

Raffiche di errori

Abbiamo fin qui considerato il caso di errori in trasmissione aventi ciascuno probabilità indipendenti l'uno dallo altro.

Consideriamo ora anche il caso in cui gli errori si presentino concentrati in intervalli temporali particolari, ovvero il caso in cui si verificano **raffiche di errori** (burst errors).

Vi sono molti casi in cui questo fenomeno si presenta tipicamente. Ad esempio, citiamo i canali temporali costituiti dai supporti di memoria magnetici e ottici.

Per definire la dimensione delle raffiche considerate, dobbiamo definire il numero massimo g di simboli corretti consecutivi ammissibili tra un simbolo errato e il successivo simbolo errato in una raffica di errori. Questo parametro, che viene detto **spazio di guardia**, viene ottenuto osservando per un intervallo di tempo molto esteso e analizzando statistica-

mente le raffiche di errori che si presentano su un canale di trasmissione.

Ad esempio, consideriamo la seguente sequenza di simboli ottenuta al ricevitore:

oggλnλn λλvrelle λiλvere

dove con il simbolo λ abbiamo indicato i simboli errati. Se valesse $g=1$ avremmo 4 raffiche

(λλnλ, λλ, λλ, λiλ)

di lunghezza 4, 2, 2, 3 rispettivamente; se valesse $g=2$ avremmo 2 raffiche

(λλnλn λλ, λλe λiλ)

di lunghezza 8 e 7 rispettivamente; se valesse $g \geq 3$ avremmo 1 sola raffica

(λλnλn λλvrelle λiλ)

di lunghezza 17.

E' ovvio che i codici fin qui visti sarebbero poco efficienti nei casi di errori a raffica perchè, o richiederebbero una ridondanza eccessiva (poiché sfruttata solo negli specifici intervalli temporali di comparsa delle raffiche di errori), o sarebbero insufficientemente affidabili al presentarsi delle raffiche.

La soluzione del problema può essere affidata ad uno dei o ad entrambi i seguenti due approcci:

- a) definizione di *codici specializzati* per la correzione di errori a raffica;
- b) realizzazione di tecniche trasmissive che consentano la *scomposizione delle raffiche di errori in errori singoli* (casuali).

Nel prosieguo del corso vedremo due codici specializzati per la correzione di errori a raffica, i codici di Fire e i codici di Reed-Solomon; inoltre, anche i codici convoluzionali possono essere utilizzati a tal fine sotto opportune condizioni.

Il secondo approccio può essere realizzato ad esempio interfogliando (**interleaving**) le parole di codifica sul canale di trasmissione, ovvero inviando le parole codificate in modo tale da essere protette solo per le raffiche di lunghezza b/t (intero), dove t è il parametro che rappresenta la **profondità** dell'interleaving.

Per proteggere da raffiche di errori di lunghezza b si raggruppano le parole di codifica in blocchi di t parole e si inviano prima tutti i simboli di testa delle t parole, poi i simboli in seconda posizione, etc. In questo modo l'eventuale raffica di b errori si distribuisce su t parole protette per correggere al più b/t errori.

Ad esempio, supponiamo di trasmettere la seguente sequenza di simboli (indicando lo spazio con "-"):

la-trasm
issione-
con-erro
ri-a-raf
fica-vie
ne-migli
orata-co
n-l'inte
rleaving

inviando le colonne invece delle righe cioè:

licrfnonrasoiier-l-sn-c-ale ...

Se si verificasse una raffica di lunghezza ≤ 9 come la seguente (=8):

licrfnonxxxxxxxx-l-sn-c-ale ...

avremmo al ricevitore:

lx-trasm
ixsione-
cxn-erro
rx-a-raf
fxca-vie
nx-migli
oxata-co
n-l'inte
xleaving

in cui per ogni riga c'è al più un errore, per cui ci basterebbe un codice capace di correggere un errore per riga.

Questa tecnica è tipicamente usata, per esempio, nella codifica dell'informazione sui supporti ottici.

In generale possiamo dire che dato un generico codice (n,k) in grado di correggeri raffiche di b errori, il suo interleaving di profondità t genera un

codice (nt, kt) in grado di correggere raffiche di errori di lunghezza bt .

Da questo approccio sono derivati gli **array codes** (cfr. Farrell, P.G., A SURVEY OF ARRAY ERROR CONTROL CODES, Atti Advanced School on "Geometrics, Codes and Cryptography", Udine, giugno 1989).

Inoltre, le tecniche di interleaving consentono di dimostrare semplicemente che la *capacità di correzione di raffiche di errori è indipendente dalla distanza di Hamming del codice*. In pratica, consentono una migliore distribuzione temporale della ridondanza garantendo una miglior efficienza per i codici considerati.

Codifica dell'informazione nei dischi ottici

I codici binari derivati da codici di Reed-Solomon hanno trovato ampio uso negli ultimi vent'anni per la codifica dell'informazione nei canali temporali (memorie di massa) di tipo ottico e magneto-ottico.

Nelle memorie di massa ottiche viene utilizzato un metodo di codifica chiamato Cross Interleaved Reed-Solomon Code (CIRC); esso è basato su un codice di Reed-Solomon avente distanza minima 5 codificato a sua volta da un secondo codice di Reed-Solomon anch'esso con distanza minima 5.

Nel caso del CD-DA, gli errori che non sono corretti dal CIRC vengono generati per interpolazione lineare.

Nel caso del CD-ROM (dove l'interpolazione non ha significato), viene aggiunto un codice ciclico di controllo (CRC) al CIRC; il risultato viene a sua volta codificato da un doppio codice di Reed-Solomon (cioè un codice codificato da un secondo codice, entrambi aventi distanza minima 3), del tipo usato per le memorie di massa magnetiche. In totale, l'informazione viene codificata con cinque codici combinati tra loro in cascata.

Per dischi come CD-I, WORM e magneto-ottici sono usati analoghi schemi di codifica, ma diversi in ogni singolo caso in dipendenza dalle specifiche caratteristiche di occorrenza degli errori nei diversi canali temporali.

Ricordiamo a titolo esemplificativo, che, in generale, la codifica della informazione nei dischi ottici viene effettuata codificando blocchi di 8 bit di informazione in parole di codifica di 14 bit concatenate tra loro (**EFM**, Eight-to-Fourteen Modulation) mediante parole aggiuntive di 3 bit (**merging** bits) ciascuna. I criteri adottati sono i seguenti:

a) il passaggio da 8 bit a 14 bit è basato su una tabella di corrispondenza tra le 256 possibili parole di 8 bit e 256 parole delle 267 parole di 14 bit che soddisfano la proprietà di avere sequenze di zeri lunghe almeno 2 e al massimo 10 tra una cifra uno e la successiva; 14 bit è appunto la minima lunghezza della parola di codifica che soddisfa tale proprietà; allora, le parole

00100101001000

01000000000000

non sono accettabili mentre le seguenti
lo sono

01001000100000

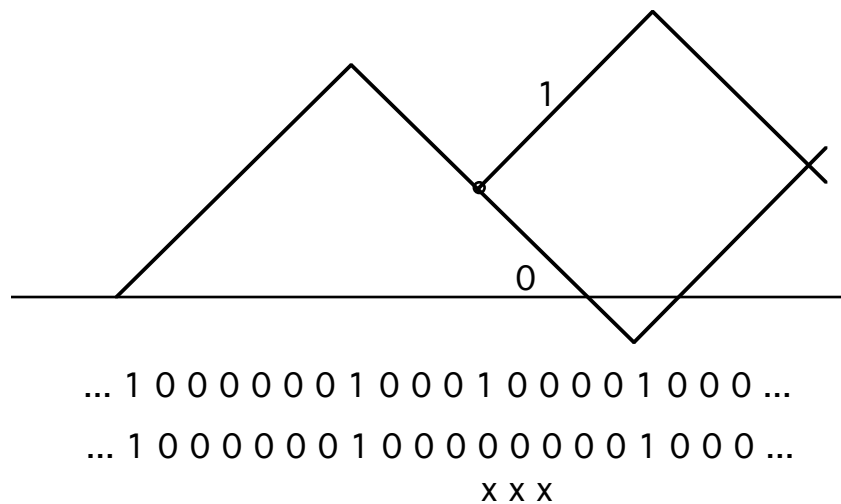
10000100000000

I tre bit aggiuntivi tra una parola di codifica e la successiva servono per evitare di non rispettare la proprietà su indicata nell'area della giunzione.

Le triple possibili sono allora solo le seguenti

000 001 010 100

e tra queste viene scelta di volta in volta quella che minimizza la **DSV** (Digital Sum Value) ovvero, interpretando la lunghezza delle sequenze di zeri (run-length) alternativamente come valori positivi e negativi, la tripla che minimizza l'area del grafico corrispondente (cfr. figura, dove le triple possibili sono 100, 010, 001, 000).



La ridondanza per la correzione degli errori viene aggiunta a queste parole di 14+3 bit nella misura di un terzo rispetto all'informazione codificata ovvero 1 byte di ridondanza ogni 3 byte di informazione.

Infine, ai **frame**, ovvero ai blocchi di codice (24 parole di informazione + 8 parole di ridondanza, ognuna nel formato 14+3 bit + altri codici di servizio per l'unità di controllo del supporto fisico, per un totale di 588 bit per frame), viene applicata la tecnica di interleaving vista precedentemente, in modo tale da distribuire eventuali raffiche di errori (dovute ad imperfezioni e/o

danneggiamenti del supporto fisico) in
differenti parole di codifica.