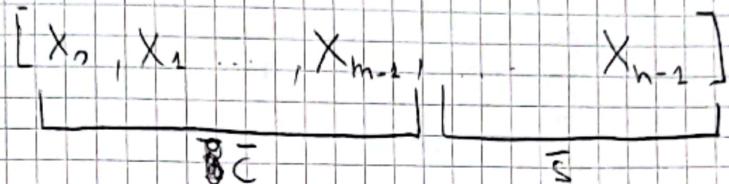


Tesine

menejano $\bar{U} [U_0, U_1, \dots, U_{k-1}]$ k bit
 ↓
 codifica

(a) per le parole di codice $\bar{X} [X_0, X_1, \dots, X_{n-1}]$ n bit



$$\bar{X} = [\bar{C} \mid \bar{S}] \quad \bar{C} = \text{check bit seguenti } (m) = n - k$$

\bar{S} = menejano di codice (m)

$$H = \begin{bmatrix} h_{00}, h_{01}, \dots, h_{0n} \\ \vdots \\ h_{m0}, h_{m1}, \dots, h_{mn} \end{bmatrix}^n \quad m \times n$$

$$= \left[\begin{array}{c|c} h_{00} & h_{01} \\ \hline A & B \\ h_{m0} & h_{mm} \end{array} \right] \quad \begin{array}{l} A \text{ matrice quadrata } m \times m \\ B \text{ restante } (n-m) \times m \\ k \times m \end{array}$$

per avere una parola di codice, in base alla def di

$$H \cdot \bar{X}^\top = 0 \quad \circ \quad \bar{X} \cdot H^\top = 0$$

$$A\bar{C} + B\bar{S} = 0 \Rightarrow \bar{C} = A^{-1} \cdot B \cdot \bar{S}$$

~~matrice~~

$$\text{entra } \bar{S} \Rightarrow \bar{C} = A^{-1} \cdot B \cdot \bar{S} \Rightarrow \bar{X} = [\bar{C}, \bar{S}]$$

G non è esplicito! $G: \bar{U} \cdot G = \bar{X}$

Altri metodi per dare tracce di esplosa

$$\bar{x} = \bar{U} \cdot G$$

$$= \bar{U} \cdot \left[\underbrace{I_K}_{\text{I densità di}} \quad | \quad F \right]_n$$

I densità di
riportare il messaggio
nei primi K bit

$$\bar{U} \cdot I_K$$

$$\bar{U} \cdot F$$

$$= \left[\underbrace{\bar{U}}_{\text{per sistematiche}} \quad | \quad \underbrace{\bar{U} \cdot F}_{\text{da aggiungere alle}} \right]$$

Il messaggio

Chiamiamo $\bar{U} \cdot F = \bar{C}$ dove è la parte di check aggiunta

$$\Rightarrow \bar{C} = \bar{U} \cdot \bar{F}$$

$$F \in (K, x (n-K) = m)$$

$$\bar{H} \cdot x^T = \bar{C}$$

$$H = \left[\begin{array}{c|c} A & B \\ \hline n & m \end{array} \right]^T$$

$$B = m \times m$$

$$A = m \times (n-m) = k$$

$$\Rightarrow \bar{H} \cdot x^T = [A \mid B] \cdot x^T = [A \mid B] \cdot [U, C]^T$$

$$= A \cdot U^T + B \cdot C^T = 0$$

$$(H \cdot x^T)^T = \bar{U} \cdot A^T + \bar{C} \cdot B^T = 0$$

$$\Rightarrow (H \cdot x^T)^T \cancel{B^{-1T}} = \bar{U} \cdot A^T \cancel{B^{-1T}} + \bar{C} \cancel{B^{-1T}} = 0$$

moltiplicare

$\times B^{-1T}$ e da es

$$\bar{C} = \bar{U} \cdot (A^T \cdot B^{-1})$$

non c'è il meno sinistra n. 2]

Sono ormai visti, sono
punti da mettere con A, B
invertiti.

Ricavo G esplicito

scrivendo che $\bar{U} \cdot F = \bar{C}$

$$\Rightarrow F = A^T \cdot B^{-1}$$

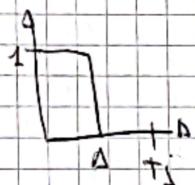
$$\Rightarrow G = \left[\begin{array}{c|c} I_k & A^T \cdot B^{-1} \end{array} \right]$$

Tressmissione

AWGN usata modulazione 2PAM

Q_i simboli antipodici

$$b_i \rightarrow Q_L \sim u_B f(t)$$



$$x(t) = \sum a_i f(t - i\tau_s)$$

$$b = 1 \rightarrow 1.0 + j0$$

$$r = \sigma_x^2 \cdot \alpha$$

~~Spettro~~

a V.A. gaussiana
con media 0 e $\sigma_x^2 = 1$

r V.A. con

$$\mu_r = \sigma_x \cdot \alpha = 0$$

$$\sigma_r^2 = \sigma_x^2 \cdot \alpha^2 = \sigma_x^2$$

$$\sigma_d^2 = 1$$

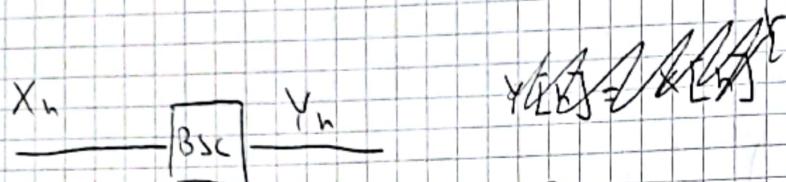
BSC

nel canale verso X V.A.

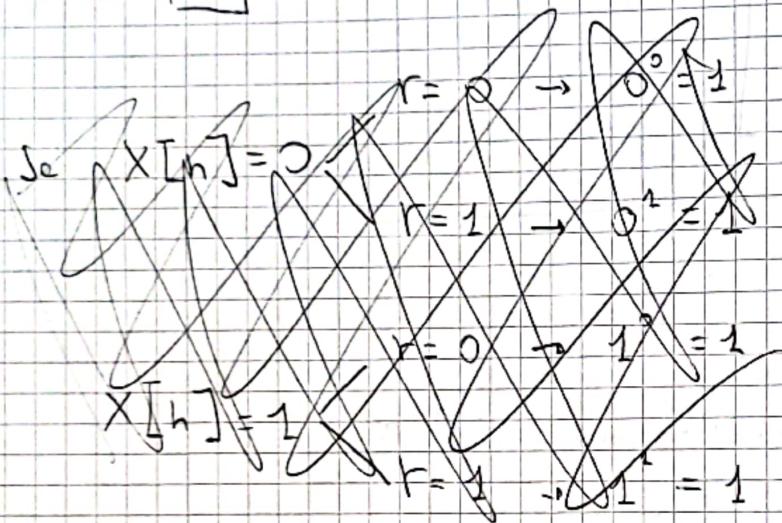
errore di scarto
?

V.A uniforme $[0, 1]$

$$r = \begin{cases} 0 & \text{se } 2 < p_e \\ 1 & \text{se } 2 \geq p_e \end{cases}$$



~~Se $X_n = 1$ e $r = 1$ allora $Y_n = 0$~~



$$Y_n = X_n \oplus r$$

$$X_n \quad r \quad | \quad Y_n$$

0	0	0
1	0	1

Se $r = 0$ bit inalterato

0	1	1
1	1	0

$r = 1$ X_n scontrollato

Codici LDPC (Low density parity check)

Sono codici a blocchi lineari

$$\bar{X} = \bar{J} \cdot G \rightarrow \mathcal{C} \left\{ \bar{X}; \bar{X} = \bar{J} \cdot G + \bar{U} \right\}$$

matrice generatrice

In alternativa

$$\mathcal{C} = \left\{ \bar{X}; \bar{X} \cdot H^T = 0 \right\}$$

matrice di controllo per le righe

per le righe che sono orthonormali

de H

per le righe che sono ortogonalmente derivate da G

È una prop. di H si userà per la sindrome

$$X \cdot H^T = 0 \Rightarrow \begin{cases} m \text{ eguali} \\ n \text{ indipendenti} \end{cases}$$

$$\Rightarrow \text{Se il rango}(H) = m = n - k$$

→ posso avere che $m > n - k = \text{rango}(H)$ alcune equazioni sono ridondanti.

$$q^{n-m} = q^k \text{ bloccini per cod. (1)}$$

- Low density ~~Dense~~ è riferito al numero di '1' in H è molto scarsi, è una matrice sparsa. Questo facilita la decodifica.

Le prestazioni sono simili a ML encoding

- LDPC codice regolare

r = peso di tutte le righe costante di H

$C = u \quad v \quad w \quad \vdots \quad z$ colonne

$$R_c = \frac{k}{n} \geq \frac{n-m}{n} = 1 - \frac{m}{n}$$

matrice sparsa

$$\frac{r}{n} = \frac{c}{m} \ll 1 \quad \text{es } 0.1$$

$\frac{1}{\text{peso righe}} \quad \frac{\text{numeri righe}}{\text{peso righe}}$

- LDPC Codice irregolare

peso di righe e colonne ≠ tra loro

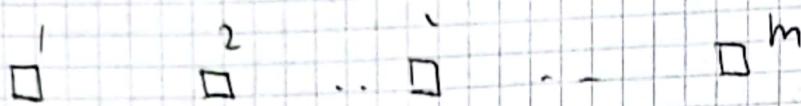
sono più performanti solo quelli con opportuna distribuzione di 1 tra righe e colonne

Come si rappresentano?

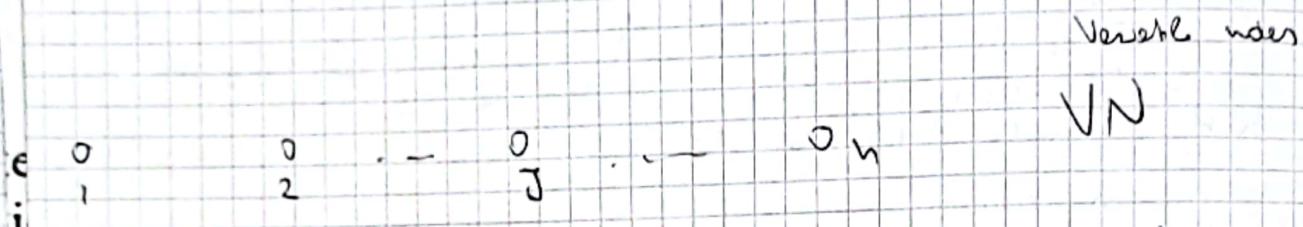
Grafo di Tanner: i grafi sono importanti perché ci fanno ricavare gli algoritmi

è un grafo bipartito

Variable nodes }
check nodes }
 2 classi di nodi



check nodes
CN

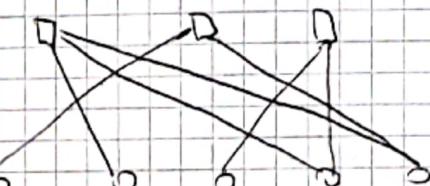


VN

\exists Un VN per ogni elemento di $\bar{X} = (x_1 \dots x_n)$
 \exists Un CN per ogni riga di $H \Rightarrow$ check equation $\sum_{j=1}^n x_j h_{ij} = 0$

Esempio

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



m=5

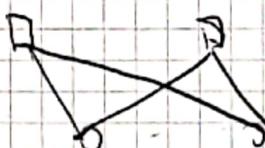
n=3

metti un collegamento dove $h_{ij} \neq 0$

Se grafo è il supporto per algoritmo di decodifica.

Dov'è esistere i cicli

$$H = \begin{bmatrix} & & & & \\ 1 & - & 1 & & \\ & \vdots & & \vdots & \\ 1 & - & 1 & & \end{bmatrix}$$



ciclo lungo

Quando disegni su vertici → di un rettangolo

Girth: la minima lunghezza di un ciclo nel grafo (se ci sono)

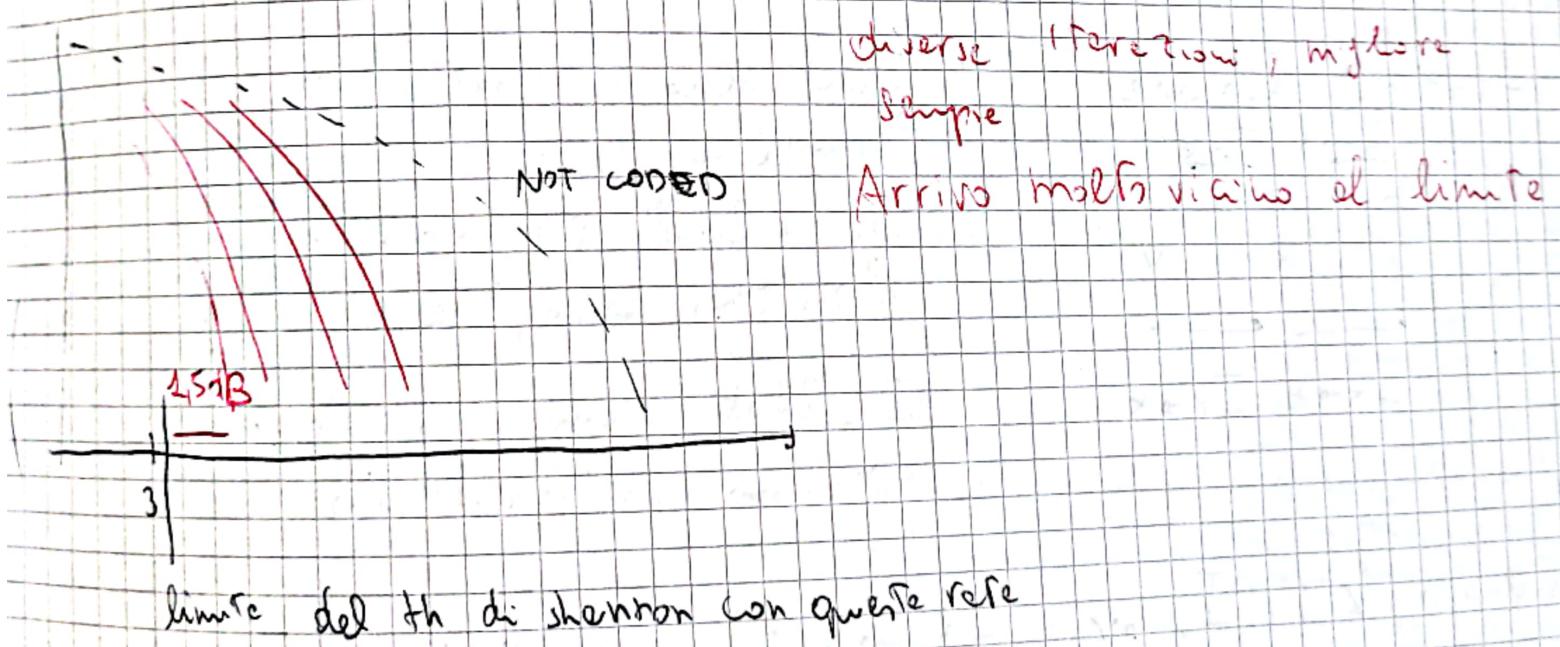
Se ci sono cicli doppie questo grande

Se ci sono cicli e sono costanti l'algoritmo funziona male

- COSTRUZIONE DEL CODICE
 - random con vincoli sui pesi r e c ad eventuali presenze di cicli girth $> h$
 - ⇒ è difficile da codificare deve risolvere sisteme di H eq in m incognite. Oppure la matrice $H \rightarrow G$ poi con G costituito. Ma operazioni cicliche con numeri grandi non è da trascurare
 - una struttura del codice ciclico [H ha shift di righe]
 - Codice quasi-ciclico

tuttavia $H \rightarrow$ grafo di tenne → COSTRUZIONE DEC.

Esempio codice LDPC (6564, 6036) anche qui i codici vengono generati solo lunghi famiglie QC-IRA. Quasi ciclico irregolare repeat accumulate è un esempio



Algoritmo di decodifica

Procedura del Proximity distribuito

Abbiamo una micro unità di DSC che in ogni modo del grafico di Terreni. Usiamo i rami per scambiare info tra i diversi nodi.

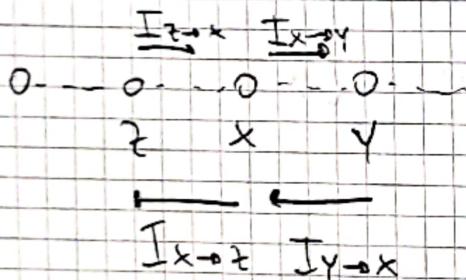
In ogni modo c'è un piccolo algoritmo di decodifica

Alg distribuiti di tipo message-passing

- rete di nodi
- Ogni nodo applica un algoritmo di calcolo
- i nodi connesi scambiano informazioni ["nodi vicini"] usate negli algoritmi

Esempio: conteggio dei soldati in formez ordinate

- formaz. lineare



Quanti sono i soldati? Non usiamo del formattore.

Ogni soldato può implementare un algoritmo semplice

I soldati vicini si scambiano info
operano in parallelo

• Soldato X: $I_{x \rightarrow y} = I_{z \rightarrow x + 1}$

Somme 1 all'inf. che ti dà il tuo vicino e all'altro tuo vicino

Nei 2 versi

$$I_{z \rightarrow x} = I_{x \rightarrow y} + 1$$

↓ ↓
OUTPUT istante $i+1$ INPUT istante i

Dopo un tot di calcoli ogni soldato sa il numero totale prendendo info da $I_{x \rightarrow y}$, inf. da $I_{z \rightarrow x+1}$ e aggiungendo 1

$$I_{y \rightarrow x} + I_{z \rightarrow x+1} = 5 \text{ per tutti i regimi}$$

time	0	0	0	0
0	0	1	1	1
1	1	2	2	2
2	1	2	3	3
3	1	2	3	4
4	1	2	3	4

dx₀₊₁ + dx₀₊₁ + dx₀₊₁ = 3 per tutti i arrivati a regime

dx₀₊₁ + dx₀₊₁ + dx₀₊₁ = 3 per tutti i arrivati a regime

Cosa succede nell'altra direzione dopo le colpi di clock ottenuta questa situazione nell'altra direzione, è quella sopra riconosciuta

Attenzione tutte le volte che un nodo conferisce info per quelli
vicini $I_x \rightarrow Y$ NON DI PENSE MAI delle info del nodo ricevere
verso X $I_Y \rightarrow X$

I_{x-y} NON DIPENDE MAI DI I_{y-x}
condizione che serve per fare in modo che vede e regole e
bien fine

- former how linear

$$z \times y \\ 0 \quad 0 \ldots 0 \cdot - 0 \\ \downarrow \\ 0 \\ \text{W}$$

$$\text{Solder.} x : I_{x \rightarrow z} = I_{y \rightarrow x} + I_{w \rightarrow x} + 1$$

domani del rincaro tranne
il ricevante

In general

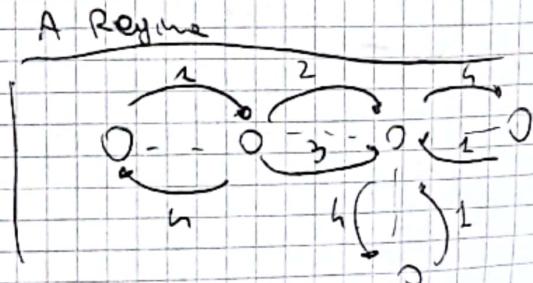
$$I_{x \rightarrow y} = \sum_{a \in N(x) - \{y\}} I_{a \rightarrow x} + I_x$$

\downarrow
not view x

excluso y

$$N_k = \sum_{d \in N(k)} I_{d \rightarrow x} + I_x$$

→ numbers divided:

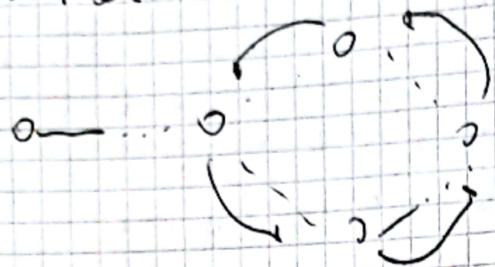


$I_{x,y}$ = info esterna

I_x = info intrinseca

N_x = info totale

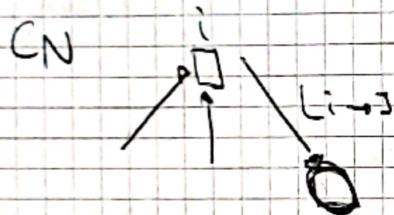
formazione di cicli



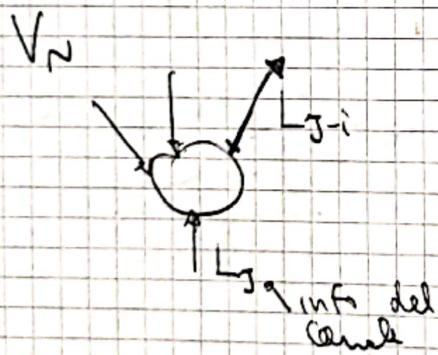
l'algoritmo non converge
cicli dominanti

Esempi di algoritmi usati per LDPC

- alg. iterativo e distribuito sul grafo di Tanner
- accanto a SPA (sum product Algorithm)



Alg. che stimare un bit basandosi
sull'equazione di controllo di
perdita $\sum_j x_j h_{ij} = 0$



Alg stimare i bit inviando che uno
sia inviato in copie uguali su
codificatori e controlli su punti
e sul canale

In effetti, qui calcoliamo una matrice ispirata al criterio di
decisiva MAP su ogni bit $X \in \{0, 1\}$
MAX prob e prob.

Cerca $\max_{X \in \{0, 1\}} P(X | \bar{F}) \Rightarrow$ confronto $P(0 | \bar{F}) \leq P(1 | \bar{F})$

Una sarà più grande dell'altra

$\circ \frac{P(0 | \bar{F})}{P(1 | \bar{F})} \leq 1$

metriche moltiplicate dell'algoritmo, $\log \left(\frac{P(0 | \bar{F})}{P(1 | \bar{F})} \right) \leq 0$

metrice LLR Log likelihood Ratio

 $\Rightarrow L(X | \bar{F})$

In particolare

$$VN: L_{j \rightarrow i} \sum_{d \in N(j) - \{i\}} L_{d \rightarrow j} + L_j \quad L_j = L(X_j | \bar{r})$$

Somma infi intrinseca con quelle ricevute da CN

CN:

$$L_{i \rightarrow j} = 2 \tanh^{-1} \left[\prod_{d \in N(i) - \{j\}} \tanh \left(\frac{1}{2} L_{d \rightarrow i} \right) \right]$$

È tipo il conteggio dei soldati nel VN è quella nel CN
è più complicato.

Parte intrinseca L_j si calcola

$$\text{Caso HD (come BSC)} \quad L_j = L(X_j | g_j) = (-1)^{g_j} \log \left(\frac{1 - p_e}{p_e} \right)$$

$$\text{Caso SD (come AWGN)} \quad L_j = L(X_j | r_j) = 2r_j / \sigma^2$$

Perando da qui in letteratura ci sono molti veicoli per
simplificare e/o migliorare questo