Appunti di Teoria dell'Informazione

Pietro Piram
Francesco Romani

Versione 2.5, Gennaio 2007

Premessa

Questo libro è stato scritto principalmente come testo di riferimento per il corso di Teoria dell'Informazione previsto del nuovo corso di laurea in Informatica.
Il testo è in forma incompleta; ogni suggerimento in merito ad errori e oscurità è benvenuto.

e-mail: romani@di.unipi.it

Notazione

Notazioni

- $\log x = \log_2 x$, $\ln x = \log_e x$, per continuità assumiamo valida la relazione:

  $0 \log 0 = 0$.

- La notazione $\binom{n}{k}$ indica il coefficiente binomiale $\frac{n!}{(n-k)!\,k!}$ per cui vale:

  $$(a+b)^n = \sum_{k=1}^{n} \binom{n}{k} a^k b^{n-k} .$$

- Il coefficiente multinomiale viene indicato con $\frac{n!}{n_1!n_2!\ldots n_k!}$ per esso vale:

  $$(a_1 + a_2 + \ldots + a_k)^n = \sum_{n_1+n_2+\ldots+n_k=n} \frac{n!}{n_1!n_2!\ldots n_k!} a_1^{n_1} a_2^{n_2} \ldots a_k^{n_k}$$

- I vettori sono da intendere vettori colonna. I vettori riga sono indicati col simbolo di trasposizione: x è un vettore colonna, x^T è un vettore riga. Se A è una matrice di dimensioni compatibili con x, avremo:

  $y = A\,x$ è un vettore colonna;

  $y^T = x^T A^T$ è un vettore riga.

- Se x e y sono vettori di uguale lunghezza

$<x,y> = x^T y$

indica il loro prodotto scalare

- Un esperimento finito viene denotato con l'insieme dei suoi possibili risultati $X=\{x_1, x_2, \ldots, x_k\}$, le probabilità associate sono $\{p(x_1), p(x_2), \ldots, p(x_k)\}$. Quando non ci sono rischi di ambiguità le probabilità possono venire abbreviate in $\{p_1, p_2, \ldots, p_k\}$. Ricordiamo che per ogni distribuzione di probabilità deve valere

  $$\sum_{i=1}^{k} p(x_i) = 1, \qquad p(x_i) \geq 0, i = 1,2,\ldots,k.$$

  Se z è una qualunque quantità indipendente da i vale:

  $$z = \sum_{i=1}^{k} z\ p(x_i),$$

  nel seguito questa identità sarà spesso usata in entrambi i sensi.

- L'entropia associata ad un esperimento X si indicherà indifferentemente come $H(X)$ oppure $H(p(x_1), p(x_2), \ldots, p(x_k))$.

Stringhe e loro rappresentazioni

La lunghezza di una stringa x si denota con $|x|$. Una stringa di lunghezza n su un alfabeto di q simboli $\{0,1,..., q\text{-}1\}$, può essere interpretata in vari modi, ognuno dei quali presenta diversi vantaggi matematici;

- un vettore di $\mathbf{R}^n$;
- un cammino di lunghezza n in un albero q-ario;
- un polinomio di grado al più n-1:
  $p(x) = a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\ldots+a_1x+ a_0$;
- un elemento di un campo di Galois con q^n elementi;
- un numero rappresentato in base q con notazione posizionale.

Osservazione. Quando si rappresenta una stringa con un polinomio e viceversa occorre un buon accordo sulla numerazione delle cifre. Il metodo sopra adottato presenta l'indubbio vantaggio che la cifra di posto i è il coefficiente del termine di grado i e che se la stringa è vista come un numero a in base q si ha $p(q) = a$.

Nel seguito useremo l'una o l'altra rappresentazione secondo le necessità del caso.

1
Incertezza di un esperimento finito

1.1 Definizione di Entropia

Consideriamo un esperimento X con k possibili risultati $\{x_1, x_2, \ldots, x_k\}$ e probabilità $p(x_1)$, $p(x_2)$, …, $p(x_k)$. Stiamo cercando una misura dell'**Incertezza** associata all'esperimento; è ragionevole chiedere che tale misura soddisfi i seguenti assiomi.

I tre assiomi dell'Entropia.

I $H(p_1, p_2, \ldots, p_k)$ è una funzione continua delle p_i.

II Dati due esperimenti A e B con rispettivamente k e $k+1$ possibili risultati equiprobabili l'esperimento B ha maggiore incertezza dell'esperimento A, ovvero:

$$H(\underbrace{\frac{1}{k},\frac{1}{k},\ldots,\frac{1}{k}}_{k \text{ volte}}) < H(\underbrace{\frac{1}{k+1},\frac{1}{k+1},\ldots,\frac{1}{k+1}}_{k+1 \text{ volte}}).$$

III Si consideri un esperimento X con n possibili risultati scomposto in due fasi successive:

fase 1: un esperimento Y con $h<n$ risultati y_j, j=1,2, …, h;

fase 2: uno tra h esperimenti $Z^{(j)}$, j=1, 2, …, h (indipendenti da Y) eseguito in base al risultato di Y (il numero di uscite di ogni $Z^{(j)}$ è n_j, e vale $n_1+n_2+\ldots n_h = n$);

allora vale:

$$H(X) = H(Y) + \sum_{j=1}^{h} p(y_j) H(Z^{(j)}),$$

cioè l'incertezza di X è uguale alla somma pesata delle incertezze degli esperimenti in cui X viene scomposto.

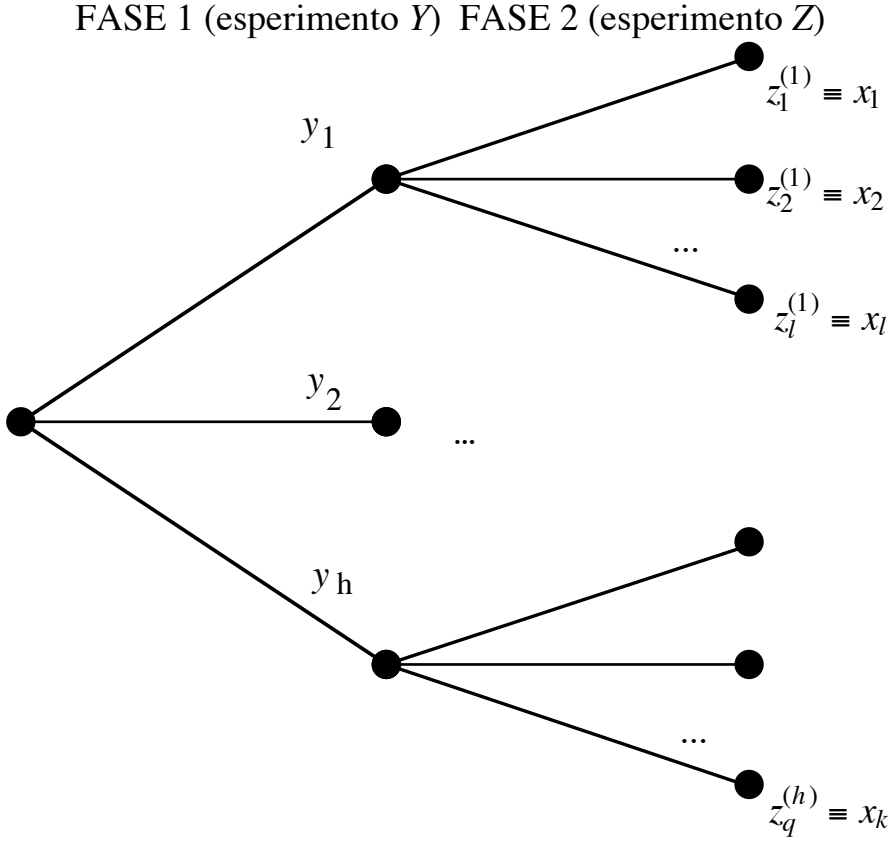


Figura 1.1. Un esperimento X scomposto in due fasi.

Lemma 1.1. Se f è una funzione positiva monotona crescente per cui per ogni $t>1$ e ogni n intero vale $f(t^n) = n\, f(t)$, allora f è della forma

$$f(t) = C \log(t).$$

Dim. Per ogni coppia n e t esiste un intero m per cui

$$m \le n \log t < m+1 \qquad (1.1)$$

ovvero

$$2^m \le t^n < 2^{m+1}.$$

Applicando f e sfruttandone le proprietà si ottiene:

$$f(2^m) \le f(t^n) < f(2^{m+1}),$$

$$m\, f(2) \le n\, f(t) < (m+1)\, f(2). \qquad (1.2)$$

Dividendo per n i membri della (1.1) e per $n\, f(2)$ i membri della (1.2) si ottiene

$$\frac{m}{n} \le \log t < \frac{m}{n} + \frac{1}{n},$$
$$\frac{m}{n} \le \frac{f(t)}{f(2)} < \frac{m}{n} + \frac{1}{n}.$$

Per ipotesi quanto scritto vale per ogni n, passando al limite per n che va all'infinito segue $f(t) = f(2) \log t$. ❑

Il seguente lemma mostra un risultato fortemente intuitivo che discende dall'assioma III: *se considero 10 estrazioni del Lotto mi aspetto un'incertezza pari a 10 volte l'incertezza di una sola estrazione*.

Lemma 1.2 In base all'assioma **III** l'incertezza associata ad un esperimento ripetuto n volte in modo indipendente è pari ad n volte l'incertezza del singolo esperimento, ovvero $H(Y^n) = n\, H(Y)$.
Dim. Se H soddisfa **III** e si considera l'esperimento Y ripetuto n volte in modo indipendente (indicato nel seguito con Y^n) , vale

$$H(Y^n) = H(Y) + \sum_{i=1}^{k} p(y_i) H(Y^{n-1}) = H(Y) + H(Y^{n-1}).$$

Da cui segue la tesi. ❑

Lemma 1.3 In base agli assiomi **II** e **III** l'incertezza associata ad un esperimento con k uscite equiprobabili vale $C \log k$.
Dim. Dato un esperimento Y con k uscite equiprobabili la sua incertezza è funzione solamente di k, e possiamo indicarla con $h(k)$. L'esperimento Y^n ha k^n uscite equiprobabili e si può applicare il Lemma 1.2 ottenendo

$$h(k^n) = H(Y^n) = n\, H(Y) = n\, h(k) .$$

In base all'assioma **II** $h(k)$ è una funzione crescente in k. Applicando il Lemma 1.1 si ottiene la tesi. ❑

Possiamo ora dimostrare che esiste una sola possibile formula per l'incertezza (a meno di una costante moltiplicativa che tiene conto dell'unità di misura).

Teorema 1.1. L'unica funzione H che soddisfa gli assiomi **I**, **II**, **III** è la seguente:

$$H(p_1, p_2, \ldots, p_k) = -C \sum_{i=1}^{k} p_i \log p_i \qquad (1.3)$$

Dim.
1) *La* (1.3) *soddisfa i tre assiomi*. Il primo è soddisfatto in modo ovvio. Il secondo segue dal fatto che per un esperimento con tutti i risultati equiprobabili vale

$$H(\frac{1}{k},\frac{1}{k},\ldots,\frac{1}{k}) = C\log k < C\log(k+1).$$

Per quanto riguarda l'assioma **III** si nota (Figura 1.1) che per l'esperimento scomposto vale, per opportuni valori di *i* ed *l*:

$$p(x_i) = p(y_j)\,p(z_l^{(j)}).$$

Allora si ha

$$\begin{aligned}
H(X) &= -C\sum_{j=1}^{h}\sum_{l=1}^{n_j} p(y_j)p(z_l^{(j)})\log\Big[p(y_j)p(z_l^{(j)})\Big] = \\
&= -C\sum_{j=1}^{h}\sum_{l=1}^{n_j} p(y_j)p(z_l^{(j)})\Big[\log p(y_j) + \log p(z_l^{(j)})\Big] = \\
&= -C\sum_{j=1}^{h}\sum_{l=1}^{n_j} p(y_j)p(z_l^{(j)})\log p(y_j) - C\sum_{j=1}^{h}\sum_{l=1}^{n_j} p(y_j)\,p(z_l^{(j)})\log p(z_l^{(j)}) = \\
&= -C\sum_{j=1}^{h}\Big(\sum_{l=1}^{n_j} p(z_l^{(j)})\Big)p(y_j)\log p(y_j) - C\sum_{j=1}^{h} p(y_j)\sum_{l=1}^{n_j} p(z_l^{(j)})\log p(z_l^{(j)}) = \\
&= -C\sum_{j=1}^{h} p(y_j)\log p(y_j) - C\sum_{j=1}^{h} p(y_j)\sum_{l=1}^{n_j} p(z_l^{(j)})\log p(z_l^{(j)})
\end{aligned}$$

2) *Se una funzione soddisfa i tre assiomi allora è della forma* (1.3). Consideriamo un esperimento *Y* le cui probabilità sono tutti numeri razionali, posti ad un comune denominatore *m*:

$$p(y_j) = m_j/m,\ \mathrm{j}=1,2\ldots,h.$$

A questo esperimento può essere fatta seguire una serie di esperimenti $Z^{(j)}$ ognuno di essi con m_j risultati equiprobabili ciascuno di probabilità $1/m_j$, per cui (in base al Lemma 1.3) vale $H(Z^{(j)}) = C\log m_j$. L'esperimento complessivo *X* consistente in *Y* seguito dall'opportuno $Z^{(j)}$ (vedi ancora la Figura 1.1) è un esperimento con *m* uscite equiprobabili, ovvero $H(X) = C\log m$. Per l'assioma **III** vale

$$C\log m = H(X) = H(Y) + \sum_{j=1}^{h} p(y_j)H(Z^{(j)}) = H(Y) + C\sum_{j=1}^{h} p(y_j)\log m_j$$

da cui

$$H(Y) = C\log m - C\sum_{j=1}^{h}\frac{m_j}{m}\log m_j = -C\sum_{j=1}^{h}\frac{m_j}{m}\log\frac{m_j}{m}$$

e per la continuità di *H* (assioma **I**) questo risultato deve valere anche se le probabilità di *Y* sono numeri reali. ❑

Definizione 1.1. L'incertezza associata ad un esperimento X è detta **Entropia** di X e vale:

$$H(X) = -C\sum_{i=1}^{k} p(x_i)\log p(x_i). \ \square$$

Nel seguito si pone $C = 1/\log 2$ e l'Entropia si misura in **bit**. Poiché $H(1/2,1/2) = 1$ un bit è per definizione la quantità di incertezza associata all'esperimento con due sole uscite equiprobabili.

Esempio 1.1. Nel caso di un esperimento con due soli risultati l'entropia ha la forma $-p \log p -(1-p)\log(1-p)$ e il grafico di $H(p,1-p)$ in funzione di p è il seguente:

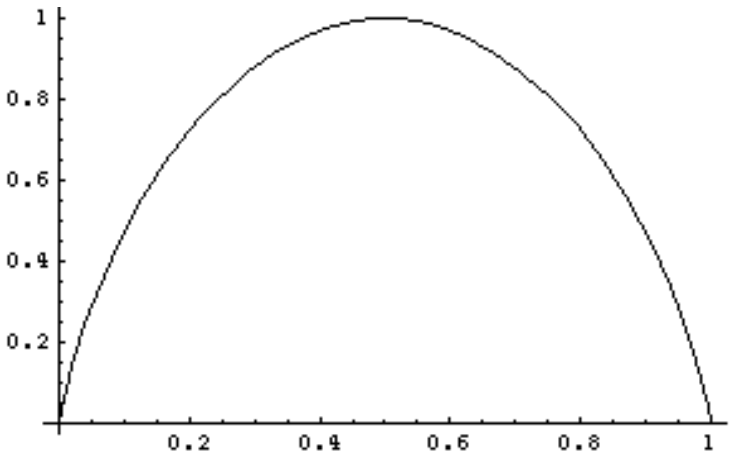


Figura 1.2. Il grafico di $H(p,1-p)$.

Il massimo vale 1 ed è raggiunto per $p = 1-p = 1/2$. ❑

Lemma del logaritmo

Lemma 1.4. Date due distribuzioni di probabilità $\{p_1, p_2, \ldots, p_k\}$ e $\{q_1, q_2, \ldots, q_k\}$ vale

$$-\sum_{i=1}^{k} p_i \log p_i \le -\sum_{i=1}^{k} p_i \log q_i$$

Dim. La disequazione non cambia se usiamo i logaritmi naturali (entrambi i membri vengono moltiplicati per una costante positiva). Per la funzione logaritmo vale

$$\ln x \le x-1$$

su tutto l'asse reale positivo.

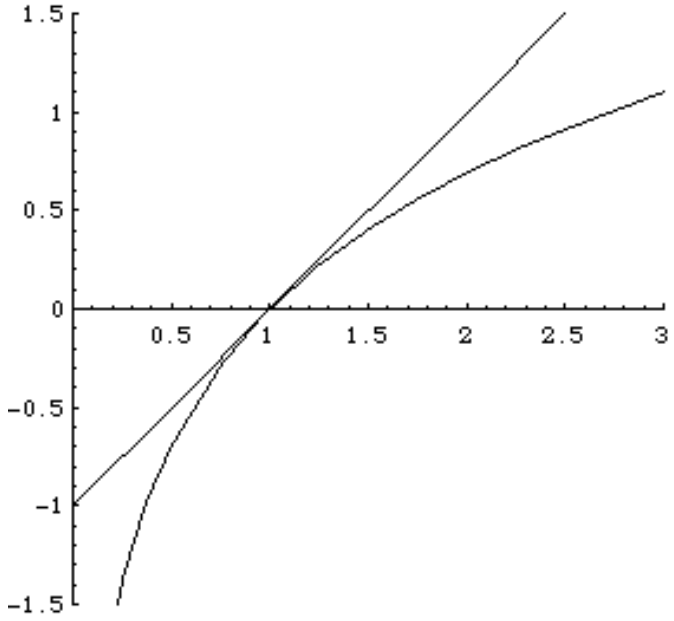


Figura 1.3. Il grafico di x-1 e ln x.

Allora

$$\ln \frac{q_i}{p_i} \le \frac{q_i}{p_i} - 1$$

moltiplicando per p_i e sommando si ha

$$\sum_{i=1}^{k} p_i \left(\ln q_i - \ln p_i\right) \le \sum_{i=1}^{k} q_i - \sum_{i=1}^{k} p_i = 0$$

da cui segue la tesi. ❑

1.2 Proprietà dell'Entropia

Teorema 1.2. Sia X un esperimento con k possibili risultati e distribuzione di probabilità $\{p_1, p_2, \ldots, p_k\}$. Vale:

$$0 \le H(X) \le \log k$$

con la prima uguaglianza che vale se e solo se uno dei $p_i = 1$ e la seconda se e solo se tutti i p_i sono uguali a $1/k$.

Dim. L'entropia è non negativa perché -log p_i non è mai negativo. Il valore massimo si ottiene dal Lemma del logaritmo:

$$-\sum_{i=1}^{k} p_i \log p_i \le -\sum_{i=1}^{k} p_i \log \frac{1}{k} = \log k\,.\ ❑$$

Teorema 1.3. Siano X e Y due esperimenti con, rispettivamente, k ed h uscite, se con (X,Y) si indica l'esperimento congiunto (con hk uscite). Vale:

$$H(X,Y) \le H(X) + H(Y) \tag{1.4}$$

dove l'uguaglianza vale se e solo se X e Y sono indipendenti.

Dim. Abbiamo

$$\begin{aligned}
H(X,Y) &= -\sum_{i=1}^{k} \sum_{j=1}^{h} p(x_i, y_j) \log p(x_i, y_j) \\
H(X) &= -\sum_{i=1}^{k} p(x_i) \log p(x_i) = \\
&= -\sum_{i=1}^{k} \left(\sum_{j=1}^{h} p(x_i, y_j) \right) \log p(x_i), \\
H(Y) &= -\sum_{j=1}^{h} p(y_j) \log p(y_j) = \\
&= -\sum_{j=1}^{h} \left(\sum_{i=1}^{k} p(x_i, y_j) \right) \log p(y_j), \\
H(X) + H(Y) &= -\sum_{i=1}^{k} \sum_{j=1}^{h} p(x_i, y_j) \left[\log p(x_i) + \log p(y_j) \right] = \\
&= -\sum_{i=1}^{k} \sum_{j=1}^{h} p(x_i, y_j) \log \left[p(x_i) p(y_j) \right]
\end{aligned}$$

e poiché

$$\sum_{i=1}^{k} \sum_{j=1}^{h} p(x_i) p(y_j) = 1$$

la tesi segue dal Lemma del logaritmo. ❑

Entropia Condizionata

Definizione 1.2 Dati due esperimenti X e Y l'incertezza associata all'esperimento X una volta noto il risultato di Y è detta **Entropia Condizionata di *X* dato *Y*** e vale:

$$H(X|Y) = \sum_{j=1}^{h} p(y_j) H(X|y_j)$$

Teorema 1.4. Siano X e Y due esperimenti. Vale:

a) $H(X,Y) = H(Y) + H(X|Y)$; (1.5)

b) $H(X,Y) = H(X) + H(Y|X)$;

c) $H(X|Y) \le H(X)$
con l'uguaglianza che vale se e solo se X e Y sono indipendenti.

Dim. Scrivendo l'espressione di $H(X,Y)$ e utilizzando la relazione

$$p(x_i, y_j) = p(y_j)\, p(x_i / y_j)$$

si ottiene

$$\begin{aligned}
H(X,Y) &= -\sum_{i=1}^{k}\sum_{j=1}^{h} p(x_i, y_j)\log p(x_i, y_j) = \\
&= -\sum_{i=1}^{k}\sum_{j=1}^{h} p(y_j)\, p(x_i|y_j)\log\Big[p(y_j)\, p(x_i|y_j)\Big] = \\
&= -\sum_{j=1}^{h} p(y_j)\sum_{i=1}^{k} p(x_i|y_j)\Big[\log p(y_j) + \log p(x_i|y_j)\Big] = \\
&= -\sum_{j=1}^{h} p(y_j)\sum_{i=1}^{k} p(x_i|y_j)\log p(y_j) + \\
&\quad -\sum_{j=1}^{h} p(y_j)\sum_{i=1}^{k} p(x_i|y_j)\log p(x_i|y_j) = \\
&= -\sum_{j=1}^{h} p(y_j)\log p(y_j)\Big(\sum_{i=1}^{k} p(x_i|y_j)\Big) + \sum_{j=1}^{h} p(y_j) H(X|y_j) = \\
&= -\sum_{j=1}^{h} p(y_j)\log p(y_j) + \sum_{j=1}^{h} p(y_j) H(X|y_j) =
\end{aligned}$$

da cui segue la relazione (a).
La relazione (b) segue poiché, per simmetria, $H(X,Y) = H(Y,X)$ che usando ciò che abbiamo appena dimostrato vale $H(X) + H(Y|X)$.
Infine la relazione (c) si ottiene confrontando la (a) e la (1.4). ❑

Informazione Reciproca

Dati due esperimenti X e Y, l'informazione su di X data da Y può essere definita come la riduzione di incertezza sul risultato di X data dalla conoscenza del risultato di Y.

Definizione 1.3. Dati due esperimenti X e Y, la quantità non negativa

$$I(X|Y) = H(X) - H(X|Y)$$

è detta **Informazione reciproca** tra i due esperimenti. ❑

Teorema 1.5. Vale $I(X|Y) = I(Y|X)$.
Dim. Applicando il Teorema 1.4 si ha

$$H(X|Y) = H(X,Y) - H(Y)$$

da cui sostituendo nella Definizione 1.3

$$I(X|Y) = H(X) + H(Y) - H(X,Y)$$

che è un'espressione simmetrica in X e Y. ❑

Osservazione. In base alla relazione (c) del Teorema 1.4, $I(X|Y)$ vale 0 se e solo se i due esperimenti sono indipendenti.

1.3 Sequenze tipiche ed equipartizione asintotica

Se consideriamo il solito esperimento X e immaginiamo di ripeterlo "molte" volte ci aspettiamo una distribuzione dei risultati proporzionale alle loro probabilità. Sequenze di risultati che soddisfano questa distribuzione sono dette "tipiche". Si ricordi che spesso nelle scienze sperimentali le probabilità dei risultati di un esperimento sono determinate empiricamente proprio contando le frequenze dei vari risultati in una varietà di esperimenti.
Queste considerazioni informali possono essere formalizzate nel teorema dell'equipartizione asintotica, semplice conseguenza del teorema di Bernoulli. In queste pagine preferiamo un approccio più diretto anche se meno preciso. Vediamo prima un esempio nel caso di due possibili risultati, 0 con probabilità p e 1 con probabilità $1-p$. La probabilità di avere esattamente k zeri in n tentativi vale:

$$\binom{n}{k} p^k (1-p)^{n-k}$$

Questo valore si forma dal prodotto di due fattori:

- $\binom{n}{k}$ conta il numero di sequenze lunghe n con k zeri;
- $p^k (1-p)^{n-k}$ dà la probabilità di ognuna di esse.

Il primo fattore è massimo quando k è circa $n/2$ come si vede dalla Figura 1.4. Supponendo p=0.7 il secondo fattore è massimo quando k è uguale ad n, ovvero la sequenza più probabile è quella formata da n volte il simbolo più probabile, come si vede dalla Figura 1.5.

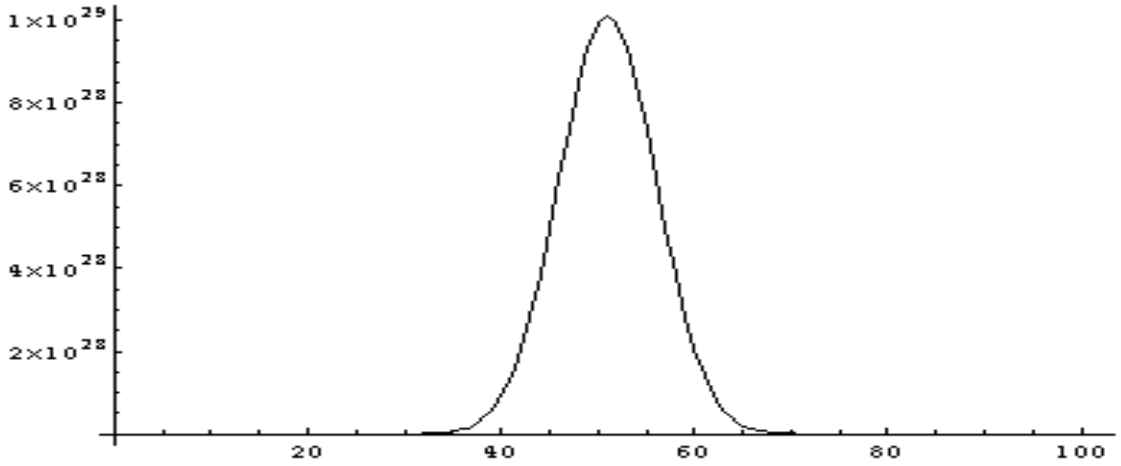


Figura 1.4. Il grafico di $\binom{n}{k}$, in funzione di k, per n=100.

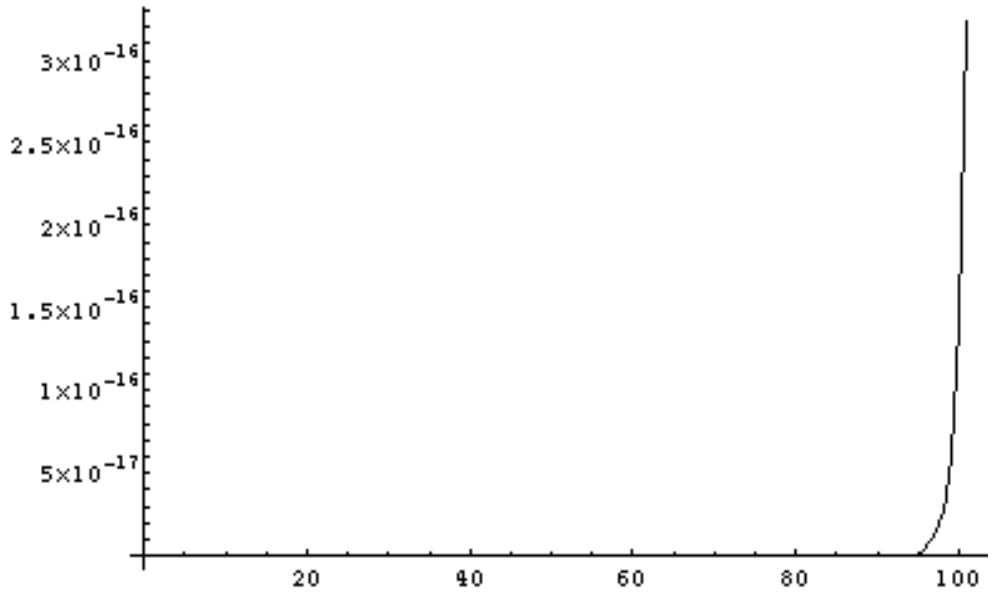


Figura 1.5. Il grafico di $p^k(1-p)^{n-k}$, in funzione di k, per p=0.7, n=100.

Ma di sequenze di tutti 0 ce n'è una sola; la tipologia di sequenze più probabile è invece quella con circa il 70% di zeri e il 30% di uni come si vede dalla Figura 1.6. Infine, nella Figura 1.7 abbiamo in ascissa la frequenza relativa degli zeri e in ordinata i valori della probabilità delle varie classi moltiplicate per n (avendo diviso per n i valori in ascissa si è dovuto moltiplicare per n i valori in ordinata per conservare le aree). Le aree sottese alle curve sono tutte unitarie ma con l'aumentare di n la campana si stringe e tende ad un picco di altezza infinita. Quindi, asintoticamente, la probabilità che la sequenza che si presenta sia tipica tende ad 1.

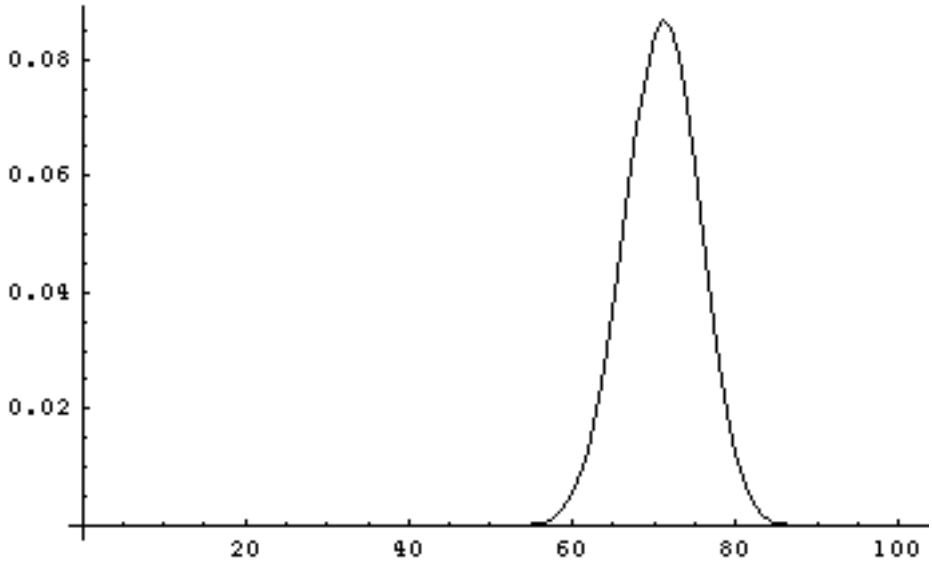


Figura 1.6. Il grafico di $\binom{n}{k} p^k (1-p)^{n-k}$, in funzione di k, per p=0.7, n=100.

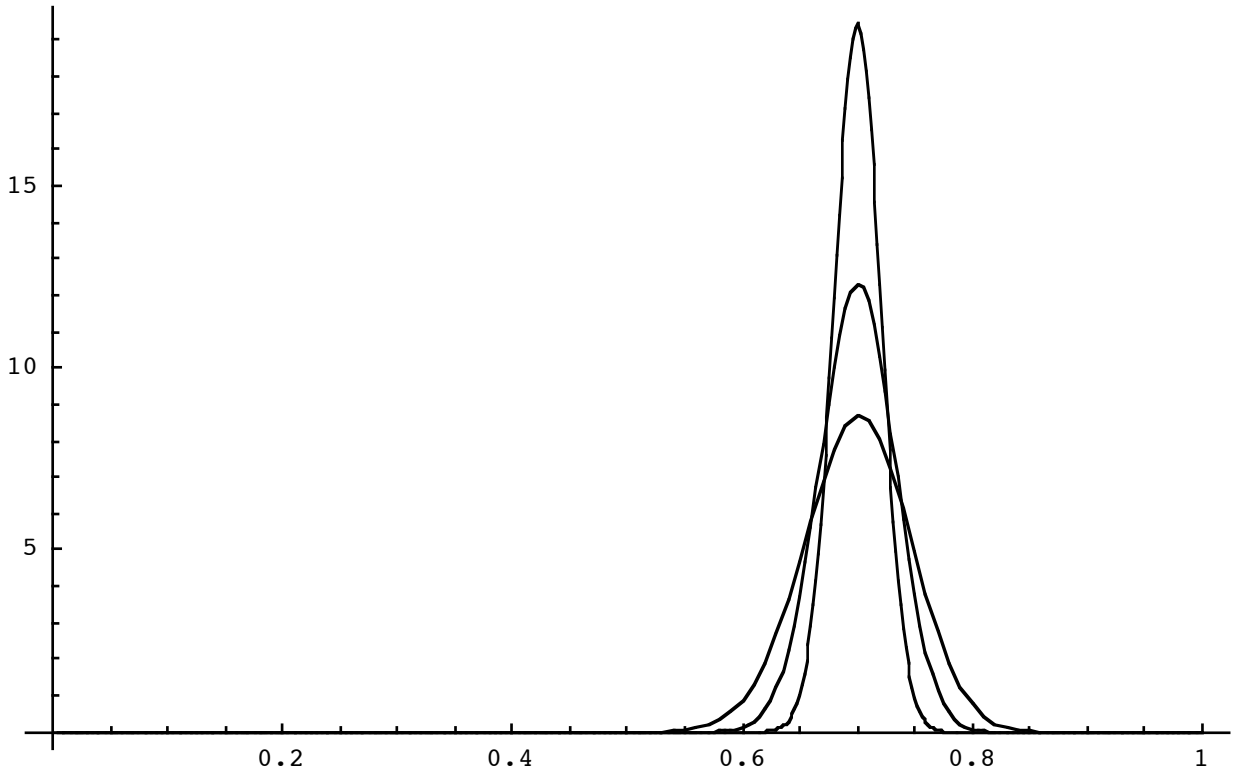


Figura 1.7. Le curve relative a p=0.7, n=100, 200, 500, in scala normalizzata.

Il fatto più interessante di queste considerazioni è che la probabilità delle sequenze tipiche è legata all'entropia dell'esperimento. Ognuna delle sequenze con esattamente np zeri e $n(1-p)$ uni ha probabilità

$$p_{\text{tipica}} = p^{np}\,(1-p)^{n(1-p)}$$

e il suo logaritmo è

$$\log p_{\text{tipica}} = n\,p \log p + n\,(1\text{-}p) \log (1\text{-}p)$$

da cui

$$\log p_{\text{tipica}} = -n\,H(X).$$

È probabilmente un'osservazione di questo genere che ha suggerito a Shannon la forma della funzione entropia.

Più in generale se i possibili risultati sono k invece che 2, si può dimostrare il seguente teorema.

Teorema 1.6. Dato un esperimento X con possibili risultati $\{x_1, x_2, \ldots, x_k\}$ con probabilità di uscita $\{p_1, p_2, \ldots, p_k\}$ e entropia $H(X)$ le sequenze di uscita di n esperimenti ripetuti, per n che tende all'infinito, si possono dividere in due classi

1) circa $2^{n\,H(X)}$ sequenze **tipiche** con probabilità uniforme $2^{-n\,H(X)}$
2) tutte le altre.

Poiché la probabilita della prima classe è circa 1 la probabilità che una sequenza non sia tipica è trascurabile.

Dim. (informale). Dallo sviluppo della potenza n-esima di una somma di k termini si ricava che la probabilità di avere una sequenza con n_i risultati pari a x_i in n tentativi vale esattamente:

$$\frac{n!}{n_1!n_2!\ldots n_k!}\,p_1^{n_1}\,p_2^{n_2}\ldots p_k^{n_k}\,,\ \text{con}\ n_1 + n_2 + \ldots + n_k = n\,.$$

Prendiamo in considerazione separatamente il logaritmo dei due fattori.

i) $$\log \frac{n!}{n_1!n_2!\ldots n_k!} = \log n! - \sum_{i=1}^{k} \log n_i!$$

e usando la formula di Stirling:

$$\log \frac{n!}{n_1!n_2!\ldots n_k!} \approx n \log n - \sum_{i=1}^{k} n_i \log n_i = \tag{1.6}$$

$$= n \sum_{i=1}^{k} \frac{n_i}{n} \log n - n \sum_{i=1}^{k} \frac{n_i}{n} \log n_i =$$

$$= -n \sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

Quindi le sequenze per cui n_i/n è vicino a p_i (quelle tipiche) sono circa $2^{n\,H(X)}$.

(1.7)

ii) $$\log\left(p_1^{n_1}\,p_2^{n_2}\ldots p_k^{n_k}\right) = \sum_{i=1}^{k} n_i \log p_i = n \sum_{i=1}^{k} \frac{n_i}{n} \log p_i$$

Quindi le sequenze tipiche hanno circa la probabilità

$$2^{n\sum_{i=1}^{k} p_i \log p_i} = 2^{-nH(X)}$$

e sono pressoché equiprobabili. ❑

Osservazione. Una tesi più precisa si può enunciare e dimostrare utilizzando il teorema di Bernoulli. Vedi (Gestri, p. 16, Ash, p. 197).

1.4 Esercizi

Esercizio 1.1. Date due variabili casuali X e Y, se accade che è sempre $X > Y$, si può dedurre che è $H(X) > H(Y)$?

Esercizio 1.2. Sapendo che $H(5/12,5/12,1/6) = 1.484$ e che $\log_2 5 = 2.322\ldots$, quale proprietà dell'Entropia posso utilizzare (e come) per calcolare $\log_2 12$ senza ricorrere a tabelle o calcolatrici?

Esercizio 1.3. Dati r vettori di probabilità a q componenti $p_1, p_2, \ldots, p_r$ e r numeri non negativi $a_1, a_2, \ldots, a_r$ tali che la loro somma sia 1, il vettore

$$p = \sum_{i=1}^{r} a_i p_i$$

è un vettore di probabilità?

Esercizio 1.4. Dati due esperimenti X e Y, vale sempre $H(X,Y)>H(X/Y)$? In caso contrario, quando vale $H(X,Y)=H(X/Y)$? Quando vale $H(X,Y)<H(X/Y)$?

Esercizio 1.5. Calcolare $H(2^{-1}, 2^{-2}, 2^{-3}, \ldots, 2^{-k}, 2^{-k})$, per k=4, 6, 8, 100.

Esercizio 1.6. Data una moneta che produce testa (T) con probabilità 0.1 e croce (C) con probabiltà 0.9, dire quali delle sequenze

a= TTTTTTTTTC, b= CCCCCCCCCC, c = CCCCTCCCCC.

sono tipiche e calcolare la probabilità di uscita di ciascuna di esse.

Esercizio 1.7. Determinare due valori di x e y per cui $H(2^{-1}, 2^{-2}, x, y)$ sia un numero razionale

Esercizio 1.8. Determinare i valori di x e y che massimizzano l'entropia dell'esperimento con probabilità {1/2, x,1/8, y}.

2
Sorgenti discrete

2.1 Classificazione delle sorgenti

Le sorgenti rappresentano un modello di come vengono generati i dati che si vogliono trasmettere. Per una sorgente S viene definito un alfabeto finito di uscita e un modello probabilistico di come i simboli di uscita vengono generati.

Definizione 2.1. Una **sorgente** S su un alfabeto $A=\{a_1, a_2, \ldots, a_r\}$ è un "dispositivo" che ad ogni istante t emette un simbolo $x_t \in A$. In altre parole una sorgente può essere vista come una sequenza di esperimenti $X_1, X_2,\ldots,X_t$, ognuno con possibili risultati nell'insieme A. ❑

È utile classificare le sorgenti in base alle proprietà a cui soddisfano.

Definizione 2.2. Una sorgente è **stazionaria** se per ogni simbolo generato e per ogni n la probabilità che una sequenza $x_{t+1}, x_{t+2}, \ldots, x_{t+n}$ appartenga alla stringa di uscita è indipendente da t. ❑

Le sorgenti stazionarie sono particolarmente importanti perché il loro comportamento non dipende dal tempo. In particolare per una sorgente stazionaria la distribuzione delle probabilità di emissione dei vari simboli è indipendente dal tempo.

Definizione 2.3. Una sorgente S è **senza memoria** se per ogni istante t e per ogni simbolo a_i di S generato la probabilità che x_t sia uguale ad a_i non dipende dai simboli $x_0, x_1, \ldots, x_{t-1}$ usciti in precedenza. ❑

Definizione 2.4. Una sorgente S ha **memoria finita** se esiste un intero m tale che la probabilità del simbolo in uscita dipende al più dalla sequenza degli m simboli usciti in precedenza. Il minimo intero m per cui tale proprietà è verificata è detto **memoria** di S. ❑

Esempio 2.1. Un esempio di sorgente stazionaria è fornito dalla ripetizione dello stesso esperimento (per esempio il lancio di una moneta) in modo indipendente. Un esempio di sorgente non stazionaria è fornito dalla ripetizione alternata di due esperimenti diversi (per esempio una moneta con probabilità {1/2,1/2} e una moneta con probabilità {1/3,2/3}). In questo caso le probabilità di uscita dei simboli dipendono dal fatto che t sia pari o dispari ma non dal risultato precedente (testa o croce). Entrambe queste sorgenti sono quindi senza memoria. ❑

Definizione 2.5. Data una sorgente stazionaria S con memoria m e probabilità $\{p_1, p_2, \ldots, p_r\}$ di generare $\{a_1, a_2, \ldots, a_r\}$ la **sorgente adiacente** $\overline{S}$ è una sorgente senza memoria con le stesse probabilità. ❑

Definizione 2.6. Data una sorgente S la sorgente **estensione *k*-esima** di S si indica con S^k e si ottiene considerando la sorgente con r^k simboli ottenuta raggruppando k a k i simboli emessi da S.

Teorema 2.1. Se S ha memoria m la memoria di S^k è $\left\lceil \frac{m}{k} \right\rceil$.

Dim. Basta considerare una sequenza di simboli di S raggruppati k a k come nella Figura 2.1. ❑

Figura 2.1. Memoria della sorgente estensione nel caso $m = 7, k = 3$.

È possibile costruire sorgenti in cui il risultato di ogni simbolo dipende dai risultati di tutti gli esperimenti precedenti, in tal caso la sorgente ha **memoria infinita**. Un semplice esempio didattico è il seguente.

Esempio 2.2. (Gestri, p.40). Si hanno due monete una con probabilità di uscita {1/2,1/2} e l'altra con probabilità {1/4, 3/4} si inizia a lanciare la prima moneta e si continua fino a che non esce croce. In questo caso si continua ad usare indefinitamente la seconda moneta. Se le ultime m estrazioni sono tutte testa non è dato sapere quale sia la probabilità di emettere testa o croce e la sorgente ha memoria maggiore di m. ❑

Nella pratica i linguaggi naturali possono essere considerati sorgenti con memoria infinita. Un modo per trattare un tale tipo di sorgenti è approssimarle con modelli di memoria finita che "ricordano" solo un numero finito di simboli precedenti.

Definizione 2.7. Data una sorgente S stazionaria la sorgente **approssimazione di S di ordine k** si indica con S_k e si ottiene considerando la sorgente stazionaria con memoria al più k che ha le stesse probabilità condizionali di S di emettere un simbolo dati i k precedenti. In particolare S_0 è l'adiacente di S. Inoltre se S ha memoria $m \le k$ allora S_k coincide con S. ❑

2.2 Sorgenti con memoria finita e catene di Markov

Uno strumento efficace per descrivere le sorgenti con memoria finita è la **catena di Markov**. Sia $A=\{a_1, a_2, \ldots, a_r\}$ l'alfabeto della sorgente. Se la sorgente ha memoria m il suo comportamento dipende dagli ultimi m simboli emessi. Si associa ad ogni possibile m-pla di simboli un possibile **stato** della sorgente. Vi sono quindi $s=r^m$ stati $\sigma_1, \sigma_2, \ldots, \sigma_s$, e il comportamento della sorgente è definito dalle r^{m+1} probabilità condizionate:

$$p(a_j|\sigma_i), \quad j=1,2,\ldots,r, \quad i=1,2,\ldots,r^m.$$

La quantità $p(a_j|\sigma_i)$ rappresenta la probabilità di emettere il simbolo a_j essendo nello stato σ_i

Alla catena di Markov risulta naturalmente associata una matrice Π di elementi $\{\pi_{ij}\}$ e dimensioni $r^m \times r^m$ il cui elemento $\pi_{ij} = p(\sigma_j|\sigma_i)$ rappresenta la probabilità di passare dallo stato σ_i allo stato σ_j. Poiché viene emesso un simbolo alla volta ogni riga della matrice può contenere solo r elementi diversi da zero; la matrice contiene quindi almeno $r^{2m} - r^{m+1}$ zeri e, per m grande, è **sparsa**.

La probabilità $p(\sigma_j^{(t)})$ di essere, al tempo t nello stato σ_j vale

$$p(\sigma_j^{(t)}) = \sum_{i=1}^{s} p(\sigma_i^{(t-1)}) p(\sigma_j|\sigma_i) = \sum_{i=1}^{s} p(\sigma_i^{(t-1)}) \pi_{ij} \qquad (2.1)$$

e la probabilità di avere in uscita il simbolo a_j al tempo t vale

$$p(a_j) = \sum_{i=1}^{s} p(\sigma_i^{(t-1)}) p(a_j | \sigma_i)$$

La catena di Markov che schematizza la sorgente può anche essere definita dal grafo associato alla matrice Π (si vedano gli esempi seguenti).

Esempio 2.3. Sorgente con memoria 1.

$p(0|0) = 0.2,\ p(0|1) = 0.6.$

	0	1
0	0.2	0.8
1	0.6	0.4

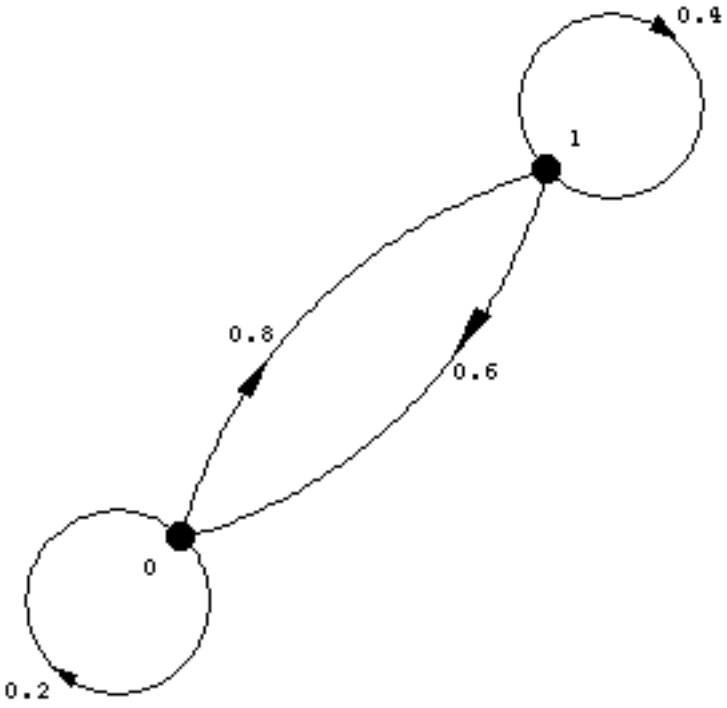


Figura 2.2. Grafo della sorgente dell'Esempio 2.3 ❑

Esempio 2.4. Sorgente con memoria 2.

$p(0|00) = 0.3$ $p(0|01) = 0.9$

$p(0|10) = 0.2$ $p(0|11) = 0.8$

	00	01	10	11
00	0.3	0.7	0	0
01	0	0	0.9	0.1
10	0.2	0.8	0	0
11	0	0	0.8	0.2

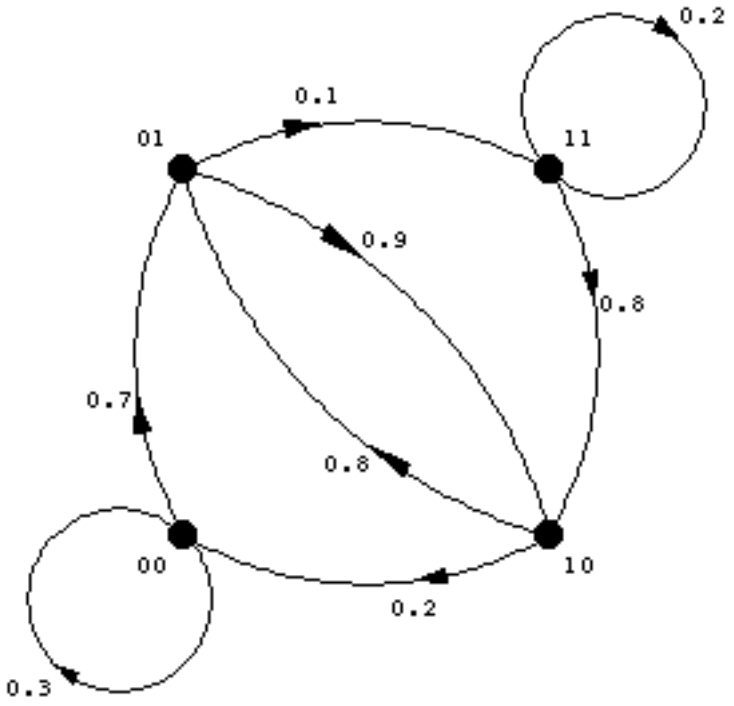


Figura 2.3. Grafico della sorgente dell'Esempio 2.4 ❑

Esempio 2.5. Sorgente con memoria 3.

$p(0|000) = 0.9$ $p(0|001) = 0.4$
$p(0|010) = 0.7$ $p(0|011) = 0.9$
$p(0|100) = 0.9$ $p(0|101) = 0.8$
$p(0|110) = 0.7$ $p(0|111) = 0.6$

	000	001	010	011	100	101	110	111
000	0.9	0.1	0	0	0	0	0	0
001	0	0	0.4	0.6	0	0	0	0
010	0	0	0	0	0.7	0.3	0	0
011	0	0	0	0	0	0	0.9	0.1
100	0.9	0.1	0	0	0	0	0	0
101	0	0	0.8	0.2	0	0	0	0
110	0	0	0	0	0.7	0.3	0	0
111	0	0	0	0	0	0	0.6	0.4

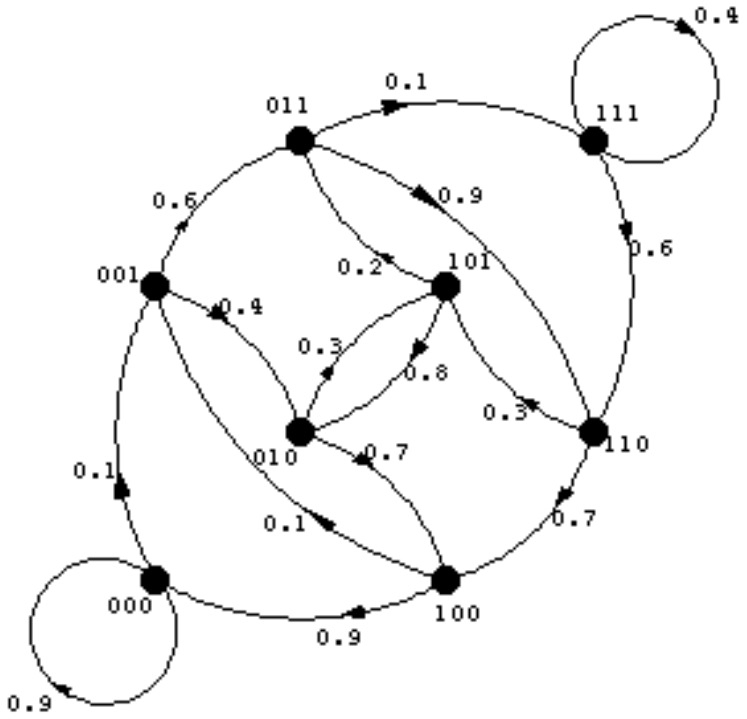


Figura 2.4. Grafico della sorgente dell'Esempio 2.5 ❑

Distribuzione stazionaria di una Sorgente di Markov

Sia S una sorgente con alfabeto di cardinalità r e memoria m; S può essere descritta da una catena di Markov con matrice Π di dimensioni $s \infty s$ con $s=r^m$.
In base alla (2.1) se all'istante iniziale le probabilità degli stati della sorgente sono date dal vettore x_0, all'istante successivo la distribuzione è

$$x_1^T = x_0^T \Pi$$

e all'istante n la distribuzione è

$$x_n^T = x_{n-1}^T \Pi = x_0^T \Pi^n.$$

È molto importante determinare sotto quali condizioni una sorgente definita da una catena di Markov è stazionaria. La nostra trattazione userà per quanto possibile gli strumenti standard dell'algebra delle matrici.

La prima proprietà da notare è che gli elementi di Π sono delle probabilità e quindi sono non negativi e che la somma degli elementi di ogni riga è 1. Questo permette di dimostrare subito il seguente teorema.

Teorema 2.2. La matrice di Markov Π ha almeno un autovalore 1; tutti gli altri autovalori sono in modulo non maggiori di 1 (ovvero il raggio spettrale $\rho(\Pi) = 1$).
Dim. Segue dal fatto che $\Pi u = u$, dove $u^T = (1,1,\ldots 1)$ e $\rho(\Pi) \leq \|\Pi\|_\infty = 1$ (Menchi, *p.*

119). ❑

Poiché gli autovalori di A sono gli stessi di quelli di A^T, la presenza di un autovalore 1 implica l'esistenza di vettori w che sono autovettori sinistri di A associati all'autovalore 1 (e contemporaneamente autovettori destri di A^T associati allo stesso autovalore) per cui vale la relazione

$$w^T \Pi = w^T \qquad (2.2)$$

Se all'istante 0 gli stati di S hanno distribuzione di probabilità w e per il vettore w vale la (2.2) allora la distribuzione rimane la stessa all'istante 1 e per tutti gli istanti successivi. La sorgente è quindi stazionaria e w viene detta **distribuzione stazionaria** di S.

Vedremo in seguito che ogni sorgente con memoria finita ha almeno una distribuzione stazionaria.

Teorema 2.3. Le distribuzioni stazionarie di una sorgente con matrice di Markov Π sono tutti e soli i vettori di probabilità nello spazio generato dagli autovettori sinistri relativi agli autovalori di Π uguali ad 1.
Dim. Segue dal fatto che $x^T \Pi = x^T$ se e solo se x è un autovettore sinistro relativo ad un autovalore 1. ❑

Sorgenti decomponibili

Definizione 2.8. Un insieme B di stati tale che ogni elemento di B è raggiungibile da ogni altro elemento di B e che nessun elemento fuori di B è raggiungibile da un elemento di B è detto **insieme essenziale**. ❑

Definizione 2.9. Una sorgente è detta **indecomponibile** se ha un solo insieme essenziale, **decomponibile** se ne ha più di uno. ❑

Se la matrice di Markov è irriducibile (Menchi *p*. 17) da ogni stato si può andare in un altro stato in un numero finito di passi con probabilità non nulla. In tal caso tutti gli stati della sorgente formano un solo insieme essenziale. Più precisamente si può dimostrare il seguente Teorema.

Teorema 2.4. Se la matrice di Markov Π è irriducibile,
1) esiste un solo autovalore uguale ad 1 e il corrispondente autovettore sinistro è tutto positivo;
2) S ammette un'unica distribuzione stazionaria.
Dim. 1) segue dal teorema di Perron-Frobenius (vedi Menchi, p. 303); 2) discende dalla 1) e dal Teorema 2.3. ❑

Se la matrice di Markov è riducibile allora, attraverso una permutazione di righe e di colonne, può essere messa nella forma:

$$\Pi = \begin{pmatrix} A & \mathrm{O} \\ B & C \end{pmatrix}$$

con A matrice quadrata irriducibile, C matrice quadrata.

Poiché gli elementi delle righe di A contengono tutti gli elementi non nulli delle corrispondenti righe di Π la matrice A è una matrice di Markov, e i suoi stati formano un insieme essenziale. La matrice A ha un autovalore unitario e un solo autovettore sinistro w_A associato ad esso. Il corrispondente autovettore sinistro di Π ha la forma $w^{\mathrm{T}} = [w_A{}^{\mathrm{T}} \mid \mathrm{O}]$.

Se in ogni riga di B c'è almeno un elemento positivo, da un qualunque stato di C si può passare nell'insieme essenziale formato dagli stati di A, gli stati di C sono allora detti **transitori** (nel senso che da essi prima o poi si esce per non più ritornarvi). La sorgente in questo caso è indecomponibile.

Se invece la matrice B ha alcune righe nulle allora, se C è riducibile, la matrice originaria si può scrivere

$$\begin{pmatrix} A & \mathrm{O} & \mathrm{O} \\ \mathrm{O} & D & \mathrm{O} \\ E & F & G \end{pmatrix}$$

con A e D quadrate e irriducibili. Gli stati associati alle righe di A e D formano 2 insiemi essenziali. La sorgente in questo caso è decomponibile.

Per G si può applicare lo stesso ragionamento usato per C, individuando eventualmente altri insiemi essenziali.

Si noti che ogni vettore di probabilità appartenente allo spazio generato dagli autovettori associati agli autovalori della matrice Π uguali ad 1 è una distribuzione stazionaria; in altre parole se vi sono almeno 2 insiemi essenziali vi sono infinite distribuzioni stazionarie.

Queste considerazioni mostrano che una sorgente con memoria finita ammette almeno una distribuzione stazionaria. La seguente tabella esemplifica le varie possibilità di combinazione tra riducibilità della matrice, decomponibilità della sorgente e numero delle distribuzioni stazionarie.

Matrice	Autovalori uguali a 1	Insiemi essenziali	Sorgente	Distribuzioni stazionarie
Irriducibile	1	1	Indecomponibile	1
Riducibile	1	1	Indecomponibile	1
Riducibile	k	k	Decomponibile	∞

Esempio 2.6. La sorgente con memoria 2

$p(0|00) = 1$ $\quad$ $p(0|01) = 0.9$

$p(0|10) = 0.2$ $\quad$ $p(1|11) = 1$

è decomponibile con due insiemi essenziali come si vede dal grafo orientato.

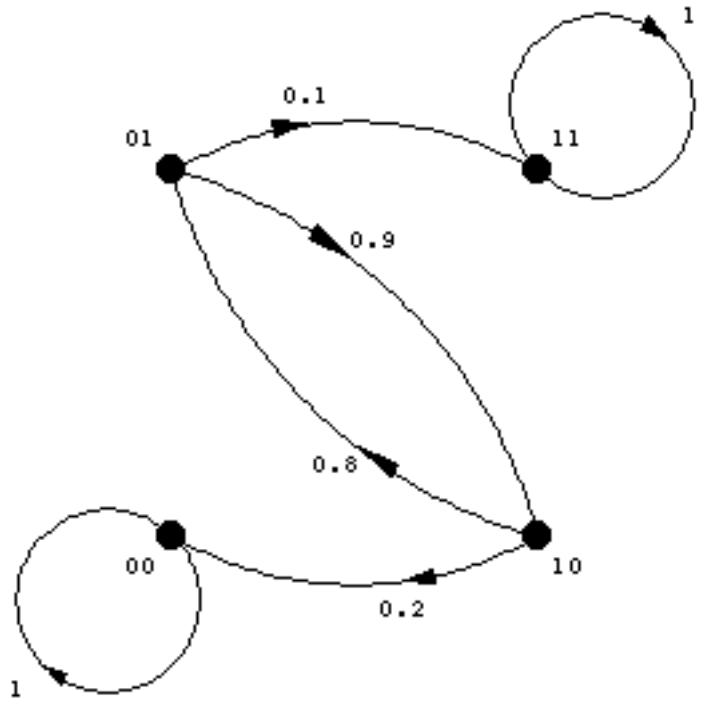


Figura 2.5. Grafico della sorgente dell'Esempio 2.6

La matrice Π è la seguente:

	00	01	10	11
00	1	0	0	0
01	0	0	0.9	0.1
10	0.2	0.8	0	0
11	0	0	0	1

ed è riducibile. Spostando la quarta riga (e colonna) fra la prima e la seconda si ottiene la matrice:

	00	11	01	10
00	1	0	0	0
11	0	1	0	0
01	0	0.1	0	0.9
10	0.2	0	0.8	0

Gli autovalori sono {1.,1.,0.85…, -0.85…}. Due autovettori sinistri linearmente indipendenti di Π associati agli autovalori unitari sono (1,0,0,0) e (0,0,0,1). Gli insiemi essenziali sono {00} e {11}. Ogni vettore di probabilità con solo la prima e l'ultima componente non nulle è combinazione lineare degli autovettori associati agli autovalori 1 e quindi una distribuzione stazionaria. ❑

Sorgenti regolari

Definizione 2.10. Una sorgente è detta **regolare** se esiste un vettore di probabilità w tale che per ogni vettore di probabilità w_0 vale la relazione:

$$\lim_{n \to \infty} w_0^T \Pi^n = w^T$$ ❑

Le sorgenti regolari hanno molte proprietà interessanti:

Teorema 2.5. Se Π è la matrice di Markov di una sorgente regolare allora:

1) Esiste una sola distribuzione stazionaria w.
2) A regime (ovvero per n molto grande) la sorgente è stazionaria qualunque sia la distribuzione iniziale[1] .
3) La sorgente è indecomponibile
4) $\lim_{k \to \infty} \Pi^k = W$ dove le righe di W sono tutte uguali a w^T.
5) Π ha un solo autovalore uguale a 1 e tutti gli altri autovalori sono in modulo minori di 1.

Dim.

1) il vettore $w^T\Pi$ deve coincidere con w^T e quindi w è una distribuzione stazionaria. Per ogni distribuzione stazionaria z vale

$$z^T \Pi^k = z^T$$

e quindi, per la Definizione 2.10, $z=w$.

[1] Da un punto di vista strettamente matematico la sorgente non è mai stazionaria a meno che non si parta con la distribuzione $w_0 = w$.

2) Per qualunque distribuzione iniziale w_0 vale (per n grande) $w_0^T \Pi^n \approx w^T$

3) Se la sorgente fosse decomponibile avrebbe più di una distribuzione stazionaria, violando la proprietà 1).

4) Si dimostra prendendo come vettore w_0 il vettore che ha un solo uno in posizione i e zero altrove e ricavando la riga i-esima della matrice limite W ottenendo sempre lo stesso risultato w^T, in notazione matematica si ha:

$$\lim_{k \to \infty} \Pi^k = \lim_{k \to \infty} I\,\Pi^k = \begin{pmatrix} \lim_{k \to \infty} (1,0,\ldots,0)\,\Pi^k \\ \hline \lim_{k \to \infty} (0,1,\ldots,0)\,\Pi^k \\ \hline \ldots \\ \hline \lim_{k \to \infty} (0,0,\ldots,1)\,\Pi^k \end{pmatrix} = \begin{pmatrix} w^T \\ \hline w^T \\ \hline \ldots \\ \hline w^T \end{pmatrix} = W$$

5) Gli autovalori di W sono i limiti degli autovalori di Π^n, ma W ha rango 1 e quindi un solo autovalore 1 e tutti gli altri sono nulli[1]. Ne segue che Π può avere un solo autovalore di modulo unitario. ❑

Due di queste proprietà (la 4 e la 5) caratterizzano completamente le sorgenti regolari. Valgono infatti i seguenti teoremi.

Teorema 2.6. Se

$$\lim_{k \to \infty} \Pi^k = W$$

dove le righe di W sono tutte uguali ad uno stesso vettore w^T allora Π è la matrice di Markov di una sorgente regolare.

Dim. Per ogni vettore di probabilità w_0 vale la relazione $w_0^T W = w^T$ e la sorgente è regolare in base alla Definizione 2.10. ❑

Teorema 2.7. Se Π ha un solo autovalore uguale a 1 e tutti gli altri autovalori sono in modulo minori di 1 allora Π è la matrice di Markov di una sorgente regolare.

Dim. Al limite Π^n tende ad una matrice Z di rango 1 con un autovalore 1 e tutti gli altri nulli. Poiché Z ha rango 1 le righe di Z sono tra loro proporzionali; sia $u^T=(1,1,\ldots 1)$ da $\Pi u = u$, segue $Zu = u$ e quindi le righe di Z hanno la stessa somma. Allora le righe di Z sono uguali tra loro e si può applicare il Teorema 2.6. ❑

Per le altre tre proprietà (la 1, la 2 e la 3) si trovano esempi di sorgenti non regolari che le soddisfano.

[1] Una matrice quadrata di rango 1 ha al più un solo autovalore non nullo.

Esempio 2.7. La sorgente con memoria 1 definita dalla probabilità:
$p(0|0) = 0$ $p(0|1) = 1$:

	0	1
0	0	1
1	1	0

Ha un solo insieme essenziale {0,1}, una sola distribuzione stazionaria (1/2,1/2), è indecomponibile, con matrice irriducibile, ma non è regolare. Gli autovalori sono 1 e -1. Le potenze pari di Π sono la matrice identica di ordine 2 e le potenze dispari di Π sono Π. Questo esempio mostra che le proprietà 1 e 3 non sono sufficienti a garantire la regolarità.

Esempio 2.8. La sorgente con memoria 1 definita dalla probabilità:
$p(0|0) = 1$ $p(0|1) = 0$:

	0	1
0	1	0
1	0	1

Ha 2 insiemi essenziali, ogni vettore di probabilità è una distribuzione stazionaria e quindi la sorgente è sempre stazionaria ma non è regolare. Questo esempio mostra che la proprietà 2 non è sufficiente a garantire la regolarità. ❑

Vediamo altri esempi "patologici".

Esempio 2.9. La sorgente con memoria 1 definita dalla probabilità:
$p(0|0) = 1$ $p(0|1) = 0.6$

	0	1
0	1	0
1	0.6	0.4

Ha un solo insieme essenziale (lo stato 0), una sola distribuzione stazionaria (1,0), è regolare ma la matrice è riducibile (è già ridotta).

Esempio 2.10. La sorgente con memoria 1 definita dalla probabilità:
$p(0|0) = 0$ $p(1|0) = 1$
$p(0|1) = 1$
$p(0|2) = 1/3$ $p(1|2) = 1/3$

	0	1	2
0	0	1	0
1	1	0	0
2	1/3	1/3	1/3

Ha un solo insieme essenziale, una sola distribuzione stazionaria (0.5,0.5,0), non è regolare, la matrice è riducibile (già ridotta). ❑

Schema riassuntivo delle combinazioni

sorgente		matrice	
indecomponibile	regolare	irriducibile	Es. 2.3, 2.4, 2.5
indecomponibile	non regolare	irriducibile	Es. 2.7
indecomponibile	regolare	riducibile	Es. 2.9
indecomponibile	non regolare	riducibile	Es. 2.10
decomponibile	regolare	irriducibile	CASO IMPOSSIBILE
decomponibile	non regolare	irriducibile	CASO IMPOSSIBILE
decomponibile	regolare	riducibile	CASO IMPOSSIBILE
decomponibile	non regolare	riducibile	Es. 2.6, 2.8

Proprietà delle sorgenti in base ad autovalori e autovettori.

L'impostazione che abbiamo usato, basata su autovalori e autovettori della matrice di Markov permette di determinare facilmente in modo algoritmico le proprietà della sorgente.

Algoritmo 2.1. Per determinare le proprietà di una sorgente a cui è associata una matrice di Markov Π

1) si calcolano gli autovalori e gli autovettori di Π;
2) se vi sono $k>1$ autovalori uguali a 1 allora la sorgente è decomponibile, non regolare, con k insiemi essenziali;
3) altrimenti se vi sono più autovalori uguali, in modulo, a 1 allora vi è un solo insieme essenziale e una sola distribuzione stazionaria ma la sorgente non è regolare;
4) se vi è un solo autovalore uguale a 1 e tutti gli altri sono, in modulo, minori di 1 allora la sorgente è regolare e la distribuzione a regime è ottenibile normalizzando l'autovettore associato.

Esempio 2.11. (Sorgente adiacente)
Vediamo ora un esempio di come data una sorgente regolare a memoria m si possa trovare la sorgente adiacente e le varie approssimazioni. Si consideri la sorgente con memoria 3 dell'Esempio 2.5. Le probabilità che definiscono la sorgente sono

$$p(0|000) = 0.9 \qquad p(0|001) = 0.4$$
$$p(0|010) = 0.7 \qquad p(0|011) = 0.9$$
$$p(0|100) = 0.9 \qquad p(0|101) = 0.8$$
$$p(0|110) = 0.7 \qquad p(0|111) = 0.6$$

Gli autovalori della matrice di Markov sono: (1., -0.4…, 0.38… , 0.05… ,-0.06… , 0, 0, 0}. L'autovettore associato all'autovalore 1 una volta normalizzato dà la distribuzione stazionaria

$$w^{\mathrm{T}}=(0.658\ldots,0.073\ldots,0.054\ldots,0.050\ldots,0.073\ldots,0.031\ldots,0.050\ldots,0.008\ldots).$$

La relazione $w^{\mathrm{T}} = w^{\mathrm{T}}\,\Pi$ si può scrivere:

$$w_1\,p(0|000) + w_5\,p(0|100) = w_1, \qquad w_1\,p(1|000) + w_5\,p(1|100) = w_2,$$
$$w_2\,p(0|001) + w_6\,p(0|101) = w_3, \qquad w_2\,p(1|001) + w_6\,p(1|101) = w_4,$$
$$w_3\,p(0|010) + w_7\,p(0|110) = w_5, \qquad w_3\,p(1|010) + w_7\,p(1|110) = w_6,$$
$$w_4\,p(0|011) + w_8\,p(0|111) = w_7, w_4\,p(1|011) + w_8\,p(1|111) = w_8.$$

La probabilità di emettere 0 è:

$$w_1\,p(0|000) + w_2\,p(0|001) + w_3\,p(0|010) + w_4\,p(0|011) + w_5\,p(0|100) + w_6\,p(0|101) + w_7\,p(0|110) + w_8\,p(0|111) = 0.837\ldots$$

La sorgente adiacente (ovvero l'approssimazione S_0) emette quindi 0 con probabilità 0.837…. e 1 con probabilità 0.163…
Un metodo alternativo per ottenere la sorgente adiacente consiste nel sommare le probabilità stazionarie degli stati che hanno lo stesso ultimo simbolo, ovvero:

$$p(0) = w_1 + w_3 + w_5 + w_7 = 0.837\ldots,$$
$$p(1) = w_2 + w_4 + w_6 + w_8 = 0.163\ldots \qquad \square$$

2.3 Entropia delle sorgenti

Definizione 2.11. Indicando con X_i l'esperimento che genera il simbolo i-esimo della sorgente si definisce

$$H(S) = \lim_{n \to \infty} H(X_n | X_1, X_2, \ldots, X_{n-1})$$

Teorema 2.8. Per una sorgente stazionaria l'entropia esiste.
Dim. Utilizzando il Teorema 1.4 ed il fatto che la sorgente è stazionaria si ha

$$H(X_n|X_1,X_2,\ldots,X_{n-1}) \le H(X_n|X_2,\ldots,X_{n-1}) = H(X_{n-1}|X_1,\ldots,X_{n-2})$$

e il limite che definisce l'entropia essendo applicato su una successione decrescente e non negativa esiste finito. ❑

È ovvio il seguente risultato.

Teorema 2.9. Se S è una sorgente senza memoria che emette da un alfabeto di r simboli in modo indipendente con la distribuzione stazionaria $w^T=\{w_1, w_2, \ldots, w_r\}$ la sua entropia è $H(S)= H(w_1, w_2, \ldots, w_r)$. ❑

Più interessante è il caso di una sorgente con memoria in una distribuzione stazionaria oppure di una sorgente regolare.

Teorema 2.10. Sia S una sorgente con memoria m che emette r simboli $\{a_1, a_2,\ldots,a_r\}$ con matrice di Markov Π. Se S è in una distribuzione stazionaria $w^T=\{w_1, w_2, \ldots, w_s\}$, oppure S è regolare con distribuzione limite w, la sua entropia è:

$$H(S) = -\sum_{i=1}^{s} w_i \sum_{j=1}^{s} \pi_{ij} \log \pi_{ij}$$

Dim. Siano $\{\sigma_1, \sigma_2,\ldots,\sigma_s\}$ gli $s=r^m$ stati della catena di Markov associata alla sorgente, per l'entropia dello stato σ_i vale l'identità:

$$\sum_{j=1}^{s} \pi_{ij} \log \pi_{ij} = \sum_{j=1}^{r} p(a_j|\sigma_i) \log p(a_j|\sigma_i), \qquad (2.3)$$

infatti il secondo membro si ottiene dal primo eliminando termini per cui $\pi_{ij}=0$.
In base alla definizione di $H(S)$ e sfruttando il fatto che S ha memoria m si ha

$$H(S) = \lim_{n \to \infty} H(X_n|X_{n-m},X_{n-m+1},\ldots,X_{n-1}) =$$
$$= \lim_{n \to \infty} H(X_n|\Sigma_n)$$

dove Σ_n è l'esperimento che ha come uscita gli ultimi m simboli emessi. L'uscita di Σ_n può essere vista come uno stato della catena di Markov. Svolgendo l'espressione di $H(X_n|\Sigma_n)$ in base alla definizione di entropia condizionata e usando la (2.3) si ha:

$$H(S) = \lim_{n \to \infty} \sum_{i=1}^{s} p(\sigma_i) H(X_n | \sigma_i) =$$

$$= - \lim_{n \to \infty} \sum_{i=1}^{s} p(\sigma_i) \sum_{j=1}^{r} p(a_j | \sigma_i) \log p(a_j | \sigma_i) =$$

$$= - \lim_{n \to \infty} \sum_{i=1}^{s} p(\sigma_i) \sum_{j=1}^{s} \pi_{ij} \log \pi_{ij}$$

Se la sorgente è stazionaria $p(\sigma_i) = w_i$ e l'argomento del limite è indipendente da n, se la sorgente è regolare $p(\sigma_i) \varnothing w_i$ per qualunque distribuzione iniziale. ❑

Teorema 2.11. L'entropia $H(S)$ di una sorgente stazionaria S, con memoria finita, è non maggiore della entropia della sua adiacente $\bar{S}$, con l'uguaglianza che sussiste se e solo se S è senza memoria.

Dim. Usando la (2.3) si ha:

$$H(S) = - \sum_{i=1}^{s} w_i \sum_{j=1}^{s} \pi_{ij} \log \pi_{ij} =$$

$$= - \sum_{i=1}^{s} w_i \sum_{j=1}^{r} p(a_j | \sigma_i) \log p(a_j | \sigma_i) .$$

Per il lemma del logaritmo si ha:

$$- \sum_{j=1}^{r} p(a_j | \sigma_i) \log p(a_j | \sigma_i) \le - \sum_{j=1}^{r} p(a_j | \sigma_i) \log p(a_j)$$

con l'uguaglianza che vale solo se la sorgente è senza memoria e quindi:

$$H(S) \le - \sum_{i=1}^{s} w_i \sum_{j=1}^{r} p(a_j | \sigma_i) \log p(a_j) = - \sum_{j=1}^{r} \log p(a_j) \sum_{i=1}^{s} w_i p(a_j | \sigma_i)$$

Sfruttando la relazione:

$$p(a_j) = \sum_{i=1}^{s} w_i p(a_j | \sigma_i)$$

si ha:

$$H(S) \le - \sum_{j=1}^{r} p(a_j) \log p(a_j) = H(\bar{S}) . ❑$$

Lemma 2.1. Data una sorgente S, con memoria m, in una distribuzione stazionaria $w=\{w_1, w_2, \ldots, w_s\}$ e la sua estensione k-esima con $k \ge m$ si ha

$$H\left(\overline{S^k}\right) = k\, H(S) + \left[H\left(\overline{S^m}\right) - m H(S) \right] .$$

Dim. (vedi Gestri, p. 38) ❑

Teorema 2.12. Data una sorgente S, con memoria m, in una distribuzione stazionaria $w = \{w_1, w_2, \ldots, w_s\}$, la seguente relazione lega l'entropia della sorgente e quella della adiacente della sua estensione k-esima:

$$\lim_{k \to \infty} \frac{H\left(\overline{S^k}\right)}{k} = H(S).$$

Dim. Si ottiene dal Lemma 2.1 dividendo per k e passando al limite. ❑

2.4 Esercizi

Esercizio 2.1. Che memoria ha una sorgente che emette due simboli A e B con le seguenti probabilità? (l'ultimo simbolo emesso è quello più a destra).

$$P(A|ABB) = 0.6, \quad P(A|BBA) = 0.4,$$
$$P(A|AAA) = 0.2, \quad P(A|BBB) = 0.6,$$
$$P(A|ABA) = 0.4, \quad P(A|BAB) = 0.3$$
$$P(A|AAB) = 0.3, \quad P(A|BAA) = 0.2$$

Esercizio 2.2. Che relazione c'è tra sorgenti regolari e sorgenti indecomponibili?

Esercizio 2.3. Consideriamo una sorgente S senza memoria che emette 0 con probabilità 1/4 e 1 con probabilità 3/4. Descrivere la sorgente adiacente della estensione seconda di S.

Esercizio 2.4. Una sorgente discreta con memoria può essere sempre rappresentata da una matrice di Markov. Sotto quali condizioni vale il viceversa?

Esercizio 2.5. Dare un esempio di sorgente regolare con memoria 2. Dare un esempio di sorgente con memoria infinita.Dare un esempio di sorgente decomponibile con memoria 2. Dare un esempio di sorgente indecomponibile con memoria 3. Dare un esempio di sorgente non regolare con memoria 1 e almeno 3 simboli. Dare un esempio di sorgente non regolare, indecomponibile, con memoria 1 e almeno 7 simboli.

Esercizio 2.6. Calcolare la sorgente adiacente della sorgente definita dalla matrice:

	0	1
0	1	0
1	0.6	0.4

Esercizio 2.7. Calcolare la sorgente S_0 adiacente della sorgente S definita dalla matrice:

	0	1
0	0.5	0.5
1	0.5	0.5

Dire se S_0 coincide con S?

Esercizio 2.8. Calcolare la sorgente estensione seconda della sorgente definita dalla matrice:

	0	1
0	1	0
1	0.5	0.5

Esercizio 2.9. Calcolare la sorgente S^2 estensione seconda della sorgente S definita dalla matrice:

	0	1
0	0.5	0.5
1	0.5	0.5

Dire se S^2 coincide con S?

3
Codifica in assenza di rumore

3.1 Codifica di una sorgente

Definizione 3.1. Data una sorgente S con alfabeto di uscita $\{s_1, s_2, \ldots, s_k\}$ e un alfabeto finito $\Gamma = \{a_1, a_2, \ldots, a_r\}$, si dice **codice a blocchi** una funzione che associa ad ogni simbolo di S (detto anche **messaggio)** una e una sola sequenza di simboli di Γ. L'insieme Γ è detto **alfabeto del codice**. Le sequenze di simboli di Γ si chiamano **parole del codice** e possono avere lunghezze diverse tra loro.❑

Di norma si ha $r \leq k$. Infatti non avrebbe molto senso codificare con un alfabeto del codice Γ con più simboli del numero complessivo dei messaggi da codificare.

Ad ogni simbolo s_i di S si associa una stringa $x_i = \alpha_1^{(i)}\alpha_2^{(i)}\cdots\alpha_{n_i}^{(i)}$ di simboli di Γ, Se p_k è la probabilità stazionaria di s_k si può definire la **lunghezza media del codice**:

$$\bar{n} = \sum_{i=1}^{k} p_i |x_i|$$

che si misura nelle unità del codice. In particolare se $r=2$, $\bar{n}$ si misura in bit, se $r=3$ in cifre ternarie, se $r=8$ in cifre ottali, se $r=16$ in cifre esadecimali, se $r=256$ in byte, e cosi via. Il problema "naturale" della codifica è abbassare quanto possibile la lunghezza media conservando la possibilità di decifrare le sequenze emesse da S. In altre parole l'uscita del codificatore dovrebbe essere una forma "compressa" dell'uscita della sorgente.

Per confrontare codifiche su alfabeti di diversa cardinalità basta misurare la lunghezza media in bit moltiplicando $\bar{n}$ per $\log r$. Questa operazione è perfettamente comprensibile se $\log r$ è un numero intero (3 *byte* e 8 cifre ottali corrispondono ovviamente a 24 bit) ma conserva il suo senso anche per valori non interi di $\log r$ in quanto in generale si presuppone di trasmettere non un numero finito di uscita della sorgente ma un "flusso di dati" di lunghezza indefinita.

Definizione 3.2. Un codice si dice **univocamente decifrabile** se data una qualunque sequenza di simboli del codice, esiste una sola sequenza di simboli della sorgente che può averla generata. Un codice si dice **istantaneo** se non esiste una parola di codice che sia prefisso di un'altra parola di codice. ❑

Per esempio, i codici {00, 01, 10, 11} e {1, 01, 001} sono univocamente decifrabili. Il codice {0, 01, 001} non è univocamente decifrabile. I codici {00, 01, 10, 11} e {1, 01,001} sono istantanei; il codice {1, 10, 100} è univocamente decifrabile ma non istantaneo.

Osservazione 3.1. In un albero **r-ario** ogni nodo non foglia ha esattamente r figli. Esempi tipici sono gli alberi **binari**, gli alberi **ternari**, ecc. Un codice istantaneo su un alfabeto di *r* elementi può essere rappresentato come un sottoinsieme delle foglie di un albero *r*-ario. Un codice istantaneo è detto **completo** se tutte le foglie dell'albero *r*-ario fanno parte del codice. ❑

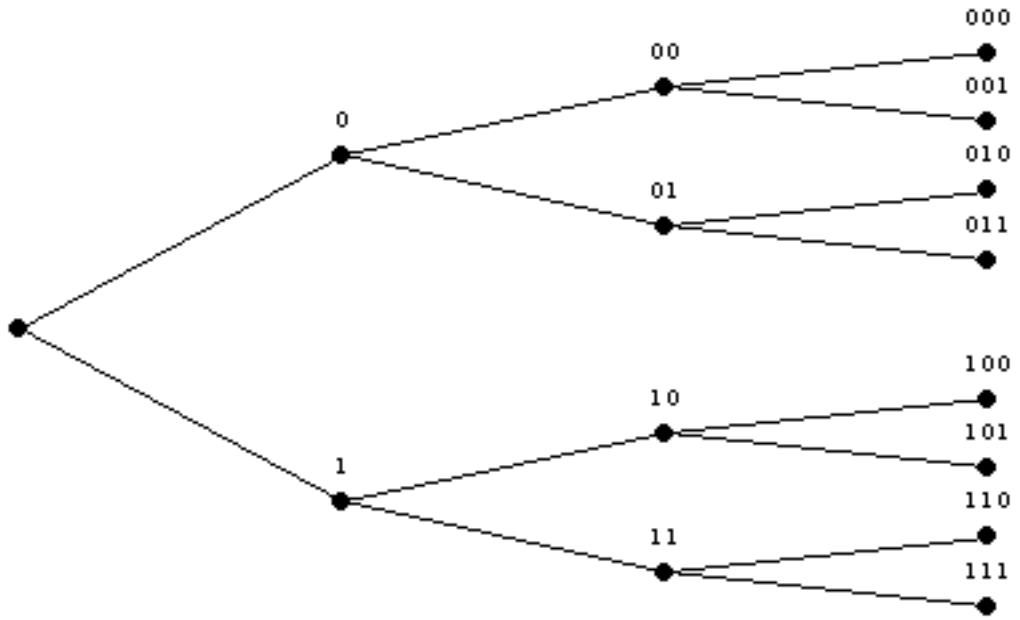


Figura 3.1. Corrispondenza stringhe-nodi di un albero binario.

Nel caso r=2 la corrispondenza albero-insieme delle stringhe binarie, è mostrata in Figura 3.1. In Figura 3.2 è mostrato il codice istantaneo {1, 01, 001}. In Figura 3.3 è mostrato il codice istantaneo {00, 01, 10, 11}.

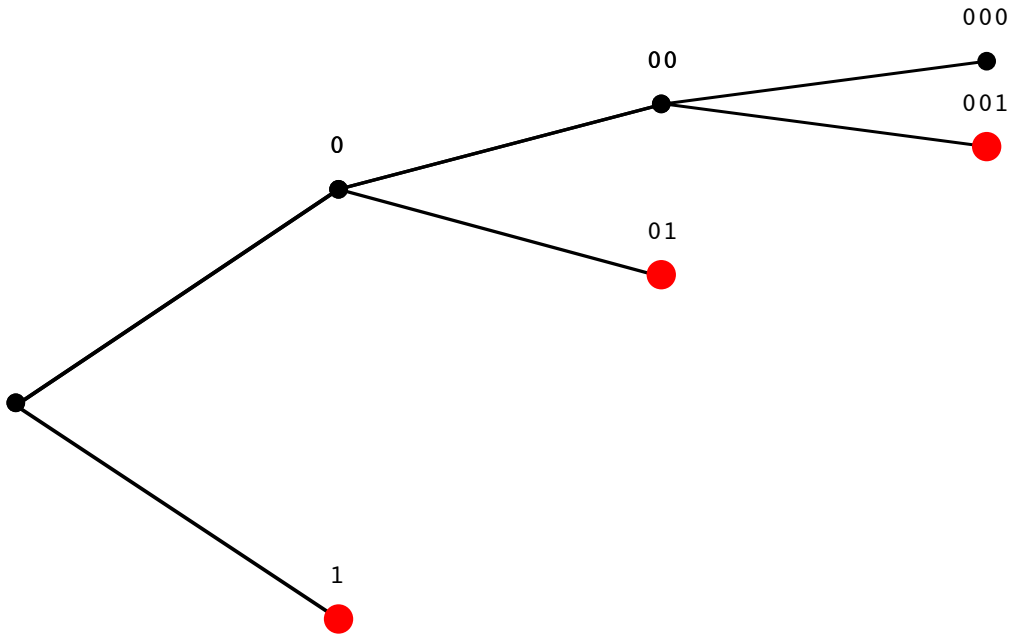


Figura 3.2. Rappresentazione su un albero binario del codice {1,01,001}; le parole di codice sono indicate dai pallini più grandi

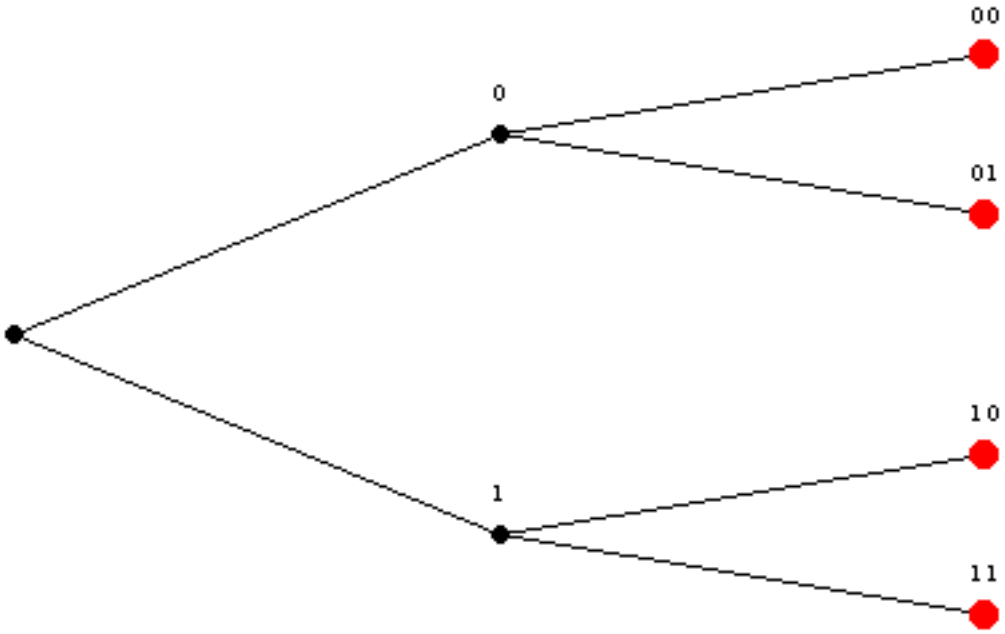


Figura 3.3. Codice {00,01,10,11}.

Teorema 3.1. I codici istantanei sono univocamente decifrabili.
Dim. Basta seguire l'albero associato al codice scegliendo il percorso in base ai simboli in arrivo. Quando si arriva ad una foglia, è stata decodificata in modo univoco una parola del codice e si riparte dalla radice per decodificare la successiva. ❑

3.2 Alcune proprietà dei codici

Disuguaglianza di Kraft

Lemma 3.1. Sia A un albero r-ario; siano f_i, i=1,2,…,k le foglie dell'albero, e sia l_i la distanza tra una foglia e la radice. Vale la seguente relazione.

$$\sum_{i=1}^{k} r^{-l_i} = 1.$$

Dim. Per un albero composto dalla sola radice vale k=1, l_1=0 e poiché $r^0 = 1$ la relazione è vera. Si suppone che la relazione valga per ogni albero con s nodi non foglia e si aggiungono r foglie ad una foglia di livello h. Il valore di s viene aumentato di *1* e k cresce di r-1. Nel bilancio dell'equazione si perde un termine di peso r^{-h} e si guadagnano r termini di peso $r^{-(h+1)}$, ovvero se vale

$$\sum_{i=1}^{k} r^{-l_i} = 1 \qquad \text{per } s \text{ nodi non foglia;}$$

allora vale

$$\sum_{i=1}^{k+r-1} r^{-l_i} = 1 - r^{-h} + r\, r^{-h-1} = 1 \qquad \text{per } s\text{+1 nodi non foglia.}$$

quindi il bilancio non viene alterato e il Lemma è dimostrato. ❑

N.B. Questa dimostrazione è un esempio tipico di applicazione della induzione agli alberi.

Teorema 3.2 (Disuguaglianza di Kraft). Dato un codice istantaneo su un alfabeto di r elementi siano x_i, i=1,2,…,k, le parole del codice, con $|x_i|=n_i$. Vale la seguente relazione.

$$\sum_{i=1}^{k} r^{-n_i} \le 1 \qquad (3.1)$$

Viceversa è sempre possibile costruire un codice istantaneo con k parole le cui lunghezze soddisfano la (3.1).

Dim. La prima parte del teorema deriva dall'Osservazione 3.1 e dal Lemma 3.1. Infatti al codice istantaneo si può associare un sottoinsieme di foglie di un albero r-ario e nella (3.1) vale il segno di uguaglianza se e solo se il codice istantaneo è completo. Per la seconda parte, senza perdere di generalità supponiamo $n_1 \le n_2 \le \ldots \le n_k$. Per la condizione che nessuna parola può essere prefisso di un'altra, ogni scelta di una parola di lunghezza h elimina r^{i-h} possibilità di lunghezza $i \ge h$. Scegliamo x_1 come una parola arbitraria di lunghezza n_1. Dopo questa operazione sono state eliminate $r^{n_2-n_1}$ possibili parole di lunghezza n_2. Continuiamo a scegliere x_i come una parola possibile di lunghezza n_i avendo ogni volta un numero di possibilità di scelta dato da:

$$r^{n_i} - \sum_{j=1}^{i-1} r^{n_i - n_j} = r^{n_i}\left(1 - \sum_{j=1}^{i-1} r^{-n_j}\right) \qquad (3.2)$$

Poiché vale la (3.1), la (3.2) è sempre positiva per tutti gli i≤k e almeno una scelta è sempre possibile. Dopo avere effettuato la k-esima scelta se il codice è completo la (3.2) diviene nulla per ogni possibile valore n_{k+1}; altrimenti è ancora possibile aggiungere al codice parole sufficientemente lunghe da continuare a soddisfare la (3.1). ❑

Disuguaglianza di MacMillan

Teorema 3.3 (Disuguaglianza di MacMillan). Per ogni codice univocamente decifrabile su un alfabeto di r elementi vale la relazione (3.1). Viceversa è sempre possibile costruire un codice univocamente decifrabile con k parole le cui lunghezze, soddisfano la (3.1).

Dim. Siano x_i, i=1,2,…,k le parole del codice, con $|x_i|=n_i$. Senza perdere di generalità supponiamo $p=n_1 \le n_2 \le \ldots \le n_k=q$. Consideriamo la potenza s-esima del primo membro della (3.1). Essa è una sommatoria di k^s termini ognuno dei quali rappresenta una delle possibile sequenze di s parole del codice, in quanto è del tipo

$$r^{-n_{i_1}} r^{-n_{i_2}} \ldots r^{-n_{i_s}} = r^{-h} \quad \text{con} \quad h = n_{i_1} + n_{i_2} + \ldots + n_{i_s}$$

Denotiamo con ω_h il numero di sequenze di lunghezza h. Contando e raggruppando le sequenze di uguale lunghezza si ha:

$$\left(\sum_{i=1}^{k} r^{-n_i}\right)^s = \sum_{h=sp}^{sq} \omega_h r^{-h} .$$

Poiché il codice è univocamente decifrabile le sequenze di lunghezza h non possono essere più di r^h (altrimenti avremmo che a due sequenze di s simboli della sorgente corrisponde la stessa sequenza di simboli del codice). Vale allora:

$$\left(\sum_{i=1}^{k} r^{-n_i}\right)^s \le \sum_{h=sp}^{sq} r^h r^{-h} = sq - sp \le sq$$

Da cui prendendo la radice s-esima:

$$\sum_{i=1}^{k} r^{-n_i} \le \sqrt[s]{sq} .$$

Poiché la relazione vale per qualunque s essa vale anche al limite ed essendo

$$\lim_{s \to \infty} \sqrt[s]{sq} = 1$$

il Teorema è dimostrato. Il viceversa è banale perché basta costruire un codice istantaneo (vedi Teorema 3.2) che è anche univocamente decifrabile. ❑

Corollario 3.1 Se esiste un codice univocamente decifrabile con lunghezze n_1, n_2, …,n_k su un alfabeto con r simboli allora esiste un codice istantaneo con le stesse lunghezze.

Dim. Le lunghezze $n_1, n_2, \ldots, n_k$ devono soddisfare la disuguaglianza di MacMillan che coincide con quella di Kraft, quindi in base al Teorema 3.2 si può costruire un codice istantaneo con le stesse lunghezze di parola. ❑

Nel seguito parleremo quindi solo di codici istantanei, senza perdere di generalità.

3.3 Codifica ottima in assenza di rumore

Vediamo ora alcune proprietà che devono essere soddisfatte dalla lunghezza media $\bar{n}$ di un codice istantaneo.

Teorema 3.4. Data una sorgente S che emette k simboli, e un codice istantaneo per S formato da parole di lunghezza n_1, n_2, …,n_k su un alfabeto con r simboli, per la sua lunghezza media $\bar{n}$ vale.

$$\frac{H(S)}{\log r} \le \bar{n}$$

Dim.

Costruiamo un vettore di probabilità q_i normalizzando le r^{-n_i}:

$$q_i = \frac{r^{-n_i}}{\sum_{j=1}^{k} r^{-n_j}},$$

allora vale

$$\begin{aligned}
\frac{H(S)}{\log r} &\le \frac{H(\bar{S})}{\log r} = \\
&= -\sum_{i=1}^{k} p_i \frac{\log p_i}{\log r} = -\sum_{i=1}^{k} p_i \log_r p_i \le -\sum_{i=1}^{k} p_i \log_r q_i = \\
&= -\sum_{i=1}^{k} p_i \log_r r^{-n_i} + \sum_{i=1}^{k} p_i \log_r \left(\sum_{j=1}^{k} r^{-n_j} \right) = \\
&= \sum_{i=1}^{k} p_i n_i + \log_r \left(\sum_{j=1}^{k} r^{-n_j} \right) \le \bar{n}
\end{aligned}$$

dove le tre disuguaglianze derivano rispettivamente dal Teorema 2.11, dal Lemma del Logaritmo e dalla disuguaglianza di Kraft. ❑

Teorema. 3.5. (Codifica in assenza di Rumore) Per ogni sorgente S che emette k simboli si può trovare un codice con r simboli per l'adiacente della sua estensione t-esima tale che per la sua lunghezza media $\bar{n}^{(t)}$ vale:

$$\lim_{t\to\infty} \frac{\bar{n}^{(t)}}{t} = \frac{H(S)}{\log r}$$

Dim.
Siano p_i, i=1,2,…,k^t le probabilità stazionarie dei simboli di S^t, allora si possono scegliere k^t interi n_i tali che:

$$-\log_r p_i \le n_i < -\log_r p_i + 1 \tag{3.3}$$

Verifichiamo che tali interi soddisfano la disuguaglianza di Kraft.

$$-\log_r p_i \le n_i \quad \Rightarrow \quad -n_i \le \log_r p_i \quad \Rightarrow$$

$$\Rightarrow r^{-n_i} \le p_i \quad \Rightarrow \quad \sum_{i=1}^{k^t} r^{-n_i} \le \sum_{i=1}^{k^t} p_i = 1$$

Allora si può costruire un codice istantaneo per cui vale la (3.3). Moltiplicando la (3.3) per p_i e sommando si ottiene

$$-\sum_{i=1}^{k^t} p_i \log_r p_i \le \sum_{i=1}^{k^t} p_i n_i < -\sum_{i=1}^{k^t} p_i \log_r p_i + 1$$

ovvero

$$\frac{H\left(\overline{S^t}\right)}{\log r} \le \bar{n}^{(t)} < \frac{H\left(\overline{S^t}\right)}{\log r} + 1$$

Dividendo per t e passando al limite si ottiene

$$\lim_{t\to\infty} \frac{\bar{n}^{(t)}}{t} = \lim_{t\to\infty} \frac{H\left(\overline{S^t}\right)}{t \log r} = \frac{H(S)}{\log r}$$

dove si è applicato il Teorema 2.12. ❑

Algoritmo 3.1 (Algoritmo di Huffman, caso binario)
Sia S una sorgente stazionaria che emette un insieme di parole $\{s_1, s_2, \ldots, s_k\}$ con distribuzione di probabilità $\{p_1, p_2, \ldots, p_k\}$ e si supponga $p_1 \le p_2 \le \ldots \le p_k$. Le due parole più lunghe del codice che verrà generato saranno associate a s_1 e s_2. Usiamo un bit per distinguere tra i due eventi e consideriamo l'evento [viene emesso s_1 o s_2]. Tale evento ha probabilità p_1+p_2 e l'insieme $\{p_1+p_2, p_3, \ldots, p_k\}$ è ancora una distribuzione di k–1 probabilità. In pratica si considerano i k eventi elementari come k alberi formati dalla sola radice. Ogni volta che i due eventi meno probabili vengono raggruppati,

l'evento risultante è un albero formato da un nodo e dai due sottoalberi relativi agli eventi che vengono raggruppati. Iterando il procedimento si ottiene un unico albero binario a cui è associato un codice istantaneo. Si può dimostrare che tale codice ha una lunghezza media r tale che:

$$H(p_1, p_2, \ldots, p_k) \le r \le \sum_{i=1}^{k} p_i \lceil -\log p_i \rceil .$$

e che non è possibile trovare un codice con una lunghezza media inferiore. ❑

Esempio 3.2
Si consideri una distribuzione di probabilità

$p_1 = 1/6,$
$p_2 = 1/3,$
$p_3 = 1/2.$

Gli eventi s_1 e s_2 vengono raggruppati e la nuova distribuzione è

$p_1+p_2 = 1/2$
$p_3 = 1/2.$

Un possibile codice istantaneo di Huffman è quindi

$c_1 = 00,$
$c_2 = 01,$
$c_3 = 1,$

che ha lunghezza media 1/2 + 2/3 + 2/6 = 1.5. ❑

3.5 Compressione di una sorgente

La **velocità relativa di trasmissione** R di un codice con r simboli e lunghezza media $\bar{n}$, utilizzato per codificare (e trasmettere) k simboli di una sorgente con entropia $H(S)$ è definita come segue:

$$R = \frac{H(S)}{\bar{n}} \qquad (3.4)$$

R è quindi la quantità di informazione per ogni simbolo di codice trasmesso ovvero l'informazione portata mediamente da ogni simbolo del codice. Il Teorema 3.4 ci assicura che $R \le \log r$. Ma questo è anche un risultato ovvio perché è evidente che ogni simbolo del codice non può portare una informazione maggiore del logaritmo della cardinalità dell'alfabeto cui appartiene. Se l'alfabeto del codice è binario si ha $R \le 1$. In tal caso R rappresenta la quantità di informazione che mediamente porta con sé ogni bit

che viene trasmesso, cioè ogni bit del codice.
Essendo $H(S) \leq \log k$ dalla (3.4) si ha anche

$$R \leq \frac{\log k}{n}$$

Anche se i simboli della sorgente non sono equiprobabili possono sempre essere rappresentati da sequenze di simboli di Γ di lunghezza costante pari a $\lceil \log_r k \rceil$ (ci riferiremo a tali sequenze chiamandole **codifica banale** della sorgente) e poi essere codificati, per essere trasmessi, con un codice (non banale, sempre di alfabeto Γ) di lunghezza media $\bar{n} < \lceil \log_r k \rceil$.

Se viceversa le parole di codice hanno tutte la stessa lunghezza n (n è allora intero), deve essere $k < r^n$, ovvero $\log k < n \log r$. Se i messaggi sono equiprobabili $\log k/n$ rappresenta R e si ha

$$\mathrm{R} = \log k/n < \log r$$

Abbiamo già osservato che una sequenza di simboli di codice (che è anche una sequenza di parole del codice) che costituisce una codifica banale di una sequenza di simboli di una sorgente generica (cioè con simboli non equiprobabili e indipendenti) può essere ricodificata in una nuova sequenza di simboli di codice più corta. In tal caso si dirà che la sequenza iniziale di simboli di codice è compattabile. Quindi una sequenza in cui *ogni gruppo* di $\log_r k$ simboli porta con sé una informazione $\mathrm{H}(S) < \log_r k$ simboli (ovvero meno di $\log k$ bit) o equivalentemente una sequenza in cui *ogni simbolo* porta con sé una informazione inferiore a $\log k/\log_r k$, cioè inferiore a $\log r$ bit, può essere compattata. Viceversa una sequenza di simboli di un alfabeto A con cardinalità r in cui *ogni simbolo* porta con sé l'informazione di $\log r$ bit non può essere compattata e una sequenza che non può essere compattata (cioè ricodificata con una sequenza più corta) è costituita da simboli *ognuno dei quali* porta l'informazione di $\log r$ bit, cioè è costituita da simboli equiprobabili e indipendenti.

Codificando opportunamente blocchi di t simboli della sorgente si può sempre ottenere, per t sufficientemente grande, con qualsiasi approssimazione, una sequenza di simboli del codice in cui *ogni simbolo* porta con sé l'informazione di $\log r$ bit. Una tale sequenza non è compattabile ed è costituita da simboli equiprobabili e indipendenti.

Esempio 3.1
Si consideri una sorgente S che emette {a,b,c} con probabilità {1/2, 1/4, 1/4}. Abbiamo $H(S) = 1.5$. Il codice istantaneo {a→0, b→10, c→11}, ha lunghezza media 1.5 e soddisfa banalmente (anche con t=1) la tesi del Teorema 3.5. Sia C la sorgente che emette i simboli dopo la codifica. Si nota che

$$p(0) = \frac{(p(a) + p(b))}{\frac{3}{2}} = \frac{2}{3}\left(\frac{1}{2} + \frac{1}{4}\right) = \frac{1}{2}$$

$$p(1) = \frac{(p(b) + 2p(c))}{\frac{3}{2}} = \frac{2}{3}\left(\frac{1}{4} + \frac{2}{4}\right) = \frac{1}{2}$$

da cui $H(C) = 1$. ❑

3.6 Esercizi

Esercizio 3.1. Si deve costruire un codice istantaneo binario che abbia 20 parole di lunghezza non superiore a 5. E' possibile far sì che questo codice abbia almeno due parole di lunghezza 2?

Esercizio 3.2. Supponiamo di dover codificare con 0 e 1 esattamente 97 messaggi di cui 96 hanno probabilità p_1=1/128 e uno probabilità p_2=1/4. Quanti bit per messaggio si dovranno impiegare?

Esercizio 3.3. Supponiamo di dover codificare con 0 e 1 dieci messaggi di cui nove hanno probabilità p_1=1/100 e uno probabilità p_2=91/100. Quanti bit per messaggio si dovranno mediamente impiegare codificando blocchi di messaggi estesi a piacere?

Esercizio 3.4. Dare un esempio di codice istantaneo e uno di codice univocamente decifrabile non istantaneo.

Esercizio 3.5. Dare il codice ottimale per una sorgente senza memoria con simboli {*a*,*b*,*c*,*d*,*e*,*f*} e probabilità {1/2,1/4,1/8,1/16,1/32,1/32}. Calcolare la probabilità che dopo la codifica venga emesso uno zero.

Esercizio 3.6. Dare il codice binario di Huffman per una sorgente senza memoria con simboli {*a*,*b*,*c*,*d*,*e*,*f*,*g*} e probabilità {1/5,1/5,1/5,1/10,1/10,1/10,1/10}. Calcolare la probabilità che dopo la codifica venga emesso uno zero.

4
Il canale discreto senza memoria

4.1 Definizione di canale discreto senza memoria

Per canale si intende un sistema che riceve un simbolo in entrata e rende un simbolo in uscita secondo una qualche legge casuale.

Definizione 4.1 Dato un alfabeto di ingresso $A= \{a_1, a_2, \ldots, a_r\}$ e un alfabeto di uscita $B = \{b_1, b_2, \ldots, b_s\}$ il **canale discreto senza memoria** è definito dalla matrice delle probabilità condizionate Uscita-Entrata che ha come elementi le probabilità di avere in uscita il simbolo b_j se è entrato il simbolo a_i. ❑

Data la matrice Uscita-Entrata e una distribuzione di ingresso $E = \{p(a_1), p(a_2), \ldots, p(a_r)\}$ si può trovare facilmente la distribuzione di uscita $U = \{p(b_1), p(b_2), \ldots, p(b_s)\}$:

$$p(b_j) = \sum_{i=1}^{r} p(b_j|a_i)\, p(a_i), j = 1,2,\ldots,s.$$

e la matrice delle probabilità condizionate $E|U$, ovvero delle "probabilità all'indietro".

$$p(a_i|b_j) = \frac{p(b_j|a_i)\, p(a_i)}{p(b_j)} = \frac{p(b_j|a_i) p(a_i)}{\sum_{k=1}^{r} p(b_j|a_k)\, p(a_k)}, i = 1,2,\ldots,r,\ j = 1,2,\ldots,s..$$

Analogamente si possono definire le entropie dei quattro esperimenti E, U, $E|U$ e $U|E$.

$$H(E) = -\sum_{i=1}^{r} p(a_i) \log p(a_i)$$

$$H(U) = -\sum_{j=1}^{s} p(b_j)\log p(b_j)$$

$$H(E|U) = \sum_{j=1}^{s} p(b_j)H(E|b_j)$$

$$H(U|E) = \sum_{i=1}^{r} p(a_i)H(U|a_i)$$

In base alla definizione di informazione reciproca vale

$$I(E|U) = H(E) - H(E|U) = H(U) - H(U|E) = I(U|E)$$

Definizione 4.2. La **capacità del canale** è la quantità:

$$C = \max_{E} I(E|U)$$

ovvero il massimo dell'informazione reciproca al variare della distribuzione di ingresso. ❑

Osservazione. Valgono banalmente le relazioni:

$C \le H(E) \le \log r$;

$C \le H(U) \le \log s$. ❑

4.2 Classificazione dei canali

Canale Lossless

Se nella matrice c'è un solo elemento diverso da 0 in ogni colonna il canale è detto **privo di perdite** (o **lossless**). Vale:

$$p(a_i|b_j) = \begin{cases} 1 & \text{se } p(b_j|a_i) > 0 \\ 0 & \text{altrimenti} \end{cases}.$$

e quindi $H(E|U)=0$. Allora $I(E|U) = H(E)$ che è massimo se le entrate sono equiprobabili e quindi la capacità è $\log r$. Un canale privo di perdite non introduce errori durante il passaggio dei dati in quanto è sempre possibile ricostruire l'entrata una volta nota l'uscita.

Esempio 4.1. Dati A= {0,1}, B={0,1,x,y} e la matrice di probabilità:

	0	1	x	y
0	1/2	0	1/2	0
1	0	2/3	0	1/3

il canale è privo di perdite. ❑

Canale Deterministico

Nella matrice c'è un solo elemento diverso da 0 in ogni riga. e quindi tale elemento vale 1. Per ogni entrata il simbolo in uscita è univocamente determinato. Vale dunque $H(U|E)$=0. Allora $I(E|U) = H(U)$. Non è detto che tutti i simboli in uscita siano possibili, ed esiste certamente una distribuzione di ingresso per cui tutti i simboli possibili sono equiprobabili. La capacità è quindi log s, dove s è il numero dei simboli possibili in uscita.

Esempio 4.2. Dati A= {0,1,2,3}, B={a,b,c,d} e la matrice di probabilità:

	a	b	c	d
0	1	0	0	0
1	0	1	0	0
2	0	1	0	0
3	0	1	0	0

il canale è deterministico e la capacità è log 2 = 1. Anche in questo caso è possibile trasmettere senza errori messaggi di un bit purché si usi in trasmissione solo le cifre 0 e 1. ❑

Canale Uniforme

Ogni riga è permutazione della prima riga e ogni colonna è permutazione della prima colonna. In questo caso il calcolo della capacità è abbastanza semplice: L'entropia condizionata $H(U|a_i)$

$$H(U|a_i) = -\sum_{j=1}^{s} p(b_j|a_i)\log p(b_j|a_i)$$

non dipende da i perché le righe della matrice $U|E$ contengono tutte gli stessi elementi. Indichiamo tale entropia con H', vale

$$H(U|E) = \sum_{i=1}^{r} H(U|a_i)\,p(a_i) = \sum_{i=1}^{r} H'\,p(a_i) = H' \ .$$

Quindi il massimo di $I(E|U) = H(U) - H'$ si ottiene massimizzando $H(U)$. Se i simboli di ingresso sono equiprobabili anche i simboli in uscita sono equiprobabili (infatti anche le colonne hanno tutte gli stessi valori). La capacità del canale uniforme è quindi:

$$\log s - H',$$

e viene raggiunta per una distribuzione di ingresso equiprobabile.

Esempio 4.3. Dati A= {0,1}, B={a,b,cd} e la matrice di probabilità:

	a	b	c	d
0	1/3	1/3	1/6	1/6
1	1/6	1/6	1/3	1/3

il canale è uniforme. La capacità vale log 4 - H(1/3,1/3,1/6,1/6) = 0.082… e viene raggiunta per $p(0)=p(1)=1/2$. ❑

Canale rSC

Un importante caso particolare di canale uniforme è quello con uguale alfabeto di simboli in ingresso e uscita in cui:

$$p(b_j|a_i) = \begin{cases} p, & \text{se } i = j \\ \dfrac{1-p}{r-1} & \text{altrimenti} \end{cases}.$$

Questo canale fornisce un modello della situazione abbastanza naturale in cui ogni simbolo "passa inalterato" con probabilità p e si trasforma in uno qualunque degli altri con probabilità 1-p.

Canale BSC (Canale Binario Simmetrico)

Il caso più interessante è il canale binario simmetrico, la cui matrice è data da:

	0	1
0	p	1-p
1	1-p	p

La capacità viene raggiunta per la distribuzione di ingresso equiprobabile e vale 1-H(p,1-p), vedi Figura 4.1.

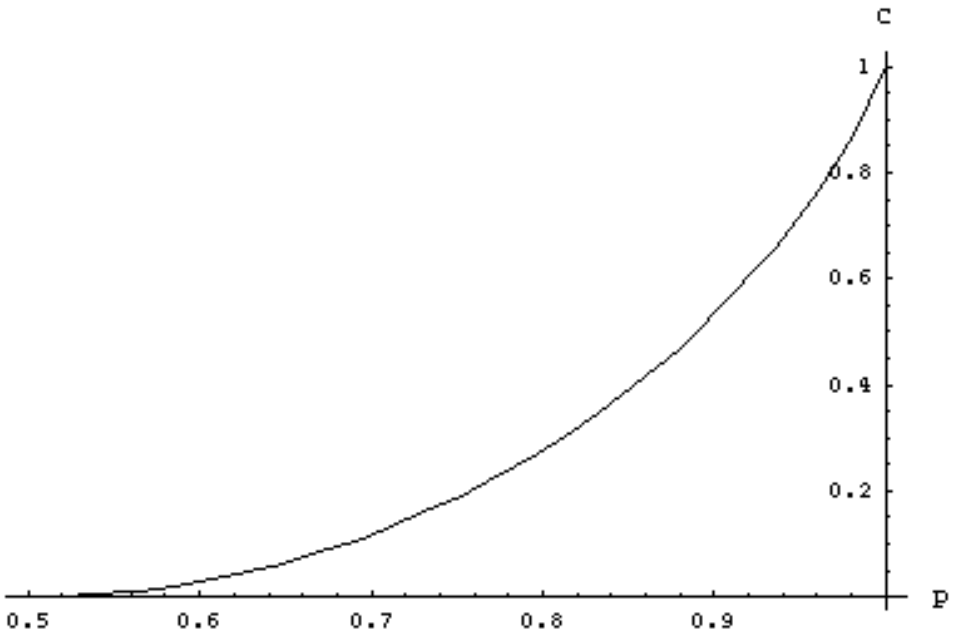


Figura 4.1. Grafico della capacità del BSC per p>0.5.

Esempio 4.4. Dati A= {0,1}, B={0,1} e la matrice di probabilità:

	0	1
0	$1\text{-}10^{-5}$	10^{-5}
1	10^{-5}	$1\text{-}10^{-5}$

il canale è un BSC con probabilità di errore 10^{-5}. La capacità vale 0.999819. ❑

4.3 Schemi di decisione e probabilità di errore

Definizione 4.3. Dato un canale con r simboli in ingresso uno **schema di decisione** è una partizione dell'insieme dei simboli di uscita B in r sottoinsiemi disgiunti $B_1, B_2,\ldots, B_r$, di cui qualcuno può essere vuoto. Più formalmente vale

$$B_u \cap B_v = \varnothing \text{ per } u \neq v, \text{ e } B = \bigcup_{i=1}^{r} B_i .$$

Sia dec una funzione da $\{1, 2, \ldots, s\}$ in $\{1, 2, \ldots, r\}$ tale che $dec(j)$ è l'unico i per cui $b_j \in B_i$, allora il simbolo in uscita b_j viene decodificato come $a_{dec(j)}$. ❑

Esempio 4.5. Dati A= $\{a_1, a_2\}$, B=$\{b_1, b_2, b_3, b_4\}$ e la matrice di probabilità:

	b_1	b_2	b_3	b_4
a_1	1/3	1/3	1/6	1/6
a_2	1/6	1/6	1/3	1/3

Un possibile schema di decisione è il seguente:

$B_1 = \{b_1, b_2\}, B_2 = \{b_3, b_4\};$

ovvero

$dec(1) = 1, dec(2) = 1, dec(3) = 2, dec(4) = 2;$

ovvero

b_1 si decodifica come a_1,
b_2 si decodifica come a_1,
b_3 si decodifica come a_2,
b_4 si decodifica come a_2. ❑

Osservazione. Da un punto di vista pratico non ha senso usare canali con $r>s$ (alcuni simboli di ingresso non potrebbero essere mai decodificati). Per questo, nel seguito assumiamo $r \leq s$.

Probabilità di errore

Una volta applicato l'algoritmo di decodifica si commette errore se il simbolo trasmesso non era quello decodificato. La probabilità di errore una volta ricevuto b_j vale

$$p_{err}\left(b_j\right) = 1 - p\left(a_{dec(j)} | b_j\right)$$

La probabilità totale di errore può essere sviluppata in funzione dei $p_{err}(b_j)$:

$$p_{err} = \sum_{j=1}^{s} p(b_j)\left(1 - p(a_{dec(j)} | b_j)\right), \qquad (4.1)$$

Analogamente si può ottenere la probabilità totale di errore in funzione della probabilità di errore dato un simbolo a_i. La probabilità di errore una volta trasmesso a_i vale

$$p_{err}\left(a_i\right) = \sum_{b_j \notin B_i} p(b_j \mid a_i)$$

e quella totale

$$p_{err} = \sum_{i=1}^{r} p(a_i) \sum_{b_j \notin B_i} p(b_j \mid a_i)$$

Schema dell'Osservatore Ideale

Se si sceglie la funzione di decodifica $dec(j)$ in modo che

$$p(a_{dec(j)} | b_j) = \max_i \; p(a_i | b_j)$$

allora la probabilità di errore è minima, in quanto viene minimizzato ogni termine della

(4.1). Questo schema è detto dell'**Osservatore Ideale**; per poterlo realizzare è necessario conoscere la matrice $p(a_i|b_j)$ e quindi la distribuzione di entrata $p(a_i)$.

Schema della massima verosimiglianza

Se la distribuzione di entrata è quella con tutti i simboli equiprobabili si ha

$$p(a_i|b_j) = \frac{p(b_j|a_i)}{r\, p(b_j)}$$

e trovare il massimo su i delle probabilità all'indietro equivale a massimizzare le probabilità in avanti ovvero trovare il valore $dec(j)$ tale che

$$p(a_{dec(j)}|b_j) = \frac{1}{r p(b_j)} \max_i p(b_j|a_i)$$

Questo schema detto della **Massima Verosimiglianza** si può applicare anche quando non si conosce la distribuzione di entrata, ma in questo caso non necessariamente minimizza la probabilità di errore.

Osservazione

Se una sorgente S con una distribuzione di uscita non uniforme X viene connessa ad un canale la cui capacità C è raggiunta per una distribuzione di ingresso non uniforme X' con $X' \neq X$ e si usa lo schema della massima verosimiglianza si va incontro a tre diversi inconvenienti.

- Se X non è uniforme la sorgente non è compressa al massimo; se si fosse usato correttamente il teorema della codifica in assenza di rumore si poteva ottenere una maggiore velocità di trasmissione (ovvero minore lunghezza media del codice) con i simboli della sorgente codificata indipendenti ed equiprobabili;
- $X' \neq X$ implica che il canale non è usato al meglio delle sue possibilità;
- lo schema della massima verosimiglianza produce una probabilità di errore maggiore dell'ottimo raggiungibile invece con lo schema dell'osservatore ideale.

Se invece la sorgente subisce un codifica ottima e il canale è uniforme (come è auspicabile che accada in pratica) non si verifica nessuno dei tre inconvenienti suddetti.

4.4 Codifica del canale in presenza di rumore

Velocità di trasmissione

Consideriamo un canale con alfabeto di ingresso di r simboli, alfabeto di uscita di s simboli (con $r \leq s$) e capacità C. Supponiamo di usare il canale 1 volta per ogni unità di tempo (per esempio 1 volta al secondo). Il passaggio di informazione attraverso il

canale è al più:

$$\log r \text{ bit/sec.}$$

Consideriamo tutte le parole di n simboli in ingresso al canale (ve ne sono r^n); se di queste ne utilizziamo solamente un numero $q \le r^n$ la velocità di tramissione diviene:

$$R = \frac{\log q}{n} \text{ bit / sec}$$

con $R \le \log r$. Ad ognuna delle $q=2^{nR}$ parole di codice in ingresso nel canale corrisponde un insieme di possibili parole in uscita ed è quindi necessario uno schema di decisione che permetta di ricostruire la parola trasmessa.

Definizione 4.4 La **codifica del canale** consiste nella scelta di:
- un numero $n \ge 1$
- q parole di lunghezza n sull'alfabeto di ingresso del canale (il codice)
- uno schema di decisione che associa ad ogni parola di n simboli di uscita del canale una delle q parole del codice. ❑

Ogni codifica del canale permette di raggiungere una determinata velocità di trasmissione con associata una determinata probabilità di errore.

Esempio 4.6 Si consideri una sorgente S che emette 0 e 1 in modo indipendente ed equiprobabile e un canale BSC con probabilità $p>0.5$ di trasmettere correttamente il bit in input. Lo schema di trasmissione banale si può esemplificare nel modo seguente:

Simbolo di S	parola di codice	parole in uscita associate	prob. errore
0	0	0	$1-p$
1	1	1	$1-p$

la probabilità di errore totale è $1-p$, la velocità di trasmissione è 1. ❑.

Esempio 4.7 Si consideri una sorgente che emette 0 e 1 in modo indipendente ed equiprobabile e un canale BSC con probabilità $p>0.5$ di trasmettere correttamente il bit in input. Uno schema di trasmissione con velocità di trasmissione 1/3 e una probabilità di errore migliore di quella dell'esempio precedente è il seguente:

Simbolo di S	parola di codice	parole in uscita associate
0	000	000 001 010 100
1	111	111 101 110 011

Per calcolare la probabilità di errore si deve prendere in considerazione la probabilità di verificarsi delle varie uscite in funzione delle entrate. Nella tabella seguente le configurazioni erronee sono evidenziate.

	000	001	010	011	100	101	110	111
0	p^3	$p^2(1-p)$	$p^2(1-p)$	$p(1-p)^2$	$p^2(1-p)$	$p(1-p)^2$	$p(1-p)^2$	$(1-p)^3$
1	$(1-p)^3$	$p(1-p)^2$	$p(1-p)^2$	$p^2(1-p)$	$p(1-p)^2$	$p^2(1-p)$	$p^2(1-p)$	p^3

La probabilità di errore è $(1-p)^3 + 3(1-p)^2p = 1 - 3p^2 + 2p^3$. ❑

Confrontiamo le due probabilità di errore al variare di p tra 0.5 e 1.

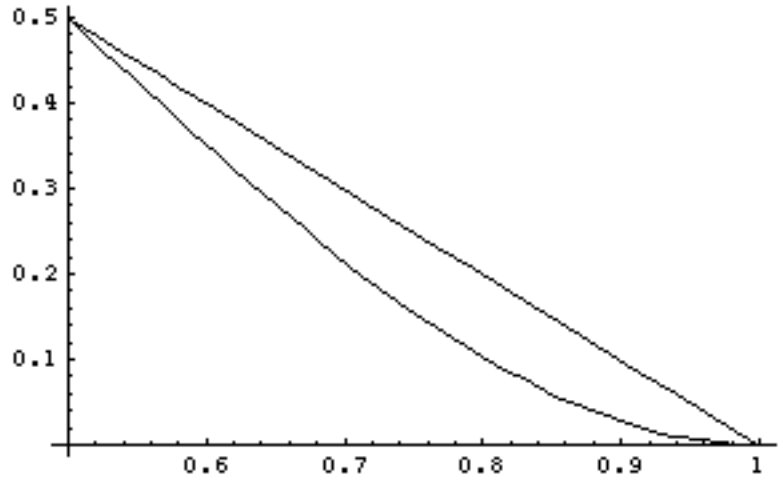


Figura 4.2. Grafico di $1-p$ e $(1-p)^3 + 3(1-p)^2p$ tra 0.5 e 1.

Nella realizzazioni pratiche il valore di p è molto vicino ad 1 e, in tali condizioni, potrebbe sembrare che le due codifiche siano pressoché equivalenti. Se plottiamo in scala doppio logaritmica la probabilità di errore in funzione di $1-p$, si nota invece la differenza di ordine di infinitesimo con cui le due probabilità di errore tendono a zero.

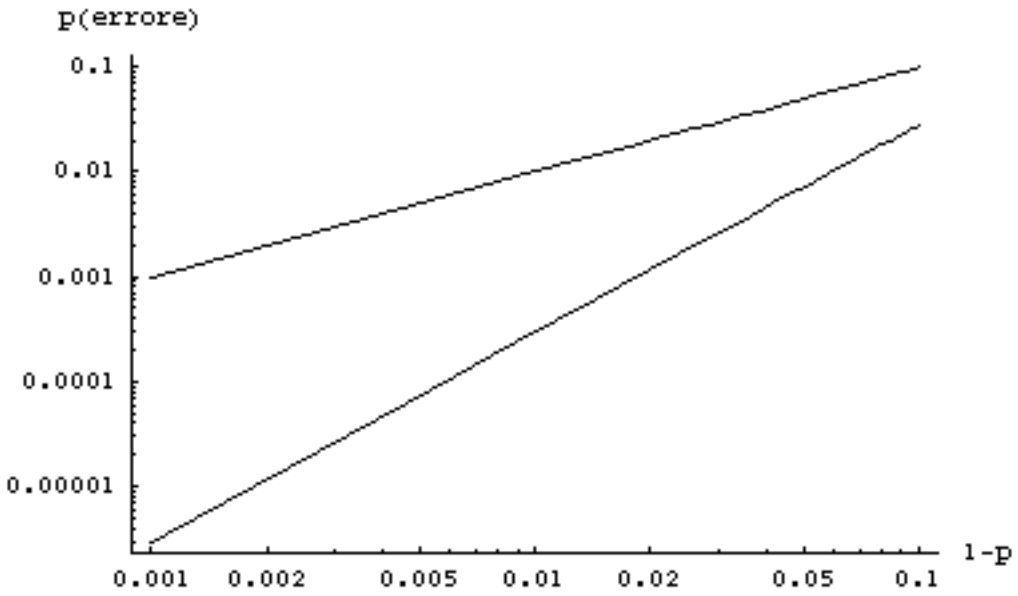


Figura 4.3. Grafico della probabilità di errore in funzione di 1-p.

❑

Il teorema seguente, detto per la sua importanza *Teorema Fondamentale*, mette in relazione con la capacità del canale la massima velocità di trasmissione, compatibile con una probabilità di errore nulla. Si noti che sia nell'enunciato che nella dimostrazione non sono presi in considerazione né l'alfabeto di ingresso né quello di uscita ma solo la velocità di trasmissione e la capacità del canale.

Teorema 4.1. (Teorema Fondamentale) Sia dato un canale con r simboli in ingresso e s simboli in uscita e capacità C. Per qualunque $R < C$ è possibile trovare una sequenza di codici di lunghezza crescente n con velocità di trasmissione R e probabilità di errore che tende a 0.

Dimostrazione informale.

Consideriamo n sufficientemente grande per poter usare sistematicamente il teorema della equipartizione asintotica (Teorema 1.6).

Sia W una sorgente stazionaria senza memoria con r simboli in uscita la cui distribuzione X raggiunge la capacità del canale. Sia Y la distribuzione corrispondente di uscita dal canale. Vale:

$$C = H(X) - H(X|Y)$$

Trasmettendo sequenze di n simboli di W e restringendosi alle sequenze tipiche, vi sono $2^{nH(X)}$ sequenze tipiche (equiprobabili) in ingresso al canale a cui corrispondono $2^{nH(Y)}$ sequenze tipiche (equiprobabili) di uscita. Considerando l'esperimento congiunto vi sono $2^{nH(X,Y)}$ coppie tipiche (equiprobabili) ingresso-uscita. Per ogni $y \in Y$ vi sono quindi:

$$\frac{2^{nH(X,Y)}}{2^{nH(Y)}} = 2^{nH(X,Y)-nH(Y)} = 2^{nH(X|Y)}$$

valori di ingresso "tipici" che possono averlo generato.
Si noti che poiché

$$R < C \leq H(X)$$

vale la relazione

$$q = 2^{nR} < 2^{nH(X)} \leq r^n$$

con la seconda disuguaglianza che diviene una uguaglianza se e solo se X è la distribuzione che contiene r^n parole equiprobabili.
Effettuiamo la codifica nel modo seguente. Ogni parola del codice è estratta a sorte con probabilità uniforme tra le $2^{nH(X)}$ parole tipiche di W^n. Immaginiamo di trasmettere x e ricevere y.

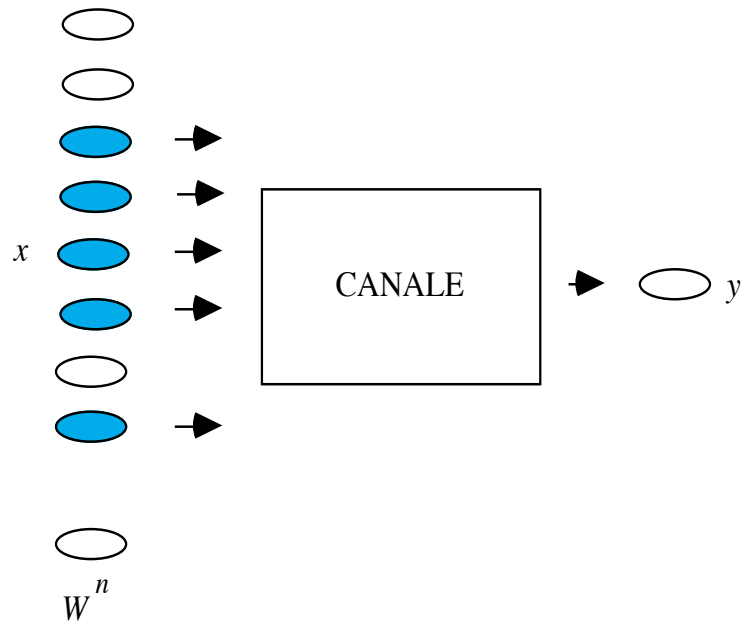


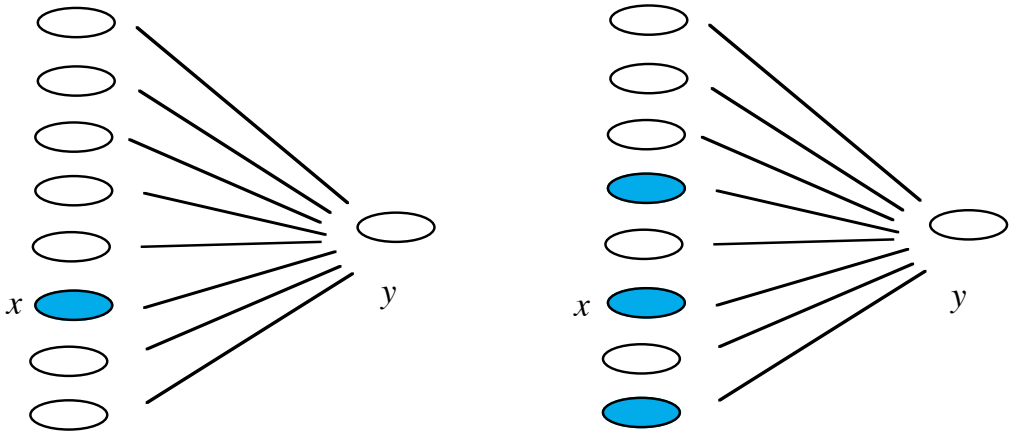
Figura 4.4 Schema di trasmissione, le parole in grigio sono le parole di codice.

Con riferimento alla Figura 4.4 chiamiamo "grigie" le parole di W^n a cui è stata assegnata una delle q parole di codice. La probabilità che una parola tipica di W^n sia "grigia" è:

$$p(x \text{ nel codice }) = \frac{2^{nR}}{2^{nH(X)}}$$

Nelle $2^{nH(X|Y)}$ parole che possono avere generato y vi è la parola x che è stata trasmessa perché la coppia (x,y) è tipica. Se questa è l'unica parola "grigia" la decodifica avviene

senza errori altrimenti essendo equiprobabili le $2^{nH(X|Y)}$ parole possibili la decodifica non può essere certa (Figura 4.5).



decodifica senza errori decodifica non corretta

Figura 4.5.

La probabilità di errore vale

$$p_{err} = \sum_{i=1}^{2^{nH(Y)}} p(y_i)\, p(\text{errore dopo aver ricevuto } y_i)$$

Gli y_i sono equiprobabili e la probabilità di cattiva decodifica è la stessa per ognuno di loro per cui

$$p_{err} = p(\text{errore dopo aver ricevuto } y)$$

Ma la probabilità di errore dopo aver ricevuto y è uguale alla probabilità che vi sia almeno un altra parola grigia tra quelle che possono aver generato y. Poiché la probabilità di "almeno 1 tra k eventi" è minore della somma delle probabilità dei k eventi, si può scrivere la disuguaglianza

$$p_{err} = p(\text{almeno un 'altra parola grigia oltre } x) \le$$

$$\le \left(2^{nH(X/Y)} - 1\right)\frac{2^{nR}}{2^{nH(X)}} < \frac{2^{nR}}{2^{n(H(X)-H(X/Y))}} =$$

$$= \frac{2^{nR}}{2^{nC}}$$

Quindi se $R<C$ allora $n\to\infty$ implica $p(e)\to 0$ e il teorema è dimostrato.
Si noti che la seconda maggiorazione è molto forte, tanto che il secondo membro non rappresenta più una probabilità e può anche raggiungere e superare 1 (come in effetti accade se le ipotesi del teorema sono violate e vale $R\ge C$). ❑

Esistono forme dell'inverso del Teorema Fondamentale che diamo senza dimostrazione.

Teorema 4.2 (Inverso debole).
1) Per ogni n finito $p_{err} = 0$ implica $R \leq C$.
2) Se $R>C$ non esistono sequenze di codici con probabilità di errore che tende a 0 per n che tende all'infinito. ❑

Teorema 4.3 (Inverso forte). Se $R>C$, e n tende all'infinito allora $p(e)$ tende a 1.

❑

4.5 Esercizi

Esercizio 4.1. Dato il seguente canale discreto

	b_1	b_2
a_1	1/3	2/3
a_2	3/4	1/4

enunciare tutti i possibili schemi di decisione e dare la probabilità di errore di quello migliore nel caso che a_1 e a_2 siano equiprobabili.

Esercizio 4.2. Dato il seguente canale discreto

	b_1	b_2	b_3
a_1	0	0.7	0.3
a_2	0.5	0.3	0.2

supponendo che i simboli di ingresso siano trasmessi con probabilità 0.3 e 0.7 trovare lo schema di decisione dell'osservatore ideale e la probabilità di errore.

Esercizio 4.3. Dato il seguente canale discreto

	b_1	b_2
a_1	0.3	0.7
a_2	0.5	0.5

supponendo che i simboli di ingresso siano trasmessi con probabilità 0.3 e 0.7 trovare lo schema di decisione dell'osservatore ideale e quello della massima verosimiglianza e la probabilità di errore per entrambi gli schemi.

5
Codici a correzione di errore

5.1 La distanza di Hamming

In questo Capitolo si considera il canale binario simmetrico definito dalla matrice

$$\begin{pmatrix} p & 1-p \\ 1-p & p \end{pmatrix}, \quad p>1/2,$$

con probabilità di ingresso {1/2,1/2} (la distribuzione che raggiunge la capacità del canale).

Inoltre tratteremo indifferentemente le stringhe di n bit come vettori di $\mathbf{R}^n$ o di $\mathbf{Z}_2^n$ a seconda delle necessità. L'insieme di tutte le stringhe di n bit viene denotato con Ω_n.

Definizione 5.1 (peso di una parola binaria). Si definisce **peso** di una parola $x \in \Omega_n$ il numero dei bit uguali ad 1 (ovvero la norma 1 in $\mathbf{R}^n$ del vettore corrispondente). Nel seguito il peso di x sarà indicato con $|x|$. ❑

Definizione 5.2. Si definisce **distanza di Hamming** tra due parole $x,y \in \Omega_n$ il valore

$|x-y|$

(ovvero il numero di bit in cui differiscono le due parole). ❑

Poiché supponiamo i bit in ingresso equiprobabili, nel nostro contesto lo schema

dell'osservatore ideale coincide con quello della massima verosimiglianza ed ha una forma molto semplice.

Teorema 5.1. Dato un codice binario $X = \{x_1, x_2, \ldots, x_q\}$ formato da q parole di Ω_n, lo schema di decodifica dell'osservatore ideale è il seguente:

se si riceve y si decodifica una delle x_i per cui

$$|x_i - y| = \min_j |x_j - y|.$$

Dim. Se si riceve y si dovrebbe trovare la parola x_i per cui è massimo $p(x_i|y)$ ma poiché i simboli di ingresso sono equiprobabili basta massimizzare $p(y|x_i)$. Per $p>1/2$ vale:

$$p(y|x_i) = p^{n-|x_i-y|}(1-p)^{|x_i-y|}$$

e ogni bit di differenza tra x_i e y contribuisce per una unità all'esponente di $(1-p)$ e diminuisce di una unità l'esponente di p. L'espressione di $p(y|x_i)$ è quindi massima quando la distanza tra x_i e y è minima. ❑

Essendo la distanza di Hamming una norma, vale la disuguaglianza triangolare:

$$|x-z| \le |x-y| + |y-z|.$$

Ora possiamo determinare le capacità correttive di un codice binario.

Definizione 5.3. Un codice binario **corregge** e errori se ogni configurazione con al più e errori viene corretta e non tutte le configurazioni con $e+1$ errori vengono corrette. ❑

Definizione 5.4. Dato un codice binario formato da q parole di Ω_n, si definisce **distanza minima** tra le parole del codice la quantità:

$$d = \min_{\substack{1\le i,j\le q \\ i \ne j}} |x_i - x_j|. \text{ ❑}$$

Teorema 5.2. Dato un codice binario con distanza minima d.

1) se $d=2e+1$, il codice corregge e errori;
2) se $d=2e$, il codice rivela e errori e corregge almeno $e-1$ errori.

Dim. Se y è una parola con al più e errori e x è la parola del codice ad essa più vicina vale $|x-y| \le e$.
Nell'ipotesi 1) per ogni altra parola z vale

$$2e+1 = d \le |x-z| \le |x-y| + |y-z| \le e + |y-z|,$$

e quindi $|y-z| > e$, ovvero e errori vengono rilevati e corretti.
Analogamente nell'ipotesi 2) vale $|y-z| \ge e$, ovvero e errori vengono rilevati ma non

corretti. ❑

Purtroppo aumentare la distanza minima tra le parole del codice ne diminuisce il numero e quindi la velocità di trasmissione.

Teorema 5.3. (Limitazione di Hamming). Per i codici binari vale la seguente relazione tra numero q di parole del codice, lunghezza del codice n, velocità di trasmissione R e numero di errori corretti e:

$$q = 2^{nR} \le \frac{2^n}{\sum_{i=0}^{e} \binom{n}{i}}$$

Dim. Segue dal fatto che le parole a distanza esattamente i da una parola data sono $\binom{n}{i}$ e quelle a distanza al più e sono $v(e) = \sum_{i=0}^{e} \binom{n}{i}$. Per correggere e errori le 2^n parole di Ω_n devono essere partizionate in q insiemi disgiunti ognuno con cardinalità almeno $v(e)$, quindi $qv(e) \le 2^n$, da cui la tesi. ❑

5.2 Considerazioni Asintotiche nel caso del BSC

Dato un BSC con probabilità di trasmissione corretta p>0.5, il numero di errori nell'i-esimo bit è una variabile casuale

$$\tau_i = \begin{cases} 0 & \text{con prob. } p \\ 1 & \text{con prob. } 1\text{-}p \end{cases}$$

con media 1-p e varianza p(1-p). Il numero di errori su n bit è quindi una variabile casuale $\xi_n = \sum_{i=0}^{n} \tau_i$ con media n(1-p) e varianza np(1-p).
È importante la seguente condizione, sufficiente perché una successione di codici abbia probabilità di errore che tende a zero per $n \to \infty$.

Teorema 5.4. Data una successione di codici che corregge e_n errori con parole di lunghezza n, se

$$\lim_{n\to\infty} \frac{e_n}{n} = L > 1 - p$$

allora

$$\lim_{n\to\infty} p_{err} = 0 \,.\, ❑$$

Definizione 5.5 Un codice per cui vale

$$\lim_{n\to\infty} \frac{e_n}{n} = L > 0$$

con velocità di trasmissione costante R è detto **asintoticamente buono** e permette di raggiungere quanto promesso dal Teorema Fondamentale su quei canali BSC per cui $1\text{-}p < L$. ❑

5.3 I codici a controllo di Parità

Definizione 5.6. Un codice a controllo di parità (CCP nel seguito) è un insieme di parole binarie di Ω_n che soddisfano al sistema lineare in $\mathbf{Z}_2$.

$$Ax = 0 \qquad (5.1)$$

dove A è una matrice di rango t con t righe e n colonne, $t<n$, detta **matrice a controllo di parità.** ❑

Dato che A ha rango t si possono individuare t colonne linearmente indipendenti. Se le colonne linearmente indipendenti sono le ultime t il codice è detto **sistematico**. La matrice si può scrivere $A = (B \mid Q)$ con Q matrice quadrata non degenere. Allora il sistema (5.1) può essere messo nella forma

$$Q^{-1}A\,x = 0$$

ovvero

$$(P|I_t)\begin{pmatrix} r \\ z \end{pmatrix} = 0$$

da cui

$$Pr + z = 0$$

e poiché si lavora modulo 2

$$z = P\,r.$$

Ogni parola del codice si può quindi decomporre in due parti:

r formata da $k = n\text{-}t$ **bit di informazione** scelti liberamente;
z formata da t **bit di parità** (o di **controllo**) che soddisfano la relazione
$Pr + z = 0$.

Il codice ha quindi 2^k parole.

Una caratterizzazione alternativa è attraverso la **matrice generatrice del codice** $G=(I_k \mid P^T)$, di k righe e n colonne.
Per ogni sequenza r di k bit si ha infatti che $x^T=(r^T \mid z^T) = r^TG = (r^T \mid r^T P^T)$ è parola di codice, vale infatti la relazione $z = P\,r$ vista precedentemente.

Poiché

$$Ax = 0, Ay = 0 \;\Rightarrow\; A(x+y) = 0$$

la somma di due parole del codice è ancora una parola di codice e i CCP sono detti anche **codici lineari**; poiché le parole formano un gruppo rispetto alla somma modulo 2, i CCP sono detti anche **codici di gruppo**.

Osservazione. Vale sempre la relazione matriciale

$$A\,G^{\mathrm{T}} = \mathrm{O},$$

dove O è una matrice di soli zeri.

Osservazione. Se la matrice G si può scrivere come $(Q_k \mid P^{\mathrm{T}})$ con Q_k matrice non singolare di dimensione k, i bit di informazione sono ancora i primi k e i bit di controllo gli ultimi n-k ma la relazione tra i bit di informazione e i bit controllo varia con la scelta della matrice Q.

Quanti sono i CCP?

Fissati n e k per ogni matrice G di rango k con k righe e n colonne si ha una diversa corrispondenza tra l'informazione r e la parola di codice generata x. Le possibili codifiche con i CCP sono quindi al più 2^{nk}. Se ci restringiamo ai codici sistematici con $G=(I_k \mid P^{\mathrm{T}})$ le k prime colonne di G sono bloccate e le possibilità sono 2^{tk}.
Permutare le righe di P^{T} porta allo stesso codice (nel senso delle insieme delle parole) ma con una diversa corrispondenza tra le cifre di informazione e le parole del codice. Permutare le colonne di P^{T} porta ad un codice diverso ma con le stesse capacità correttive, quindi restano al più $2^{tk}/t!k!$ possibilità "interessanti". In ogni caso un approccio esaustivo alla ricerca di codici buoni è impraticabile per valori di n elevati. Per esempio se n=50, k=30 vale $2^{600}/20!\ 30! \approx 6.43\ldots\ 10^{129}$.

5.4 Codifica e Decodifica per i CCP

Codifica

La codifica di un CCP è molto semplice i bit di informazione r^{T} vengono moltiplicati per la matrice P^{T} e il vettore $r^{\mathrm{T}}P^{\mathrm{T}}$ costituisce l'insieme dei bit di parità che viene aggiunto in coda ad r^{T}.

Decodifica

Si definisce **sindrome** di una parola y di lunghezza n il vettore c di lunghezza t definito dalla relazione:

$$c = A\,y$$

Vale $c = 0$ se e solo se y è una parola di codice.

Teorema 5.5. Lo schema di decodifica ottimo può essere implementato come segue: si decodifica y sommandovi una sequenza di peso minimo con la stessa sindrome; in altre parole se si riceve y si decodifica $y + z_0$ dove

$$|z_0| = \min_{Az = Ay} |z|$$

Dim. Se x è parola di codice, sia $z = x - y$, allora vale

$$Az = A(x-y) = Ax - Ay = Ay$$

e la parola di codice x da scegliere è quella per cui $|z| = |x-y|$ è minimo. ❑

Lemma 5.1. Sia dato un codice a controllo di parità $S \in \Omega_n$ con 2^k parole definito da una matrice A; allora:
1) la proprietà delle sequenze di Ω_n di avere la stessa sindrome è una relazione di equivalenza che, quindi, partiziona Ω_n in classi disgiunte dette **classi laterali**;
2) per ogni $y \in \Omega_n$ la sua classe laterale è data dall'insieme

$$\{x+y \mid x \in S\};$$

3) Le classi laterali sono $2^t = 2^{n-k}$

Dim. 1) La proprietà di avere la stessa sindrome è simmetrica, riflessiva e transitiva e definisce quindi una relazione di equivalenza. 2) x è parola di codice se e solo se $Ax=0$. Allora la relazione

$$Ay = Ax + Ay = A(x+y)$$

vale solo per le parole di codice x.
3) il numero delle parole del codice è 2^k e le classi laterali sono quindi 2^t . ❑

La decodifica di y avviene con una tabella che ad ogni sindrome associa una parola di peso minimo della classe laterale corrispondente. Si consideri una matrice $2^t \infty 2^k$ di elementi ξ_{ij} i=0,1,…,2^t-1, j=0,1,…2^k-1 costruita con le seguenti proprietà:

- ξ_{0j} sono le parole del codice;
- $A\xi_{ij} = i_B$ (la rappresentazione binaria di i) ovvero tutte le parole della riga i-esima, hanno la stessa sindrome;
- $|\xi_{i0}| = \min \{|\xi_{ij}|, j=1,\ldots 2^k-1\}$.

La prima riga è formata dalle parole di codice, tutte con sindrome 0, la prima colonna (detta dei ***leader***) viene costruita prendendo ogni volta una parola di peso minimo tra tutte quelle non ancora collocate. Il resto della riga viene ottenuto per somma bit a bit tra il *leader* e la parola del codice nella colonna corrispondente e infine le righe vengono ordinate in base alla sindrome.

Si noti che basta memorizzare la prima colonna della tabella di decodifica (detta colonna dei *leader*), se arriva $y = \xi_{ij}$ se ne calcola la sindrome e si decodifica y come

$$x = \xi_{0j} = \xi_{ij} + \xi_{i0}.$$

Esempio 5.1. La matrice

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

definisce un CCP di lunghezza 5 con 3 bit di parità e 2 bit di informazione. La matrice G corrispondente è:

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Le parole del codice sono {00000, 01110, 10011, 11101}; possono essere ottenute con le due righe di G più la loro somma e la parola tutta di 0.
La tabella di decodifica è la seguente:

sindrome	**leader**				
000	**00000**	01110	10011	11101	parole di codice
001	**00001**	01111	10010	11100	
010	**00010**	01100	10001	11111	
100	**00100**	01010	10111	11001	
110	**01000**	00110	11011	10101	
011	**10000**	11110	00011	01101	
101	**00101**	01011	10110	11000	
111	**01001**	00111	11010	10100	

Si noti che vengono corrette tutte e sole le configurazioni di errore che sono nella tabella dei *leader*. La probabilità di errore è quindi

$$1 - (p^5 + 5\,p^4(1-p) + 2\,p^3\,(1-p)^2) = 1 - 2p^3 - p^4 + 2p^5. \;\square$$

Teorema 5.6. Se z_i rappresenta un generico *leader* della tabella di decodifica di un CCP la probabilità di errore è

$$p_{err} = 1 - \sum_{i=1}^{2^t} p^{n-|z_i|}(1-p)^{|z_i|}$$

Dim. Poiché vengono corrette tutte e sole le configurazioni di errore che sono nella tabella dei *leader*, la probabilità di trasmissione corretta è data dalla probabilità di

queste configurazioni di errore. ❑

Si può anche dimostrare direttamente per i CCP il limite di Hamming. Infatti le classi laterali sono 2^t e le sequenze di al più e errori sono $v(e)$ (vedi Teorema 5.3), vale ovviamente

$$v(e) = \sum_{i=0}^{e} \binom{n}{i} \leq 2^t \qquad (5.2)$$

Osservazione. La limitazione di Hamming può essere scritta anche come limitazione inferiore al numero di bit di parità

$$\log\left[\sum_{i=0}^{e} \binom{n}{i}\right] \leq t;$$

oppure come limitazione superiore alla velocità di trasmissione:

$$R \leq 1 - \frac{1}{n}\log\left[\sum_{i=0}^{e} \binom{n}{i}\right]. \text{ ❑}$$

5.5 Capacità correttive dei CCP

Abbiamo visto che la distanza minima tra le parole di un codice determina una limitazione inferiore alla capacità correttiva. Per i CCP è facile trovare la distanza minima senza dover controllare tutte le coppie di parole di codice.

Teorema 5.7. La distanza minima tra le parole di un codice lineare è data dal minimo peso delle parole del codice esclusa la parola nulla o.
Dim. Siano x e y due parole di codice per cui si raggiunge la distanza minima $d=|x-y|$, per ogni parola w del codice vale $d \leq |w-o|$ e poiché la parola $z = x+y = x-y$ appartiene al codice vale $d = |z|$. ❑

Teorema 5.8. La distanza minima tra le parole di un codice lineare è la cardinalità del più piccolo insieme di colonne linearmente dipendenti della matrice A.
Dim. Le parole del codice, per cui vale la relazione $Ax=0$, possono essere interpretate come insiemi di colonne per cui esiste una combinazione lineare nulla. Infatti una parola di codice x con j uni può essere vista come una selezione di j colonne di A a somma nulla. Se il più piccolo degli insiemi di colonne linearmente dipendenti è formato da d colonne, non esiste una parola con meno di d uni. ❑

Teorema 5.9 Limite VGS (Varshamov-Gilbert-Sacks). È possibile costruire un CCP con parole di lunghezza n che corregga e errori con

$$t \le \left\lfloor 1 + \log \sum_{i=0}^{2e-1} \binom{n-1}{i} \right\rfloor \tag{5.3}$$

cifre di parità.

Dim. Si può costruire una matrice A di dimensioni $t \times n$ che definisce un codice con distanza minima d nel modo seguente. Si sceglie come prima colonna una t-pla non nulla, come seconda una colonna non nulla diversa dalla prima, e poi ancora al passo j una colonna linearmente indipendente rispetto a tutte le combinazioni lineari di al più $d-2$ colonne. Perché al passo n si abbia ancora almeno una possibilità di scelta è sufficiente che il numero totale di colonne non nulle 2^t-1 sia maggiore del numero delle combinazioni lineari distinte che a sua volta è al più

$$\sum_{i=1}^{d-2} \binom{n-1}{i}.$$

Deve quindi essere soddisfatta la relazione

$$2^t > 1 + \sum_{i=1}^{d-2} \binom{n-1}{i} = \sum_{i=0}^{d-2} \binom{n-1}{i} = \sum_{i=0}^{2e-1} \binom{n-1}{i} \tag{5.4}$$

in cui l'ultimo passaggio è stato ottenuto sostituendo d con $2e+1$.
Il più piccolo intero t per cui questa costruzione è sempre possibile è dato dalla (5.3). ❑

Combinando Il limite VGS con quello di Hamming si dimostra il seguente.

Corollario 5.1. Esistono sempre CCP per cui

$$\left\lceil \log \sum_{i=0}^{e} \binom{n}{i} \right\rceil \le t \le \left\lfloor 1 + \log \sum_{i=0}^{2e-1} \binom{n-1}{i} \right\rfloor$$

Ovvero

$$\frac{2^n}{1 + \sum_{i=0}^{2e-1} \binom{n-1}{i}} \le s = 2^{nR} = 2^k \le \frac{2^n}{\sum_{i=0}^{e} \binom{n}{i}}.$$

La seconda relazione si ricava direttamente dalla (5.4). ❑

Esempio 5.2. Il codice {0000000000, 0011111000, 0100001111, 0111110111, 1001010101, 1010101101, 1101011010, 1110100010} definito dalla matrice

$$A = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

ha n=10, k=3, t=7, e=2, soddisfa il limite di Hamming ed è migliore di quanto promesso dal limite VGS, infatti vale:

$$\left\lfloor \log\left[\sum_{i=0}^{2}\binom{10}{i}\right]\right\rfloor = 6 < t = 7 < 8 = \left\lfloor 1 + \log\left[\sum_{i=0}^{3}\binom{9}{i}\right]\right\rfloor \quad \square$$

Codici perfetti

Si dice **perfetto** un CCP che raggiunga esattamente il limite di Hamming. Si conoscono solo pochi codici perfetti:

- I codici con due sole parole e lunghezza $2m$+1 con le parole che differiscono in tutte le posizioni. Vale k=1, t=2m, e=m. La limitazione di Hamming si scrive

  $$2^{2m} \geq \sum_{i=0}^{m}\binom{2m+1}{i} \tag{5.5}$$

  Sfruttando la simmetria dei coefficienti binomiali e l'identità

  $$\sum_{i=0}^{2m+1}\binom{2m+1}{i} = 2^{2m+1}$$

  si verifica che nella (5.5) vale l'uguaglianza.

- I codici di Hamming con n=2^t-1, k=2^t-1-t, in cui la matrice A contiene tutti i possibili vettori di t bit tranne quello nullo. Vale e=1,

  $$\binom{2^t-1}{0} + \binom{2^t-1}{1} = 1 + \left(2^t - 1\right) = 2^t$$

- Due codici ciclici (vedi Cap. 6) con n=23, k=12, e=3, trovati da Golay.

5.6 Codici lineari non binari

Estendiamo l'alfabeto del codice ad campo finito GF[q]. I principali risultati visti per i CCP valgono ancora con alcune modifiche dovute al fatto che non si lavora più in GF[2].

Definizione 5.8. Un codice di gruppo è un insieme di parole di lunghezza n ad elementi in GF[q] che soddisfano al sistema lineare

$$Ax = 0$$

dove A è una matrice di rango t con t righe e n colonne, $t<n$ ad elementi in GF[q]**.** ❑

Se le colonne linearmente indipendenti sono le ultime t il codice è detto **sistematico**, e il sistema può essere messo nella forma

$$(P|I_t)\begin{pmatrix} r \\ z \end{pmatrix} = 0$$

da cui

$$z = -P\,r.$$

Ogni parola del codice si può quindi decomporre in due parti:

r formata da $k = n-t$ **cifre di informazione** scelte liberamente;

z formata da t **cifre di parità** che soddisfano la relazione $P\,r + z = 0$.

Il codice ha quindi q^k parole. Una caratterizzazione alternativa è attraverso la **matrice generatrice del codice** $G = (I_k \mid -P^{\mathrm{T}})$.

Per ogni parola r di k cifre si ha infatti che $x = r^{\mathrm{T}}G = (r^{\mathrm{T}} \mid -r^{\mathrm{T}}\,P^{\mathrm{T}})$ è parola di codice

Poiché

$$Ax = 0, Ay = 0 \quad \text{implica} \quad A(x+y) = 0$$

la somma di due parole del codice è ancora una parola di codice.

Definizione 5.6. Si definisce **peso di Hamming** di una sequenza x e si indica con $|x|$ il numero delle cifre diverse da 0. Si definisce **distanza di Hamming** tra due sequenze x e y il peso della sequenza x-y. ❑

Se la trasmissione è effettuata attraverso un canale uniforme simmetrico lo schema dell'osservatore ideale coincide ancora con quello della massima verosimiglianza e vale ancora il Teorema.

Teorema 5.11. Dato un codice lineare $X = \{x_1, x_2, \ldots, x_s\}$ lo schema di decodifica dell'osservatore ideale è il seguente:

se si riceve y si decodifica una delle x_i per cui

$$|x_i - y| = \min_j |x_j - y|. \; ❑$$

Esempio 5.3. La matrice su GF[3]

$$A = \begin{pmatrix} 2 & 1 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

definisce un codice sistematico. Si può ricavare la matrice:

$$G = \begin{pmatrix} 1 & 0 & 1 & 0 & 2 \\ 0 & 1 & 2 & 1 & 0 \end{pmatrix}$$

Il codice che si ottiene moltiplicando a sinistra G per tutti i 9 vettori di 2 cifre ternarie è: {00000, 01210, 02120, 10102, 11012, 12222, 20201, 21111, 22021}. ❑

5.7 Esercizi

Esercizio 5.1 Definire un codice a controllo di parità, sistematico, di lunghezza 15 che corregga 1 errore.

Esercizio 5.2. Definire un codice a controllo di parità, sistematico, di lunghezza 7 che corregga 1 errore.

Esercizio 5.3. Illustrare come può essere realizzato lo schema di decodifica della minima distanza per un codice a controllo di parità.

Esercizio 5.4. Trovare il codice a controllo di parità più piccolo che contiene le seguenti parole: 0010101, 0100111, 1001111.

Esercizio 5.5 Dimostrare che k parole di lunghezza n, linearmente indipendenti (tali cioè nessuna di esse è ottenibile come combinazione lineare delle altre) generano un codice a controllo di parità di 2^k parole.

Esercizio 5.6. Come si può ricavare il limite di Hamming per i codici a controllo di parità utilizzando le caratteristiche dei medesimi?

Esercizio 5.7. Cosa succede in un CCP se si permutano le righe di A, le colonne di A, le righe di G, le colonne di G?

6
Codici ciclici

6.1 Codici ciclici

Nel seguito prendiamo in considerazione una classe molto importante di codici lineari: i codici ciclici. L'alfabeto del codice deve essere un campo di Galois GF[q]. Nel caso più semplice q è un numero primo p e le parole del codici hanno elementi in GF[p]. Qualora invece q sia della forma p^m le parole del codice sono sequenze di elementi di GF[p^m]. In pratica, poiché ad ogni elemento di GF[p^m] può essere fatto corrispondere un vettore di m cifre di GF[p], i codici su GF[p^m] possono essere visti come insiemi di parole su GF[p]. È però importante ricordare che tutte le operazioni aritmetiche vanno effettuate in GF[p^m] che è un campo.

Caratterizzazione con i vettori

Definizione 6.1. Un **codice ciclico** su GF[q] è un codice lineare su GF[q] con l'ulteriore proprietà:

$a_{n-1}a_{n-2}\ldots a_0$ è parola di codice se e solo se $a_{n-2}a_{n-3}\ldots a_0a_{n-1}$ è parola di codice. ❑

Esempio 6.1. Si può facilmente verificare che il CCP {0000, 0101, 1010, 1111} è ciclico. ❑

Le forti proprietà algebriche dei codici ciclici risultano chiare solo con altre tecniche di rappresentazione dei suoi elementi.

Caratterizzazione con i polinomi

Rappresentiamo la parola $a_{n-1}a_{n-2}\ldots a_0$ con il polinomio

$$a(x) = a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\ldots+a_1x+ a_0 \qquad (6.1)$$

con coefficienti in GF[q].

L'operazione di scorrimento di un posto a sinistra è equivalente a moltiplicare il polinomio per x modulo x^n-1. Infatti ogni esponente viene aumentato di 1 tranne quello di x^{n-1} che diviene n (e poi 0 perché $x^n = 1$ modulo x^n-1). In altre parole se $a(x)$ è definito dalla (6.1) e rappresenta $a_{n-1}a_{n-2}\ldots a_0$, allora

$$x\ a(x) \text{ modulo } x^n\text{-}1 = a_{n-2}x^{n-1}+a_{n-3}x^{n-2}+\ldots+a_0x+ a_{n-1},$$

e rappresenta $a_{n-2}a_{n-3}\ \ldots a_0a_{n-1}$.

Nel seguito tutte le operazioni di moltiplicazioni tra polinomi sono da intendere modulo x^n -1, tutte le operazioni tra coefficienti sono da intendere in GF[q].
Se il polinomio $w(x)$ appartiene al codice allora:

- $x^s\ w(x)$, s>0, appartiene al codice per la proprietà di ciclicità;
- $b\ w(x)$, b∈GF[q] appartiene al codice per la proprietà di linearità;
- Se $a(x)$ è un qualunque polinomio, $a(x)\ w(x)$ appartiene al codice per le due considerazioni precedenti.

Teorema 6.1. Se $g(x)$ è il polinomio monico di grado minimo t di un codice ciclico C, allora tutti gli elementi di C possono essere rappresentati dalla espressione

$$w(x) = a(x)\ g(x) \qquad (6.2)$$

dove $a(x)$ è un polinomio di grado al più n-t-1. Il polinomio $g(x)$ è detto **polinomio generatore del codice**.
Dim. Per l'algoritmo di Euclide, dato $g(x)$ e una qualsiasi parola di codice $w(x)$ si può scrivere.

$$w(x) = a(x)\ g(x) + r(x)$$

con $r(x)$ di grado minore di $g(x)$. Ma poiché per quanto visto sopra a(x) $g(x)$ appartiene al codice allora $r(x)$ appartiene anch'esso al codice e per le proprietà di $g(x)$ può essere solo $r(x)$=0. ❑

Corollario 6.1. Se $g(x)$ ha grado t le parole del codice sono tante quanti i polinomi $a(x)$ di grado minore di k=n-t, ovvero q^k; le cifre di informazione sono k e le cifre di parità t. ❑

Teorema 6.2. Il polinomio generatore del codice 1) è unico; 2) divide x^n-1.
Dim. 1) se esistesse nel codice un altro polinomio monico $g'(x)$ del grado di $g(x)$

dovrebbe essere $g'(x) = a(x)\ g(x)$, l'unico $a(x)$ per cui ciò è possibile è la costante 1 e quindi i due polinomi coincidono.
2) per l'algoritmo di Euclide, vale:

$$x^n\text{-}1 = h(x)\ g(x) + r(x)$$

ovvero

$$h(x)\ g(x) = \text{-}r(x) \text{ modulo } x^n\text{-}1.$$

Ma poiché, anche in questo caso, $r(x)$ appartiene al codice può essere solo $r(x)=0$ e

$$x^n\text{-}1 = h(x)\ g(x). \qquad (6.3)$$

e il polinomio $h(x)$ è monico e di grado k. ❑

Corollario 6.2. 1) Per ogni GF[q] ed n i possibili codici ciclici sono esattamente tanti quanti i fattori di x^n-1 con coefficienti in GF[q] (contando anche i fattori non irriducibili). 2) Il polinomio $h(x)$ definito dalla (6.3) individua un codice ciclico detto il **duale** del codice definito da $g(x)$. ❑

Per le tabelle di fattorizzazioni di x^n-1 per q=2 e q=3 si veda l'Appendice A.

Osservazione. Il Teorema 6.2 ci permette di affermare che esiste una parola di codice x_g, (corrispondente a $g(x)$) che è sufficientemente "ricca" da generare tutto il codice, questo ci farà comodo quendo introdurremo la matrice G, generatrice del codice in analogia con quanto fatto per i CCP. ❑

Esempio 6.2. Tutti i possibili codici ciclici binari di 5 bit sono ricavabili nel modo seguente. La fattorizzazione di x^5-1 in polinomi irriducibili è

$$x^5\text{-}1 = (1+x)\ (1+x+x^2+x^3+x^4),$$

esistono 3 codici ciclici di 5 bit, derivati dalle seguenti scelte:
1 $g(x) = 1$
2 $g(x) = 1+x$
3 $g(x) = 1+x+x^2+x^3+x^4$
Il primo codice è formato da tutte le parole di 5 bit, ha quindi 0 bit di parità e 5 bit di informazione.
Il secondo codice ha k=4, t=1 e le parole sono:
{00000, 00011, 00101, 00110, 01001, 01010, 01100, 01111, 10001, 10010, 10100, 10111, 11000, 11011, 11101, 11110}
Il terzo codice ha due sole parole 00000 e 11111 (ovvero k=1, t=4). ❑

Caratterizzazione con le matrici

Per la (6.2) una generica parola di codice si può scrivere $w(x) = a(x)\ g(x)$. Questa relazione può essere vista come un prodotto scalare tra il vettore riga

$$(a_{k-1}, a_{k-2}, \ldots, a_0)$$

dei coefficienti di $a(x)$ e il vettore colonna di polinomi

$$\begin{pmatrix} x^{k-1}g(x) \\ x^{k-2}g(x) \\ \ldots \\ g(x) \end{pmatrix}$$

Sia $g(x) = g_t x^t + g_{t-1}x^{t-1} + \ldots + g_1 x + g_0$. Lo stesso prodotto può essere visto come il prodotto tra un vettore $(a_{k-1}, a_{k-2}, \ldots, a_0)$ di cifre di controllo e una matrice generatrice del codice G (di dimensioni $k \times n$, del tipo di quelle usate per i codici a controllo di parità) che ha la seguente forma di Toeplitz

$$G = \{G_{ij}\}, \quad i = 1,2,\ldots,k,\ j = 1,2,\ldots,n,$$

$$G_{ij} = \begin{cases} g_{t+i-j} & 0 \le j - i \le t \\ 0 & \text{altrimenti} \end{cases} \qquad (6.4)$$

ovvero (tenendo conto che $g(x)$ è monico e di grado t).

$$G = \begin{pmatrix} 1 & g_{t-1} & \ldots & & & 0 & \ldots & 0 & 0 \\ 0 & 1 & g_{t-1} & \ldots & & \ldots & 0 & \ldots & 0 \\ 0 & 0 & 1 & & \ldots & & \ldots & 0 & \ldots \\ & & \ldots & & & \ldots & & g_0 & 0 \\ 0 & 0 & 0 & \ldots & 1 & & \ldots & g_1 & g_0 \end{pmatrix}$$

Una matrice A tale che valga $Ax = 0$ per tutte e sole le parole del codice si può derivare in due modi diversi.

1) Scritta G come $(Q \mid B)$ con Q matrice quadrata $k \times k$, si nota che Q è non singolare. Allora

$$Q^{-1}\ G = (I_k \mid -P^{\mathrm{T}})$$

e

$$A' = (P \mid I_t).$$

2) Dalle relazioni (6.2) e (6.3) segue che, se $w(x)$ appartiene al codice, allora

$$h(x)\ w(x) = a(x)\ (x^n - 1) \qquad (6.5)$$

N.B. questo prodotto non è fatto modulo $x^n - 1$. Il primo membro della (6.5) è un

prodotto di polinomi, il suo termine di grado $s \geq k$ ha coefficiente:

$$h_s w_0 + h_{s-1} w_1 + \ldots + h_0 w_s = \sum_{j=s-k}^{s} h_{s-j} w_j = \sum_{j=0}^{k} h_j w_{s-j} \qquad (6.6)$$

(gli indici delle sommatorie tengono conto che $h(x)$ ha grado k).
Il polinomio $a(x)$ è di grado n-t-1=k-1, moltiplicandolo per x^n-1 si vede che i termini diversi da zero della (6.5) e della (6.6) sono quelli con esponente almeno n o inferiore a k; allora la (6.6) ha coefficiente zero in tutti i termini di grado compreso tra k e n-1 inclusi, ovvero:

$$\sum_{j=0}^{s} h_{s-j} w_j = 0, \quad s = k, k+1, \ldots, n-1$$

Questa relazione si può scrivere $A''w=0$ dove w è una parola di codice vista come vettore colonna ed A è la matrice di Toeplitz $t \times n$,:

$$A'' = \{A''_{ij}\}, \quad i = 1,2,\ldots,t, \ j = 1,2,\ldots,n,$$

$$A''_{ij} = \begin{cases} h_{j-i} & 0 \leq j-i \leq k \\ 0 & \text{altrimenti} \end{cases}$$

ovvero, tenendo conto che $h(x)$ è monico e di grado k:

$$A'' = \begin{pmatrix} h_0 & h_1 & \ldots & h_{k-1} & 1 & \ldots & 0 & \ldots \\ 0 & h_0 & h_1 & \ldots & & \ldots & \ldots & \ldots \\ 0 & 0 & h_0 & h_1 & \ldots & & 0 & 0 \\ \ldots & & \ldots & & & & 1 & 0 \\ 0 & 0 & 0 & \ldots & h_0 & \ldots & h_{k-1} & 1 \end{pmatrix}$$

Esempio 6.3. Per il secondo codice dell'Esempio 6.2, la stringa corrispondente a $g(x)$=1+x è 00011 e la stringa corrispondente a $h(x) = 1+x+x^2+x^3+x^4$ è 11111. Abbiamo quindi:

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \quad A' = A'' = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Per il terzo codice dell'Esempio 6.2, la stringa corrispondente a $g(x)$=1+x+x^2+x^3+x^4 è 11111, $h(x)$ vale 1+x e (tenendo conto che nella matrice A'' i coefficienti di $h(x)$ sono in ordine inverso) la stringa corrispondente vale 11000. Abbiamo quindi:

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \end{pmatrix}, \quad A' = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad A'' = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}. \; \square$$

Esempio 6.4. Vediamo alcuni codici ciclici su GF[4] di lunghezza 3. La fattorizzazione di x^3-1 in polinomi irriducibili su GF[4] è

$$x^3\text{-}1 = (x+1)\,(x+\alpha)\,(x+\alpha^2),$$

La scelta $g(x) = x+\alpha$ dà origine alle matrici

$$G = \begin{pmatrix} 1 & \alpha & 0 \\ 0 & 1 & \alpha \end{pmatrix}, \quad A = \begin{pmatrix} \alpha^2 & \alpha & 1 \end{pmatrix},$$

e le parole di codice sono {000, 01α, 0αα^2, 0α^{2}1, 1α0, 1α^{2}α, 10α^2, 111, αα^{2}0, ααα, α1α^2, α01, α^{2}10, α^{2}0α, α^{2}α^{2}α^2, α^{2}α1} ovvero {000000, 000110, …, 111001}. Questo codice ha k=2, t=1 ovvero 1 cifra di parità su GF[4] e 2 cifre di parità su GF[2]. Si noti che su GF[4] il codice è ciclico, su GF[2] è lineare ma non ciclico.

La scelta $g(x) = (x+1)\,(x+\alpha) = x^2+\alpha^2 x+ \alpha$ dà origine alle matrici

$$G = \begin{pmatrix} 1 & \alpha^2 & \alpha \end{pmatrix}, \quad A = \begin{pmatrix} \alpha^2 & 1 & 0 \\ 0 & \alpha^2 & 1 \end{pmatrix},$$

e le parole di codice sono {000, 1α^{2}α, α1α^2, α^{2}α1} ovvero {000000, 011110, 100111, 111001}. Questo codice ha k=1, t=2 ovvero 2 cifre di parità su GF[4] e 4 cifre di parità su GF[2]. Si noti che su GF[4] il codice è ciclico, su GF[2] è lineare ma non ciclico. ❑

Caratterizzazione con le radici

Richiamiamo alcuni fatti noti. Il Teorema Fondamentale dell'Algebra afferma che ogni polinomio di grado d a coefficienti complessi ha d radici in $\mathbb{C}$. A noi questo teorema interessa in una forma diversa (si veda anche l'Appendice A).

Teorema 6.3. Sia p un numero primo, ogni polinomio irriducibile $f(x)$ di grado d a coefficienti in GF[p^m] ha esattamente d radici in GF[p^{dm}]. ❑

Dato un polinomio $g(x)$ su GF[p^m] che definisce un codice ciclico su GF[p^m], sappiamo che $g(x)$ è divisore di x^n-1. In particolare $g(x)$ è il prodotto di alcuni dei fattori irriducibili su GF[p^m] di x^n-1. Sia $g(x) = f_1(x)\, f_2(x)\, \ldots\, f_r(x)$, e sia d_i il grado di f_i. Abbiamo $d_1+d_2+\ldots+d_r = t$ e $f_i(x)$ ha esattamente d_i radici in $GF[\,p^{md_i}\,]$ che verranno indicate con $\alpha_j^{(i)}$ j=1,2,…,d_i.

Le radici dei fattori di $g(x)$ hanno le seguenti proprietà.

- L'insieme di tutte le radici (contate con la loro molteplicità) individua univocamente $g(x)$ e quindi il codice.
- Se $w(x)$ appartiene al codice, allora $w\left(\alpha_j^{(i)}\right) = 0$. Infatti ogni $\alpha_j^{(i)}$ è radice dei fattori di $g(x)$ e quindi di $g(x)$ e quindi di $w(x)$.
- Poiché $x^n\text{-}1 = h(x)\, g(x)$ qualunque radice $\alpha_j^{(i)}$ di $g(x)$ lo è anche di $x^n\text{-}1$, quindi $\alpha^n=1$ e la lunghezza del codice è multiplo dell'ordine di ciascuna radice.

Nel seguito ci limitiamo al caso in cui tutti gli $f_i(x)$ hanno radici distinte e con molteplicità 1. In questo caso per definire il codice basta una radice per ognuno dei fattori irriducibili di $g(x)$. Infatti le relazioni $w(\alpha)=0$ scritte per una radice per ogni $f_i(x)$ danno origine ad una equazione matriciale con tante righe quanti sono i fattori di $g(x)$ e tante colonne quanto è la lunghezza del codice. Infatti, sia

$$w(x) = w_{n-1}x^{n-1}+w_{n-2}x^{n-2}+\ldots+w_0,$$

allora $w(\alpha)$ si può scrivere come il prodotto del vettore riga

$$(\alpha^{n-1}, \alpha^{n-2}, \ldots,1)$$

per il vettore colonna

$$\begin{pmatrix} w_{n-1} \\ w_{n-2} \\ \ldots \\ w_0 \end{pmatrix}$$

L'insieme delle relazioni $w(\alpha_i) = 0$, $i=1,2,\ldots r$, prendendo una sola radice per ogni $f_i(x)$, può essere visto come un prodotto matrice-vettore $Aw=0$, dove A è una matrice di dimensioni $r\times n$ con la seguente forma di Vandermonde

$$A = \begin{pmatrix} \alpha_1^{n-1} & \alpha_1^{n-2} & \ldots & \alpha_1 & 1 \\ \alpha_2^{n-1} & \alpha_2^{n-2} & \ldots & \alpha_2 & 1 \\ \ldots & & \ldots & & \\ \alpha_r^{n-1} & \alpha_r^{n-2} & \ldots & \alpha_r & 1 \end{pmatrix}. \text{❑}$$

Passando dalle radici ai vettori si ottiene una nuova forma della matrice A. Infatti gli elementi di A sono elementi di $GF[\,p^{m d_i}\,]$ e come tali hanno una rappresentazione come matrici di md_i righe. Sostituendo ad ogni potenza di una radice la matrice che la rappresenta si ottiene una matrice A' su GF[p] analoga a quelle definite precedentemente per i CCP. Inoltre, ogni radice contribuisce esattamente con md_i righe e il numero totale

di righe di A' è uguale a mt (il numero delle cifre di parità del codice su GF[p]) come appunto deve essere.

Esempio 6.5. Consideriamo codici di 7 bit su GF[2]. I fattori irriducibili di x^7-1 sono $1+x$, $1+x+x^3$ e $1+x^2+x^3$. La scelta $g(x) = (1+x)\,(1+x+x^3)$ genera il codice {0000000, 0011101, 0100111, 0111010, 1001110, 1010011, 1101001, 1110100}. L'unica radice di $1+x$ è $\beta_1=1$ rappresentabile in GF[2]; le radici di $1+x+x^3$ sono α, α^2, α^4 in GF[8]. Scegliamo come β_2 la prima delle tre che ha rappresentazione vettoriale 010. La matrice A è la seguente

$$A = \begin{pmatrix} \beta_1^6 & \beta_1^5 & \beta_1^4 & \beta_1^3 & \beta_1^2 & \beta_1 & 1 \\ \beta_2^6 & \beta_2^5 & \beta_2^4 & \beta_2^3 & \beta_2^2 & \beta_2 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \alpha^6 & \alpha^5 & \alpha^4 & \alpha^3 & \alpha^2 & \alpha & 1 \end{pmatrix}$$

che in forma vettoriale diviene.

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$ ❑

Definizione di un codice a partire dalle radici

Sia α un elemento di ordine n di un campo GF[p^t]. Assegnare α equivale a scegliere il polinomio minimo di α (di grado t) che fattorizza x^n-1, ovvero un codice lungo n con $k=n-t$ cifre di informazione e t cifre di controllo su GF[p]. La scelta di α può dare origine a codici diversi a seconda di quale sia il polinomio minimo di α.

Esempio 6.6. Consideriamo GF[2^3]=GF[8] rappresentato modulo $1+x+x^3$ come nell'Appendice. Un elemento primitivo β ha un polinomio minimo di grado 3 che è un fattore di x^7-1. Assegnare β equivale a dare un codice di 7 bit con 3 cifre di parità e 4 cifre di informazione. Le diverse scelte di β portano ad uno dei due codici seguenti.
1) $\beta = \alpha$ oppure $\beta = \alpha^2$ oppure $\beta = \alpha^4$; il polinomio minimo è $1+x+x^3$; il polinomio generatore è $g(x) = 1+x+x^3$; la matrice G in GF[2] è

;

la matrice A ha una forma diversa a seconda della scelta di β, in GF[8] vale

$$A = \begin{pmatrix} \beta^6 & \beta^5 & \beta^4 & \beta^3 & \beta^2 & \beta & 1 \end{pmatrix} = \begin{cases} \begin{pmatrix} \alpha^6 & \alpha^5 & \alpha^4 & \alpha^3 & \alpha^2 & \alpha & 1 \end{pmatrix}, & \text{se } \beta = \alpha \\ \begin{pmatrix} \alpha^5 & \alpha^3 & \alpha & \alpha^6 & \alpha^4 & \alpha^2 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^2 \\ \begin{pmatrix} \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^4 \end{cases}$$

mentre in GF[2] vale

$$A' = \begin{cases} \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, & \text{se } \beta = \alpha \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^2 \\ \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^4 \end{cases}$$

Le parole del codice sono {0000000, 0001011, 0010110, 0011101, 0100111, 0101100, 0110001, 0111010, 1000101, 1001110, 1010011, 1011000, 1100010, 1101001, 1110100, 1111111}.

2) $\beta = \alpha^3$ oppure $\beta = \alpha^5$ oppure $\beta = \alpha^6$; il polinomio minimo è $1+x^2+x^3$; il polinomio generatore è $g(x) = 1+x^2+x^3$; la matrice G in GF[2] è

$$G = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix},$$

la matrice A ha una forma diversa a seconda della scelta di β, in GF[8] vale

$$A = \begin{pmatrix} \beta^6 & \beta^5 & \beta^4 & \beta^3 & \beta^2 & \beta & 1 \end{pmatrix} = \begin{cases} \begin{pmatrix} \alpha^4 & \alpha & \alpha^5 & \alpha^2 & \alpha^6 & \alpha^3 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^3 \\ \begin{pmatrix} \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^5 \\ \begin{pmatrix} \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^6 \end{cases}$$

mentre in GF[2] vale

$$A' = \begin{cases} \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^3 \\ \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^5 \\ \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}, & \text{se } \beta = \alpha^6 \end{cases}$$

Le parole del codice sono {0000000, 0001101, 0010111, 0011010, 0100011, 0101110, 0110100, 0111001, 1000110, 1001011, 1010001, 1011100, 1100101, 1101000, 1110010, 1111111}. ❑

Esempio 6.7. I codici di Golay. Consideriamo GF[2^{11}]=GF[2048]. Sia α un qualunque elemento primitivo del campo e sia $\beta = \alpha^{89}$; poiché vale $23\times89=2047 = 2^{11}-1$. allora $\beta^{23} = \alpha^{2047} = 1$. Il polinomio minimo di β ha grado 11 ed è un fattore di $x^{23}-1$. Assegnare β equivale a dare un codice di 23 bit con 11 cifre di parità e 12 cifre di informazione. Diverse scelte di α portano a diverse scelte di β ma i codici risultanti sono solo due che corrispondono ai due polinomi $1+x+x^5+x^6+x^7+x^9+x^{11}$ e $1+x^2+x^4+x^5+x^6+x^{10}+x^{11}$, entrambi fattori di $x^{23}-1$. La distanza minima è 7 e i codici di Golay sono perfetti perche raggiungono il limite di Hamming:

$$\sum_{i=0}^{3}\binom{23}{i} = 1 + 23 + 253 + 1771 = 2048\,.\ \text{❑}$$

Più in generale siano $\alpha_1, \alpha_2, \ldots, \alpha_s$ elementi primitivi distinti di ordine $m_1, m_2, \ldots, m_s$ di campi $GF[p^{d_i}]$, i=1,2,…,s. Assegnare $\alpha_1, \alpha_2, \ldots, \alpha_s$ equivale a scegliere come $g(x)$ il mcm dei loro polinomi minimi che fattorizza x^n-1 con n=mcm($m_1, m_2, \ldots, m_s$).

Esempio 6.8. Consideriamo ancora GF[8], assegnare 1 ed α equivale a scegliere come $g(x)$ il mcm dei polinomi minimi $1+x$ e $1+x+x^3$ ovvero $g(x) = (1+x)\,(1+x+x^3)$. L'ordine della radice 1 è 1, l'ordine di α è 7 e quindi il codice ha n=mcm(1,7)=7 cifre su GF[2] con 4 cifre di parità e 3 cifre di informazione. Lo stesso codice è già stato presentato nell'Esempio 6.5. ❑

6.3 Codici BCH e di Reed-Solomon

Limite BCH

Teorema 6.4. (Bose-Chauduri-Hocquenghem). Dato un codice ciclico su GF[p^m] generato da $g(x)$, se $g(x)$ ha d-1 radici della forma $\alpha^r, \alpha^{r+1}, \ldots, \alpha^{r+d-2}$, allora la distanza minima tra le parole del codice è almeno d.

Dim. La matrice A che definisce il codice può essere messa nella forma

$$A = \begin{pmatrix} \alpha_1^{n-1} & \alpha_1^{n-2} & \ldots & \alpha_1 & 1 \\ \alpha_2^{n-1} & \alpha_2^{n-2} & \ldots & \alpha_2 & 1 \\ \ldots & & \ldots & & \\ \alpha_q^{n-1} & \alpha_q^{n-2} & \ldots & \alpha_q & 1 \end{pmatrix}$$

Scegliendo le d-1 righe corrispondenti a $\alpha^r, \alpha^{r+1}, \ldots, \alpha^{r+d-2}$ e d-1 colonne qualsiasi, si ha una sottomatrice quadrata della forma

$$\begin{pmatrix} (\alpha^r)^{j_1} & (\alpha^r)^{j_2} & \ldots & (\alpha^r)^{j_{d-1}} \\ (\alpha^{r+1})^{j_1} & (\alpha^{r+1})^{j_2} & \ldots & (\alpha^{r+1})^{j_{d-1}} \\ \ldots & & \ldots & \\ (\alpha^{r+d-2})^{j_1} & (\alpha^{r+d-2})^{j_2} & \ldots & (\alpha^{r+d-2})^{j_{d-1}} \end{pmatrix}$$

Denotiamo α^{j_i} con β_i per i=1,2,…,d-1 allora la matrice quadrata si può scrivere

$$\begin{pmatrix} \beta_1^r & \beta_2^r & \ldots & \beta_{d-1}^r \\ \beta_1^{r+1} & \beta_2^{r+1} & \ldots & \beta_{d-1}^{r+1} \\ \ldots & & \ldots & \\ \beta_1^{r+d-2} & \beta_2^{r+d-2} & \ldots & \beta_{d-1}^{r+d-2} \end{pmatrix} =$$

$$= \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \beta_1 & \beta_2 & \ldots & \beta_{d-1} \\ \ldots & & \ldots & \\ \beta_1^{d-2} & \beta_2^{d-2} & \ldots & \beta_{d-1}^{d-2} \end{pmatrix} \begin{pmatrix} \beta_1^r & 0 & \ldots & 0 \\ 0 & \beta_2^r & \ldots & 0 \\ \ldots & & \ldots & \\ 0 & 0 & \ldots & \beta_{d-1}^r \end{pmatrix}$$

che è non degenere in quanto prodotto di una matrice di Vandermonde per una matrice diagonale. La prima è non degenere perché tutti i β_i sono diversi tra loro, la seconda perché tutti i β_i sono diversi da 0. In tal modo per qualunque scelta di d-1 colonne di A esiste un minore (d-1)×(d-1) non degenere e quindi le d-1 colonne sono linearmente indipendenti. Quindi, per il Teorema 5.8, il codice ha distanza minima almeno d. Questa proprietà si conserva anche se la matrice viene rappresentata su GF[p] sostituendo alle

potenze di α i vettori colonna corrispondenti alla rappresentazione di α. ❑

Esempio 6.9 Consideriamo i codici di Golay introdotti nell'esempio 6.5. Per il Corollario dell'Appendice il polinomio generatore $g(x)$ (che è il polinomio minimo di β) ha radici β, β^2, β^4, β^8, β^{16}, β^{32}, β^{64}, β^{128}, β^{256}, β^{512}, β^{1024}. Poiché vale $\beta^{23} = 1$ le radici si possono scrivere β, β^2, β^4, β^8, β^{16}, β^9, β^{18}, β^{13}, β^3, β^6, β^{12} quindi le radici che sono potenze consecutive sono β, β^2, β^3, β^4 e per il teorema BCH la distanza minima è almeno 5, in realtà la distanza minima è 7. ❑

I codici BCH

La limitazione BCH può essere usata in modo costruttivo. Sia GF[q] un campo di Galois. Sia $m>1$, $r>0$, α elemento di GF[q^m] di ordine n, $2\le d\le n$ allora gli elementi α^r, α^{r+1}, …,α^{r+d-2} definiscono un codice ciclico. Sia $f_i(x)$ il polinomio minimo di α^i, $i=0,1,\ldots,d-2$
si ha

$$g(x)=\text{mcm}\{f_r(x), f_{r+1}(x),\ldots,f_{r+d-2}(x)\}$$

Poiché $(\alpha^{r+i})^n=1$, $i=0,1,\ldots,d-2$, ciascuno degli α^{r+i} è radice di x^n-1 e ciascuno dei $f_i(x)$ è fattore di x^n-1 ed ha grado al più m. La lunghezza del codice è quindi n e la distanza minima d.

Vediamo ora un importante caso particolare.
Sia $q=2$, $r=1$, α primitivo, $d=2e+1$. Poiché α^i e α^{2i} hanno lo stesso polinomio minimo vale

$$g(x)=\text{mcm}\{f_1(x), f_3(x),\ldots,f_{2e-1}(x)\}$$

Si ottiene un codice di lunghezza $n=2^m-1$ con al più $m\,e$ cifre di parità che corregge e errori.

Esempio 6.10. Scegliamo $q=2$, $m=3$, $d=7$ e rappresentiamo GF[8] modulo $1+x+x^3$ come nell'Appendice. Poiché α è un elemento primitivo vale $\alpha^7=1$ e $n=7$. Le radici scelte sono α, α^2,…, α^6, che hanno come polinomi minimi $1+x+x^3$ oppure $1+x^2+x^3$. Vale allora

$$g(x) = (1+x+x^3)\,(1+x^2+x^3) = 1+x+x^2+x^3+x^4+x^5+x^6.$$

Le radici α, α^2, α^4 hanno lo stesso polinomio minimo e così pure α^3 e α^6, per costruire A basta usare α e α^3:

$$A = \begin{pmatrix} \alpha^6 & \alpha^5 & \ldots & \alpha & 1 \\ \alpha^{18} & \alpha^{15} & \ldots & \alpha^3 & 1 \end{pmatrix} = \begin{pmatrix} \alpha^6 & \alpha^5 & \alpha^4 & \alpha^3 & \alpha^2 & \alpha & 1 \\ \alpha^4 & \alpha & \alpha^5 & \alpha^2 & \alpha^6 & \alpha^3 & 1 \end{pmatrix} \quad \text{-❑}$$

Esempio 6.11. Scegliamo q=2, m=4, d=5 e rappresentiamo GF[16] modulo $1+x+x^4$ come nell'Appendice. Poiché α è un elemento primitivo vale $\alpha^{15}=1$ e n=15. Le radici scelte per definire il codice sono α, α^2, α^3, α^4, che hanno come polinomi minimi, rispettivamente $1+x+x^4$, $1+x+x^4$, $1+x+x^2+x^3+x^4$, $1+x+x^4$. Per costruire la matrice A basta allora usare α ed α^3 che hanno polinomi minimi distinti,vale:

$$g(x) = (1+x+x^4)\,(1+x+x^2+x^3+x^4),$$

$$A = \begin{pmatrix} \alpha^{14} & \alpha^{13} & \dots & \alpha & 1 \\ \alpha^{42} & \alpha^{39} & \dots & \alpha^3 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} \alpha^{14} & \alpha^{13} & \alpha^{12} & \alpha^{11} & \alpha^{10} & \alpha^9 & \alpha^8 & \alpha^7 & \alpha^6 & \alpha^5 & \alpha^4 & \alpha^3 & \alpha^2 & \alpha & 1 \\ \alpha^{12} & \alpha^9 & \alpha^6 & \alpha^3 & 1 & \alpha^{12} & \alpha^9 & \alpha^6 & \alpha^3 & 1 & \alpha^{12} & \alpha^9 & \alpha^6 & \alpha^3 & 1 \end{pmatrix}$$

Codici di Reed-Solomon

È possibile trovare una famiglia particolare di codici BCH detti di **Reed-Solomon**. Sia $q=p^s$, e α un elemento primitivo di GF[p^s]. Tutte le potenze di α sono radici di $x^{p^s-1}-1$ che ammette la semplice fattorizzazione

$$x^{p^s-1}-1 = \prod_{i=0}^{p^s-2}\left(x-\alpha^i\right).$$

Il polinomio generatore

$$g(x) = (x-\alpha)(x-\alpha^2)\dots(x-\alpha^{d-1})$$

definisce un codice BCH con r=1, m=1, $n = p^s-1$, $t=d-1$.
Si noti che l'alfabeto del codice è formato dagli elementi di GF[p^s] che sono rappresentabili come potenze di α, polinomi oppure stringhe di s elementi di GF[p]. In altre parole un codice di Reed-Solomon con lunghezza p^s-1, d-1 cifre di controllo e distanza minima d può essere visto come un codice su GF[p] di lunghezza $s(p^s-1)$ e $s(d-1)$ cifre di controllo. Sia $d= 2e+1$ questo codice può correggere gruppi di se cifre di GF[p] consecutive (i cosiddetti errori a raffica).

Esempio 6.8. Sia p=2, s=3, d=5. Si ha n=7, t=4, k=3. Vale

$$g(x) = (x-\alpha)(x-\alpha^2)(x-\alpha^3)(x-\alpha^4) = x^4+\alpha^3x^3+x^2+\alpha x+\alpha^3.$$

La seconda uguaglianza si ricava sviluppando il prodotto e semplificando i coefficienti delle potenze della x secondo le regole delle operazioni su GF[8]. La matrice G che genera il codice è la seguente:

$$G = \begin{pmatrix} 1 & \alpha^3 & 1 & \alpha & \alpha^3 & 0 & 0 \\ 0 & 1 & \alpha^3 & 1 & \alpha & \alpha^3 & 0 \\ 0 & 0 & 1 & \alpha^3 & 1 & \alpha & \alpha^3 \end{pmatrix}$$

Le parole del codice sono $8^3 = 512$ si ottengono moltiplicando $g(x)$ per tutti i polinomi di grado al più 2 con coefficienti in GF[8], ovvero moltiplicando G per i vettori di 3 elementi di GF[8], (quindi abbiamo 512 parole). In binario le parole sono lunghe 21, un esempio di parola del codice è 1 α^3 1 α α^3 0 0 ovvero 001 011 001 010 011 000 000. Siccome il codice corregge due errori una "raffica" di 4 bit errati viene sempre corretta, se la "raffica" occupa due gruppi contigui si possono correggere fino a 6 bit errati consecutivi.

Si noti che le limitazioni per i codici a controllo di parità con parole lunghe 21 (vedi Corollario 5.1) afferma che il numero t di cifre di parità per correggere 4 errori deve essere superiore a 13 e può essere inferiore a 18. Ne segue che le parole non possono essere più di 256. Il codice di Reed-Solomon visto nell'esempio non è un codice ciclico su GF[2] e ottiene un migliore risultato limitandosi a correggere le raffiche di 4 bit. ❑

6.4 Esercizi

Esercizio 6.1. Dimostrare che in un codice ciclico (definito come un codice lineare a cui appartengono tutte le permutazioni cicliche di ogni parola del codice) esiste sempre un polinomio generatore.

Esercizio 6.2. Dire quali codici ciclici derivano dalla fattorizzazione di x^4-1 (dando ad es. il polinomio generatore di ognuno di essi) e per almeno uno trovarne tutte le parole.

Esercizio 6.3. Quanti codici ciclici di lunghezza 3 esistono? Quanti codici ciclici di lunghezza 6 esistono?

Esercizio 6.4. Sulla base delle vostre conoscenze, quanti codici binari perfetti di lunghezza 3 esistono? Tra questi ce ne sono di ciclici?

Esercizio 6.5. Dare le linee generali della dimostrazione del limite BCH per i codici ciclici.

Esercizio 6.6. Data la fattorizzazione $x^4-1 = (x+1)^4$ su GF[2] definire un codice ciclico di lunghezza 4 su {0,1}.

Esercizio 6.7. Trovare il codice ciclico più piccolo su {0,1} che contiene le parole 01010101 e 00110011.

Appendice: I campi di Galois.

A.1 Esempi di campi di Galois

Le seguenti tabelle mostrano la struttura di alcuni campi della forma GF[n], $n=p^k$. Per ognuno di essi viene dato il polinomio irriducibile f(α) (su GF[p]) che genera il campo, e una tabella delle potenze dell'elemento generatore α.

- La prima colonna contiene
- $e = \alpha^i$, i=0,1,…,n-2.
- Nella seconda colonna $p(\alpha)$ è la rappresentazione di e come polinomio:
- $p(\alpha) = e \bmod f(\alpha)$.
- Nella terza colonna è riportata la stringa s dei valori dei coefficienti di $p(\alpha)$.
- La quarta colonna contiene il polinomio minimo di e ovvero il polinomio $\pi(x)$ di grado minimo per cui $\pi(e)=0$.
- La quinta colonna contiene l'ordine di e ovvero il minimo j per cui $e^j=1$.

Si noti che i polinomi minimi sono i fattori irriducibili di $x^{n-1}-1$ (vedi Tabelle A1 e A.2), che ognuno di essi ha tante radici quanto è il suo grado, e che vale la relazione

$$x^{n-1} - 1 = \prod_{i=0}^{n-2} \left(x - \alpha^i \right)$$

In altre parole gli α^i, i=0,1,…,n-2. sono tutte e sole le radici di $x^{n-1}-1$ su GF[n].

GF[4]

p = 2, k = 2, GF[4]
polinomio irriducibile $f(\alpha)=1+\alpha+\alpha^2$

Tabella delle potenze di α

e	p(α)	s	π(x)	j
1	1	01	$1+x$	1
α	α	10	$1+x+x^2$	3
α^2	$1+\alpha$	11	$1+x+x^2$	3

GF[8]

p = 2, k = 3, GF[8]
polinomio irriducibile $f(\alpha)=1+\alpha+\alpha^3$

Tabella delle potenze di α

e	p(α)	s	π(x)	j
1	1	001	$1+x$	1
α	α	010	$1+x+x^3$	7
α^2	α^2	100	$1+x+x^3$	7
α^3	$1+\alpha$	011	$1+x^2+x^3$	7
α^4	$\alpha+\alpha^2$	110	$1+x+x^3$	7
α^5	$1+\alpha+\alpha^2$	111	$1+x^2+x^3$	7
α^6	$1+\alpha^2$	101	$1+x^2+x^3$	7

GF[9]

p = 3, k = 2, GF[9]
polinomio irriducibile $f(\alpha)=2+\alpha+\alpha^2$

Tabella delle potenze di α

e	p(α)	s	π(x)	j
1	1	01	$2+x$	1
α	α	10	$2+x+x^2$	8
α^2	$1+2\alpha$	21	$1+x^2$	4
α^3	$2+2\alpha$	22	$2+x+x^2$	8
α^4	2	02	$1+x$	2
α^5	2α	20	$2+2x+x^2$	8
α^6	$2+\alpha$	12	$1+x^2$	4
α^7	$1+\alpha$	11	$2+2x+x^2$	8

GF[16]

p = 2, k = 4, GF[16]
polinomio irriducibile $f(\alpha)=1+\alpha+\alpha^4$

Tabella delle potenze di α

e	p(α)	s	π(x)	j
1	1	0001	$1+x$	1
α	α	0010	$1+x+x^4$	15
α^2	α^2	0100	$1+x+x^4$	15
α^3	α^3	1000	$1+x+x^2+x^3+x^4$	5
α^4	$1+\alpha$	0011	$1+x+x^4$	15
α^5	$\alpha+\alpha^2$	0110	$1+x+x^2$	3
α^6	$\alpha^2+\alpha^3$	1100	$1+x+x^2+x^3+x^4$	5
α^7	$1+\alpha+\alpha^3$	1011	$1+x^3+x^4$	15
α^8	$1+\alpha^2$	0101	$1+x+x^4$	15
α^9	$\alpha+\alpha^3$	1010	$1+x+x^2+x^3+x^4$	5
α^{10}	$1+\alpha+\alpha^2$	0111	$1+x+x^2$	3
α^{11}	$\alpha+\alpha^2+\alpha^3$	1110	$1+x^3+x^4$	15
α^{12}	$1+\alpha+\alpha^2+\alpha^3$	1111	$1+x+x^2+x^3+x^4$	5
α^{13}	$1+\alpha^2+\alpha^3$	1101	$1+x^3+x^4$	15
α^{14}	$1+\alpha^3$	1001	$1+x^3+x^4$	15

A.2 Fattori irriducibili di x^n-1 modulo 2

$$x^2-1 = (1+x)^2$$

$$x^3-1 = (1+x)(1+x+x^2)$$

$$x^4-1 = (1+x)^4$$

$$x^5-1 = (1+x)(1+x+x^2+x^3+x^4)$$

$$x^6-1 = (1+x)^2(1+x+x^2)^2$$

$$x^7-1 = (1+x)(1+x+x^3)(1+x^2+x^3)$$

$$x^8-1 = (1+x)^8$$

$$x^9-1 = (1+x)(1+x+x^2)(1+x^3+x^6)$$

$$x^{10}-1 = (1+x)^2(1+x+x^2+x^3+x^4)^2$$

$$x^{11}-1 = (1+x)(1+x+x^2+x^3+x^4+x^5+x^6+x^7+x^8+x^9+x^{10})$$

$$x^{12}-1 = (1+x)^4(1+x+x^2)^4$$

$$x^{13}-1 = (1+x)(1+x+x^2+x^3+x^4+x^5+x^6+x^7+x^8+x^9+x^{10}+x^{11}+x^{12})$$

$$x^{14}-1 = (1+x)^2(1+x+x^3)^2(1+x^2+x^3)^2$$

$$x^{15}-1 = (1+x)(1+x+x^2)(1+x+x^4)(1+x^3+x^4)(1+x+x^2+x^3+x^4)$$

$$x^{16}-1 = (1+x)^{16}$$

$$x^{17}-1 = (1+x)(1+x^3+x^4+x^5+x^8)(1+x+x^2+x^4+x^6+x^7+x^8)$$

$$x^{18}-1 = (1+x)^2(1+x+x^2)^2(1+x^3+x^6)^2$$

Tabella A.1. Fattorizzazioni modulo 2 di x^n-1.

$x^2-1 = (1+x)(2+x)$

$x^3-1 = (2+x)^3$

$x^4-1 = (1+x)(2+x)(1+x^2)$

$x^5-1 = (2+x)(1+x+x^2+x^3+x^4)$

$x^6-1 = (1+x)^3(2+x)^3$

$x^7-1 = (2+x)(1+x+x^2+x^3+x^4+x^5+x^6)$

$x^8-1 = (1+x)(2+x)(1+x^2)(2+x+x^2)(2+2x+x^2)$

$x^9-1 = (2+x)^9$

$x^{10}-1 = (1+x)(2+x)(1+x+x^2+x^3+x^4)(1+2x+x^2+2x^3+x^4)$

Tabella A.2. Fattorizzazioni modulo 3 di x^n-1.

Indice