



Elementi di codifica di canale

Fulvio Babich (babich@units.it)

DIA – Università di Trieste

Tecniche di protezione contro gli errori



- **Codici di rivelazione d'errore e algoritmi di ritrasmissione** (Automatic Repeat reQuest - ARQ).
- **Codici ad autocorrezione d'errore** (Forward Error Correction - FEC).
- **Codifica**: a k bit di informazione vengono associati n bit per la trasmissione sul canale, con $n > k$; viene così introdotta in modo sistematico della ridondanza che può essere usata in ricezione per rivelare e/o correggere errori. Nel tempo in cui vengono generati k bit d'utente devono essere trasmessi n bit di canale; pertanto si riduce il tempo per la trasmissione di un bit e, di conseguenza, aumenta la banda utilizzata.
- **Tasso del codice**: $R_c = k/n < 1$.
- **Codici a blocco**: Gli $n-k$ bit di ridondanza dipendono solo dai k bit d'utente attuali; l'operazione di codifica è, pertanto, senza memoria.
- **Codici convoluzionali**: Gli $n-k$ bit di ridondanza dipendono k bit d'utente attuali e da $(N-1)k$ bit d'utente precedenti (N è detta lunghezza di vincolo); l'operazione di codifica è, quindi, con memoria.

Codici a blocco binari



- Le sequenze di n bit prodotte dal codificatore vengono dette **parole** del codice.
- **Peso** di una parola di codice: è il numero di bit diversi da 0 della parola di codice.
- **Distanza di Hamming** fra due parole di codice: è il numero di bit in cui differiscono; è anche il peso della parola ottenuta sommando modulo 2 le parole di codice. I codici che considereremo sono **codici lineari**, nei quali la somma fra due parole di codice è ancora una parola di codice. Pertanto, il peso della parola di peso minimo rappresenta la **distanza minima**, d_{\min} , fra due parole di codice. Per i codici lineari la parola nulla è una parola di codice.
- Prestazioni di un codice: dipendono dalla distanza minima. L'operazione di decodifica (hard) avviene in genere a massima verosimiglianza, associando cioè alla n -pla ricevuta la parola di codice più vicina.
- Un codice con distanza minima d_{\min} può correggere al più errori di peso $\lfloor (d_{\min} - 1)/2 \rfloor$, e rivelare al più errori di peso $d_{\min} - 1$. Un codice viene usato per correggere errori di peso al più $t_c \leq \lfloor (d_{\min} - 1)/2 \rfloor$ e rivelare errori di peso l , dove l soddisfa la $t_c < l < d_{\min} - t_c$.

Codici a blocco a verifica di parità (*parity check codes*)



- Sono codici lineari.
- **Codifica:** si effettua utilizzando la matrice generatrice \mathbf{G} [$k \times n$]; il vettore codificato \mathbf{x} si ottiene dal vettore non codificato \mathbf{u} mediante l'operazione $\mathbf{x} = \mathbf{u} \mathbf{G}$ (operazioni modulo 2).
- **Rivelazione d'errore** (verifica di parità): si effettua utilizzando la matrice di parità \mathbf{H} [$(n-k) \times n$]; se il vettore ricevuto è $\mathbf{r} = \mathbf{x} \oplus \mathbf{e}$, dove \mathbf{e} è il vettore di errore, si calcola $\mathbf{s} = \mathbf{r} \mathbf{H}'$ (dove \oplus indica l'operazione di somma modulo 2 e $'$ indica l'operazione di trasposizione). Se non ci sono stati errori il vettore \mathbf{s} è il vettore nullo. Risulta essere $\mathbf{G} \mathbf{H}' = \mathbf{0}$. Il vettore \mathbf{s} viene denominato sindrome.
- **Correzione d'errore:** a ogni vettore \mathbf{s} diverso dal vettore nullo associo un vettore \mathbf{e}_1 , in modo che il vettore decodificato sia $\mathbf{d} = \mathbf{r} \oplus \mathbf{e}_1$. L'associazione avviene a minima distanza (a ogni sindrome associo la sequenza d'errore a peso minimo che produce quella data sindrome). Le sindromi non nulle (e quindi le sequenze d'errore correggibili) sono $2^{n-k} - 1$.
- **Codici sistemati:** i primi k bit della parola di codice sono i bit d'utente. Per essi $\mathbf{G} = [\mathbf{I}_k : \mathbf{P}]$, $\mathbf{H} = [\mathbf{P}' : \mathbf{I}_{n-k}]$.

Codici ciclici



- Sono codici lineari. Una permutazione ciclica di una parola di codice produce una parola di codice.
- A ogni sequenza binaria viene associato un polinomio i cui coefficienti sono i bit che compongono la sequenza.
- I bit di ridondanza vengono individuati utilizzando un algoritmo di divisione. Il polinomio divisore viene indicato con $g(x)$, e ha grado $n-k$. Pertanto un codice ciclico è caratterizzato dalla terna $n, k, g(x)$.
- Codifica sistemata.
 - Alla k -pla da codificare si associa un polinomio $u(x)$ di grado $n-1$, i cui k coefficienti più significativi coincidono con i bit d'utente, mentre gli altri $n-k$ bit sono posti a 0.
 - Si calcola il resto, $r(x)$, della divisione di $u(x)$ per $g(x)$. I coefficienti di $r(x)$ sono la ridondanza. La n -pla così ottenuta è associata a un polinomio $c(x)$, divisibile per $g(x)$.
- La sequenza ricevuta viene associata al polinomio $\hat{c}(x)$. Si effettua la divisione per $g(x)$.
- Se il resto (sindrome) è nullo si suppone che non ci siano stati errori.

Codici per la rivelazione d'errore



- **Codici (concatenati) a parità singola (orizzontale + verticale)**
 - La sequenza binaria viene organizzata in una matrice di N righe di k bit ciascuna (tipicamente $k=8$).
 - A ogni riga viene associato un bit di parità (i $k+1$ bit così ottenuti contengono un numero pari di bit '1'; parità orizzontale). Alle N righe viene associato un'ulteriore riga di $k+1$ bit, tale che vi sia un numero pari di bit '1' in ciascuna colonna (parità verticale).
 - Questo schema che ha un tasso (efficienza)
$$R_c = kN / [(k+1)(N+1)] \approx k / (k+1)$$
(l'efficienza tipica è pari a circa $8/9$) è in grado di rivelare tutti gli errori di peso dispari e gli errori di peso 2.
- **Codici ciclici (*Cyclic Redundancy Check* - CRC)**
 - Il polinomio generatore è del tipo $g(x)=(x+1)g'(x)$, dove $g'(x)$ è un polinomio di grado $m=n-k-1$ scelto in modo da essere un divisore di $x^{2^m-1} + 1$ ma di non essere divisore di alcun polinomio del tipo x^h+1 , con $h < 2^m-1$.
 - Con tali scelte, il codice consente di rivelare tutti gli errori di peso dispari e tutti gli errori di peso 2 purché $n < 2^m-1$. Da cui $R_c \leq (2^m-m-3)/(2^m-2)$.

Rivelazione d'errore: CRC



- Ridondanza determinata associando la sequenza d'informazione a un polinomio di grado k . La ridondanza è il resto della divisione per il polinomio $g(x)=(x+1)g'(x)$ di grado $n-k$.
- Rivela tutti gli errori di peso dispari e gli errori doppi.
- Vincoli: fissata la ridondanza $(n-k)$, $n < 2^{n-k-1} - 1$. Da cui $k < 2^{n-k-1} - (n-k) - 1$.
- Esempi:

$n - k$ [bit]	polinomio generatore	n [bit]	k [bit]	$R_c = \frac{k}{n}$	Ente
5	$g(x) = x^5 + x^3 + x + 1$	<15	<10	$<2/3$	ITU-T
8	$g(x) = x^8 + x^7 + x^3 + x^2 + 1$	<127	<119	$<119/127$	ITU-T
16	$g(x) = x^{16} + x^{12} + x^5 + 1$	<32767	<32751	≈ 1	
32	$g(x) = x^{32} + x^{26} + x^{23} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$	$<10^9$	$<10^9$	≈ 1	IEEE
64	$g(x) = x^{64} + x^4 + x^3 + x + 1$	$<10^{18}$	$<10^{18}$	≈ 1	ISO

Codici per la correzione d'errore.

I codici di Hamming



- Sono codici a verifica di parità, (possono essere ciclici).
- Parametri: $n=2^m-1$, $k=2^m-m-1$, $d_{\min}=3$.
- Sono codici a elevato tasso, in grado di **correggere errori singoli**.
- La matrice di parità è composta da tutte le sequenze di lunghezza m , eccetto la sequenza nulla.
- Esempio, $m=3$, $n=7$, $k=4$, versione sistematica e ciclica: $g(x)=x^3+x+1$.

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

- In caso di errore singolo, la sindrome coincide con la colonna di \mathbf{H} che corrisponde alla posizione dell'errore.
- Codici BCH: $n=2^m-1$, $n-k \leq mt$, $d_{\min} \geq 2t+1$

Codici per la correzione d'errore.

I codici BCH (Bose-Chaudhuri-Hocquenghem)

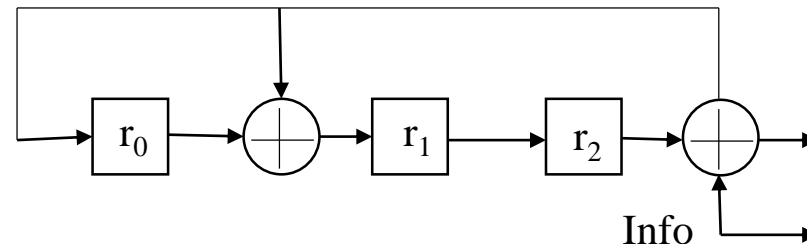


m	n	k	t	m	n	k	t	m	n	k	t	m	n	k	t	m	n	k	t
3	7	4	1	6	63	24	7	7	127	50	13	8	255	187	9	8	255	71	29
4	15	11	1			18	10			43	14			179	10			63	30
		7	2			16	11			36	15			171	11			55	31
		5	3			10	13			29	21			163	12			47	42
5	31	26	1			7	15			22	23			155	13			45	43
		21	2	7	127	120	1			15	27			147	14			37	45
		16	3			113	2			8	31			139	15			29	47
		11	5			106	3	8	255	247	1			131	18			21	55
		6	7			99	4			239	2			123	19			13	59
6	63	57	1			92	5			231	3			115	21			9	63
		51	2			85	6			223	4			107	22	9	511	502	1
		45	3			78	7			215	5			99	23			493	2
		39	4			71	9			207	6			91	25			484	3
		36	5			64	10			199	7			87	26			475	4
		30	6			57	11			191	8			79	27			466	5

Codifica di un codice ciclico

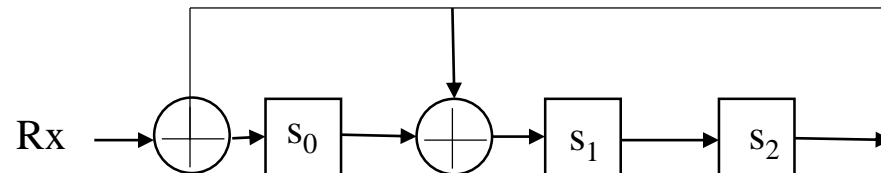


- Si può effettuare utilizzando un registro a scorrimento:
Esempio codice di Hamming (7,4) con generatore $g(x)=x^3+x+1$.



Contenuto azzerato; entrano i 4 bit di informazione; dopo il loro transito nel registro rimane il resto che esce interrompendo la reazione).

- In ricezione lo stesso dispositivo viene utilizzato per determinare la **sindrome** che, in caso di errore singolo, coincide con la colonna di **H** che corrisponde alla posizione dell'errore (codici di Hamming).



- Lo stesso dispositivo, senza ingresso e inizializzato con un contenuto diverso da 0, è utilizzato per generare le **sequenze PN**.

Esempio



- Esempio: la sequenza da codificare sia 1010. Codifica.

Info	r_0	r_1	r_2
1	1	1	0
0	0	1	1
1	0	0	1
0	1	1	0

- Quindi la parola di codice è 1010011.
- Supponiamo ora che sia ricevuta la sequenza 1110011. Calcolo sindrome.

Rx	s_0	s_1	s_2
1	1	0	0
1	1	1	0
1	1	1	1
0	1	0	1
0	1	0	0
1	1	1	0
1	1	1	1

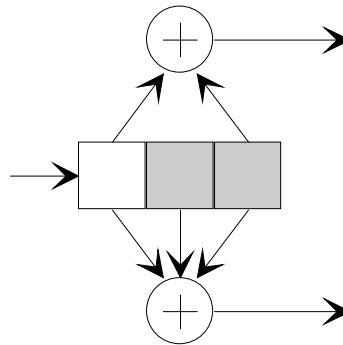
- È la seconda colonna di H.

Generatori sequenze PN

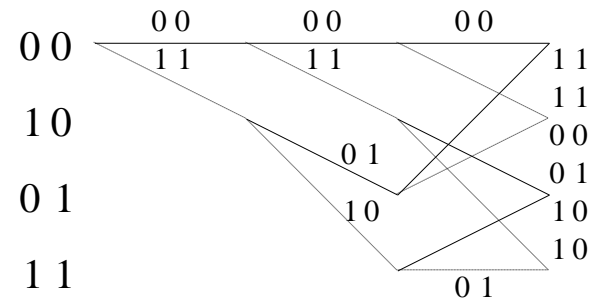
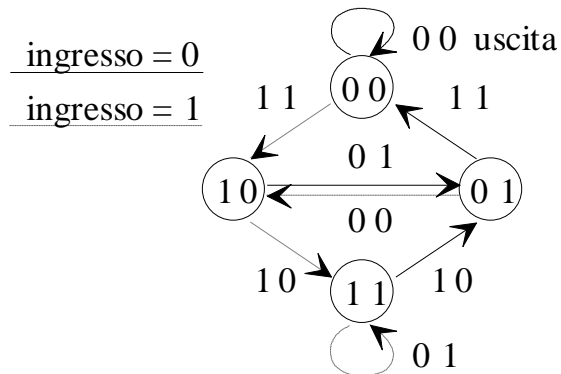
- Nella seguente tabella sono indicati in notazione ottale i generatori delle sequenze PN (divisori di x^P+1). Nella tabella $P=2^L-1$ indica il periodo della sequenza, L il numero di elementi presenti nel registro a scorrimento (corrispondenti al grado del polinomio generatore), e g (in notazione ottale), rappresenta i coefficienti del polinomio generatore, con l'esclusione del termine x^L (esempio $g=[3]=011$, con $L=3$, corrisponde al polinomio $g(x)=x^3+x+1$).

P	L	g
7	3	[3], [5]
15	4	[0 3], [1 1]
31	5	[0 5], [1 1]
63	6	[0 3], [4 1]
127	7	[0 1 1], [1 0 1]
255	8	[0 3 5], [1,6,1]
511	9	[0 4 1]
1023	10	[0 2 0 1]
2047	11	[1 0 0 1]
4095	12	[6 0 2 1]

Codici convoluzionali



- Esempio: Tasso $R_c=1/2$. Sono evidenziati i valori precedenti dell'ingresso che hanno influenza sul valore attuale dell'uscita (tali valori definiscono lo stato del sistema). Dicesi lunghezza di vincolo, L , la dimensione del registro (nell'esempio $L=3$). Le relazioni ingresso/uscita definiscono i generatori che vengono indicati in notazione ottale. Nell'esempio $g_1=(1\ 0\ 1)$ (si indica con 5), $g_2=(1\ 1\ 1)$ (si indica con 7).
- L'evoluzione si descrive mediante diagramma di stato o diagramma a traliccio. Decodificare equivale a identificare la sequenza a distanza minima sul traliccio (algoritmo di Viterbi).



Modulazione lineare (*hard decoding*)

- Si consideri una modulazione lineare a $M=2^n$ simboli
- La forma d'onda i -ma è descritta dal vettore \mathbf{s}_i . (mono o bi-dimensionale).
- Se trasmetto la forma d'onda j -ma, La forma d'onda ricevuta è rappresentata dal vettore $\mathbf{r} = \mathbf{s}_j + \mathbf{v}$, dove \mathbf{v} è il termine di rumore.
- Lo spazio vettoriale N -dimensionale cui appartiene \mathbf{r} , viene suddiviso in M regioni, a 2 a 2 disgiunte, A_m (regioni di decisione o di Voronoi).
- Criterio MAP $\mathbf{r} \in A_m : m = \arg \max_j (f(\mathbf{r}|\mathbf{s}_j)p(\mathbf{s}_j))$
- Criterio ML: $\mathbf{r} \in A_m : m = \arg \max_j (f(\mathbf{r}|\mathbf{s}_j))$ (minima distanza)
- Probabilità di corretta decisione: $p_c = \sum_m p(\mathbf{r} \in A_m | \mathbf{s}_m) p(\mathbf{s}_m) = \sum_m p(\mathbf{s}_m) \int_{\mathbf{r} \in A_m} f(\mathbf{r}|\mathbf{s}_m) d\mathbf{r}$
- Scelta binaria (MAP): $r \in A_0 : \Lambda = \log \left[\frac{f(r|s_1)}{f(r|s_0)} \right] > \log \left[\frac{p(s_0)}{p(s_1)} \right]$
 Λ è detto **rapporto di verosimiglianza logaritmico (log-likelihood ratio)**.
- Se $p(s_0)=p(s_1)$, oppure se le due probabilità non sono note si adotta il criterio ML: $r \in A_0 : \Lambda > 0$.

Square root SNR signal space



- *Square root energy signal space:* $f(\mathbf{r}|\mathbf{s}_j) = \frac{1}{(\pi N_0)^{N/2}} \exp\left(-\frac{1}{N_0} \sum_{h=1}^N (r_h - s_{jh})^2\right)$
- *Cambiamento di variabile:* $y_h = \frac{r_h}{\sqrt{N_0}} \quad x_{jh} = \frac{s_{jh}}{\sqrt{N_0}}$
- *Square root SNR signal space:* $f(\mathbf{y}|\mathbf{x}_j) = \frac{1}{(\pi)^{N/2}} \exp\left(-\sum_{h=1}^N (y_h - x_{jh})^2\right)$
- *Pairwise error probability:* $p_e = Q(d_t^y \sqrt{2})$ essendo d_t^y la distanza dalla soglia nel nuovo sistema di coordinate.
- *Log-Likelihood Ratio modulazione binaria antipodale:* $L = \log \frac{f(y|x_0)}{f(y|x_1)} = 4y \sqrt{\frac{E_s}{N_0}}$

MAP in presenza di un codice



$$\begin{aligned}\log \frac{p(x_i = +|\mathbf{r})}{p(x_i = -|\mathbf{r})} &= \log \frac{p(\mathbf{r}|x_i = +)p(x_i = +)}{p(\mathbf{r}|x_i = -)p(x_i = -)} = \log \frac{p(r_1, r_2, \dots, r_n | x_i = +)p(x_i = +)}{p(r_1, r_2, \dots, r_n | x_i = -)p(x_i = -)} = \\ &= \log \frac{p(r_1, r_2, \dots, r_{i-1}, r_{i+1}, \dots, r_n | x_i = +)p(r_i | x_i = +)p(x_i = +)}{p(r_1, r_2, \dots, r_{i-1}, r_{i+1}, \dots, r_n | x_i = -)p(r_i | x_i = -)p(x_i = -)} \\ &= \log \frac{p(r_1, r_2, \dots, r_{i-1}, r_{i+1}, \dots, r_n | x_i = +)}{p(r_1, r_2, \dots, r_{i-1}, r_{i+1}, \dots, r_n | x_i = -)} \quad \text{Codice} \\ &+ \log \frac{p(r_i | x_i = +)}{p(r_i | x_i = -)} \quad \text{Canale} \\ &+ \log \frac{p(x_i = +)}{p(x_i = -)} \quad \text{A priori}\end{aligned}$$

Codici di parità: matematica delle APP



- $$L = \log \frac{p_+}{p_-} = \log \frac{p_+}{1 - p_+} \quad p_+ = \frac{\exp(L)}{1 + \exp(L)}$$

Consideriamo una coppia di bit (antipodali), u_1, u_2 . $u_1 \bullet u_2 = +$ se $u_1 = u_2$.

$$p_+(u_1 u_2) = p(u_1^+)p(u_2^+) + (1 - p(u_1^+))(1 - p(u_2^+)) = 1 - p(u_1^+) - p(u_2^+) + 2p(u_1^+)p(u_2^+)$$

$$p_-(u_1 u_2) = p(u_1^+) + p(u_2^+) - 2p(u_1^+)p(u_2^+)$$

Sostituendo, dopo alcuni passaggi si ottiene

$$p_+(u_1 u_2) = \frac{1 + \exp(L(u_1))\exp(L(u_2))}{(1 + \exp(L(u_1)))(1 + \exp(L(u_2)))}$$

$$p_-(u_1 u_2) = \frac{\exp(L(u_1)) + \exp(L(u_2))}{(1 + \exp(L(u_1)))(1 + \exp(L(u_2)))}$$

$$L(u_1 u_2) = \log \left[\frac{1 + \exp(L(u_1))\exp(L(u_2))}{\exp(L(u_1)) + \exp(L(u_2))} \right]$$

- Generalizzando (si dimostra per induzione, partendo da $J=2$)

$$L(u_1 u_2 \dots u_J) = \log \left[\frac{\prod_{j=1}^J (\exp(L(u_j)) + 1) + \prod_{j=1}^J (\exp(L(u_j)) - 1)}{\prod_{j=1}^J (\exp(L(u_j)) + 1) - \prod_{j=1}^J (\exp(L(u_j)) - 1)} \right]$$

$$\text{Detto } x = \frac{\prod_{j=1}^J (\exp(L(u_j)) - 1)}{\prod_{j=1}^J (\exp(L(u_j)) + 1)} = \frac{\prod_{j=1}^J (\exp(L(u_j)/2) - \exp(-L(u_j)/2))}{\prod_{j=1}^J (\exp(L(u_j)/2) + \exp(-L(u_j)/2))} = \prod_{j=1}^J \tanh(L(u_j)/2)$$

$$\text{Si ottiene: } L(u_1 u_2 \dots u_J) = \log \left[\frac{1+x}{1-x} \right]$$

$$\text{da cui } x = \frac{\exp(L(u_1 u_2 \dots u_J)) - 1}{\exp(L(u_1 u_2 \dots u_J)) + 1} = \tanh(L(u_1 u_2 \dots u_J)/2)$$

$$\text{Sostituendo: } L(u_1 u_2 \dots u_J) = 2 \operatorname{arctanh}(x) = 2 \operatorname{arctanh} \left(\prod_{j=1}^J \tanh(L(u_j)/2) \right)$$

Single parity check (SPC)



- Si consideri un codice a verifica di parità (una parola di codice, \mathbf{c} , è costituita da K bit di informazione e 1 bit di ridondanza tale che il numero di bit di valore 1 nella parola è in numero pari).
- Si consideri la versione binaria antipodale $\mathbf{d}=1-2\mathbf{c}$. Nella parola \mathbf{d} , i valori pari a '-1' sono in numero pari. Pertanto valgono le relazioni: $\prod_{k=1}^{K+1} d_k = 1$, $d_k = \prod_{h=1, h \neq k}^{K+1} d_h$
- Di conseguenza la stima soft del valore k -mo derivante dalla stima degli altri valori (contributo del codice alla stima complessiva) si ottiene con la relazione determinata in precedenza (stima soft di un prodotto).
- Se si organizzano i dati una matrice bidimensionale, con R righe e C colonne, e si aggiunge una riga di parità verticale e una colonna di parità orizzontale, si ottiene un codice prodotto in cui ogni bit può essere sottoposto a una doppia stima soft (di riga e di colonna), $R_c = \frac{RC}{(R+1)(C+1)}$ consentendo una decodifica iterativa.
- Sono possibili configurazioni più complesse, per ottenere prestazioni migliori e tassi di codifica opportuni. (F. Babich, "Design of Adaptive Systems for the Fading Channel Adopting Efficient Coded Modulations", IEEE ICC 2006, Istanbul, Turkey, 11-15 June 2006.)

Probabilità d'errore di blocco (o di parola - *word error probability*); errori indipendenti.



- Decodifica mediante decisione morbida (*soft decision decoding*). Introduciamo il concetto di distanza euclidea fra due parole di codice i, j , d_{ij}^E , mentre la distanza di Hamming sia $d_{ij}^H \geq d_{\min}$.

- Modulazione binaria antipodale:
$$\left(d_{ij}^E\right)^2 = \sum_{\substack{k=1 \\ k:c_{ik} \neq c_{jk}}}^n \left(\pm 2\sqrt{E_g}\right)^2 = 4E_g d_{ij}^H$$

- In generale, la probabilità d'errore fra due parole è: $P_{eij} = Q\left(d/\sqrt{2N_0}\right)$

- Pertanto, applicato lo *union bound* e introdotto il tasso del codice $R_c = k/n$ (E_b = energia dei bit di sorgente) otteniamo:

modulazione binaria antipodale:
$$P_W \leq \left(2^k - 1\right)Q\left(\sqrt{2R_c d_{\min} E_b / N_0}\right)$$