**COMUNICAZIONI**

***MODELLO*** ***DI*** ***RAPPRESENTAZIONE*** ***DI*** ***UN*** ***PROCESSO*** ***DI***

***COMUNICAZIONE***

**La comunicazione è il procedimento per trasmettere e ricevere messaggi.**

Per gli esseri umani la comunicazione, cioè lo scambio di informazioni, è essenziale. La comunicazione avviene mediante linguaggio verbale, i gesti, la parola scritta, le immagini, la musica.

La nostra società è detta società dell’informazione perché molte attività si basano sulla trasmissione o sulla comunicazione di informazione.

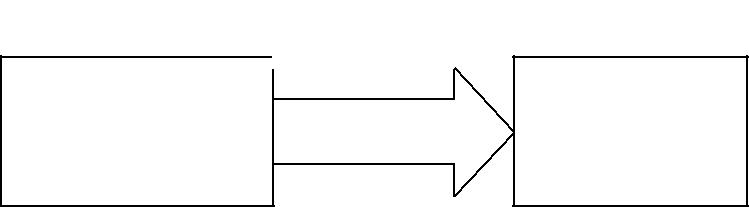
In questo contesto il computer assume un ruolo ed un’importanza sempre crescenti.

Di seguito è riportata una sintesi di concetti, la cui trattazione per esteso è nei libri di testo.

**Mezzi di Comunicazione**

* La trasmissione richiede sempre la presenza di **due interlocutori**: la sorgente o mittente, ossia la fonte del messaggio, e il destinatario che riceve il messaggio.
* Per avere **comunicazione** la trasmissione deve essere **bidirezionale**, cioè a due vie (il mittente ed il destinatario sono ruoli interscambiabili).
* La comunicazione tra gli interlocutori deve avvenire secondo regole accettate da entrambe le parti (**protocollo**).
* La comunicazione avviene in accordo a **tecniche di trasmissione** che dipendono dal mezzo usato (cavo, fibra ottica, onde e.m.)
* Il **messaggio** da comunicare deve essere **codificato** dal mittente e **decodificato** dal destinatario.

Nel modello di Jakobson (1966) i fattori comunicativi sono: mittente, destinatario, messaggio, contesto (argomento trattato), canale (mezzo trasmissivo), codice.



|  |  |  |
| --- | --- | --- |
|  | Messaggio |  |
| Sorgente | Canale | Destinatario |

**Nota:** I mezzi di comunicazione di massa (mass media) sono unidirezionali: sono mezzi ditrasmissione (**broadcast**) non di comunicazione !

**Nota:** I protocolli garantiscono che la comunicazione **non sia ambigua**, in modo che tutti ipartecipanti capiscano il significato dei messaggi trasmessi. Anche nella vita di tutti i giorni usiamo protocolli di comunicazione definiti (ad esempio saluterai una persona conosciuta dicendo ‘ciao’ o ‘buon giorno’ e poi le stringerai la mano o le dara i un bacio). Questi protocolli delle relazioni sociali cambiano in società o culture diverse (ad e s. in Giappone ci si saluta con un inchino…).

La comunicazione può essere sincrona o asincrona.

**Comunicazione Sincrona**: La sorgente e il destinatario sono contemporaneamente collegati.La trasmissione e la ricezione del messaggio avvengono quasi simultaneamente (es. telefono, citofono, TV, radio, chat)

**Comunicazione Asincrona**: La sorgente e il destinatario non sono contemporaneamente collegati.La trasmissione e la ricezione del messaggio non sono contemporanei (es. posta tradizionale, e-mail, fax).

La comunicazione inoltre può essere:

* **unicast** (il destinatario è uno solo)
* **multicast** (i destinatari sono più di uno)
* **broadcast** (i destinatari sono tutti).

Nella comunicazione è opportuno distinguere i ruoli di destinatari e di riceventi: i destinatari sono le persone a cui il messaggio è destinato, mentre i riceventi sono le persone che ricevono il messaggio. Per esempio, quando l'insegnante in classe chiama un alunno per l'interrogazione tutta la classe è ricevente, mentre solo l'interrogato è anche destinatario; quando si spedisce una e-mail il destinatario è quello posto nella casella ‘A’; i riceventi sono tutti quelli posti nella casella ‘Copy to’.

Un'altra utile distinzione stabilisce la differenza tra **messaggio e informazione**:

* per **messaggio** si intende il fatto, la notizia, quindi qualcosa di **oggettivo**
* per **informazione** si intende il significato che attribuiamo alla notizia, quindi qualcosa di **soggettivo**.

Per esempio, se non si deve prendere l'aereo, il fatto che i piloti facciano sciopero non ha alcun contenuto informativo, è solamente una notizia, viceversa se si dovesse prendere un aereo allora la notizia avrebbe un contenuto informativo.

Quando si deve comunicare qualcosa, occorre decidere come rappresentare il messaggio, in forma testuale oppure in forma grafica. Nel caso di un testo è importante decidere quale lingua usare perché se, per esempio, si deve scrivere una lettera ad un amico inglese, il messaggio deve essere codificato in lingua inglese. Si deve quindi scegliere un codice: il processo che codifica il messaggio si chiama **codificazione di sorgente (CS)**. Dalla parte del destinatario ci deve essere di conseguenza un **decodificatore di sorgente (DS)** che esegue la funzione inversa.

Un canale di trasmissione è generalmente **disturbato** da rumori (Noise), che possono essere interferenze, spike (impulsi) o distorsione. In conseguenza di questi disturbi il messaggio potrebbe essere alterato e arriverebbe a destinazione con errori e quindi non sarebbe più comprensibile al destinatario.

Per mettere il **destinatario** in condizione di **correggere eventuali errori di trasmissione**, o quantomeno di **riconoscerli** per richiedere la ritrasmissione, il messaggio viene codificato in modo ridondante (sovrabbondante), cioè vengono attivati i controlli aggiuntivi per segnalare al destinatario la correttezza o meno del messaggio ricevuto. Si introduce allora un'ulteriore codifica, detta **codifica di canale (CC) dalla parte del sorgente e decodifica di canale (DC) da parte del** **destinatario**.

Il messaggio, per poter viaggiare sul mezzo trasmissivo, deve essere trasformato da simbolico in **fisico (cioè in segnale**), in pratica in una delle seguenti forme:

* segnali elettrici
* onde elettromagnetiche
* segnali ottici

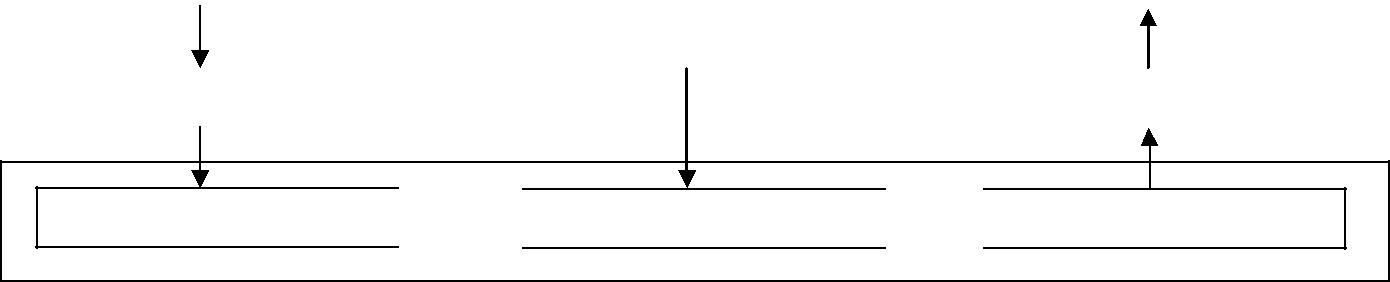
anche in dipendenza del mezzo trasmissivo utilizzato, rispettivamente: cavo elettrico, etere (aria), fibra ottica.

Il processo di trasformazione da messaggio simbolico in segnale alla partenza (lato trasmittente) si chiama **codificazione di linea (CL)**; dalla parte ricevente l'operazione inversa si chiama **decodifica** **di linea (DL).**

Dopo aver trasformato il messaggio in segnale, esso viene preso in carico da un oggetto, detto **trasmettitore (TX),** che materialmente lo manda sul canale; analogamente dalla parte deldestinatario ci deve essere un oggetto che svolge la funzione inversa, detto **ricevitore (RX)**.

Riassumendo, qualsiasi processo di comunicazione può essere rappresentato secondo il modello della figura seguente, detto **modello di Shannon**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Sorgente di informazione |  |  |  | Destinatario |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | Codifica di sorgente |  |  |  | Decodifica di sorgente |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | Cifratura |  |  |  | Decifrazione |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | Codifica di canale |  |  |  | Decodifica di canale |
|  |  |  |  |  |  |
|  |  |  | Distorsione (rumore) |  |  |
|  |  |  |  |  |  |
|  | Codifica di linea |  |  |  | Decodifica di linea |
|  |  |  |  |  |  |



Modulazione  Canale continuo  Demodulazione

La **sorgente** è l’origine del segnale analogico o numerico da trasmettere.

Il **destinatario** è il sistema finale che memorizza e utilizza l’informazione ricevuta.

Il **canale di comunicazione**, tra sorgente e destinatario, ha caratteristiche che condizionano i messaggi trasmessi.

**Segnali analogici e numerici**

Gli aspetti delle telecomunicazioni brevemente accennati sono immediatamente applicabili a **segnali** di natura **analogica**, in cui il segnale è definito per tutti gli istanti di tempo e assume valoriqualsiasi.

Nel caso invece in cui il segnale è definito solo per istanti di tempo discreti e valori discreti, si entri nell'ambito delle **trasmissioni numeriche.**

**CODIFICA Dl SORGENTE.** Si tratta del sottosistema che effettua la codifica logica da uninsieme di simboli a un altro secondo le tecniche più opportune.

**CIFRATURA.** Si tratta di un sottosistema che altera i codici in modo che il significato non siadecifrabile senza l'uso di una opportuna sequenza binaria detta chiave e di uno specifico algoritmo.

**CODIFICA Dl LINEA.** Assegna a ogni simbolo logico una particolare forma fisica, in genere informa di segnale elettrico o luminoso. È important e perché un’opportuna scelta della forma del segnale consente di distribuirne le energie sulle frequenze più opportune per le caratteristiche del canale.

**MODULAZIONE.** Non è sempre presente. È il sottosistema che, tram ite il segnale della sorgente,cambia alcuni parametri di un altro segnale (detto **portante**), le cui caratteristiche in termini di

3

spettro di frequenza si adattano meglio alle caratteristiche del canale (attenuazione e sfasatura in funzione delle frequenze).

**CANALE Dl TRASMISSIONE.** Il canale, nella realtà fisica, è costituito da un mezzo trasmissivosu cui si propaga un segnale di natura elettrica o ottica, in genere elettromagnetica nei casi di nostro interesse.

**DEMODULAZIONE.** Effettua l'operazione inversa della modulazione, ricostruendo in base alladecodifica dei parametri della portante il segnale in banda base.

**DECODIFICA DI CANALE.** Cambia la forma d'onda dei simboli elementari da quella adatta alcanale a quella utilizzata dal sistema.

**DECIFRATORE.** Utilizzando la sequenza chiave e l'algoritmo di cifratura, riporta in formacomprensibile (in chiaro) il messaggio.

**DECODIFICA Dl SORGENTE**. Ripristina la codifica logica originale della sorgente.

**Rumore (Noise)**

In un sistema di comunicazione reale, insieme al segnale utile è presente anche un segnale con **andamento del tutto casuale** che prende il nome di rumore o noise. Il rumore è un segnalecompletamente privo di informazione e influisce notevolmente, soprattutto sulla ricezione, con conseguente **perdita di qualità dell'informazione** . Alcuni esempi avvertibili di rumore possono essere il ronzio negli alimentatori, il fruscio degli altoparlanti, o più banalmente il brusio nell'aula quando si parla attraverso il canale “aria". Si dis tingue essenzialmente in:

* **rumore interno**, generato dagli apparati elettronici impiegati nella trasmissione; è semprepresente ed è molto difficile da eliminare, in quanto dipende dalla materia stessa e dalle cariche elettriche che danno origine alla corrente;
* **rumore esterno**, quali per esempio il **rumore atmosferico** (prodotto da onde elettromagnetichee da scariche naturali come i fulmini) e il **rumore cosmico**, causato dai raggi cosmici o da emissioni di onde elettromagnetiche ad alta frequenza provenienti da varie sorgenti artificiali, come l’accensione di motori, interferenze tra circuiti ecc.

***LA CODIFICA DELLE INFORMAZIONI NEI CALCOLATORI***

La condizione necessaria affinché ci sia informazione è l'esistenza di un processo di comunicazione; il **messaggio va quindi rappresentato fisicamente, modificando le grandezze fisiche con** **opportune regole**.

Questo problema viene risolto mediante l'adozione di opportuni **codici**, che stabiliscono una **corrispondenza** tra l'**informazione** significativa per il problema (l'applicazione) preso inconsiderazione **e una serie di simboli** manipolabili dal sistema di comunicazione. Nell'esempio della macchina di Von Neumann, i simboli di base sono cifre binarie, quindi ogni informazione deve essere codificata sotto tale forma per poter essere trattata dal sistema.

***I codici***

Un codice è costituito da un insieme di **simboli o segni (alfabeto),** di fatto associati a variazioni di grandezze fisiche, e dalle **combinazioni di simboli validi (parole).**

Sono esempi di alfabeto le 10 cifre arabe, oppure le 21 lettere dell’alfabeto italiano, oppure l’insieme dei tre colori rosso, verde e giallo.

Sono esempi di codici un insieme di numeri (per esempio i numeri naturali nel sistema di numerazione decimale, i numeri naturali nel sistema di numerazione binario, i numeri interi nel sistema di numerazione decimale ecc.) o il vocabolario di una lingua.

La ‘ **Teoria dei Codici**" è la **teoria matematica** che affronta i problemi essenziali della comunicazione.

E’ possibile dare una definizione astratta del concetto di codice, senza far riferimento a nessun esempio specifico.

Sia **X** un qualsiasi **insieme** (finito o infinito), che rappresenta gli **oggetti o le informazioni** che vogliamo trattare.

Sia **A** un qualsiasi **alfabeto finito di simboli**.

Con **A\*** indichiamo **l'insieme delle possibili sequenze** (ordinate da sinistra a destra) di simboli dell'alfabeto A, di lunghezza finita e infinita compresa la sequenza vuota (che non contiene nessun simbolo).

Definiamo una funzione iniettiva di codifica cod: X A\*

ed una funzione di decodifica decod: A\* X

tali che per ogni y appartenente a X decod(cod(y)) = y

Nell’utilizzo del computer tutti i codici usano come alfabeto le cifre del sistema binario 0 e l. **L’uso** **di due soli simboli riduce la possibilità di errore** (è più facile distinguere tra due simboli che tradieci simboli) e rende semplice la rappresentazione all'interno del computer, infatti due simboli possono essere rappresentati con due stati fisici distinti.

**L'insieme X, l'alfabeto A e le funzioni cod e decod formano un codice per la**

**rappresentazione di elementi di X.** La sequenza di simboli cod(y), corrispondente a unqualsiasi elemento y appartenente a X, verrà chiama ta "codifica di y" (ovvero "rappresentazione di y", secondo il codice considerato).

**Normalmente, non tutte le sequenze di simboli b appartenenti ad A\* saranno rappresentazioni di elementi appartenenti a X**. Solitamente, anzi, in Informatica avremo a chefare con **codici a lunghezza fissa**, nei quali è predeterminato il numero di simboli che costituiscono una codifica corretta, e tutte le altre sequenze (che ovviamente darebbero decod(b) = errore) non vengono neanche prese in considerazione

**Segnali**

Si definisce segnale **l'emissione nel tempo di messaggi sotto forma di simboli del codice**.

Per esempio, parlando con una persona le comunichiamo informazioni attraverso le parole, il tono della voce, i gesti, gli sguardi, che sono tutte variazioni codificate di grandezze fisiche nel tempo.

Tutti i segnali che lanciamo vengono recepiti e interpretati dal nostro interlocutore in base alle convenzioni acquisite attraverso codici impliciti creati con l'esperienza, l'educazione o altro.

Per esempio, i colori rosso, giallo e verde e la convenzione per cui i tre colori significano, rispettivamente, "non si può passare", "sta per div entare rosso", "si può passare" costituiscono il codice dei semafori.

***Definizione di codifica***

Proviamo a dare una definizione del termine codificare.

Ci occupiamo di computer e siamo abituati ad usarli per fare calcoli, per scrivere, per disegnare o ritoccare fotografie, per giocare, per vedere filmati, per sentire musica, per viaggiare in Internet, ecc. Un **elaboratore elettronico** dunque tratta per noi **informazioni di diverso tipo**, ad esempio acustiche, visive numeriche, ecc. Ma la macchina che lavora è sempre la stessa ed usa lo stesso tipo di tecnologia per gestire entità diverse. Cosa hann o in comune i numeri, le immagini, i suoni ed i testi quando sono rappresentati in un computer? La risposta è: **sono comunque un insieme di bit**, l'elaboratore in ultima analisi lavora solo con i bit. Quindi deve esistere un metodo che a partire da un'immagine crea una serie di bit ed è in grado dala serie di bit di riprodurla a schermo o di stamparla. La stessa cosa deve valere per i suoni, per i film in DVD... Partiamo da un'informazione per noi significativa e la trasformiamo in qualcos’altro che è in diretta corrispondenza con l'originale. I vincoli della tecnologia che utilizziamo stabiliscono le tecniche di rappresentazione dell'informazione.

C'è anche un'altra serie di considerazioni che interessano nell'operazione di codifica. Un conto è rappresentare del testo in chiaro da spedire con la posta elettronica e ben altro conto è il trasmettere dell'informazione segreta cifrata con qualche strano meccanismo. Nel secondo caso si deve fare in modo che il testo codificato sia leggibile solo possedendo un'opportuna chiave di lettura.

Se avete qualche esperienza di Internet vi sarà cap itato di scaricare **file «zippati»**. Pensate alla differenza di grandezza tra il file originale non compresso e l'equivalente zippato. Qualcuno ha inventato una tecnica di rappresentazione dell'informazione grazie alla quale la **stessa cosa è** **rappresentata con meno cifre binarie**. Se avete a che fare con file sonori .mp3, .wav o .mid, lostesso brano può essere riprodotto a partire da tre rappresentazioni diverse con varie qualità ed ampiezze di memoria richieste. Se usate un qualche programma di grafica avete sicuramente notato degli strani codici RGB o CMYK per gestire il colore; sono modi diversi: uno pensato per la visualizzazione con il monitor l'altro per la stampa. Quindi in generale l'operazione di codifica è definita in funzione di esigenze di ottimizzazione di una qualche prestazione piuttosto che da esigenze di riservatezza o di sicurezza.

Riassumendo:

**Il processo di codifica trasforma l'informazione al suo ingresso in modo da rappresentarla in maniera adatta alla tecnologia usata, tenendo conto di vincoli di riservatezza o sicurezza oltre a considerazioni di efficienza rispetto ad opportuni criteri di valutazione.**

Per analogia pensate anche a delle **codifiche** non tratte direttamente dal mondo dell'informatica: il codice che utilizza **bandiere** per la segnalazione in mare, il codice **Morse** che rappresenta l'alfabeto con sequenze d'impulsi di durata più o meno lunga, il **linguaggio a segni dei muti**, **l'alfabeto** **muto**, i **segnali di fumo** degli indiani, le bandiere durante le gare di formula 1, il **codice a barre** per registrare le informazioni sui prodotti venduti e quindi produrre lo scontrino. Sono tutti casi in cui si descrive qualcosa usando una forma di rappresentazione diversa.

La **ricostruzione dell'informazione originaria si chiama operazione di decodifica**. Idealmente la doppia trasformazione di codifica con la successiva decodifica dovrebbe permettere di ricostruire con fedeltà l'informazione originaria. Non è detto che ciò avvenga. Si distinguono **tecniche lossy** e **lossless**. Una tecnica di codifica è lossy quando c'è perdita di informazione tra rappresentazione

originaria, quella codificata e dunque quella ricostruita. Viceversa si parla di tecniche lossless se la versione codificata e quindi la sua ricostruzione è una replica fedele dell'originale. In caso di codifiche lossy è necessario che la perdita sia trascurabile o comunque accettabile.

Le operazioni di **codifica dei dati** sono, in un sistema informatico, **operazioni di input**, mentre la **decodifica è un’operazione di output**. Ovviamente l'informazione d'uscita è una ricostruzione diquella d'ingresso nel caso in cui non ci sia stata una qualche elaborazione dell'informazione stessa.

**Un'informazione in rappresentazione interna può sub ire delle modifiche**. Tali modifichedevono avere stretta corrispondenza con modifiche che possono in via teorica essere fatte sull'informazione originale. Se è possibile colorare un disegno, la stessa operazione deve essere possibile anche sulla sua versione in codice ed il risultato in uscita deve essere l'analogo di ciò che avviene in realtà. È il primo embrionale esempio di realtà virtuale.

In generale oltre alla fedeltà rappresentativa si r ichiede alle tecniche di codifica di permettere di conservare lo stesso tipo di **manipolabilità dell'informazione** che si ha nel mondo reale almeno per le applicazioni in cui la uso. Una rappresentazione numerica che non permette di fare somme con gli stessi risultati che si ottengono con carta e penna non può servire per far calcoli numerici. Potrebbe permettermi di gestire i numeri delle carte di credito risultando magari in ciò più efficient e di un'altra.

Possiamo, estendendo questo esempio, introdurre il **concetto di efficacia** di una rappresentazione dell'informazione. Pensiamo ai **codici cifrati**; è un caso in cui il codice segreto dovrà essere difficilmente scopribile, se non addirittura inviolabile. In questo caso l'efficacia è cruciale. Ma anche **nel caso di codici rivelatori o correttori di errore**, che in seguito vedremo, l'efficacia sarà un parametro fondamentale.

Le considerazioni sulle codifiche si applicano in generale anche ai sistemi di comunicazioni digitali (telefonia fissa, telefonia mobile, Tv satellitari sono solo alcuni esempi). In generale si parla di **sistema telematico** quando **si accoppia l'elaborazione dei dati con la trasmissione a distanza** (ilBancomat, Internet, la teleprenotazione, il collegamento tra una filiale e la sede centrale di un'azienda, il telelavoro ecc..)

***Codifica e decodifica***

**Per poter utilizzare un codice deve essere definito un processo di codifica.**

Il processo di codifica in una lingua naturale per esempio è l'associazione di ogni parola ad un oggetto.

Spesso le parole di un codice vengono definite proprio in modo che sia possibile codificare gli elementi di un insieme; per esempio le parole di una lingua sono state definite appositamente per denominare gli oggetti.

Si può applicare un processo di codifica per **passare da un codice ad un altro**, mettendo in corrispondenza le parole di un codice con quelle di un altro.

Se il **numero di simboli è uguale basta associare a ogni simbolo del primo codice un simbolo** **del secondo**

Se il numero di simboli è diverso un metodo semplice consiste **nell'associare a ogni simbolo del** **primo codice una parola del secondo codice**; per esempio a ogni cifra decimale può essereassociata una combinazione di 4 cifre binarie (codice BCD) o a ogni carattere alfanumerico può essere associata una combinazione di 7 cifre binarie (codice ASCII).

Il processo di codifica permette di rappresentare delle informazioni mediante un certo codice.

**Per ogni processo di codifica** è definito il processo inverso, o **decodifica,** che permette di estrarrele informazioni da un messaggio codificato.

Questo implica che **durante il processo di decodifica** deve essere possibile **distinguere una** **dall’altra le parole del codice, indipendentemente dalla sequenza con cui sono usate**; questacondizione è semplice da rispettare se le parole del codice hanno tutte la stessa lunghezza, cioè lo stesso numero di simboli (e se naturalmente non si usa una stessa parola per rappresentare due elementi diversi).

Se le parole hanno lunghezza diversa, **soddisfare la condizione che ogni parola sia sempre** **identificabile può non essere così semplice** .

Definendo A = 0, B = 1, C = 01, D = 10, la sequenza 010 potrebbe essere interpretate come ABA o AD o CA.

Per poter **valutare i codici bisogna fare riferimento:**

* **alla quantità di informazione associata all’insieme degli elementi che devono essere codificati**

**e**

* **alla lunghezza del codice usato per rappresentarli.**

La **misura dell'informazione** viene definita a **partire da un processo di comunicazione tra una** **sorgente che produce un messaggio da trasmettere, scelto in un insieme di possibili messaggi, e un destinatario che riceve il messaggio attraverso un canale di comunicazione**. La sorgentecodifica i messaggi da trasmettere mediante un opportuno codice e quindi i possibili messaggi della sorgente sono gli elementi che devono essere codificati.

***Costruzione di un codice***

Come già detto, per codice si intende un insieme X di oggetti chiamati **parole** che sono associati, secondo un criterio, agli elementi di un insieme Y da codificare.

Per costruire un codice bisogna prima **definire l’alfabeto** , cioè **l'insieme dei simboli** con i quali si costruiscono le parole.

A partire dai simboli si costruiscono le parole. La **parola è l'oggetto formato da una sequenza di** **uno o più simboli appartenenti all'alfabeto**.

Per esempio i numeri decimali sono le parole del codice che ha come alfabeto le cifre N = {0, 1, 2, 3, ..., 9}

Le parole della lingua italiana sono sequenze di simboli rappresentati dai caratteri appartenenti all'alfabeto formato dall'insieme {a, b, ...z; A, B, ...Z}.

**Per costruire un codice occorre definire il numero dei simboli dell'alfabeto e la lunghezza delle parole.**

Dato un codice avente un **alfabeto di S simboli** e **parole lunghe n simboli**, si possono costruire **Sn** parole diverse tra loro.

Per esempio con un codice binario e con parole formate da 5 bit si hanno 25=32 configurazioni (parole) diverse; con un codice ternario e con parole formate da 4 cifre si ottengono 34=81 parole diverse.

Si definisce **lunghezza minima di parola** il numero minimo di simboli che devono avere tutte le parole per **codificare in modo non ambiguo** gli elementi dell'insieme.

Un **codice è ambiguo** quando **una parola** del codice corrisponde **a due o più** **elementi** dell'insieme da codificare.

Il significato di ambiguità è analogo a quello della lingua parlata, nel senso che una parola può fare intendere più cose diverse.

Nel caso in cui le parole del codice fossero in numero inferiore al numero degli elementi dell'insieme da codificare saremmo costretti a codificare (cioè ad associare) almeno due elementi diversi dell'insieme alla stessa parola del codice. Questo implica il fatto che quando si riceve una parola non si è in grado di stabilire a quale degli elementi si riferisce.

(Esempi nella lingua italiana:

tasso: di cambio, bancario, di sconto, animale, albero, scrittore…. collo: bagaglio, parte del corpo umano, della giacca…)

Per definire la **lunghezza minima m** occorre conoscere il numero di elementi dell'insieme Y da codificare: per esempio, se l'insieme da codificare è formato da 25 elementi e si utilizza un codice binario, la lunghezza minima è il più piccolo numero per cui vale 2m>=25, cioè m=5; per il codice ternario si avrebbe 3m>=25, cioè m=3.

In generale

**m = lgkN**

dove **N rappresenta il numero di elementi dell'insieme da codificare e k indica la base** (numero dei simboli dell'alfabeto).

Dopo aver dato la definizione di lunghezza minima di parola, si possono **classificare** **i codici** in questo modo:

* **Ambigui**
* **Efficienti**
* **Ridondanti.**

Un codice si dice ambiguo quando la lunghezza delle sue parole è inferiore alla lunghezza minima. Un codice si dice efficiente quando la lunghezza delle sue parole è uguale alla lunghezza minima.

Un codice si dice ridondante quando la lunghezza delle sue parole è maggiore della lunghezza minima.

Nei paragrafi seguenti vengono trattati i codici più utilizzati nella trasmissione e nell'elaborazione delle informazioni, di tipo alfabetico e numerico. I codici in questo tipo di applicazioni sono di tipo binario.

in cui numeri di valore diverso richiedono un

***Il problema della codifica ottimale***

Per semplificare, consideriamo la codifica degli M simboli emessi da una sorgente S con simboli binari. Se la distribuzione è equiprobabile, il contenuto informativo del messaggio è I = log2M. Tutti gli M simboli hanno una codifica su 1 bit, ma in genere le sorgenti non sono equiprobabili ed in questo caso I è un reale ma, poiché il supportotecnologico offre solo multipli interi di bit, dobbiamo impegnare un numero intero di bit con un qualche spreco.

Codificare ha implicazioni sulle prestazioni del sistema di comunicazione per due motivi:

* in ultima analisi, il processo di codifica è un algoritmo che va eseguito da un esecutore; ciò impegna risorse fisiche e di tempo;
* se non è la più piccola possibile, la lunghezza degli elementi di codice implica un sovraccarico del mezzo trasmissivo.

Le suddette considerazioni assumono particolare rilevanza pensando a una sorgente che emette sequenze di simboli. Pensiamo a una sorgente che emette numeri decimali fino a 5 cifre, oppure sequenze di lettere a 4 cifre presenti in un vocabolario di Italiano. Emetteremo sempre i numeri con 5 cifre premettendo gli 0 o ci fermeremo all'emissione della cifra più significativa? A ogni elemento/simbolo del messaggio assegniamo sempre il numero di bit minimale oppure è possibile variare il numero di bit, per esempio in base alla probabilità a priori di presentazione di quel simbolo, probabilità chiaramente non uniforme per l e lettere dell'alfabeto? Inoltre, se una sorgente emette messaggi composti da sequenze di simboli (dette stringhe), come fa il ricevitore a capire dove finisce una stringa e inizia un'altra?

Innanzitutto facciamo alcune considerazioni sull'uso di codici dei simboli a lunghezza variabile: la

lunghezza di una stringa di simboli non equiprobabili può essere ridotta assegnando ai simboli con probabilità maggiori codici più cort i e a quelli con probabilità inferiore codici più lunghi (come nel metodo di Huffmann)

***Codici a lunghezza fissa***

Su carta siamo abituati ad applicare un principio di economicità della rappresentazione che ci porta a definire la regola: "non si scrive zero come cifra più a sinistra”. In questo modo, oltre a risparmiare carta, otteniamo anche una rappresentazione unica per ogni numero. In mancanza di questa regola, potremmo scrivere un certo numero (per esempio cinquantasette) in un numero potenzialmente infinito di modi diversi, semplicemente aggiungendo un numero arbitrario di cifre "0" a sinistra di una rappresentazione scritta dello stesso numero (ossia: 57, 057, 005 " 00057 ecc.)

D'altra parte, questa **abitudine di scrivere numeri in forma minima corrisponde**

**all'uso di un codice a lunghezza variabile**,

numero di cifre diverso per la rappresentazione

Di solito questa caratteristica del codice non crea problemi, anche perché generalmente non si scrivono numeri di valore molto grande.

Se vogliamo applicare l'algoritmo di addizione, uno dei requisiti è quello di poter allineare le rappresentazioni degli addendi a destra, in modo da porre sulla stessa verticale le cifre che rappresentano le unità, le decine ecc.

Questa considerazione dimostra la necessità di **stabilire un massimo valore rappresentabile** quando si vogliono prefissare risorse da dedicare alla realizzazione di un algoritmo.

Volendo usare un **sistema di calcolo per la realizzazione di algoritmi**, dobbiamo preoccuparci di **definire a priori le risorse dedicate alla rappresentazione dei dati**. Nel caso della rappresentazione di numeri naturali, il modo più semplice per arrivare a gestire queste risorse è quello di **definire a priori un valore massimo** **rappresentabile per ogni numero (a cui corrisponde un numero massimo di cifre impiegate per la rappresentazione)**, con l'accortezza di usare un valore massimosufficientemente grande da poter trattare tutti (o quasi) i casi di rilevanza pratica, ma sufficientemente piccolo da non sprecare inutilmente risorse costose del sistema. Una volta determinato il massimo valore rappresentabile, esso determina anche il numero di cifre (decimali, binarie, ottali, o altro) da usare per la sua rappresentazione.

**Un codice viene detto a lunghezza fissa quando viene sempre usato lo stesso numero di cifre per la rappresentazione di qualunque valore.** Ovviamente il numero di cifre da usare è, in questocaso, il numero minimo di cifre per la rappresentazione del massimo valore predeterminato.

Per esempio, se decidessimo di trattare numeri non superiori a 9999, potremmo usare una rappresentazione decimale posizionale fissa a 4 cifre. In tal caso il numero cinquantasette verrebbe rappresentato (in modo unico) come 0057.

Esempi pratici di codici a lunghezza fissa nella vita quotidiana comprendono i numeri di telefono e i vari codici di sicurezza chiamati PIN associati a Bancomat, carte di credito, telefoni cellulari, codice fiscale ecc… Chi volesse telefonare da una q uaunque località italiana dovrebbe comporre un numero di 10 cifre '0995565758" incurante del fatto che la prima cifra sia 0.

***Codici a lunghezza variabile***

Una **sorgente** si dice **discreta** quando i **simboli** che può emettere sono in **numero finito**. Un **messaggio** può essere pertanto definito come una **sequenza finita di simboli**.

Le proprietà che caratterizzano i simboli sono:

* la **probabilità di emissione** ,
* la **quantità di informazione** da essi trasportata.

La **probabilità di emissione** indica la probabilità che un simbolo venga emesso tra gli N elementi dell'alfabeto.

Se i simboli hanno tutti la **stessa probabilità di emissione** , questa vale

P = 1

N

Se la probabilità di emissione **non è uguale** per tutti i simboli, occorre calcolarla: in modo empirico essa è data dal **rapporto tra il numero di volte che il simbolo si** **viene emesso sul numero totale** **dei simboli emessi**. Più elevato è il numero dei simboli (teoricamente un numero infinito disimboli) su cui tale rapporto si calcola e più attendibile è la probabilità.

**Misura dell'informazione**

Se avessimo una **sorgente che emette** sempre lo **stesso simbolo**, si potrebbe dire che la **sorgente** **non fornisce alcuna informazione**: è come se una persona rispondesse sempre allo stesso modo aqualsiasi domanda, per esempio sempre "si".

Per **informazione** si intende tutto ciò che serve a togliere incertez za.

Se la sorgente ha un alfabeto di 2 simboli, l'emissione di un simbolo fornisce una maggiore informazione rispetto alla sorgente precedente: in questo caso è come se una persona rispondesse 'si" o "no" alle domande; la sua risposta toglie incertezza.

Si può quindi affermare che **maggiore è l'insieme dei simboli che una sorgente può** **emettere, più informazione è trasportata da un simbolo**.

La **quantità di informazione** trasportata da un simbolo è data dalla seguente formula:

I(si) = log 1\_

P(si)

Applicando la proprietà dei logaritmi, per la quale il logaritmo di un rapporto è uguale alla differenza dei logaritmi, si può scrivere:

I(si) = log 1 - log P(si) = 0 - log P(si) = - log P(si)

**Se i logaritmi sono in base 2, il risultato è espresso in bit.**

Possiamo definire il **bit come l'unità elementare di informazione** .

Se una sorgente emette **sempre lo stesso simbolo, P = 1** **e quindi** **I = 0.**

Se una sorgente emette **N simboli equiprobabili** (tutti con la stessa probabilità), l'**informazione** associata **ad un simbolo** è data da:

I(si) = log2 1\_

P(si)

e poiché P = 1/N allora I = log 1 = log N

2 2

1/N

Si può notare che più grande è N, maggiore è l'informazione associata ad ogni simbolo che viene emesso.

Nel caso minimo in cui la sorgente emetta 2 soli simboli, l'informazione vale log22 = 1 bit.

**Quindi il bit (unità di misura dell’informazione) è l’informazione associata ad un simbolo che può essere scelto tra soli due simboli equiprobabil i**.

**Per codificare N messaggi equiprobabili sono necessari k = log2N bit;** infatti, intuitivamente,con k cifre binarie si possono comporre 2k parole diverse.

Nel sistema di numerazione decimale l’informazione relativa ad ogni cifra è log210 = 3,32 bit (ogni cifra ha probabilità 1/10).

Per rappresentare ogni cifra decimale servono 4 bit (l’intero superiore a 3,32).

Per rappresentare due cifre decimali (da 0 fino a 99) servono 7 bit (3,32 + 3,32 = 6,64 e si prende l’intero superiore a 6,64).

Per rappresentare tre cifre decimali (da 0 fino a 999) servono 10 bit (3,32 \* 3 = 9,96 e si prende l’intero superiore a 9,96).

Per rappresentare cinque cifre decimali (da 0 fino a 99999) servono 17 bit.

**Esempio 1**

*Determinare la quantità di informazione trasportata da un simbolo emesso da una sorgente avente un alfabeto di 16 simboli equiprobabili*

La probabilità di emissione di un simbolo è P(s) = 1/16 e quindi l'informazione trasportata è:

I(s) = log2 1 = log2 1 = log2 16 = 4

P(s) 1/16

**Esempio 2**

*Da una sorgente binaria con simboli equiprobabili ed indipendenti viene emesso il seguente messaggio M = 1111000. Determinare la quantità di i nformazione associata al messaggio.*

Poiché i simboli sono equiprobabili, P(s) = 1/2 Quindi entrambi i simboli trasportano la stessa quantità di informazione I(s) = log 2 2 = 1 bit e, poiché il messaggio è lungo 7 simboliindipendenti, l'informazione contenuta nel messaggio è I(M) = 7\*1= 7 bit

**Esempio 3**

*Una sorgente binaria con simboli indipendenti emette i simboli con probabilità P(0)= 1/4, P(l)= 3/4.*

*Determinare la quantità di informazione associata a l seguente messaggio M=11100.*

Poiché le probabilità sono diverse, anche i contenuti informativi sono diversi e precisamente:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| I(0) = log2 | 1 | = | log2 | 1 | = log2 4 | = 2 |
|  | P(0) |  |  | 1/4 |  |  |
| I(1) = log2 | 1 | = | log2 | 1 | = log2 4 | = log2 4 - log2 3 = 2 – 1,58 = 0,42 |
|  | P(1) |  |  | ¾ | 3 |  |

Nel messaggio ci sono 3 bit di valore 1 e 2 bit di valore 0.

Poiché i bit sono indipendenti si ottiene la seguente quantità di informazione: I(M) = 3 \* I(1) + 2 \* I(0) = 3 \* 0,42 + 2 \* 2 = 1,26 + 4 = 5,26 bit

**Entropia**

Si definisce **entropia l'informazione media di una sorgente S che emette simboli indipendenti** **in sequenza.**

Viene calcolata prendendo in considerazione l’informazione totale relativa a tutti i messaggi che la sorgente può emettere calcolata come media ponderat a (rispetto alle probabilità) delle quantità di informazione dei vari messaggi.

Supponendo che l'alfabeto sia formato dai simboli indipendenti s1, s2,….., s n con probabilità p(s 1),

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| p(s2), | …. , p(s | n), **l’entropia o informazione media** della sorgente è espressa dalla **formula di** | | | |  |
| **Shannon:** | |  |  |  |  |  |
|  | n | n |  |  |  |  |
| H(S) = | ∑ p(si )\* I(si) = ∑ p(si )\* log2 | | | 1 | bit |  |
|  | i = 1 | i = 1 | p(si ) | |  |  |
|  |  |  |  |  |

L’entropia si misura in bit ed indica il **grado di incertezza di ricevere un simbolo tra gli n** **dell'alfabeto della sorgente**.

La **massima entropia** si ha quando i **simboli sono equiprobabili.** In questo caso la probabilità di

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| emissione di un simbolo vale p(si) = 1/N e l'informazione associata è I(si) = log2 | | | | | | | | |  | 1 | = log2N; |  |
| applicando la formula in questo caso si ottiene | | | | | | | | | 1/N | | |  |
| n |  |  |  | n |  |  |  |  |  |  |  |  |
| H(S) = ∑ | 1 | log2 | 1 | = ∑ ( | 1 | log2N ) = N \* ( | 1 | log2N ) = log2N |  |  |  |  |
| i = 1 | N |  | 1/N | i = 1 | N |  | N | |  |  |  |  |
|  |  |  |  |  |  |  |  |

L’entropia delle 10 cifre decimali è log210 = 3,32

**In generale**, l'entropia di una sorgente è:

**H(S) <= log2 N**

Si dimostra che **l'entropia è uguale alla lunghezza media del codice**.

**Esempio 1 sull’Entropia (quattro assi equiprobabili)**

Proviamo a utilizzare in maniera più formale la correlazione intuitiva tra probabilità di emissione e contenuto informativo di un messaggio.

Nel caso di quattro messaggi equiprobabili, per esempio i quattro assi di un mazzo di carta, possiamo dire che ciascun messaggio dà un contribut o informativo pari a:

I = ¼ \* log 4 = ½ .

i 2

Il contenuto informativo totale della sorgente è (per i da 1 a 4):

I = ∑ Ii= 2 bit

i

**Esempio 2 sull’Entropia (quattro assi con diverse probabilità)**

In genere le possiamo attribuire a ciascun messaggio una probabilità P i e quindi vale la formula di Shannon dell’entropia, che esprime analiticamente l’entropia cioè la quantità media di informazione emessa da una sorgente che può emettere n messaggi appartenenti all'alfabeto A, ciascuno con una probabilità P i (∑ Pi =1).

Pensiamo ora che sia più facile far vedere l'asso di cuori, meno quello di quadri e raramente gli vedano gli altri due. Un osservatore se ne accorge, perché rilevando le frequenze di emissione nota una distribuzione non uniforme; ritiene quindi opportuno assegnare le seguenti probabilità:

PH = 1/2; PQ = 1/4; PP = PF = 1/8.

Notiamo che la sommatoria delle probabilità è sempre 1 = 1/2 + 1/4 + 1/8 + 1/8

Calcoliamo ora l'entropia:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| H = ½ \* log2 | 1 | + ¼ \* log2 | 1 | + 1/8 \* log2 |  | 1 | + 1/8 \* log2 | 1 | = |
| ½ | | ¼ | |  | 1/8 | |  | 1/8 |  |

= ½ \* 1og 2 + ¼ \* log 4 + 1/8 \*log 8+ 1/8 \*log 8 =

2 2 2 2

= ½ \* 1 + ¼ \* 2 + 1/8 \* 3 + 1/8 \* 3 = ½ + ½ + ¾ = 7/4 = 1,75

Come era lecito aspettarsi, H = 1,75 è minore dell'entropia della sorgente equiprobabile; infatti questa nuova sorgente è sicuramente più prevedibile della precedente.

Ricordiamo che il contenuto di informazione di un messaggio è una quantità ben definita e misurabile, ma non si riferisce al significato del messaggio trasmesso (semantica) e quindi al suo possibile valore affettivo, artistico, operativo o altro; **la quantità di informazione di un messaggio**

**è legata alla probabilità che l'evento oggetto del messaggio ha di verificarsi e quindi che venga trasmesso proprio quel dato messaggio**. Il contenuto di informazione è in funzione dalla maggioreo minore prevedibilità del messaggio che può essere trasmesso. In un insieme di possibili messaggi riguardanti un dato evento, **il più alto contenuto di informazione viene assegnato a quello che ha** **la minore probabilità di venire trasmesso perché meno prevedibile degli altri**. **Un messaggio che può essere previsto con certezza ha contenuto d i informazione 0.**

**Lunghezza media di un codice**

Se una sorgente emette parole lunghe l1, l2,…. l n in bit ciascuna di probabilità p 1, p2, …. , p n la lunghezza media del codice è data dalla seguente formula:

n

L = p1 \* l1 + p2 \* l2 + ….. + p n \* ln = ∑ pi \* li bit/parola

i = 1

dove pi è la probabilità di una parola di codice

li è la lunghezza di una parola di codice

Per parole equiprobabili e della stessa lunghezza, la lunghezza media del codice è pari alla lunghezza di una parola; se N è il numero di parole equiprobabili la lunghezza del codice è log2N.

Per il teorema di Shannon la lunghezza di un codice non può mai essere inferiore all’entropia

**L ≥ H**

Si definisce **efficienza** di un codice il **rapporto H/L** che è sempre < 1 e misura percentualmente l’efficienza.

Un codice ottimale ha efficienza che si avvicina a 1.

Es. usando 4 cifre binarie per rappresentare ogni cifra decimale:

L = 4 lunghezza del codice

H = 3.32 entropia delle cifre decimali

Efficienza = H/L = 3.32/4 = 0,83 (83%)

***Codice CICLICO DI RIDONDANZA o CONTROLLO POLINOMIALE (CRC)***

Il sistema fin qui adottato non è efficace in casi particolari, come quando nell'intero blocco considerato si presentano più errori di ordine pari che occupano la stessa posizione di ciascun carattere sia in orizzontale sia in verticale. In questo caso si deve usare il CRC (Controllo Ridondanza Ciclica), che permette la rilevazione di un'alta percentuale di errori con commutazione simultanea; esso utilizza una **ridondanza non troppo elevata** aggiungendo in coda al blocco una serie di bit.

Il CRC considera il **messaggio** in forma di **polinomio completo**, avente come coefficienti le cifre binarie (0,1); esso è chiamato polinomio messaggio; ricordiamo che un polinomio di grado (n-1) ha n coefficienti (in questo caso bit).

**M(x) = bn-1 \* xn-1 + ….. + b** **2\*x2 + b1\*x1 + b0\*x0** polinomio di grado (n-1) definito da n bit

Inoltre utilizza un polinomio **G(x)**, detto **polinomio generatore**, sempre a coefficienti binari, scelto base ad alcuni standard internazionali imposti dal CCITT, quali:

|  |  |  |  |
| --- | --- | --- | --- |
| G(x) = x16 + xl2 + x5 +1 | | | (CRC CCITT) |
| G(x) = x16 + x12 | | + x2 +1 | (CRC l6) |
| G(x) = x12 | + x1l +x3 +x2 +x+ 1 | | (CRC-12) |
| G(x) = x8 + x2 +x+ 1 | | | (CRC-8 ATM(HEC)) |
| G(x) = x16 | + x11 | + x4 + 1 | (CRC-CCITT inverso) |
| G(x) = x16 | + x14 | + x + 1 | (CRC-16inverso) |
| G(x) = x32 | + x26 | + x23 + x22 + x16 + x12 + x1l + x10 + x8 + x7 + x5 + x4 + x2 + x + 1 | |
|  |  |  | (CRC 32 Ethernet FDDI) |

Le sigle numeriche 16 e 12 indicano il numero di bit del carattere di controllo da aggiungere alla fine del blocco. Nel campo di applicazioni speciali, dove la precisione dei dati ricevuti è critica, si usane polinomi generatori di 32 bit (CRC-32).

Un CRC a 16 bit corregge:

* errori singoli e doppi;
* errori di numero dispari di bit;
* errori burst di lunghezza < = 16;
* 99,997% di burst lunghi 17;
* 99,998% di burst lunghi 18

Questi risultati valgono nell'ipotesi che gli m bit del messaggio siano distribuiti casualmente.

Le cifre ridondanti vengono aggiunte in coda a M(x), in modo da rendere il polinomio messaggio M'(x) così ottenuto sempre divisibile per il polinomio generatore G(x), ovvero tale che il resto R(x) sia uguale a zero. In ricezione infatti si effettua la stessa operazione tra i polinomi M’(x)/G(x) e si controlla che R(x) sia uguale a zero, nel qual caso non sono rilevati errori.

Tutte le **operazioni** (somme, sottrazioni, prodotti e divisioni) devono essere eseguite in **modulo 2**, ossia eseguendo **XOR bit a bit** senza generazione di riporti o prestiti. (Vedremo negli esempi come si fa).

Il polinomio generatore **G(x) deve essere di grado inferiore** al polinomio M(x) e deve avere il **termine noto uguale a 1**.

Dalla parte della sorgente (trasmittente) si esegue il seguente algoritmo (tutte le somme e differenze si effettuano in modulo 2, ossia senza generare riporti o prestiti):

1. Supponendo che il grado del polinomio generatore G(x) sia m, bisogna **aggiungere m zeri** in coda al messaggio M(x) da trasmettere. Se n sono i bit di M(X) il messaggio ottenuto, che chiameremo **M’(x)** avrà **(n+m) bit**.
2. Si divide M’(x) per G(x) e si **calcola il resto R(x)** che ha tanti bit quanto è il grado di G(x).
3. Si sottrae R(x) da M’(x) e si ottiene il messaggio T(x) da trasmettere. Il messaggio da trasmettere è composto dal messaggio M(x) originale seguito dal resto espresso in modulo 2.

4)

Dalla parte del ricevente si deve fare l'operazione inversa: il messaggio ricevuto T(x) si divide per il polinomio G(x), e si controlla che il resto ottenuto sia uguale a zero (Trasmissione corretta).

Se si verifica il caso di un resto diverso da zero, allora risulta un errore di trasmissione.

***Codici di HAMMING***

I codici di Hamming sono codici con h = 3 o h = 4 usati come rivelatori d'errore o come autocorrettori.

Dai Teoremi 1 e 2 discende che: (r è la molteplicità dell’errore)

I **codici con h=3** sono in grado di **rivelare errori semplici e doppi**, se usati come rivelatori di

|  |  |
| --- | --- |
| errore (EDC) in quanto vale h > r + 1 | ossia r < 2 , o di **correggere errori singoli**, in quanto |
| vale h > 2 c + 1ossia c < 1 | interpretando ogni configurazione non significativa come |

originata da quella significativa piu' vicina. In tal caso tuttavia il codice **non e' in grado di rivelare** **gli errori doppi.**

I codici con **h = 4** individuano gli **errori semplici, doppi e tripli** quando usati come **EDC**. Se usati come auto correttori **ECC, correggono gli errori singoli e individuano quelli doppi**.

In generale, se **r** e' la **molteplicita' massima degli errori rilevabili** e **c la molteplicità** degli errori correggibili, si ha:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| per i **codici rivelatori di errori**: | r = h-1 |  | (vale infatti | | h > r + 1 ) |
| per i **codici auto correttori**: | c < h/2 | | e | c + r | < h |
| (nel caso specifico di h = 3 e codice ECC | | c < 3/2 ossia prendendo la parte intera c = 1 | | | |
|  |  |  | 1 + r < 3 | ossia r < 2) | |
| (nel caso specifico di h = 4 e codice ECC | | c < 4/2 ossia prendendo la parte intera c = 1 | | | |
|  |  |  | 1 + r < 4 | ossia r < 3) | |

Il codice di **Hamming con h=3** si ottiene **aggiungendo agli n bit di un codice** **efficiente k bit di controllo** ognuno dei quali sia destinato al **controllo di parita' pari di un gruppo opportuno degli m = n + k** bit risultanti.

Piu' esattamente **l'i-esimo bit di controllo** va situato **nella posizione 2i-1** della parola e **controlla la parita' di gruppi alternati di 2i-1** **bit** a cominciare **dalla posizione 2i-1**.

Il primo bit di controllo va quindi nella prima posizione e controlla la parita' dei bit 1,3,5,7,9,...(in questo caso ogni gruppo è composto da un solo bit);

il secondo bit di controllo va in posizione 2 e controlla la parita' dei bit nelle posizioni (2,3),(6,7),(10,11),....;

il terzo bit va nella quarta posizione e controlla la parita' dei bit nelle posizioni (4,5,6,7), (12,13,14,15), e cosi' via.

**In ricezione si verifica la parita' di ciascun gruppo, partendo dal 1° bit di controllo e procedendo con gli altri bit di controllo, e per ogni verifica esatta si scrive uno 0, per ogni verifica errata un 1.**

La stringa cosi' ottenuta, letta in senso inverso, forma il cosidetto **numero di controllo Nc ,**

**che e' nullo nel caso in cui non vi siano errori.**

In caso contrario, ognuno dei 2k -1 possibili valori di **Nc indica in quale delle m** **posizioni si e' verificato un errore di molteplicita' 1**.

ESEMPIO

Per quanto su spiegato, la composizione della parola con h = 3, indicando con **k i bit di controllo** e con **b i bit del codice efficiente** di partenza, sarà' quindi:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| posizione |  | 1 2 3 4 | 5 | 6 | 7 |
|  |  | **k1 k2 b1 k3 b2 b3 b4** | | | |
| Ad esempio, il messaggio 1 0 0 | | | 1 andrà ad occupare le posizioni mostrate di seguito: | | |
| posizione |  | 1 2 3 4 | 5 | 6 | 7 |
|  |  | **k1 k2 b1 k3 b2 b3 b4** | | | |
|  |  | 1 | 0 | 0 | 1 |

L’applicazione del codice di Hamming darà' luogo al la parola di codice (codeword da trasmettere):

|  |  |  |  |
| --- | --- | --- | --- |
| posizione |  | 1 2 | 3 4 5 6 7 |
|  |  | **k1 k2 b1 k3 b2 b3 b4** | |
| codeword |  | 0 0 | 1 1 0 0 1 |

Si supponga che la parola ricevuta sia invece:

**k1 k2 b1 k3 b2 b3 b4**

0 0 0 1 0 0 1

I primi due controlli di parita' (sui bit 1,3,5,7 e (2,3) e (6,7) rispettivamente) danno risultato errato quindi 1 e 1, mentre il terzo controllo (sui bit 4,5,6,7) e' esatto quindi 0, ossia al ricevitore la verifica genera 1 1 0 . Il numero di controllo e' quindi:

Nc = 0 1 1

corrispondente alla posizione (3 in decimale) del bit errato.

La ridondanza del codice e' R = 7/4 = 1,75

Per quanto riguarda la probabilita' di errore non rivelato, anche ammettendo che tutte le configurazioni a distanza 3 da una parola del codice siano esse stesse parole del codice, si ha

con p = 0,01:

P t < 15\* 0,013 = 0,0015 %

**I codici di Hamming con h=4 si ottengono da quelli con h=3 aggiungendo in ultima posizione un ulteriore bit di controllo che verifichi la parita' di tutti i bit che lo precedono.**

La verifica di un messaggio si fa controllando tutti i bit di parita'. Se i primi (k - 1) controlli non sono tutti esatti mentre l'ultimo lo e' allora si e' in presenza di un errore doppio non correggibile. Supponendo di aver ottenuto dal codice h=3 appena illustrato quello h=4, si ha:

R = 8/4 = 2 Pt  15.p 4 = 1,5.10 -7

**Per il corretto funzionamento del codice di Hamming, ossia su n bit (di codice efficiente), si devono impiegare almeno r o in altra forma n < 2r – 1**

**per correggere un singolo errore check bit con r >= log2 (n + 1)**

**Si dicono ottimi quei codici per cui la relazione appena scritta e' verificata con l'uguaglianza.**

**Si supponga di voler codificare 16 simboli. In questo caso:**

**n = 4** **r > log2 (4 + 1) = 3** **m = 4 + 3 = 7**

(E’ il caso dell’esempio precedentemente svolto)

***Codice di HUFFMAN***

Consideriamo una sorgente che "legge" testi formati dalle 21 lettere maiuscole dell'alfabeto italiano (pensiamo a uno scrittore molto superficiale che non usa punteggiatura, scrive a stampatello e non usa cifre!).

Utilizzando esempi concreti si può ricavare una sta tistica di presentazione dei 21 caratteri.

Creiamo una tabella dei caratteri che associa a ciascuno la relativa probabilità e ordiniamo l'elenco in ordine decrescente di probabilità