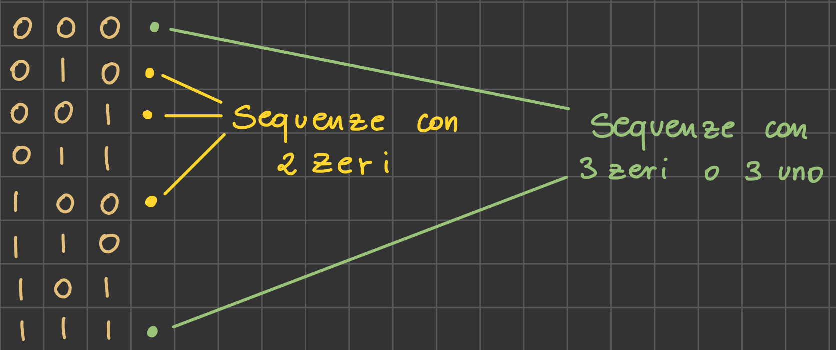
Codificatore di canale e modulatore

Piccolo recap della lezione precedente



Tutte le sequenze sono equiprobabili prese singolarmente, ma a grandi numeri di parole, alcune sequenze hanno meno probabilità di comparire.

Poniamo il caso di avere 8 sequenze di bit; in queste sequenze avremo 3 diverse sequenze in cui compaiono 2 zeri, mentre solo una dove gli zeri sono 3.

Di conseguenza la probabilità di ogni sequenza è la seguente:

000 → $1/8$

010, 001, 100 → $3/8$

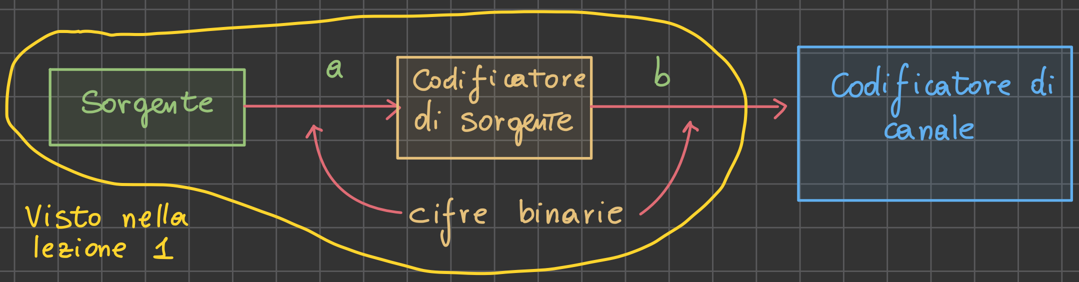
Ora poniamo il caso di avere a disposizione 200 bit e quindi 2^{200} parole. In questo caso la probabilità di una parola composta da soli zeri diventa $1/2^{200}$, ovvero bassissima.

La probabilità delle parole con $n < 2^{200}$ zeri invece va diminuendo quanti più zeri sono presenti (circa) secondo la legge del coefficiente binomiale.

Cosa pensò Shannon?

Siccome alcune seq sono molto rare, nella compressione possiamo direttamente non trasmetterle.

Riprendiamo lo schema visto



Codificatore di canale

Anche in questo caso il cod di can riceve e "sputa" in binario. Il nostro obbiettivo è quello di inviare una certa sequenza lato sender, e poi poterla recuperare, lato receiver, il più fedele possibile.

Quindi, il CDC introduce ridondanza; perché? Viene aggiunta proprio per permettere al receiver di recuperare la sequenza in modo fedele nonostante il rumore dovuto al canale.

Teorema di capacita' di Shannon

Possiamo rendere assolutamente immune il segnale dagli errori dovuti al canale di comunicazione? Possiamo rendere la probabilita' di errore piccola a nostro piacimento?

Risposta: " Si, ma ... "

La velocita' di trasmissione in bit/s deve essere piu' piccola di una grandezza definita come CAPACITA' DEL CANALE.

In parole povere Se la velocita' di trasmissione (RATE, R) e' Minore della capacita' del canale (C), allora la probabilita' di errore e' arbitrariamente piccola.

$\rightarrow$ $R < C \Rightarrow P(e)$ E' piccola

ESEMPIO

Considero una sequenza di 100 bit; li codifico (aggiungo ridondanza) con un'altra seq lunga (per esempio) 120 bit.
Trasmetto la sequenza.
In ricezione decodifico non il singolo bit, ma tutta la sequenza.

Shannon disse che si puo' ottenere probabilita' di errore zero se si trasmettono seq. infinitamente lunghe.

Riassumendo... Schematicamente

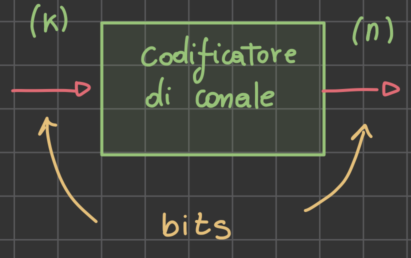
$R < C$ $\longrightarrow$ Quanto piu' e' lunga (codificata) e' la seq. meno probabilita' di errori dovuti al canale avro'.

$R > C$ $\longrightarrow$ Piu' e' lunga la sequenza, peggio e' (tende ad 1?).

Shannon fece un ragionamento: doveva rendere la sequenza quanto piu' a prova di errore; la lunghezza della sequenza e' "n".
Codifichiamo la sequenza nel seguente modo:

ad una sottosequenza di k bit, ne associamo una lunga n, con $n > k$. Stiamo quindi "allungando" la sequenza.

$k \longrightarrow n \qquad n > k$



Facciamo un altro esempio.
Prendiamo una seq di len = 2; associamo ad essa una seq di len 3. La seq. di len. 2 ha 4 configurazioni. Siccome l'altra seq. ha 2^3 conf:

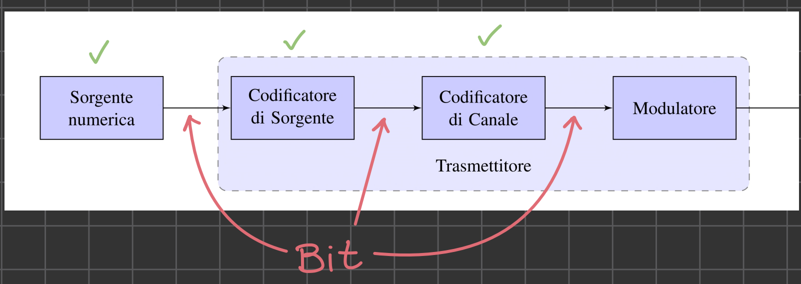
00 → 001
01 → 010
10 → 100
11 → 111

Ci accorgiamo che la sequenza 000 non è prevista, così come altre 3 sequenze che non vengono usate.

Se uno di questi bit viene alterato dal rumore (ad esempio invece di ricevere 001 riceviamo 000, che è un valore non previsto), ce ne accorgiamo subito.

La fase di codifica è un processo semplice, mentre quella di decodifica è più complessa, anche perchè gli algoritmi sono iterativi ed occupano molta memoria.

Facciamo il punto della situazione



Modulatore

Il modulatore va a configurare una forma d'onda adatta al canale di trasmissione. Esso prende in esame un pezzo di una sequenza di bit:

10110111110010...

A seconda del tipo di modulatore, va a raggruppare un certo numero di cifre binarie (k cifre) ed associa ad ogni possibile configurazione una diversa forma d'onda.

Ad esempio:

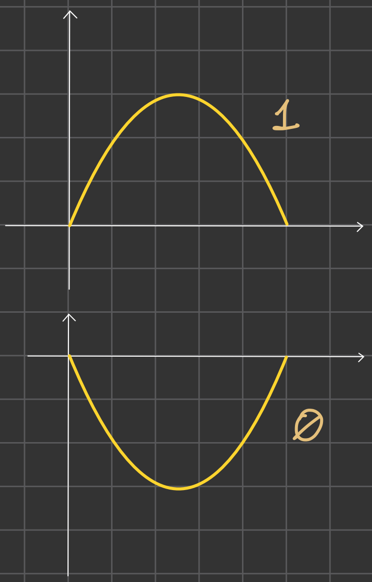
000 → $S_1(t)$
001
010 → $S_2(t)$
011 ⋮
100
101
110
111 $S_n(t)$

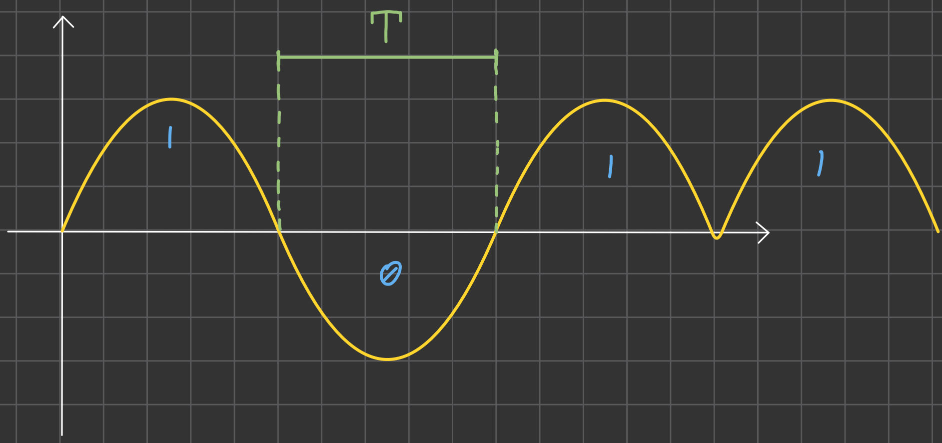
il modulatore (esempio) considera 3 cifre per volta ed ad ogni seq associa una F. d'O. $S(t)$

dove $n = 2^k$

La forma d'onda generata durerà un determinato tempo T. DURATA
Se volessi trasmettere con il modulatore 1 Mb/s; potremmo prendere una sola cifra per volta, quindi $n = 2^1 = 2$ forme d'onda.

Quindi la f.O. risultante sarebbe:





Se vogliamo trasmettere 1 Mb/S, allora T dovrà durare 1 milionesimo di secondo.

Esempio 2: voglio trasmettere 2 bit alla volta $\Rightarrow$ $n = 2^2 = 4$; di conseguenza mi servono 4 forme d'onda.

$\Rightarrow$ Siccome trasmetto 2 bit per volta, T può "allungarsi", quindi la forma d'onda può durare il doppio del tempo.