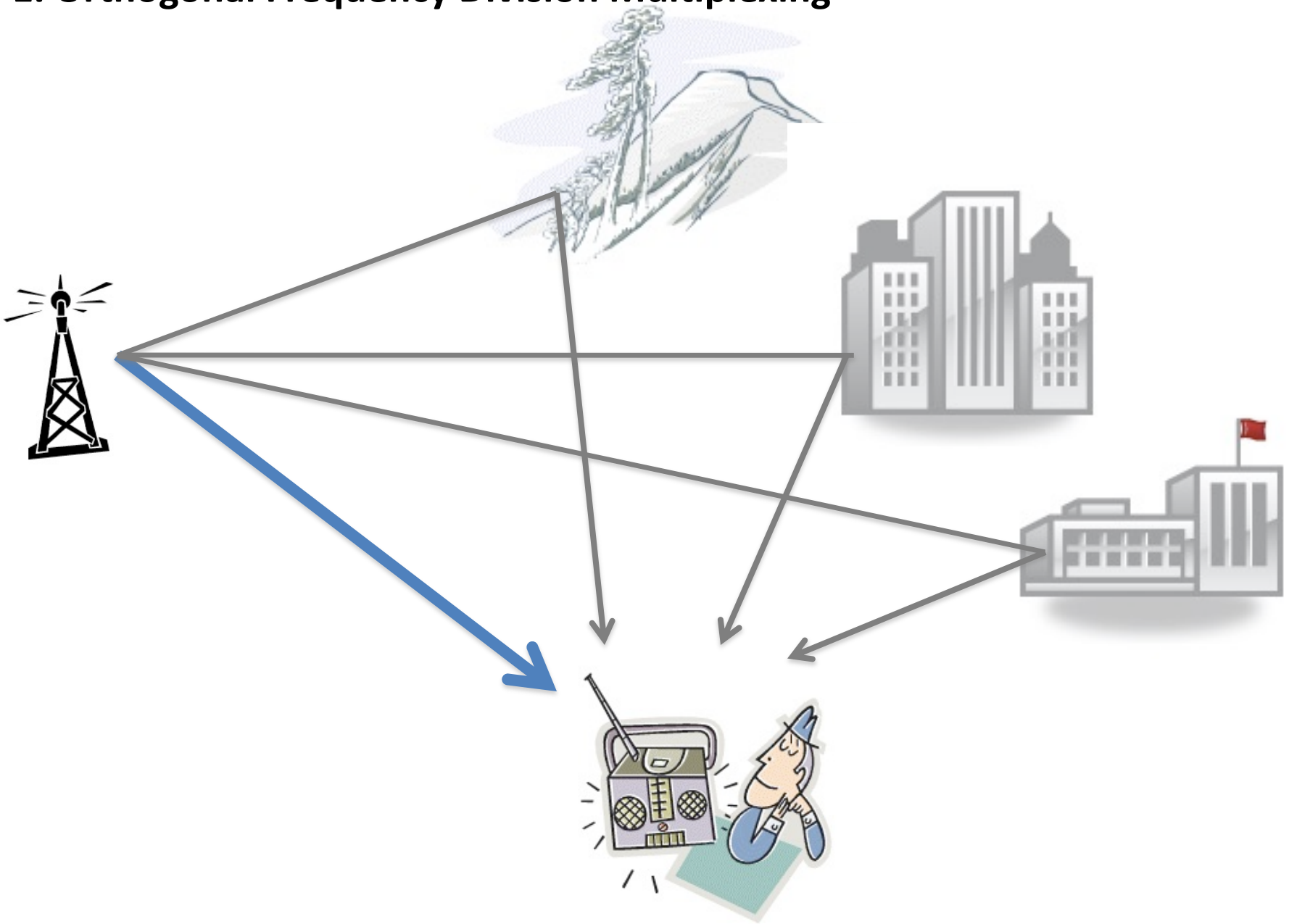
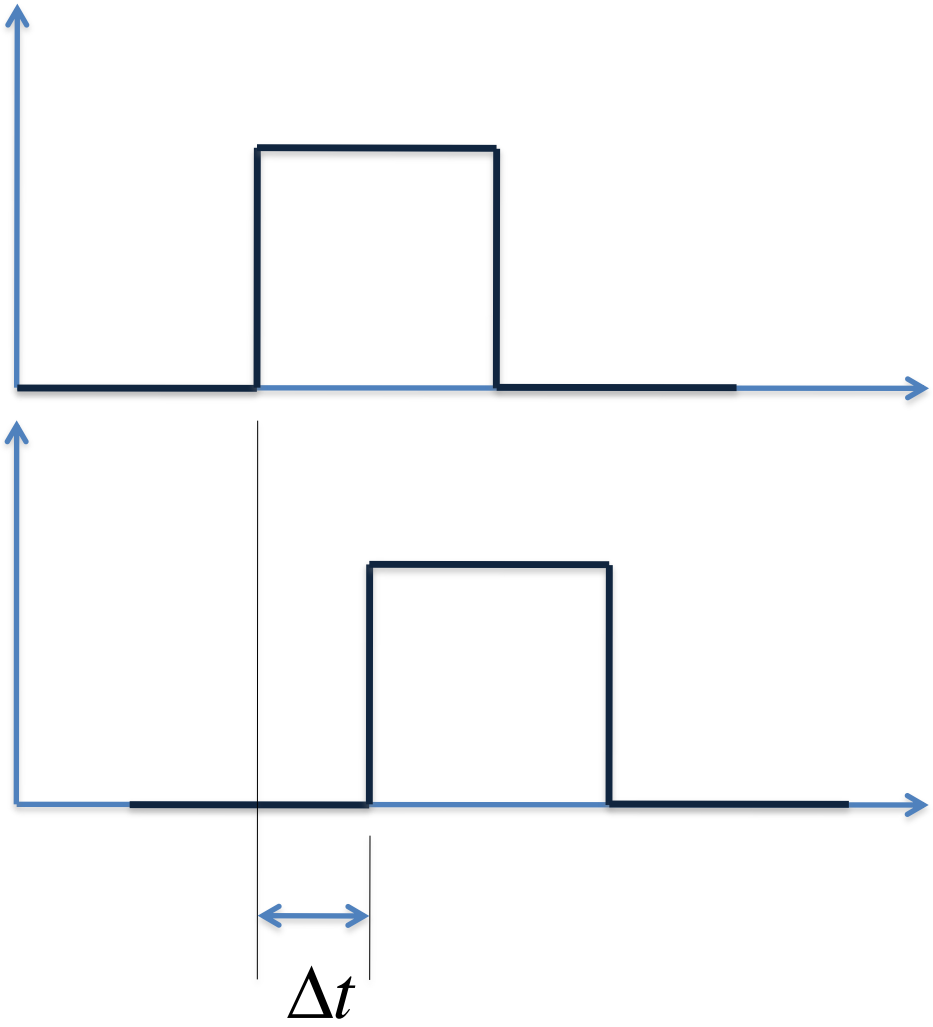
Argomenti associati alle DFT FFT e DCT

Edoardo Milotti

Corso di Fondamenti Fisici di Tecnologia Moderna

1. Orthogonal Frequency Division Multiplexing




segnale diretto
riflessione
Δt

Come fare ad ottenere un banco di filtri *estremamente* stretti?

Basta realizzare una DFT e prendere come frequenze da filtrare un insieme di frequenze che appartengono al pettine di frequenze!

Quindi

- durata del campionamento: T
- numero di campioni: 2N
- numero totale di componenti di Fourier indipendenti (spettro unilatero): N
- frequenze del pettine:

$$\omega_k = \frac{2\pi k}{T}$$

- frequenze trasmesse:

$$\omega_k = \frac{2\pi (k + n_0)}{T}$$

Segnali campionati nel tempo T (N campioni)

Frequenza del pettine di frequenze della DFT

Anche la portante fa parte del pettine di frequenze della DFT

Singolo bit

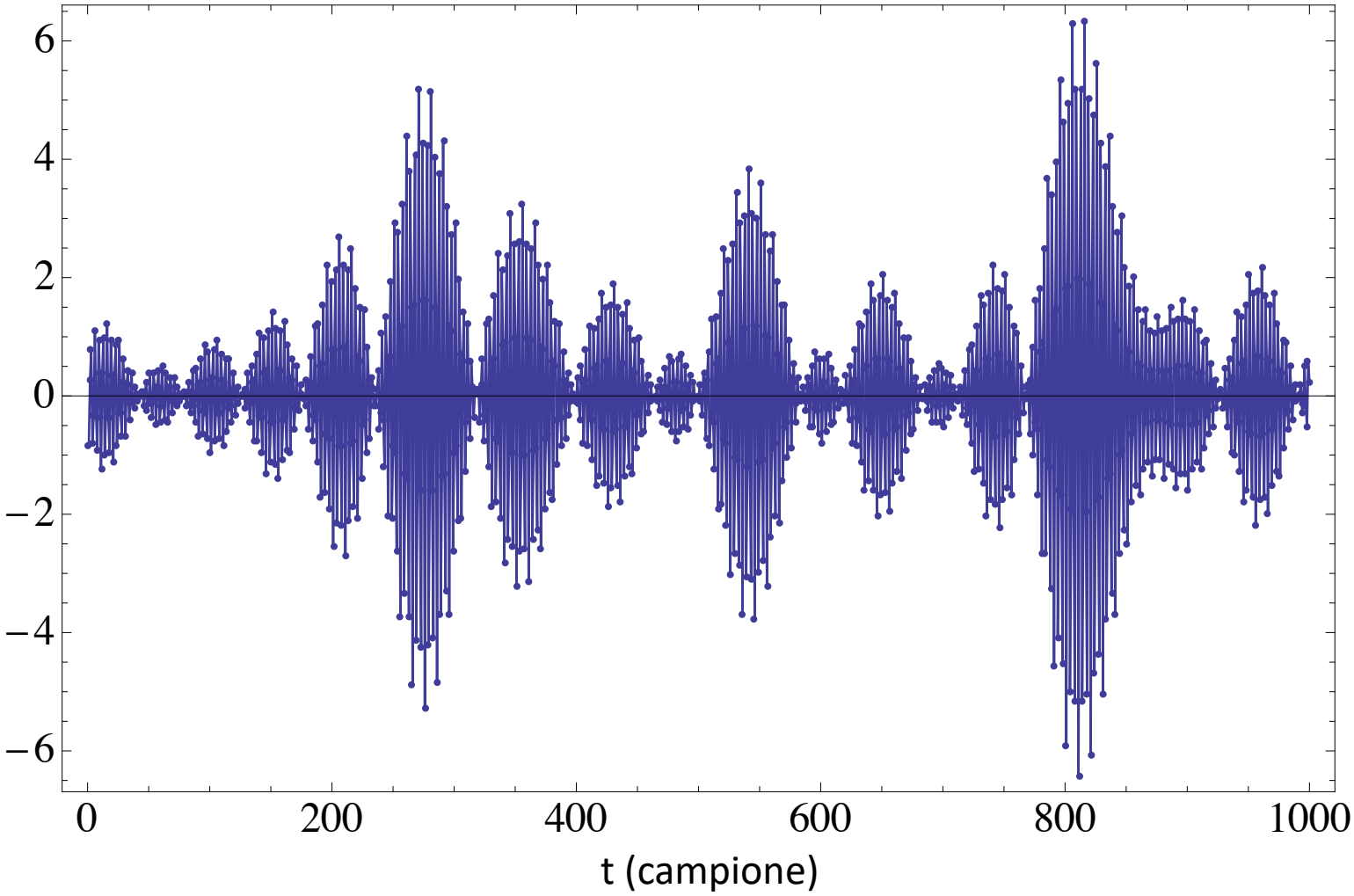
$$b_k \cos\left(\frac{2\pi k}{T}t + \varphi_k\right)\cos(\omega_C t)$$

Array di bits

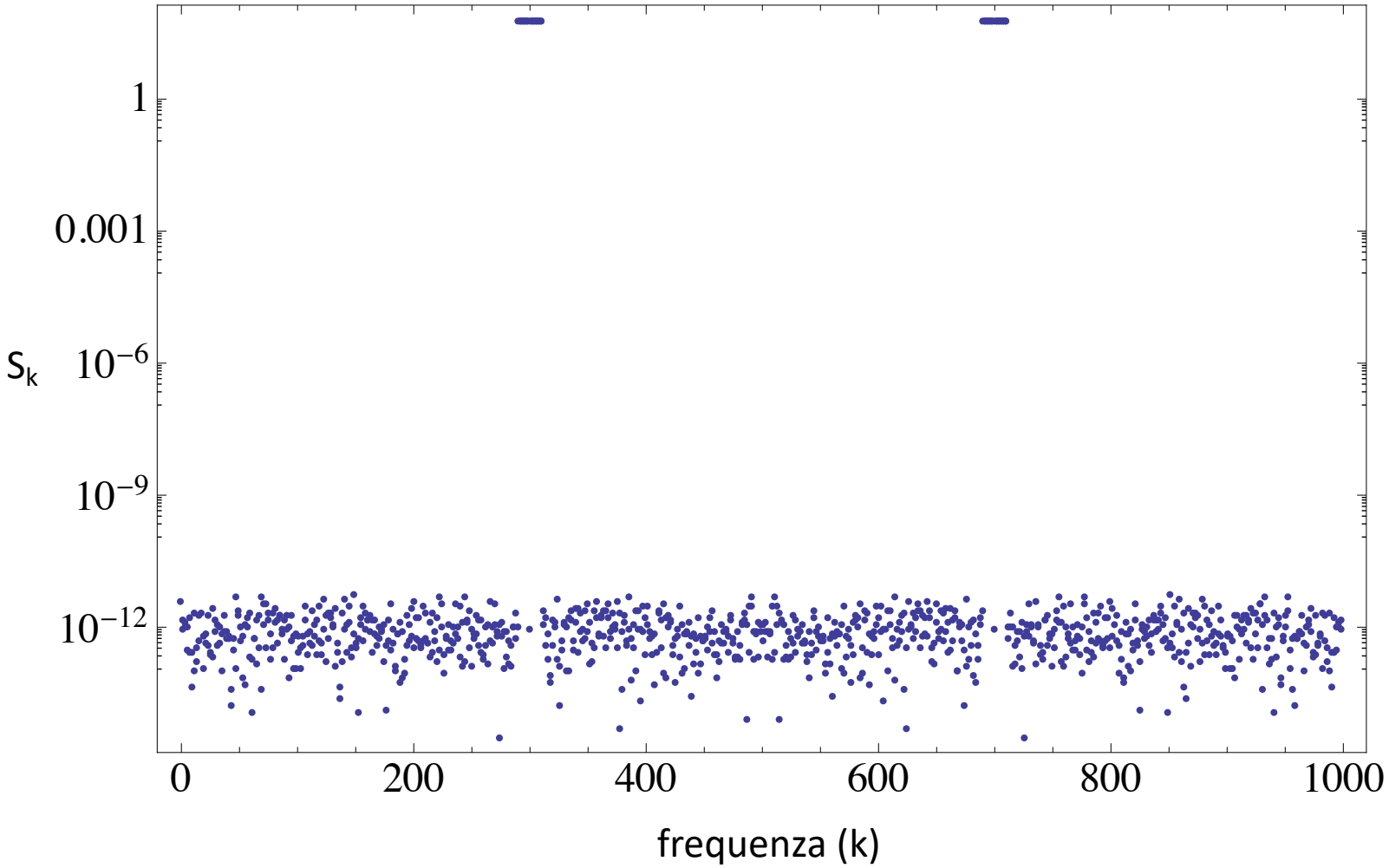
$$\sum_k b_k \cos\left(\frac{2\pi k}{T}t + \varphi_k\right)\cos(\omega_C t)$$

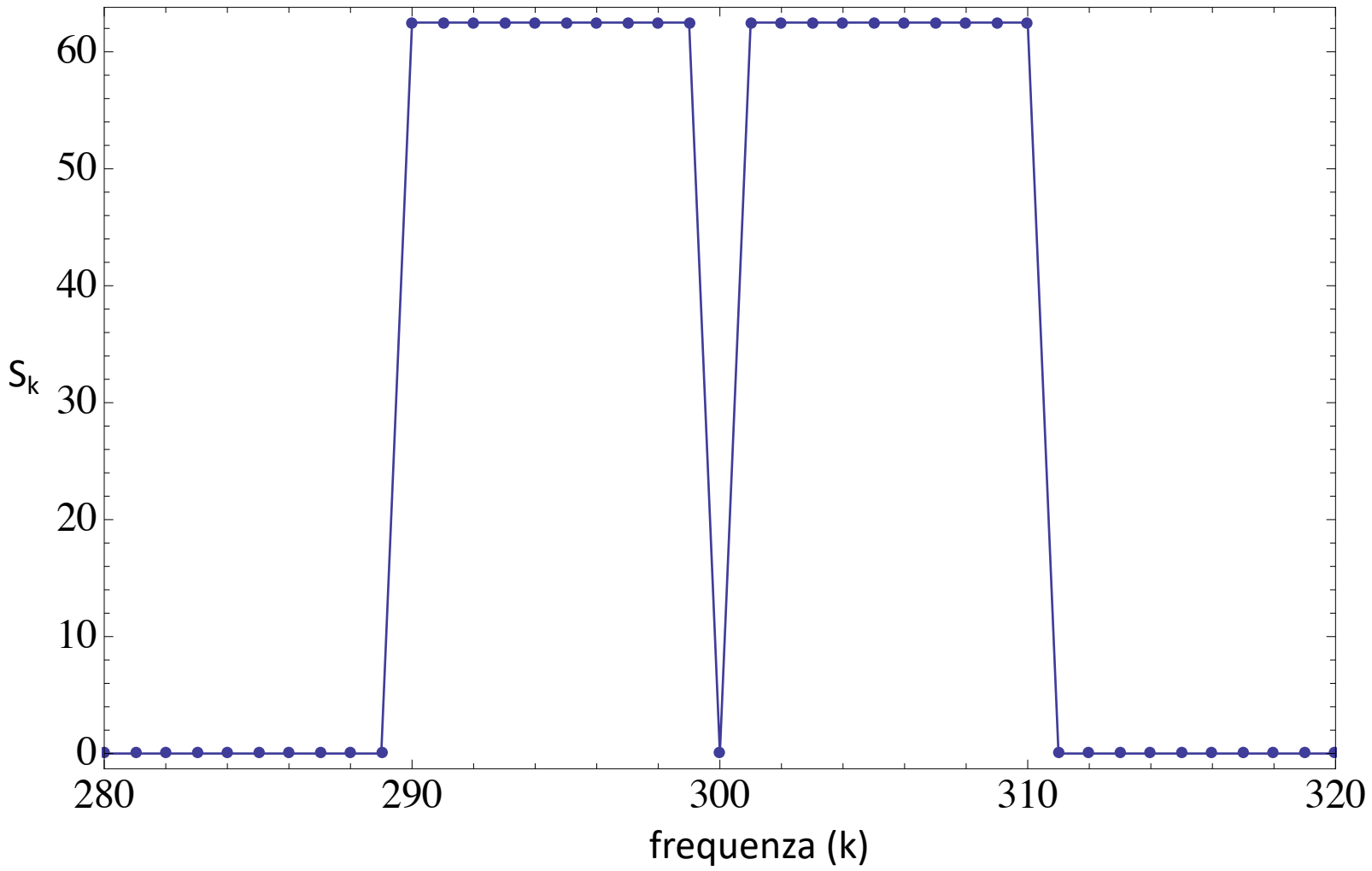
una portante + 10 segnali che portano ciascuno l'informazione corrispondente ad un bit alto

$$\sum_{k=1,10} \cos\left(\frac{2\pi k}{T}t + \varphi_k\right) \cos\left(300\frac{2\pi}{T}t\right)$$



spettro bilatero

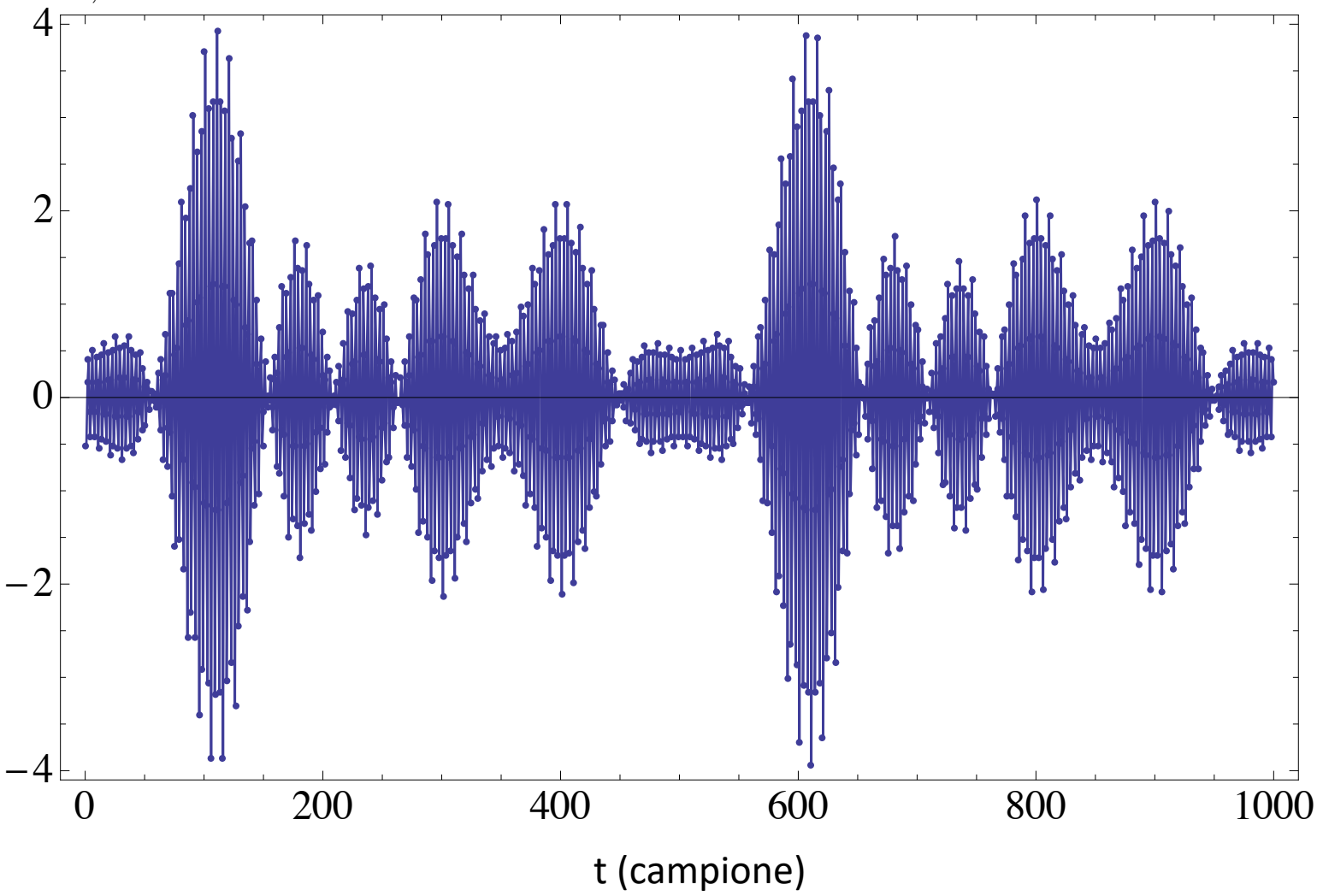


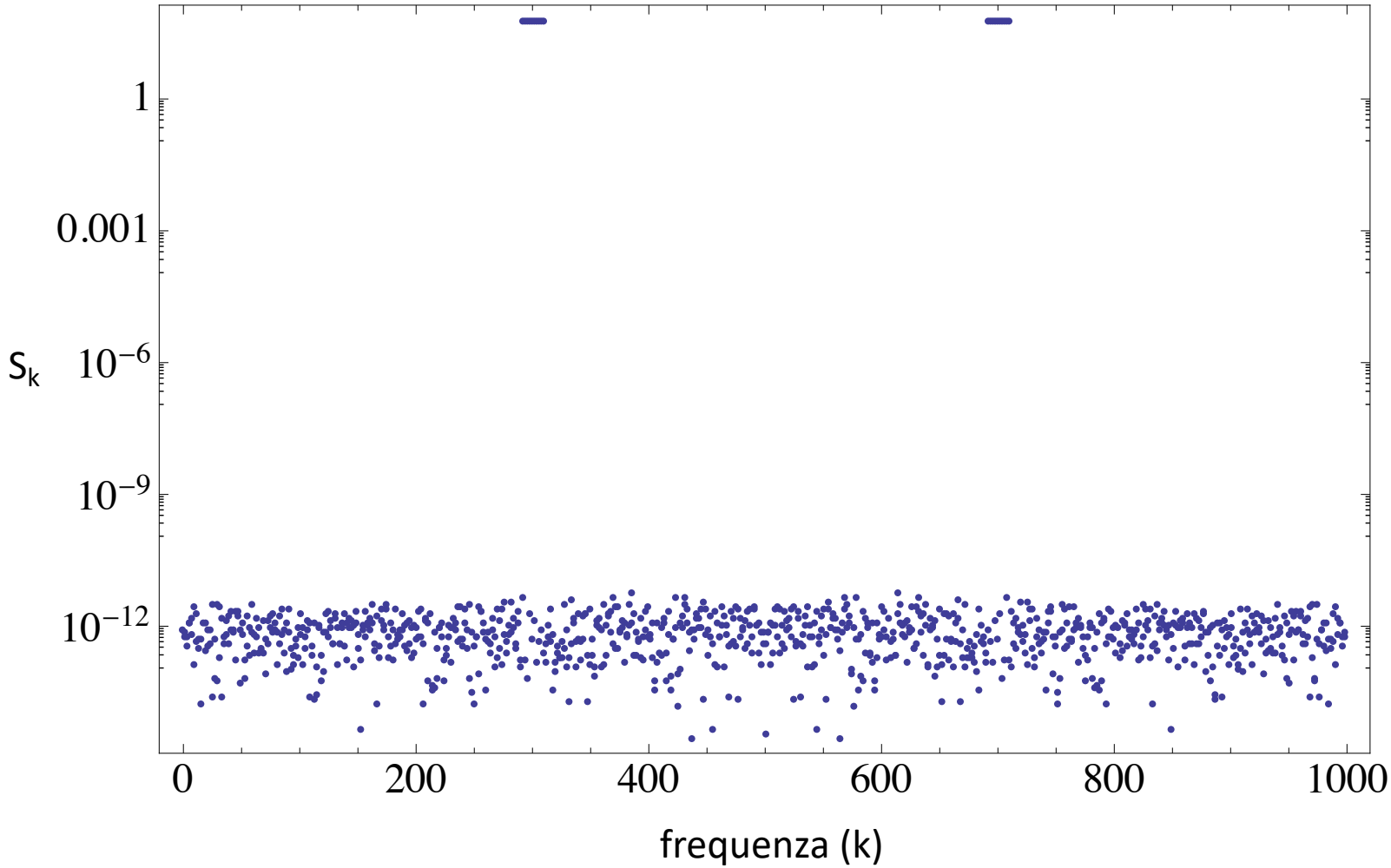

60
50
40
30
20
10
0
S_k
280
290
300
310
320
frequenza (k)

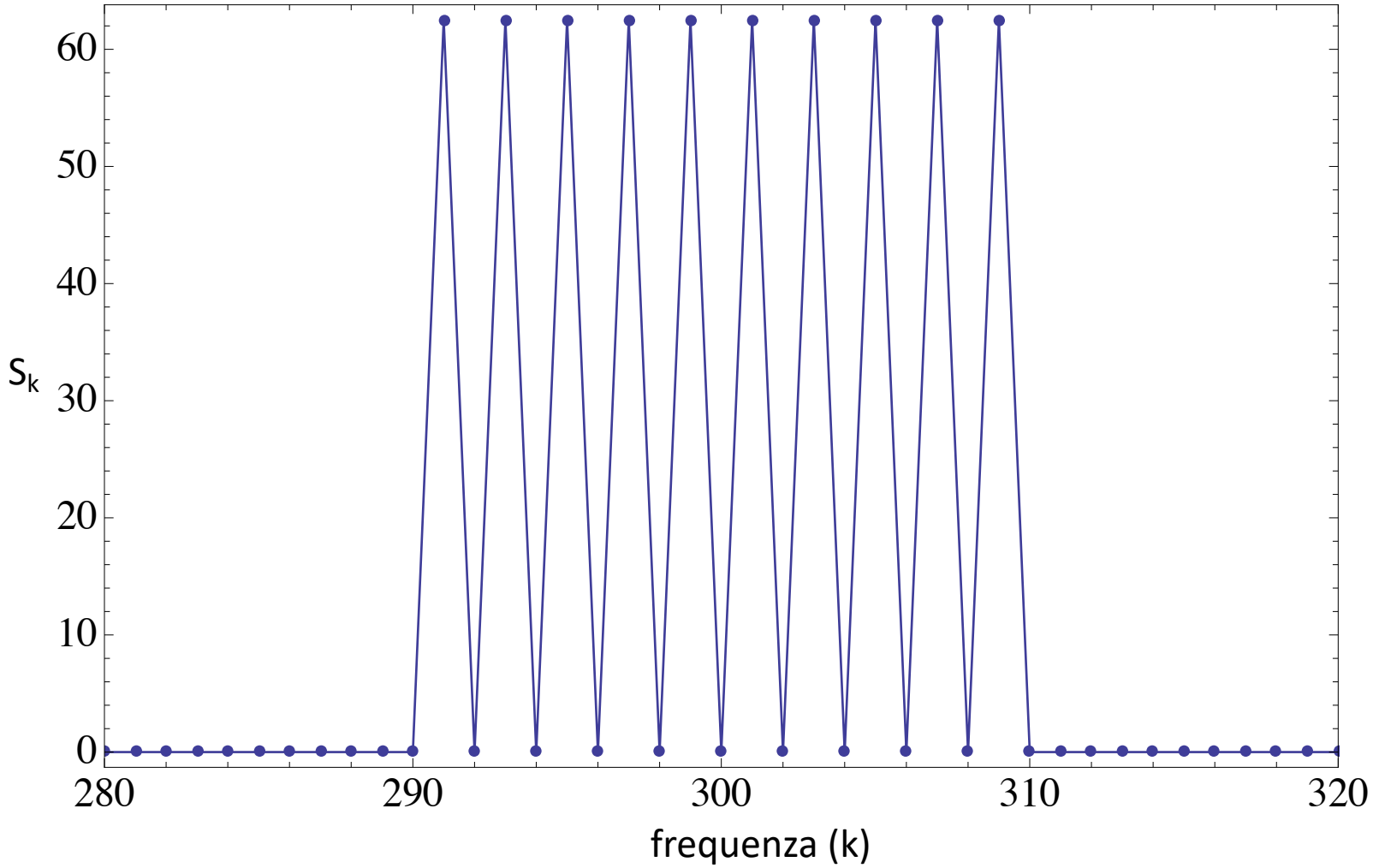
una portante + 10 segnali che portano ciascuno l'informazione di un bit

$$\sum_{k=1,10} b_k \cos\left(\frac{2\pi k}{T}t + \varphi_k\right) \cos\left(300\frac{2\pi}{T}t\right)$$

bit alternati (alto basso)




1
0.001
10^{-6}
10^{-9}
10^{-12}
S_k
0
200
400
600
800
1000
frequenza (k)


60
50
40
30
20
10
0
S_k
280
290
300
310
320
frequenza (k)

Esempi di sistemi che usano il metodo OFDM

- ADSL su cavo
- DVB-C2 (TV digitale su cavo)
- WLAN con standard IEEE 802.11
- DAB+ e Digital Radio Mondiale
- DVB-T

2. Compressione dati tramite l'algoritmo di Huffman (Huffman coding)

Esempio di algoritmo di compressione: Run-length encoding

a. Sequenza di pixel bianchi (W) e neri (B)

$$BWWW\ BBWW\ WWWW\ WWWW\ BWWW\ WWBB\ WBBB\ WWWW$$

b. Rappresentazione binaria

$$0111\ 0011\ 1111\ 1111\ 0111\ 1100\ 1000\ 1111$$

c. Conteggio di bianchi e neri

$$1B3W2B10W1B5W2B1W3B4W = 1 \text{ black, } 3 \text{ whites, } 2 \text{ blacks, } \ldots$$

d. Rappresentazione compatta

$$1,\ 3,\ 2,\ 10,\ 1,\ 5,\ 2,\ 1,\ 3,\ 4$$

Entropia di Shannon

– Analoga all'entropia di Boltzmann

– Quantifica l'informazione trasmessa da una sorgente di simboli

– L'informazione portata da un simbolo emesso con probabilità p è definita da $\log_2 1/p$

– L'entropia di Shannon è *l'informazione media* trasmessa da una sorgente

$$H = -\sum_k p_k \log_2 p_k$$

Il risultato è un numero che va interpretato come bit/simbolo.

Claude Elwood Shannon (30 aprile 1916 – 24 febbraio 2001) è stato matematico, ingegnere elettrico, crittografo. È noto come il "padre della teoria dell'informazione", che egli ha fondato con un articolo epocale "A Mathematical Theory of Communication", pubblicato nel 1948.

Ha fondato anche la teoria dei circuiti digitali nel 1937, quand'era un ventunenne studente di master al Massachusetts Institute of Technology (MIT). Nella sua tesi dimostrò che tramite le applicazioni elettriche dell'algebra di Boole è possibile costruire tutte le relazioni logiche/numeriche. Contribuì anche al campo della criptanalisi per la difesa nazionale durante la Seconda Guerra Mondiale, con contributi fondamentali nel campo della analisi dei codici segreti e delle telecomunicazioni sicure. (adattato da Wikipedia, si veda anche il recente film biografico "The Bit Player" https://www.youtube.com/watch?v=JP1Ljp8X6hg)

Problema: che distribuzione di probabilità massimizza l'entropia (e cioè l'informazione media trasmessa per simbolo)?

Si deve massimizzare la funzione entropia tenendo conto della condizione di normalizzazione delle probabilità e per questo si usa un moltiplicatore di Lagrange e si massimizza la funzione ausiliaria

$$H - \lambda \sum_k p_k = -\sum_k p_k \log_2 p_k + \lambda \sum_k p_k$$

Derivando ed eguagliando a zero si trova

$$\frac{\partial}{\partial p_j}\left(H - \lambda \sum_k p_k\right) = -\log_2 p_j - 1 - \lambda = 0$$

e quindi le probabilità che massimizzano l'entropia per sorgente che emette *N* simboli sono quelle di una distribuzione uniforme

$$p_j = 2^{\lambda - 1} = 1/N$$

Quindi l'entropia massima di una sorgente che trasmette *N* simboli è

$$H = \log_2 N$$

Se ci sono 2 soli simboli, 0 e 1, l'entropia massima vale esattamente 1 e questo definisce il bit.

Ad esempio se ci sono 8 simboli l'entropia massima vale 3 bit e se ci sono 2^n simboli l'entropia massima vale *n* bit.

Problema per casa: si calcoli quanto vale l'entropia di una sorgente che emette 8 simboli con probabilità

$$p_k \propto \frac{1}{k}$$

(*k* = 1 .... 8)

Compressione dell'informazione

Consideriamo la seguente frase in lingua italiana:

la compressione dei dati permette la riduzione del numero di bit necessari alla rappresentazione in forma digitale di una informazione

Nella frase vengono utilizzate le lettere dell'alfabeto italiano (21) più lo spazio e quindi in totale 22 simboli. *Poiché $2^4 < 22 < 2^5$ dobbiamo usare 5 bit per rappresentare individualmente ciascun carattere.*

In totale nella frase ci sono 134 caratteri e dunque dobbiamo utilizzare 670 bit.

D'altra parte l'entropia associata a questa frase è 3.87592 bit e quindi dovrebbe essere possibile utilizzare un minimo di 520 bit per trasferire la stessa informazione.

Il metodo proposto da Huffman dà una risposta a questo problema.

Huffman, D.A. 1952, "A Method for the Construction of Minimum-Redundancy Codes," Proceedings of the Institute of Radio Engineers, vol. 40, pp. 1098–1101.

PROCEEDINGS OF THE I.R.E.

A Method for the Construction of Minimum-Redundancy Codes*

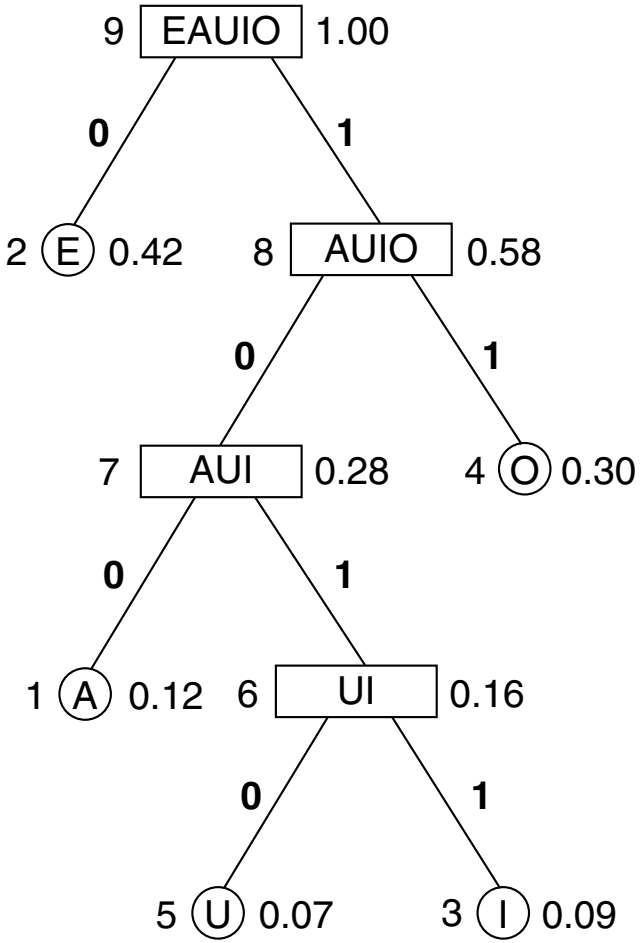
DAVID A. HUFFMAN†, ASSOCIATE, IRE

***Summary*—An optimum method of coding an ensemble of messages consisting of a finite number of members is developed. A minimum-redundancy code is one constructed in such a way that the average number of coding digits per message is minimized.**

Semplice esempio (da Numerical Recipes, si veda anche http://numerical.recipes)

In questo caso ci sono solo 5 simboli, che compaiono con diverse frequenze nel messaggio da trasferire, la tabella e l'albero binario illustrano il metodo di costruzione del codice.

Node	Stage:	1	2	3	4	5
1	A:	0.12	0.12 ■			
2	E:	0.42	0.42	0.42	0.42 ■	
3	I:	0.09 ■				
4	O:	0.30	0.30	0.30 ■		
5	U:	0.07 ■				
6		UI:	0.16 ■			
7			AUI:	0.28 ■		
8				AUIO:	0.58 ■	
9					EAUIO:	1.00



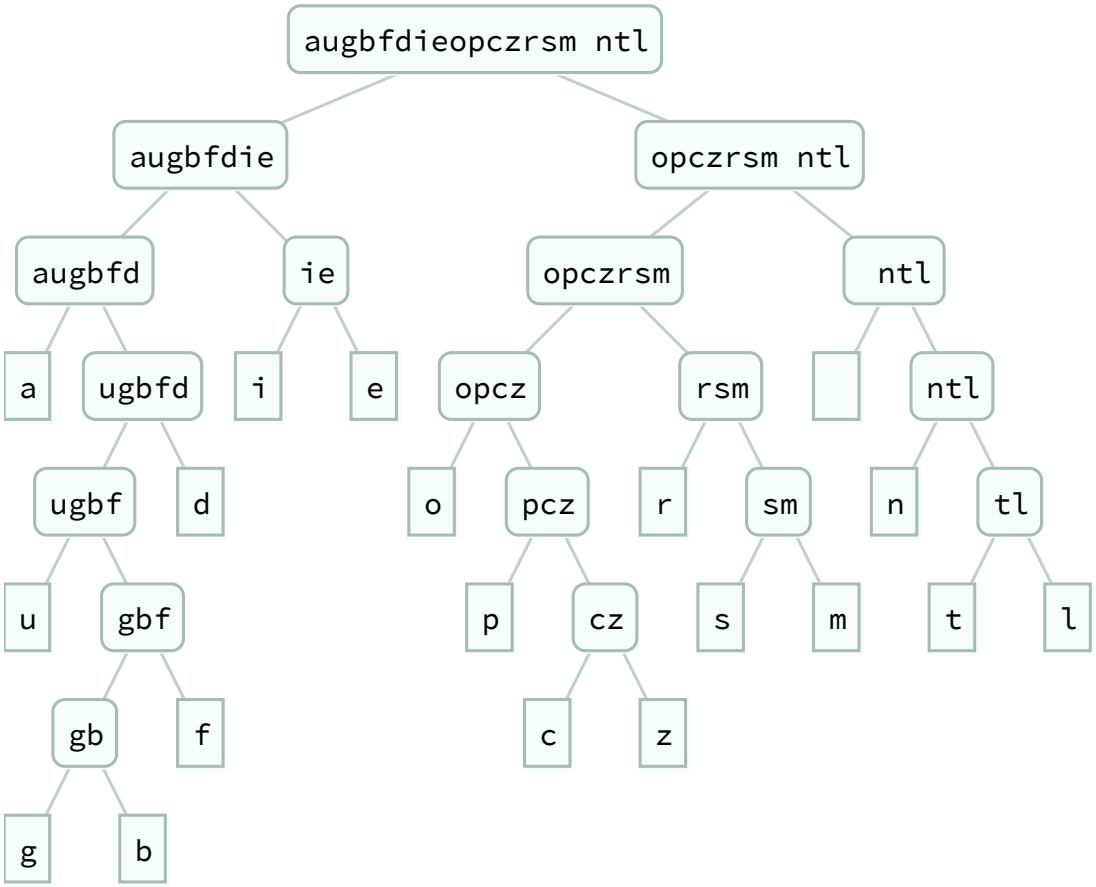
CODICE RISULTANTE

A	100
E	0
I	1011
O	11
U	1010

“la compressione dei dati permette la riduzione del numero di bit necessari alla rappresentazione in forma digitale di una informazione”

‘ ‘	19
e	16
i	15
a	12
n	10
r	9
o	8
d	7
l	6
t	6
m	5
s	5
p	4
u	3
z	3
c	2
f	2
b	1
g	1

a: 000
b: 0010101
c: 100110
d: 0011
e: 011
f: 001011
g: 0010100
i: 010
l: 11111
m: 10111
n: 1110
o: 1000
p: 10010
r: 1010
s: 10110
t: 11110
u: 00100
z: 100111
‘ ‘: 110

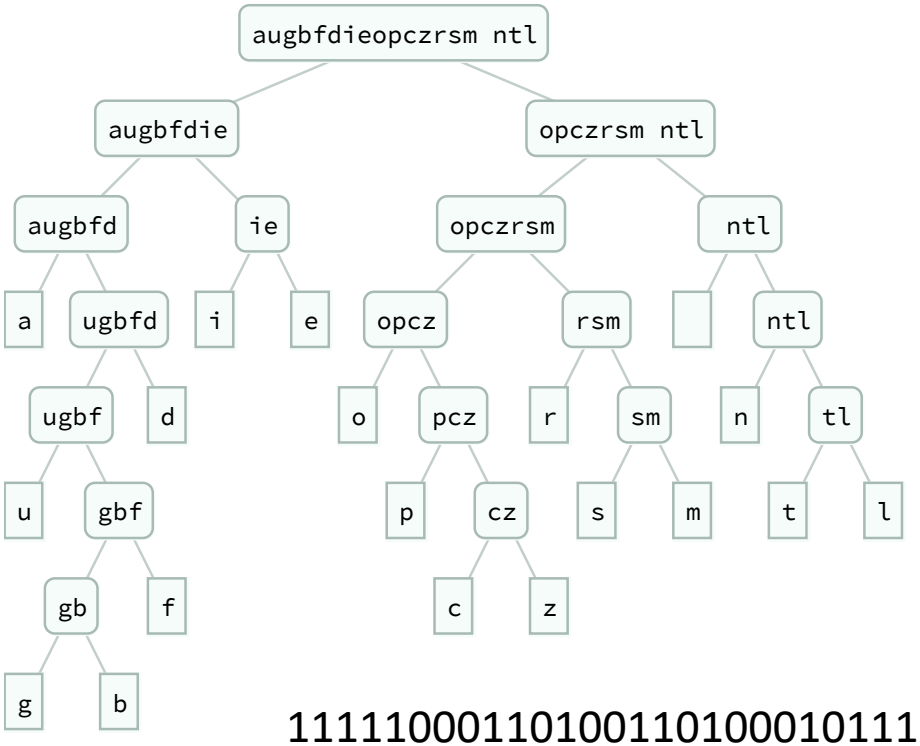


“la compressione dei dati permette la riduzione del numero di bit necessari alla rappresentazione in forma digitale di una informazione”

a:	000
b:	0010101
c:	100110
d:	0011
e:	011
f:	001011
g:	0010100
i:	010
l:	11111
m:	10111
n:	1110
o:	1000
p:	10010
r:	1010
s:	10110
t:	11110
u:	00100
z:	100111
‘ ‘:	110

11111 000 110 100110 1000 10111 10010 1010 011
10110 10110 010 1000 1110 011 110 0011 011 010
110 0011 000 11110 010 110 10010 011 1010 10111
011 11110 11110 011 110 11111 000 110 1010 010
0011 00100 100111 010 1000 1110 011 110 0011
011 11111 110 1110 00100 10111 011 1010 1000
110 0011 010 110 0010101 010 11110 110 1110 011
100110 011 10110 10110 000 1010 010 110 000
11111 11111 000 110 1010 000 10010 10010 1010
011 10110 011 1110 11110 000 100111 010 1000
1110 011 110 010 1110 110 001011 1000 1010
10111 000 110 0011 010 0010100 010 11110 000
11111 011 110 0011 010 110 00100 1110 000 110
010 1110 001011 1000 1010 10111 000 100111 010
1000 1110 011

523 bit, 3.902985 bit/simbolo



Per la decodifica, si percorrono i rami dell'albero dalla radice fino ai nodi finali.

Quando viene raggiunto un nodo finale si segna il valore e si riparte.

11111000110100110100010111100101010011101101011001010001110011111000110110101100011000111100101101001001110101011101111110111001111011111000110101001000110010010011101010001110011110001101011000110001111001011010010011101010111011111101110011110111110001101010010001100100100111010100011100111100011010110001100011110010110100100111010101110111111011100111101111100011010100100011001001001110101000111001111000110101100010011100001100101110001010101111000111110111100011010110001001110000110010111000101010111000100111010100011100111

Edoardo Milotti - Fondamenti Fisici di Tecnologia Moderna