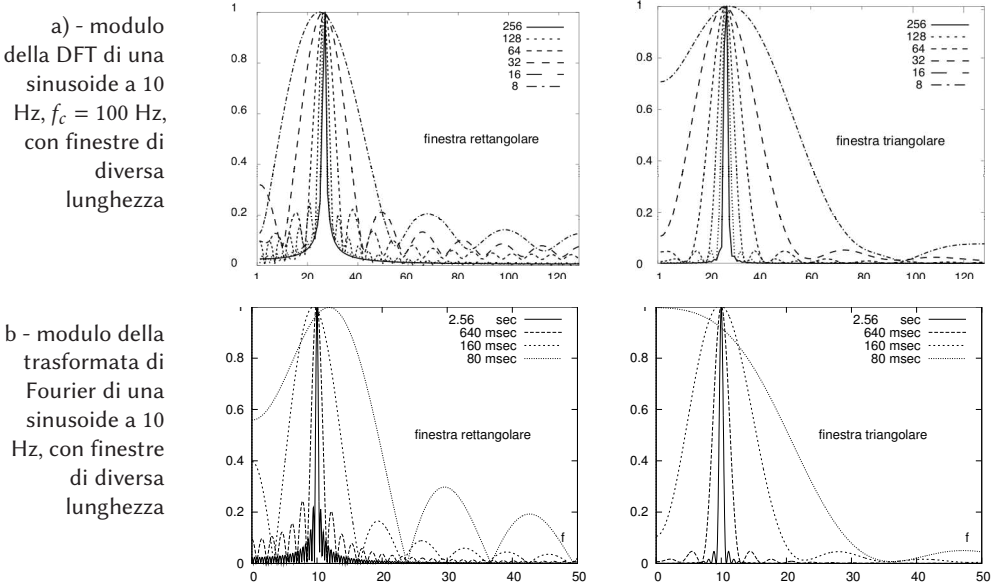


Figura 4.11: Confronto tra DFT ed  $\mathcal{F}$ -trasformata con uguale estensione temporale

Il passaggio dai campioni nel tempo  $x_n$  a quelli in frequenza  $X_m$  è invertibile<sup>40</sup>, ricorrendo alla INVERSE DISCRETE FOURIER TRANSFORM (IDFT)

$$x_n = \frac{1}{N} \sum_{m=0}^{N-1} X_m e^{j2\pi \frac{m}{N} n} \tag{4.13}$$

che per  $n$  esterno a  $[0, N - 1]$  continua a valere, ed assume valori periodici, coerentemente a quanto accade per lo sviluppo in serie di Fourier. Infatti il legame tra DFT e serie di Fourier è molto stretto, in quanto i valori  $X_m$  rappresentano una approssimazione<sup>41</sup>

<sup>40</sup>Sostituendo infatti la (4.10) nella (4.13), otteniamo

$$x_n = \frac{1}{N} \sum_{m=0}^{N-1} \left( \sum_{k=0}^{N-1} x_k e^{-j2\pi \frac{m}{N} k} \right) e^{j2\pi \frac{m}{N} n} = \frac{1}{N} \sum_{k=0}^{N-1} x_k \sum_{m=0}^{N-1} e^{j2\pi \frac{m}{N} (n-k)}$$

ma, dato che  $\sum_{m=0}^{N-1} e^{j2\pi \frac{m}{N} (n-k)} = \begin{cases} N & \text{se } k = n \\ 0 & \text{altrimenti} \end{cases}$ , nella sommatoria esterna sopravvive solo il termine  $x_n$ , dimostrando l'uguaglianza.

<sup>41</sup>La relazione (4.14) si dimostra combinando le relazioni (3.4) e (4.11):

$$X_n \simeq f_c X \left( \frac{n}{N} f_c \right) = f_c X \left( \frac{n}{NT_c} \right) = f_c X \left( \frac{n}{T} \right) = f_c X (nF) = f_c T X_n^{SF} = \frac{1}{T_c} T X_n^{SF} = N X_n^{SF}$$

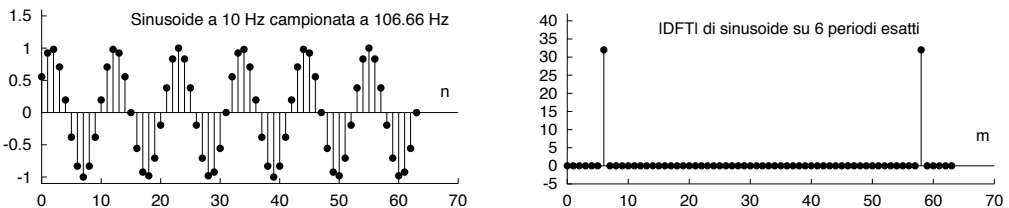


Figura 4.12: Sinusoide campionata su di un numero intero di periodi e relativo modulo di DFT

dei rispettivi coefficienti della serie di Fourier  $X_m^{SF} = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x_T(t) e^{-j2\pi \frac{m}{T} t} dt$ , calcolati a partire da un segmento  $x_T(t)$  estratto da  $x(t)$ , e moltiplicati per  $N$ :

$$X_m \simeq N \cdot X_m^{SF} \tag{4.14}$$

Per approfondire i risvolti di questo risultato, affrontiamo la sezione successiva.

### 4.5.1 Relazione tra DTFT, DFT e trasformata zeta

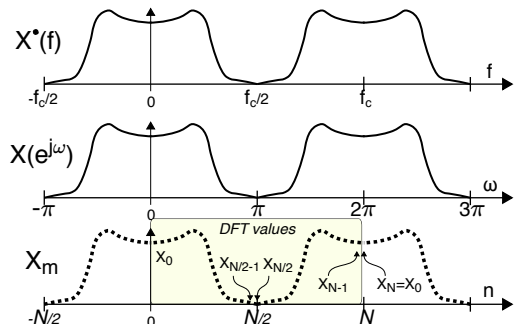
Così come per i segnali analogici sussiste una relazione (vedi pag. 119) tra la trasformata di FOURIER e quella di LAPLACE, così nel contesto delle sequenze, esistono legami tra DTFT e *trasformata zeta*, definita quest'ultima come  $X(z) = \mathcal{Z}\{x_n\}$ , funzione complessa della variabile complessa *zeta*, e valutata come

$$X(z) = \sum_{n=-\infty}^{\infty} x_n z^{-n} \tag{4.15}$$

che, nel caso in cui per  $|z| = 1$  la (4.15) converga, può essere fatta corrispondere alla DTFT (4.8) della stessa sequenza  $x_n$  semplicemente ponendo  $z = e^{j\omega}$ , ovvero calcolando  $X(z)$  sul *cerchio unitario*:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x_n e^{-j\omega n} = X(z)|_{z=e^{j\omega}} = X^*(f)|_{f=\frac{\omega}{2\pi T_c}} \tag{4.16}$$

Infatti, nelle consuete condizioni in cui gli  $x_n$  sono i campioni di un segnale  $x(t)$  limitato in banda tra  $\pm W$  e prelevati con ritmo  $f_c \geq 2W$ , la (4.16) effettivamente coincide (per  $-\pi \leq \omega < \pi$ ) con la  $X^*(f)$  (eq. (4.8), per  $-f_c/2 \leq f \leq f_c/2$ ) in cui si ponga  $f = \frac{\omega}{2\pi T_c}$ , mettendo cioè in corrispondenza le frequenze  $\pm f_c/2$  di  $X^*(f)$  con le pulsazioni  $\pm \pi$  di  $X(e^{j\omega})$ . Al di fuori dell'intervallo  $-\pi \leq \omega < \pi$  la  $X(e^{j\omega})$  è periodica in  $\omega$  con periodo  $2\pi$ , analogamente a ciò che risulta (con periodo  $f_c$ ) per la trasformata di Fourier  $X^*(f)$  di sequenze; se invece  $x_n$  è sottocampionata, ossia  $f_c < 2W$ , anche  $X(e^{j\omega})$  è affetta da aliasing, così come avviene per  $X^*(f)$ .



**Esempio** Consideriamo la sequenza  $x_n = \begin{cases} a^n & \text{se } n \geq 0 \\ 0 & \text{altrimenti} \end{cases}$  il cui andamento per  $a = 0.7$

è mostrato in fig. 4.13, la cui trasformata zeta  $X(z) = \sum_{n=-\infty}^{\infty} a^n z^{-n}$  risulta pari a<sup>42</sup>  $X(z) = \frac{1}{1-az^{-1}}$ , ed il cui modulo, dopo aver scritto la variabile complessa  $z$  come  $z = x+jy$ , è espresso come  $|X(z)| = 1/\sqrt{\left(\frac{x^2-ax+y^2}{x^2+y^2}\right)^2 + \left(\frac{ay}{x^2+y^2}\right)^2}$ . Facendo ora variare  $z$  nell'intervallo  $[-2-j2, 2+j2]$  si ottiene per il modulo di  $X(z)$  l'andamento mostrato nella figura a lato, in cui è anche raffigurato un cilindro di raggio unitario, la cui intersezione con

<sup>42</sup>Il risultato si ottiene ricordando che  $\sum_{n=0}^{\infty} a^n = \frac{1}{1-a}$  qualora  $|a| < 1$ .

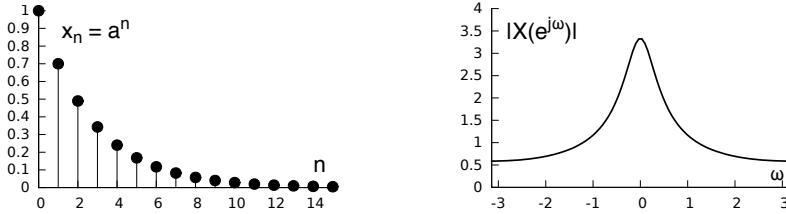
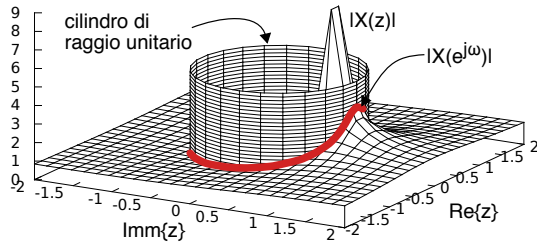


Figura 4.13: Sequenza  $x_n = a^n$  e modulo della relativa trasformata di Fourier a tempo discreto

$|X(z)|$  individua la forma di

$$|X(e^{j\omega})| = \frac{1}{\sqrt{1 + a^2 - 2a\cos\omega}} \tag{4.17}$$

ossia della DFT della sequenza  $a^n$ , che a sua volta è mostrata in fig. 4.13 per  $-\pi < \omega < \pi$ , e nella figura a lato come una linea rossa.



Se la  $X(e^{j\omega})$  ottenuta per una sequenza  $x_n$  aperiodica nel tempo è campionata in  $N$  punti equispaziati e disposti sul cerchio unitario, ossia ponendo  $\omega = 2\pi \frac{m}{N}$  con  $m = 0, 1, \dots, N - 1$ , allora si ottiene una sequenza periodica in frequenza<sup>43</sup>

$$\tilde{X}_m = \sum_{n=-\infty}^{\infty} x_n e^{-j2\pi \frac{m}{N} n} = X(e^{j\omega}) \Big|_{\omega=2\pi \frac{m}{N}} = X^*(f) \Big|_{f=f_c \frac{m}{N}} \tag{4.18}$$

che può coincidere con la sequenza  $X_m$  ottenuta calcolando la DFT (4.10) di una sequenza  $x_n$ , qualora questa abbia una durata limitata  $\leq N$ . D’altro canto, è possibile applicare la IDFT (4.13) ad un periodo della sequenza  $\tilde{X}_m$ , ed ottenere quindi una nuova sequenza di valori nel tempo, anch’essa periodica di periodo  $N$ , espressa come

$$\tilde{x}_n = \frac{1}{N} \sum_{m=0}^{N-1} \tilde{X}_m e^{j2\pi \frac{m}{N} n} \tag{4.19}$$

Infatti, i valori  $\tilde{x}_n$  dipendono da quelli  $x_n = x(t)|_{t=nT_c}$  del segnale originario  $x(t)$ , campionato agli istanti  $t = nT_c$ , mediante la relazione<sup>44</sup>

$$\tilde{x}_n = \sum_{r=-\infty}^{\infty} x_{n+rN} \tag{4.20}$$

<sup>43</sup> Infatti  $e^{-j2\pi \frac{m+N}{N} n} = e^{-j2\pi \frac{m}{N} n} e^{-j2\pi n}$ , ed il secondo termine vale 1 per qualsiasi  $n$ . Indichiamo qui ed al prossimo §, una sequenza periodica mediante la tilde  $\tilde{\cdot}$ .

<sup>44</sup> Infatti, sostituendo la (4.18) in (4.19), otteniamo  $\tilde{x}_n = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{h=-\infty}^{\infty} x_h e^{-j2\pi \frac{k}{N} h} e^{j2\pi \frac{k}{N} n}$ . Scambiando ora l’ordine delle sommatorie risulta

$$\tilde{x}_n = \sum_{h=-\infty}^{\infty} x_h \left( \frac{1}{N} \sum_{k=0}^{N-1} e^{-j2\pi \frac{k}{N} (h-n)} \right)$$

Dato che  $\frac{1}{N} \sum_{k=0}^{N-1} e^{-j2\pi \frac{k}{N} (h-n)} = \begin{cases} 1 & \text{se } h = n + rN \\ 0 & \text{altrimenti} \end{cases}$ , con  $r$  intero, si ottiene il risultato (4.20).

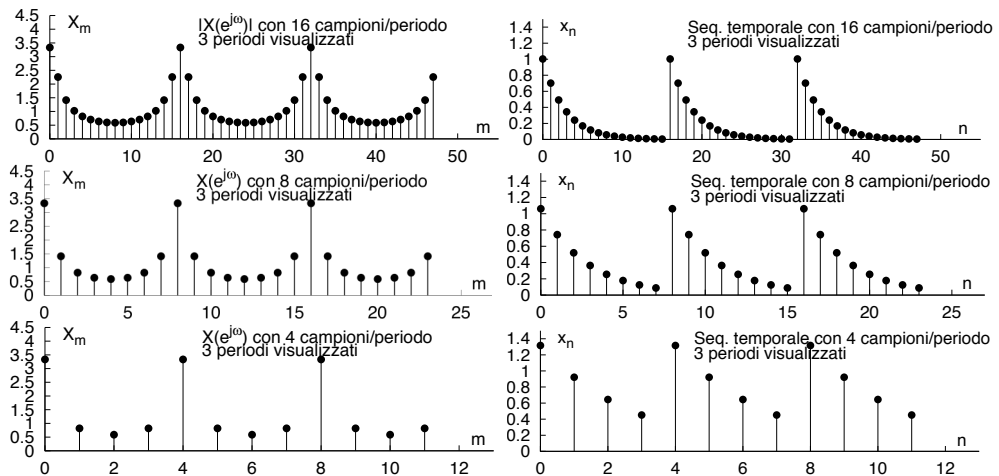


Figura 4.14: Aliasing temporale al diminuire della risoluzione del campionamento in frequenza

e quindi i primi  $N$  valori di  $\tilde{x}_n$  coincidono con i campioni di  $x(t)$  solo se quest'ultimo ha durata limitata, con estensione minore di  $NT_c$ , ossia se  $N$  è sufficientemente elevato in modo che  $NT_c$  copra tutta la durata di  $x(t)$ , e la (4.15) si riconduca alla somma di un numero finito di termini. D'altra parte, se  $x(t)$  ha durata maggiore di  $NT_c$ , ovvero  $X(z)$  è stata campionata su di un numero di campioni troppo ristretto, allora l'applicazione della IDFT (4.19) ad  $\tilde{X}(k)$  provoca il fenomeno di *aliasing temporale*.

**Esempio** Nella parte sinistra di fig. 4.14 viene mostrato il modulo della sequenza  $X_m$  ottenuta campionando la  $X(e^{j\omega})$  dell'esempio precedente, utilizzando 16, 8 o 4 campioni/periodo. Nella parte destra della stessa figura sono quindi rappresentate le corrispondenti sequenze  $x_n$  ottenute mediante IDFT. Si può notare che, mentre con 16 campioni/periodo la ricostruzione della sequenza  $x_n = a^n$  è piuttosto fedele, con 8 campioni si inizia a verificare il fenomeno di aliasing temporale, che diviene ancor più evidente per 4 campioni/periodo.

**Riepilogo** La figura 4.15 riassume le relazioni che legano la trasformata di Fourier per segnali limitati in banda ai suoi campioni ed alla relativa DTFT, così come viene illustrata la relazione tra DTFT, DFT, e trasformata zeta.

### 4.5.2 Fast Fourier Transform

La sigla FFT descrive una classe di algoritmi di calcolo della DFT e della sua inversa, caratterizzati dall'uso di un numero molto ridotto di operazioni, rendendo così computazionalmente praticabile l'elaborazione numerica dei segnali. Analizziamo innanzitutto come il calcolo di ognuno degli  $N$  termini  $X_m$  della (4.10), considerando i valori  $e^{-j2\pi \frac{m}{N}n}$  precalcolati (vedi (4.12)), richieda  $N$  moltiplicazioni complesse (equivalenti ognuna ad 4 moltiplicazioni e 2 somme reali) ed  $N - 1$  somme complesse (ognuna pari a 2 somme reali): pertanto una DFT richiede  $N(N(4+2) + 2(N-1)) = N(8N-2) \approx 8N^2$  operazioni.

Al contrario, gli algoritmi FFT più efficienti riducono il numero di operazioni ad  $8N \log_2 N$ : ad esempio, ponendo  $N = 1024$ , si ottiene un miglioramento di  $2^3(2^{10})^2/2^{3 \cdot 2^{10}} =$

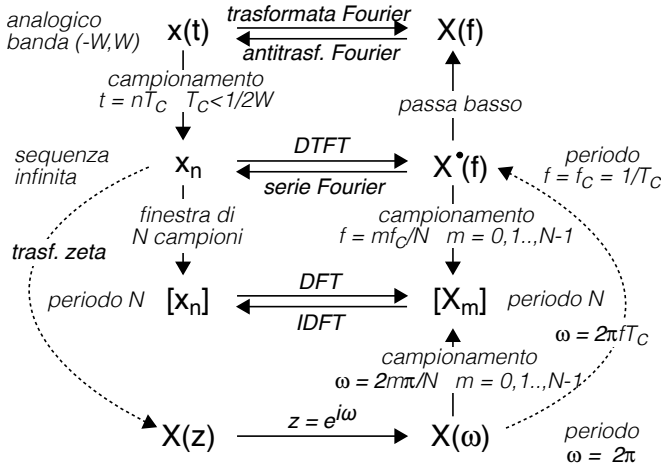


Figura 4.15: Grafo di corrispondenza tra trasformate continue e discrete

$2^{10}/10 \approx 100$  volte! Queste prestazioni sono legate all'adozione di un valore di  $N$  che sia una potenza di due (ossia  $N = 2^M$  con  $M$  intero), ma successivamente sono stati individuati metodi<sup>45</sup> che permettono una efficienza di calcolo comparabile anche per finestre di analisi di lunghezza qualsiasi.

### 4.5.3 Relazione tra DFT e DCT

Anche per la DFT risulta valida la proprietà di simmetria coniugata (§ 3.3) e quindi, se i valori della sequenza  $x_n$  di lunghezza  $N$  che compare nella (4.10) sono reali anziché complessi, allora i coefficienti di DFT  $X_m$  presentano parte reale pari e parte immaginaria dispari. In particolare, se immaginiamo di *estendere* la lunghezza della sequenza a  $2N$  punti, ottenuti ribaltando sugli indici negativi la sequenza di partenza come  $x_{-n} = x_n$  (vedi prima riga di fig. 4.16), allora siamo nelle condizioni di sequenza *reale pari*, che determina una trasformata solo reale (e pari), con parte immaginaria nulla.

Per arrivare a definire la DISCRETE COSINE TRANSFORM (DCT) si calcola una DFT *bilatera* sulla sequenza lunga  $2N$  ottenuta *traslando* quella descritta al passo precedente in modo da renderla effettivamente pari (seconda riga di fig. 4.16). Considerando che per segnali reali pari le componenti sinusoidali della base della

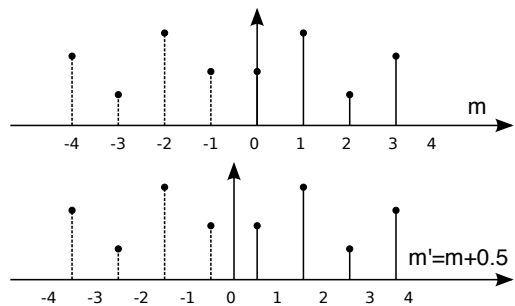


Figura 4.16: Estensione pari di sequenza reale

<sup>45</sup>Vedi ad es. [http://it.wikipedia.org/wiki/Trasformata\\_di\\_Fourier\\_veloce](http://it.wikipedia.org/wiki/Trasformata_di_Fourier_veloce)

DFT non danno contributi al risultato<sup>46</sup>, e adottando un nuovo cambio di variabile, si ottiene in definitiva la formula di calcolo della DCT come

$$X_m = \sum_{n=0}^{N-1} x_n \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) m \right] \quad (4.21)$$

a cui è associata la trasformazione inversa IDCT

$$x_n = \frac{1}{2} X_0 + \sum_{m=1}^{N-1} X_m \cos \left[ \frac{\pi}{N} \left( m + \frac{1}{2} \right) n \right] \quad (4.22)$$

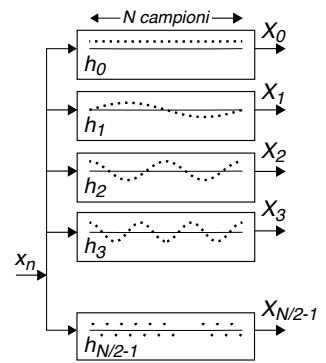
La DCT verrà usata nell'ambito della compressione di immagini (§ 10.2.4). Infatti, i valori di luminanza dei pixel in cui si scompone una immagine sono tutti valori reali.

#### 4.5.4 DFT come un banco di filtri

Un banco di filtri è qualcosa che esegue simultaneamente diverse operazioni di filtraggio sullo stesso segnale, emettendo tutti i risultati contemporaneamente. I valori

$$X_m = \sum_{n=0}^{N-1} x_n e^{-j2\pi \frac{m}{N} n}$$

di una DFT ad  $N$  punti per una sequenza  $x_n$  *equivalgono* all'output decimato di  $N$  filtri, ciascuno con risposta impulsiva  $h_m(n) = e^{j\frac{2\pi}{N} m \cdot n}$  con  $m, n = 0, 1, \dots, N-1$ , rappresentate nella figura a lato:



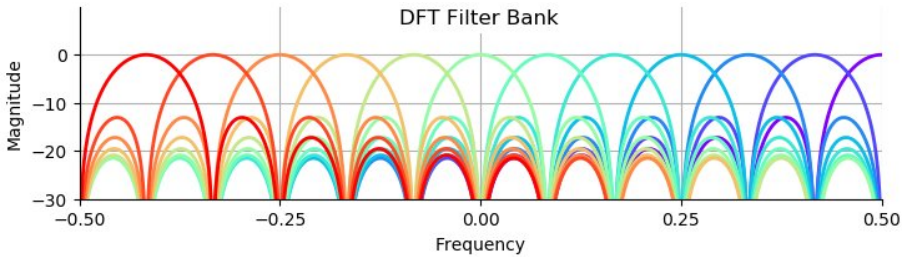
- viene mostrata solo la parte reale di  $h_m(n)$ ;
- la decimazione avviene con periodo pari alla separazione delle finestre di segnale (sovrapposte o meno);
- ogni filtro  $h_m$  ha il modulo della risposta in frequenza approssimativamente pari ad un *sinc* (dovuto ad una finestra rettangolare) centrato alla frequenza digitale  $2\pi \frac{m}{N}$  (vedi figura seguente);
- un progetto *più specifico* del filtro può ottenere una risposta in frequenza più selettiva, vedi cap. 5.

<sup>46</sup>Scriviamo la (4.10) come

$$\begin{aligned} X_m &= \sum_{n'=-N+1/2}^{N-1/2} x_{n'-1/2} e^{-j2\pi \frac{n'}{2N} m} = \sum_{n'=-N+1/2}^{N-1/2} x_{n'-1/2} \cos \left( 2\pi \frac{n'}{2N} m \right) - j \sum_{m'=-N+1/2}^{N-1/2} x_{n'-1/2} \sin \left( 2\pi \frac{n'}{2N} m \right) = \\ &= 2 \sum_{n'=1/2}^{N-1/2} x_{n'-1/2} \cos \left( 2\pi \frac{n'}{2N} m \right) = 2 \sum_{n=0}^{N-1} x_n \cos \left( 2\pi \frac{n+1/2}{2N} m \right) = 2 \sum_{n=0}^{N-1} x_n \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) m \right] \end{aligned}$$

in cui  $x_{n'}$  è quella disegnata per seconda in fig. 4.16. La quarta eguaglianza tiene conto del fatto che il termine immaginario si annulla, in quanto sommatoria bilatera di una funzione dispari (ottenuta come prodotto di  $x_{n'-1/2}$  pari e  $\sin \left( 2\pi \frac{n'}{2N} m \right)$  dispari), e del fatto che essendo i termini coseno pari, la sommatoria può essere ristretta ai soli indici positivi, raddoppiati. La penultima eguaglianza rappresenta il semplice cambio di variabile  $n = n' - 1/2$ , mentre l'ultima è (a parte il fattore 2) la definizione della DCT data in (4.21).

Un'analisi più approfondita di ciò che accade può essere fatta partendo dal filtro che produce  $X_0$  e che è un filtro *a media mobile* (pagina 133), la cui uscita è esattamente uguale a  $X_0 = \sum_{n=0}^{N-1} x_n e^{-j2\pi \frac{0}{N} n} = \sum_{n=0}^{N-1} x_n$ , che è *la media* dei valori dei campioni: pertanto il filtro risultante ha una risposta in frequenza *passa-basso* della forma mostrata a pagina 133, con  $N - 1$  zeri disposti sul cerchio unitario. Per gli altri elementi della DFT con  $m \neq 0$  la risposta impulsiva a media mobile viene *modulata* moltiplicandola per  $e^{-j2\pi \frac{0}{N} n}$ , in modo che la sua risposta in frequenza *trasli* verso l'indice  $m$ , corrispondente alla frequenza digitale  $\omega = 2\pi \frac{m}{N}$ . In definitiva, la risposta in frequenza composta viene mostrata sotto<sup>47</sup> con una scala verticale in dB, per il caso di una DFT con 12 punti.



## 4.6 Filtraggio numerico via DFT

La definizione di DFT illustrata al § 4.5 ben si presta a calcolare il risultato relativo ad un integrale di convoluzione, a patto di seguire alcune accortezze.

### 4.6.1 Convoluzione discreta

Dati due segnali  $x(t)$  e  $h(t)$  limitati in banda tra  $-W$  e  $W$ , anche il risultato della convoluzione  $y(t) = x(t) * h(t)$  è limitato in banda, ed i suoi campioni  $y_n = y(nT_c)$  (con  $T_c < \frac{1}{2W}$ ) possono essere calcolati<sup>48</sup> a partire da quelli di  $x(t)$  e  $h(t)$ , come

<sup>47</sup>Immagine presa da <https://wirelesspi.com/discrete-fourier-transform-dft-as-a-filter-bank/>

<sup>48</sup>Infatti, esprimendo l'integrale di convoluzione  $x(t) * h(t)$  nei termini dei campioni di  $x(t)$  e  $h(t)$  (eq. 4.1), e sfruttando la proprietà di ortogonalità dei segnali sinc ( $f_c(t - kT_c)$ ) (vedi § 4.1.2), per i campioni dell'uscita possiamo scrivere

$$\begin{aligned}
 y(nT_c) &= \int_{-\infty}^{\infty} x(\tau) h(nT_c - \tau) d\tau = \\
 &= \int_{-\infty}^{\infty} \left[ \sum_{k=-\infty}^{\infty} x(kT_c) \cdot \text{sinc}(f_c(\tau - kT_c)) \right] \left[ \sum_{j=-\infty}^{\infty} h(jT_c) \cdot \text{sinc}(f_c(nT_c - \tau - jT_c)) \right] d\tau = \\
 &= \sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} x(kT_c) h(jT_c) \int_{-\infty}^{\infty} \text{sinc}(f_c(\tau - kT_c)) \text{sinc}(f_c(\tau - (n - j)T_c)) d\tau = \\
 &= \frac{1}{f_c} \sum_{k=-\infty}^{\infty} x(kT_c) h((n - k)T_c) = \frac{1}{f_c} \sum_{k=-\infty}^{\infty} x_k h_{n-k}
 \end{aligned}$$

in cui alla seconda uguaglianza si è applicata la formula di ricostruzione *cardinale*  $x(t) = \sum_{k=-\infty}^{\infty} x(kT_c) \cdot \text{sinc}(f_c(t - kT_c))$  e dunque  $h(t - \tau) = \sum_{j=-\infty}^{\infty} h(jT_c) \cdot \text{sinc}(f_c(t - \tau - jT_c))$ , quest'ultima valutata per  $t = nT_c$ ; alla terza uguaglianza si è considerato che  $\text{sinc}(x)$  è una funzione *pari*, permettendo di scrivere  $\text{sinc}(f_c((n - j)T_c - \tau)) = \text{sinc}(f_c(\tau - (n - j)T_c))$ , ed alla quarta si è applicata la proprietà di

$$y_n = T_c \sum_{k=-\infty}^{\infty} x_k h_{n-k} \quad (4.23)$$

Nel caso in cui le sequenze  $x_n$  e  $h_n$  abbiano durata finita e pari rispettivamente a  $N$  ed  $M$  campioni, si otterrà una sequenza  $y_n$  di durata pari a  $N + M - 1$  campioni.

**Esempio:** il lato sinistro di fig. 4.17 mostra una sequenza triangolare ed una rettangolare con un numero di elementi non nulli rispettivamente pari a 5 e 4, la cui convoluzione si estende su 8 valori.

#### 4.6.2 Convoluzione circolare

Date due sequenze  $x_n$  ed  $h_n$  di durata finita  $N$ , il prodotto  $Y_m = X_m H_m$  delle rispettive DFT

$$X_m = \sum_{n=0}^{N-1} x_n e^{-j2\pi \frac{m}{N} n} \quad \text{ed} \quad H_m = \sum_{n=0}^{N-1} h_n e^{-j2\pi \frac{m}{N} n}$$

possiede antitrasformata  $\tilde{y}_n = \text{IDFT}\{Y_m\}$  periodica in  $n$  di periodo  $N$ , e pari a

$$\tilde{y}_n = \sum_{p=0}^{N-1} \tilde{x}_p \tilde{h}_{n-p} \quad (4.24)$$

in cui  $\tilde{x}_n$  e  $\tilde{h}_n$  sono le sequenze periodiche di periodo  $N$  ottenute replicando infinitamente le sequenze originali  $x_n$  ed  $h_n$  (<sup>49</sup>). La convoluzione (4.24) è detta *circolare* perché è possibile immaginare le sequenze  $x_n$  ed  $h_n$  *incollate* su due cilindri concentrici, e la somma svolta sui prodotti degli elementi coincidenti. Ogni valore di  $p$  corrisponde ad una diversa rotazione relativa (con angolo multiplo di  $2\pi/N$ ) dei cilindri, ed il campione di  $h_n$  che era allineato ad  $x_{N-1}$  rientra dall'altro lato, per corrispondere ad  $x_0$ .

La colonna centrale di fig. (4.17) mostra il risultato della periodicizzazione delle stesse sequenze dell'esempio precedente, assieme al risultato della convoluzione circolare tra esse, considerando un periodo  $N = 6$  per entrambe: come evidente, il risultato è abbastanza differente da quello ottenuto nel caso della convoluzione discreta.

#### 4.6.3 Convoluzione tra sequenze di durata finita via DFT

Sappiamo che la convoluzione produce un risultato di durata pari alla somma delle durate degli operandi; come anticipato, nel caso di due sequenze  $x_n$  ed  $h_n$  di durata

ortogonalità tra sinc ( $f_c t$ ) traslati di multipli di  $T_c = 1/f_c$  (vedi § 4.1.2), per cui l'integrale vale  $T_c = 1/f_c$  solo quando  $k = n - j$ , ovvero  $j = n - k$ .

<sup>49</sup>Infatti, ad  $x_n$  ed  $h_n$  corrispondono le DFT periodiche  $\tilde{X}_m$  ed  $\tilde{H}_m$ , che hanno per antitrasformata  $\tilde{x}_n$  ed  $\tilde{h}_n$ . Il prodotto  $\tilde{X}_m \tilde{H}_m$ , espresso in termini di  $\tilde{x}_n$  ed  $\tilde{h}_n$ , risulta pari a  $\tilde{Y}_m = \tilde{X}_m \tilde{H}_m = \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} \tilde{x}_p \tilde{h}_q e^{-j2\pi \frac{m}{N} (p+q)}$ , ed applicando a questo la IDFT (4.13), otteniamo:

$$\tilde{y}_n = \frac{1}{N} \sum_{m=0}^{N-1} \tilde{Y}_m e^{j2\pi \frac{m}{N} n} = \frac{1}{N} \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} \tilde{x}_p \tilde{h}_q \left( \sum_{m=0}^{N-1} e^{j2\pi \frac{m}{N} (n-p-q)} \right)$$

Dato che  $\sum_{m=0}^{N-1} e^{j2\pi \frac{m}{N} (n-p-q)} = \begin{cases} N & \text{se } q = (n-p) + lN \\ 0 & \text{altrimenti} \end{cases}$ , con  $l$  intero, risulta allora  $\tilde{y}_n = \sum_{p=0}^{N-1} \tilde{x}_p \tilde{h}_{n-p}$ , come espresso dalla (4.24).

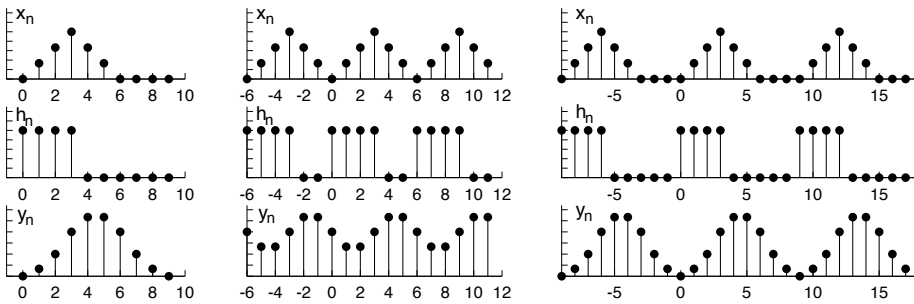


Figura 4.17: Confronto tra convoluzione discreta, circolare, e via dft

$N$  ed  $M$ , il risultato della convoluzione discreta  $y_n = \sum_{k=0}^{N-1} x_k h_{n-k}$  produce valori non nulli per indici  $n = 0, 1, \dots, N + M - 1$ . Pertanto, per fare in modo che la (4.24) produca lo stesso effetto di una convoluzione discreta, occorre costruire delle sequenze  $x'_n$  ed  $h'_n$  di lunghezza almeno pari ad  $N + M - 1$ , ottenute a partire dai valori di  $x_n$  ed  $h_n$ , a cui si aggiungono  $M - 1$  ed  $N - 1$  valori nulli, rispettivamente. In tal modo, il prodotto  $X'_m H'_m$  tra le DFT ad  $N + M - 1$  punti di queste due nuove sequenze può essere antitrasformato, per fornire il risultato corretto.

La parte destra di fig. (4.17) mostra come estendendo con zeri le due sequenze di esempio fino ad una stessa lunghezza  $N = 9$ , il risultato della loro convoluzione circolare anch'esso di periodo  $N$  ora fornisca con esattezza lo stesso risultato ottenuto calcolando direttamente la convoluzione discreta.

#### 4.6.4 Convoluzione di segnali via DFT

Due segnali  $x(t)$  e  $h(t)$  limitati in banda non possono, a rigore, essere limitati nel tempo. Viceversa, una finestra di segnale non può, a rigore, essere rappresentata dai suoi campioni: infatti, l'effetto della convoluzione in frequenza tra la trasformata della finestra (nominalmente illimitata in banda) e lo spettro del segnale, produce una dispersione frequenziale di quest'ultimo.

Ciononostante, disponendo di un numero di campioni sufficientemente elevato, si può assumere che la trasformata della finestra si attenni più velocemente in frequenza, fino a rendersi trascurabile oltre un certo valore. Inoltre, l'adozione di una frequenza di campionamento più elevata provoca un allontanamento delle repliche spettrali del segnale campionato. In queste due ipotesi è lecito ritenere l'elaborazione condotta sui campioni di segnale, equivalente a quella da svolgere sul segnale originario.

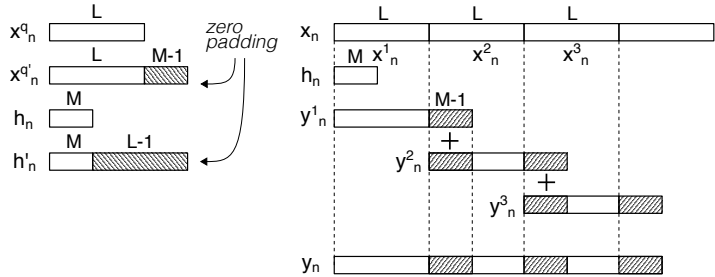
Consideriamo quindi il caso di operare su campioni prelevati alla frequenza opportuna, e di voler determinare la risposta di un filtro caratterizzato dai campioni<sup>50</sup>  $h_n = T_c \cdot h(nT_c)$  di durata finita  $M$ , ad un ingresso descritto dai campioni  $x_n = x(nT_c)$  di durata indefinita. Per applicare i risultati fin qui descritti occorre suddividere la sequenza  $x_n$  in segmenti  $x_n^q$  di lunghezza  $L$

$$x_n^q = \begin{cases} x_n & \text{per } qL \leq n < (q+1)L \\ 0 & \text{altrove} \end{cases}$$

<sup>50</sup>La normalizzazione per  $T_c$  discende dalla (4.23)

in modo da poter scrivere  $x_n = \sum_{q=-\infty}^{\infty} x_n^q$ , operare una serie di convoluzioni discrete  $y_n^q = x_n^q * h_n$ , ed infine ottenere  $y_n = x_n * h_n = \sum_{q=-\infty}^{\infty} x_n^q * h_n$  per la linearità della convoluzione.

Osserviamo ora (vedi figura) che ognuno dei termini  $y_n^q$  risulta di estensione  $N = M + L - 1$  punti, e può essere calcolato mediante la DFT inversa ad  $N$  punti del prodotto  $X_m^q H_m'$  tra le DFT ad  $N$  punti delle versioni *al-lungate con zero* (ZERO PADDED) di  $x_n^q$  ed  $h_n$ .



Infine, notiamo che l'estensione temporale  $N = M + L - 1$  dei termini  $y_n^q$  è maggiore di quella dei segmenti originali  $x_n^q$ , di lunghezza  $L$ : pertanto la sequenza  $y_n$  si ottiene sommando ai primi  $M - 1$  valori di ognuna delle  $y_n^q$ , gli ultimi  $M - 1$  valori risultanti dalle operazioni precedenti. Per questo motivo, il metodo prende il nome di OVERLAP AND ADD.

### 4.7 Riassumendo

La DFT (4.10) e la IDFT (4.13) costituiscono una coppia di relazioni invertibili che permettono di passare da una sequenza complessa di lunghezza  $N$  ad un'altra di pari lunghezza. Ma:

- calcolando la DFT (4.10) su di una finestra di  $N$  campioni  $x_n$  di un segnale  $x(t)$  limitato in una banda  $W < f_c/2$ , si ottengono delle stime  $X_m$  dei campioni della sua trasformata di Fourier  $X(f)$  per  $f = \frac{m}{N}f_c$ , ossia  $X_m \approx f_c X(\frac{m}{N}f_c)$  con  $m = 0, 1, \dots, N - 1$ ;
- calcolando la IDFT (4.13) degli  $X_m$  si ri-ottengono i campioni di  $x(t)$  di partenza;
- sia gli  $X_m$  che gli  $x_n$  sono in realtà sequenze periodiche di periodo  $N$ ;
- i calcoli indicati dalle (4.10) e (4.13) sono in realtà svolti mediante un diverso algoritmo, chiamato *Fast Fourier Transform* o FFT, che ha il vantaggio di richiedere una complessità  $O(N \log_2 N)$  ridotta rispetto a quella della DFT, che è  $O(N^2)$ .

Se la sommatoria (4.10) della DFT venisse applicata, anziché ad un numero finito  $N$  di campioni  $x_n$ , ad un loro numero infinito, allora

- si otterrebbe una sequenza periodica  $\tilde{X}_m$  (4.18) di periodo  $N$ , corrispondente al campionamento dello spettro periodico  $X^\bullet(f)$ ;
- l'applicazione della IDFT a  $\tilde{X}_m$  produrrebbe una sequenza periodica  $\tilde{x}_n$  (4.19), coincidente con la sequenza originaria  $x_n$  solo nel caso in cui questa fosse di durata finita, minore o uguale ad  $N$ ;

- segmentando un segnale  $x(t)$  in sotto-intervalli disgiunti, si può eseguire la convoluzione tra  $x(t)$  ed una  $h(t)$  di durata finita (§ 4.6.4), operando esclusivamente nel dominio digitale, e sommando tra loro le IDFT dei prodotti tra la DFT dei campioni di  $T_c \cdot h(t)$ , e le DFT dei campioni di  $x(t)$  prelevati in corrispondenza ai suoi segmenti.

L'interpretazione dei valori che risultano dalla applicazione della DFT su dei campioni di segnale, come stima della trasformata di Fourier del segnale, deve tenere conto oltre che delle fonti di approssimazioni evidenziate nella nota 36, anche dei corretti valori da assegnare alla scala delle frequenze e delle ampiezze, ossia:

#### 4.7.1 Le frequenze della DFT

Occorre tener presente il valore della frequenza di campionamento e della periodicità degli  $X_m$ . Infatti i valori  $X_m$  per  $m = 0, 1, \dots, N - 1$  corrispondono ai campioni di  $X^\bullet(f)$  per  $f = \frac{m}{N}f_c$ , ma se  $x(t)$  è reale,  $X^\bullet(f)$  oltre ad essere periodico presenta simmetria coniugata, e dunque per valori  $f > \frac{f_c}{2}$ , ossia per  $m > \frac{N}{2}$ ,  $X^\bullet(f)$  assume valori *speculari* a quelli risultanti per  $f < \frac{f_c}{2}$ .

**Esempio** Se  $N = 512$ , i primi 256 valori (con  $m$  da 0 a 255, ossia per  $m = 0, 1, 2, \dots, N/2 - 1$ ) sono da mettere in corrispondenza con quelli di  $X(f)$  con  $f = 0, \frac{1}{N}f_c, \frac{2}{N}f_c, \dots, \frac{N/2-1}{N}f_c$ ; mentre i restanti 256 valori (da 256 a 511, ossia per  $m = N/2, N/2 + 1, \dots, N - 1$ , e corrispondenti a  $f = \frac{1}{2}f_c, \frac{N/2+1}{N}f_c, \dots, \frac{N-1}{N}f_c$ ) esibiscono un comportamento speculare a quello dei precedenti.

#### 4.7.2 Le ampiezze della DFT

Come espresso dalla (4.14), i valori  $X_m$  rappresentano una approssimazione dei coefficienti della serie di Fourier calcolati sulla finestra temporale da cui provengono i campioni di segnale, e moltiplicati per il numero di campioni utilizzati nel calcolo:  $X_m \simeq N \cdot X_m^{SF}$ . Pertanto, i valori ottenuti dalla DFT devono essere normalizzati, dividendoli per  $N$ .

### 4.8 Sottocampionamento

Questo capitolo ha messo fin da subito in chiaro come per poter ricostruire un segnale limitato in banda  $x(t)$  a partire dai suoi campioni è necessario che la frequenza di campionamento  $f_c$  superi il doppio della massima frequenza  $W$  contenuta in  $x(t)$ , ovvero  $f_c \geq 2W$ . In realtà, ciò è vero solamente per segnali di banda base. Nel caso di segnali limitati in banda e centrati attorno ad una frequenza  $f_0$  detti *segnali modulati* (cap. 11), la cui massima frequenza è  $f_0 + W$ , la ricostruzione è *possibile* purché si scelga  $2W \leq f_c < f_0/k$  con  $k$  intero, nel senso che in tal caso il segnale con spettro periodico in frequenza  $X^\bullet(f)$  che si viene a creare non presenta aliasing, e quello di partenza può essere recuperato mediante una operazione di filtraggio *passa banda*.

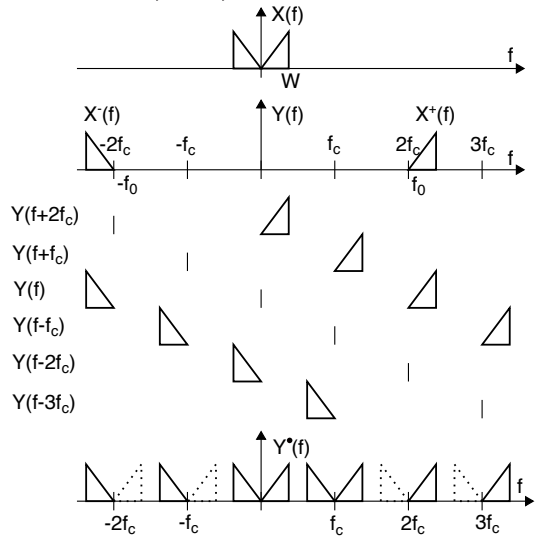
Per illustrare ciò che accade, facciamo riferimento alla figura che segue. Partiamo da un segnale  $x(t)$  limitato in banda, e consideriamo un secondo segnale con spettro

$Y(f) = X^+(f) + X^-(f)$  in cui  $X(f)$  si è scisso in due parti, quella a frequenze positive ed a frequenze negative, centrate rispettivamente ad  $f_0$  e  $-f_0$ .<sup>51</sup>

Anche se in teoria dovremmo campionare  $y(t)$  a frequenza almeno pari a  $2(f_0 + W)$ , scegliamo di adottare  $f_c = f_0/2$ . Sappiamo che il risultato sarà un segnale con spettro periodico

$$Y^*(f) = \sum_n Y(f - nf_c)$$

e dunque disegniamo le repliche di  $Y(f)$  traslate in  $\pm nf_c$ , per poi all'ultima riga sovrapporle tra loro e verificare il risultato. Osserviamo allora che non solo gli spettri  $X^+(f)$  e  $X^-(f)$  sono ancora al loro posto e possono essere recuperati mediante un filtraggio passa banda, ma addirittura si è ricomposto anche il segnale  $x(t)$  di banda base, estraibile da  $Y^*(f)$  mediante filtraggio passabasso, da realizzare ovviamente per via numerica, avendo già a disposizione i campioni.



Ci si può ora chiedere se il *trucco* funzioni solo per segnali BLU. La risposta è no, funziona anche per segnali a modulazione con *banda laterale doppia*, come ad esempio  $y(t) = x(t) \cos \omega_0 t$ , a cui corrisponde  $Y(f) = \frac{1}{2} [X(f + f_0) + X(f - f_0)]$ . Scegliendo  $f_c = f_0/k$  otteniamo

$$\begin{aligned} Y^*(f) &= \sum_n Y\left(f - n\frac{f_0}{k}\right) = \frac{1}{2} \left[ \sum_n X\left(f - n\frac{f_0}{k} + f_0\right) + \sum_n X\left(f - n\frac{f_0}{k} - f_0\right) \right] = \\ &= \frac{1}{2} \left[ \sum_n X\left(f - \left(\frac{n}{k} - 1\right)f_0\right) + \sum_n X\left(f - \left(\frac{n}{k} + 1\right)f_0\right) \right] = \\ &= \frac{1}{2} \left[ \sum_n X\left(f - (n - k)\frac{f_0}{k}\right) + \sum_n X\left(f - (n + k)\frac{f_0}{k}\right) \right] = \\ &= \frac{1}{2} \left[ \sum_m X\left(f - m\frac{f_0}{k}\right) + \sum_m X\left(f - m\frac{f_0}{k}\right) \right] = \sum_m X\left(f - m\frac{f_0}{k}\right) \end{aligned}$$

ovvero lo stesso risultato a cui saremmo pervenuti campionando direttamente il segnale  $x(t)$  di banda base di partenza.

Quanto esposto costituisce di fatto una strada alternativa per demodulare (sia in forma omodina che eterodina) un segnale modulato e passare alla sua rappresentazione digitale *in un colpo solo*, secondo un approccio chiamato *software defined radio*<sup>52</sup> in cui tutta l'elaborazione del segnale avviene in forma numerica. Oltre ad avere ovvia applicazione nella tecnologia dei nostri attuali telefonini e schede WiFi, il sottocampionamento di segnali radio è uno degli avanzamenti tecnologici che contribuiscono

<sup>51</sup>In effetti, questo è ciò che succede nel caso di modulazione in *banda laterale unica* o BLU, vedi § 12.1.2

<sup>52</sup>Vedi ad es. [https://en.wikipedia.org/wiki/Software-defined\\_radio](https://en.wikipedia.org/wiki/Software-defined_radio)

alla progettazione della telefonia 5G, in cui i segnali radio ricevuti dalle *base station* collegate alle antenne collocate sopra palazzi e torri, anziché essere elaborati in loco, vengono subito campionati e *spediti* su fibra ottica presso il *data center* più vicino, dove ciò che una volta era una *centrale telefonica* si è trasformato un *centro di elaborazione dati*.

Vi sono alcune accortezze di cui tenere conto:

- la precisione con cui è necessario generare  $f_c$  è del tutto comparabile a quella con la quale è necessario ricostruire la portante in un ricevitore omodina;
- la banda di rumore in ingresso al campionatore si estende fino a  $f_0 + W$ , mentre la funzione del filtro anti-aliasing è ora svolta da un filtro passa banda che isola la porzione di spettro su cui effettuare il sottocampionamento;
- la durata  $\tau$  della *fase di lettura* da parte del *sample & hold* deve essere una frazione del periodo di campionamento teorico  $T_c \leq 1/2(f_0+W)$ .

L'opera

# ***Trasmissione dei Segnali e Sistemi di Telecomunicazione***

è il risultato di un progetto ventennale di cultura libera, aggiornato di continuo ed evolutosi fino alla forma attuale. La sua disponibilità pubblica è regolata dalle norme di licenza CREATIVE COMMONS

*Attribuzione - Non commerciale -  
Condividi allo stesso modo*



<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.it>

e tutte le risorse relative al testo sono accessibili presso

<https://teoriadeisignali.it/libro/>

Puoi contribuire al suo successo promuovendone la diffusione e supportarne lo sviluppo attraverso una donazione, in buona parte devoluta ai progetti *open source*<sup>1</sup> che ne hanno resa possibile realizzazione e divulgazione. Ai donatori viene accordato un accesso *vitalizio* al formato PDF *navigabile* di tutte le edizioni presenti *e future*.

---

1

- . Lyx - <http://www.lyx.org/>
- . L<sup>A</sup>T<sub>E</sub>X - <https://www.latex-project.org/>
- . TeX Users Group - <https://tug.org/>
- . Inkscape - <http://www.inkscape.org/>
- . Gnuplot - <http://www.gnuplot.info/>
- . Octave - <http://www.gnu.org/software/octave/>
- . Geany - <https://www.geany.org/>
- . Linux - <https://www.linux.it/>
- . Free Software Foundation - <https://shop.fsf.org/>
- . GNOME Foundation - <https://www.gnome.org/>
- . Mozilla Foundation - <https://www.mozilla.org/it/>
- . Wikipedia - <https://it.wikipedia.org>
- . Internet Archive - <https://archive.org/about/>
- . Creative Commons - <https://creativecommons.it/chapterIT/>
- . WordPress - <https://it.wordpress.org/>
- . Phplist - <https://www.phplist.org/>