

TTI - Teoria Transmisiei Informatiei

Note de curs

Daniela Coltuc

2019

Notatii si abbrevieri

X variabila aleatoare

x_i realizare particulara a variabilei aleatoare X

$p(x = E_i) = p(E_i) = p_i$ probabilitatea ca evenimentul E_i sa se realizeze

$F(x)$ functie de repartitie sau distributie a unei variabile aleatoare

$f(x)$ densitate de probabilitate a unei variabile aleatoare

$[X]$ alfabetul sursei

$[P(X)]$ setul de probabilitati asociat alfabetului $[X]$

$H(X)$ entropia sursei cu alfabet $[X]$

v.a. variabila aleatoare

1. INTRODUCERE

Teoria Informatiei raspunde la doua intrebari fundamentale in telecomunicatii:

- **Cat de mult pot fi compresate datele?** (Shannon a aratat ca datele reprezentand procese aleatoare ca muzica sau vorbirea, nu pot fi compresate sub o anumita limita pe care a numit-o *entropie*, un termen folosit deja in termodinamica)
- **Cat de mult se poate transmite printr-un canal de comunicatie ?** (In anii 40, comunitatea stiintifica credea ca marind cantitatea de informatie transmisa printr-un canal, creste si probabilitatea eronarii ei; Shannon a surpris lumea stiintifica, aratand ca transmisia poate fi facuta cu o probabilitate de eroare oricat de mica, cu conditia ca rata de transmisie sa nu depaseasca *capacitatea canalului*)

Teoria Informatiei are aplicatii nu numai in telecomunicatii ci si in alte domenii (Fig. din *Elements of Information Theory*, Thomas M. Cover, Joy A. Thomas)

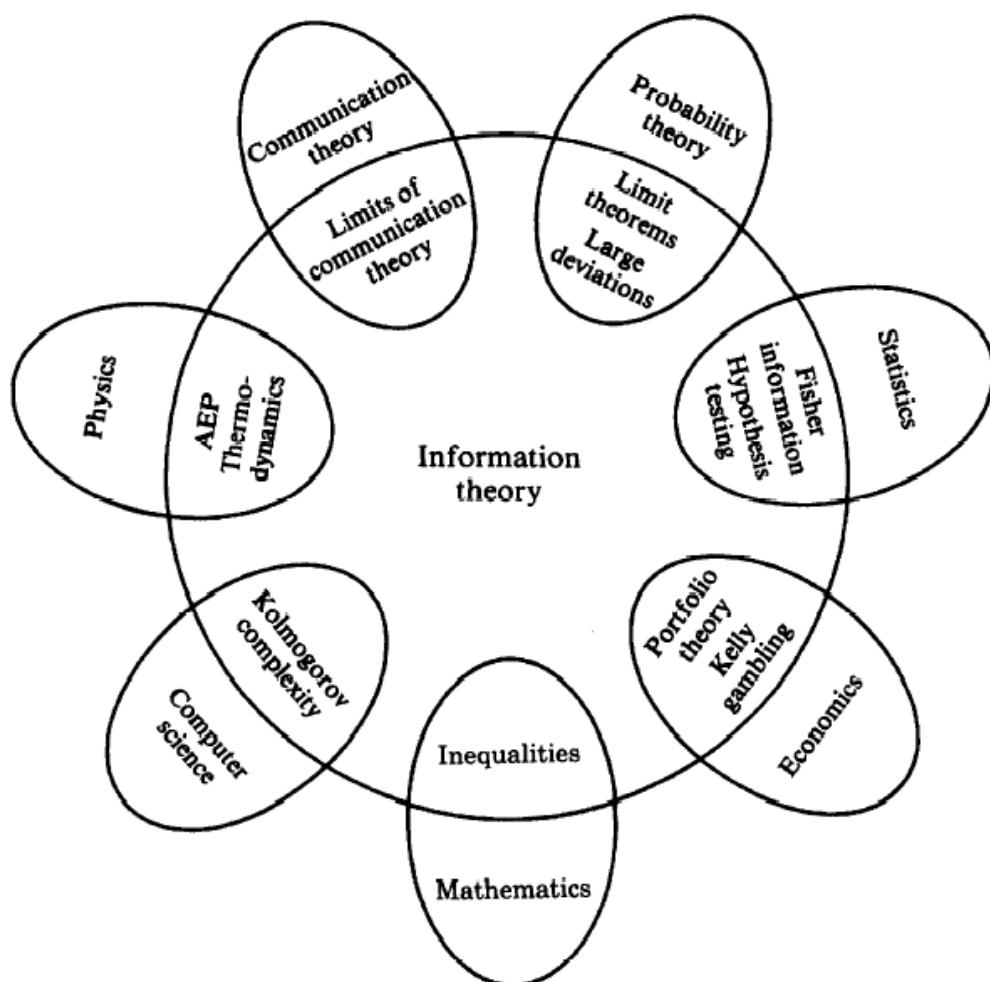


Figure 1.1. The relationship of information theory with other fields.

Teoria Informatiei a fost elaborata de Claude Shannon (1916-2001) in 1948.

Claude Shannon (1916 – 2001)

1935 Licenta la Michigan University

1936 Asistent de cercetare la MIT (Massachusetts Institute of Technology)
Master

1940 Doctorat în matematică cu teza « *Mathematics in genetics* »

15 ani la Bell Laboratories în compania unor personalități din lumea științei: John Pierce (comunicațiile prin satelit), Harry Nyquist (Teoria semnalelor), Hendrik Bode (diagramale), inventatorii tranzistorului (Shokley, Bardeee, Brattain).

1948 Shannon a elaborat *Teoria informației*, una dintre teoriile cu cel mai mare impact în secolul XX.

Shannon, Claude E. "A mathematical theory of communication, Part I, Part II." *Bell Syst. Tech. J.* 27 (1948): 623-656.

A fondat teoria informației printr-un articol de referință, intitulat „**O teorie matematică a comunicației**”, publicat în anul **1948**. În această lucrare s-a bazat pe rezultate anterioare obținute de Nyquist și Hartley, pe care le-a integrat într-o teorie completă care depăsea cu mult inteligeerea pana atunci a problemelor de comunicații. În sprijinul acestei teorii Shannon a dezvoltat noțiunea de **entropie informatională** ca măsură a incertitudinii dintr-un mesaj.

El este considerat, de asemenea, fondatorul teoriei proiectării circuitelor digitale și calculatoarelor numerice încă din **1937**, când, la vîrstă de 21 de ani, fiind student la masterat la **MIT**, a scris o teză prin care demonstra că folosind **algebra booleene**, se poate construi și rezolva orice relație logică numerică. Aceasta este una dintre cele mai importante disertații de masterat din toate timpurile.

Shannon și soția sa Betty își petreceau adesea weekendurile în **Las Vegas** împreună cu matematicianul **Ed Thorp** de la **MIT**,^[22] câștigând masiv la **ruletă** și **blackjack**. Folosindu-se de metode din **teoria jocurilor** dezvoltate împreună cu colegul lor de la Laboratoarele Bell, fizicianul **John L. Kelly Jr.**, și bazate pe principii din teoria informației.^[23] Au câștigat o avere, după cum se arată în cartea **Fortune's Formula** de **William Poundstone** și în scrierile lui **Elwyn Berlekamp**,^[24] asistent al lui Kelly între 1960 și 1962.^[14] Shannon și Thorp au aplicat aceeași teorie, ulterior cunoscută drept **criteriul Kelly**, la bursa de acțiuni, cu rezultate și mai bune.

Profesorul Alexandru Spataru (1920-2012) - o personalitate proeminentă a științei românești.

In 1953, sub coordonarea sa, au fost initiate cercetările pentru introducerea televiziunii în România. Doi ani mai tarziu, ele se concretizau într-un echipament de studio și o prima stație de emisie TV, construite integral în țară. Cercetările au fost reluate în 1962, de data aceasta pentru televiziunea în culori și, în 1964, împreună cu un grup de tineri cercetatori, Profesorul Spataru realizează prima transmisiune experimentală de imagini color din țară noastră.

Profesorul Spataru a introdus în învățământul tehnic românesc studiul Teoriei Informației, despre care unele vocile spun că ar fi teoria cu cel mai puternic impact din secolul XX. Cartea sa **Teoria Transmisiei Informației**, publicată de Editura Tehnică București, Editura Masson Paris, Editura Vieweg und Shon Braunschweig și Editura Akademie-Verlag Berlin, a format numeroase generații de ingineri. Versiunea franceză este cunoscută în țările francofone ca fiind una din primele cărți de prelucrare de semnal.

A introdus cursul de Teoria Informației la Facultatea de Electronică și Telecomunicații din Institutul Politehnic București, și aceasta la numai un an după publicarea noțiunilor fundamentale ale acestei teorii, elaborate de C. E. Shannon în 1948.

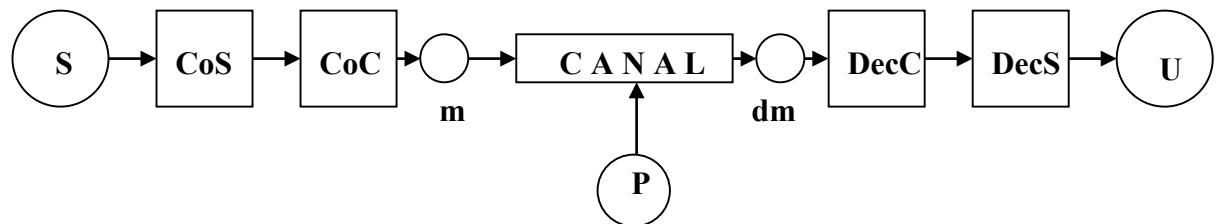
In facultatea noastră, Profesorul Spataru a fondat în 1962 Catedra de Electronică Aplicată, pe care a condus-o timp de două decenii. Toti cei care predau astăzi TTI-ul în Facultatea de Electronică din București poartă amprenta personalității sale.

Profesorul Alexandru Spataru a fost Vicepreședinte C.N.C.S. și Președinte al Comisiei Române de Activități Spațiale, a fost membru al Independent Commission for World-wide

Telecommunications Development (U.I.T.), al International Council for Computer Communication, al International Television Committee si IFAC Technical Committee on Space. A condus delegații ale României la conferințe ale Uniunii Internaționale de Telecomunicatii, U.N. Committee on Science and Technology si U.N. Committee on Outer Space.

1.1. Schema generala a unui sistem de comunicatii

Definitie : Un sistem de transmisiune (de comunicatie) este un ansamblu fizic care realizeaza transmiterea unui *mesaj* de la o *sursa* la un *utilizator*.



S sursa de informatie (fara memorie)

CoS Codor de sursa (compresia datelor)

CoC Codor de canal (protectie contra perturbatiilor)

m modulator

CANAL Canal de comunicatie

Dm demodulator

P Perturbatii

DecC Decodor de canal

DecS Decodor de sursa

U Utilizator

2. INTRODUCERE IN TEORIA PROBABILITATILOR

2.1. Experiment aleator, evenimente

Definitie : Un experiment aleator este un experiment cu mai multe rezultate posibile.

Definitie : Rezultatele unui experiment aleator se numesc **evenimente**.

Exemplu: aruncarea unui zar

Este un experiment aleator cu 6 evenimente posibile.

Multimea evenimentelor elementare: $E = [E_1, \dots, E_6]$

Multimea evenimentelor se poate lărgi adăugând: - evenimentul sigur (orice fata)
- evenimentul imposibil (fata 7)
- evenimente compuse (fata para)

Rezultatul unui experiment aleator nu este cunoscut dinainte; un eveniment se realizează cu o anumită probabilitate.

2.2. Probabilitatea unui eveniment E_i

Definitia 1 (clasica, de acum cateva secole): $p(E_i) = \frac{N_f}{N_p}$ unde N_f este numărul de cazuri favorabile evenimentului și N_p numărul de cazuri posibile.

Definitia 2 (Von Mises, inceput de sec XX): $p(E_i) = \lim_{n \rightarrow \infty} \frac{n_a}{n}$ unde n_a este numărul de aparitii ale evenimentului si n este numărul total de experimente.

Definitia 3 (Kolmogoroff, 1933): Axiomele probabilitatilor

- $p \geq 0$ probabilitatea este un număr nenegativ
- $p(S) = 1$ probabilitatea evenimentului sigur este 1
- $p(E_1 + E_2) = p(E_1) + p(E_2)$ probabilitatea a două evenimente mutual exclusive, E_1 și E_2 , este egală cu suma probabilităților evenimentelor.

2.3. Variabila aleatoare

Definitie: Variabila aleatoare (v.a.) este o funcție care asociază valori numerice fiecarui eveniment o valoare numerică.

Notăm cu X v.a. care descrie aruncarea cu zarul.

$X : E \rightarrow R$ (X asociaza fiecarui eveniment o valoare numerica)

Exemplu:

Zarul $X : E = [E_1, \dots, E_6] \rightarrow [1, 2, 3, 4, 5, 6]$

Observatie:

- a) Oricarei submultimi a multimii valorilor lui X ii corespunde un eveniment (elementar sau compus).
- b) $1, 2, 3, 4, 5, 6$ se numesc *realizari particulare* ale v.a. X .

2.4. Distributia unei v.a.

Notam probabilitatea ca un eveniment E_i sa se realizeze cu:

$$p(E_i) = p(X = x_i) = p(x_i) = p_i$$

Exemple:

- a) Aruncarea cu zarul: realizarile particulare ale lui X constituie o multime discreta.
- b) Masurarea temperaturii: realizarile particulare ale v.a. iau valori intr-un interval.

Tipuri de v.a.:

V.a. *discreta* ia valori intr-o multime discreta

V.a. *continua* ia valori intr-un interval

V.a. *mixta*

Definitie: Functia de repartitie (Functia de distributie) a unei v.a.

$$F(x) = p\{X \leq x\}$$

Definitie: Densitatea de probabilitate a unei v.a. continue (distributia v.a.)

$$f(x) = \frac{dF}{dx}$$

Exemple:

V.a. discrete: Functia de repartitie este o functie in scara

Densitatea de probabilitate este o serie de functii Dyrac

V.a. continue: Densitate de probabilitate Gaussiana (sau normala): $f(x) = \frac{x^{-(x-\mu)^2/2\sigma^2}}{\sqrt{2\pi}\sigma}$ unde μ este media v.a., iar σ^2 este varianta v.a. (σ se numeste dispersie).

Densitatea de probabilitate uniforma: $f(x) = \begin{cases} 1/a & x \in [0, a] \\ 0 & \text{in_rest} \end{cases}$

Densitatea de probabilitate exponentiala: $f(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & \text{in_rest} \end{cases}$

2.5. Variabile aleatoare conditionate

Exemplu: La aruncarea cu zarul, probabilitatea de a avea un 2 cand stim ca fata aparuta este para este $1/3$. Aceasta probabilitate este diferita de probabilitatea neconditionata a lui 2, care este $1/6$.

Definitie: Probabilitatea unui eveniment E_i , conditionat de un alt eveniment E_j , este probabilitatea de a se realiza E_i cand E_j este deja realizat

$p(E_i/E_j) = \frac{p(E_i, E_j)}{p(E_j)}$ unde $p(E_i, E_j)$ este probabilitatea ca atat E_i cat si E_j sa se realizeze.

Daca notam cu X v.a. care descrie aruncarea cu zarul, atunci probabilitatea de mai sus se poate scrie:

$$p(X = x_i / E = x_j) = \frac{p(x_i, x_j)}{p(x_j)}$$

sau mai simplu

$$p(x_i / x_j) = \frac{p(x_i, x_j)}{p(x_j)}$$

Daca E_i si E_j sunt evenimente a doua experimente diferite caracterizate de v.a. X si Y , atunci probabilitatea conditionata se scrie :

$$p(x_i / y_j) = \frac{p(x_i, y_j)}{p(y_j)}$$

Teorema lui Bayes: $p(x_i / y_j) = \frac{p(y_j / x_i)p(x_i)}{p(y_j)}$

Teorema probabilitatii totale :

$$p(x_i) = p(x_i / y_1)p(y_1) + p(x_i / y_2)p(y_2) + \dots + p(x_i / y_N)p(y_N) \quad (\text{prima forma})$$

$$p(x_i) = p(x_i, y_1) + p(x_i, y_2) + \dots + p(x_i, y_N) \quad (\text{a 2-a forma})$$

unde y_1, y_2, \dots, y_N constituie o partitie a multimii realizarilor particulare ale v.a. Y .

Observatii:

a) Functia de repartitie si densitatea de probabilitate se definesc si pentru v.a. conditionate

$$F(x/x_j) = p\{X < x/x_j\} \quad f(x/x_j) = \frac{dF(x/x_j)}{dx}$$

b) Functia de repartitie si densitatea de probabilitate se definesc si pentru 2 sau mai multe v.a.

$$F(x,y) = p\{X \leq x, Y \leq y\} \quad f(x,y) = \frac{dF(x,y)}{dxdy}$$

2.6. Notiunea de independenta statistica

Definitie: Doua evenimente, E_i si E_j , sunt independente daca

$$p(E_i, E_j) = p(E_i)p(E_j)$$

Definitie: Doua v.a., X si Y , sunt independente daca oricare dintre realizarile lor particulare sunt independente.

$$p(x_i, y_j) = p(x_i)p(y_j) \text{ unde } x_i \text{ este o realizare particulara a lui } X \text{ si} \\ y_j \text{ este o realizare particulara a lui } Y.$$

2.7. Semnalele numerice ca siruri de v.a.

Un semnal numeric este o serie de valori numerice. Cum se ajunge la un semnal numeric? Prin esantionarea si cuantizarea semnalului continuu.

Un semnal numeric poate fi modelat ca un sir de v.a.: $\dots, X_{k-1}, X_k, X_{k+1}, \dots$, unde k este indice de timp. Toate v.a. iau valori in aceeasi multime si, daca semnalul este *stationar*, au acelasi set de probabilitati.

3. SURSE DE INFORMATIE

3.1. Informatia

3.1.1. Definitii si notatii

Definitie : Informatia este **cantitatea** de incertitudine pe care o avem asupra producerii unui eveniment, rezultat in urma unui experiment aleator.

Fie un experiment aleator ale carui rezultate sunt descrise prin v.a. X , care ia valori in multimea $[X] = [x_1, x_2, \dots, x_n]$. Incertitudinea asupra evenimentului E_i , caruia ii corespunde realizarea particulara x_i , se noteaza:

$$U(E_i) = U(X = x_i) = U(x_i)$$

U de la *uncertainty*

Incertitudinea si informatia sunt, din punct de vedere cantitativ, doua notiuni echivalente. Vorbim despre incertitudine inainte de producerea evenimentului si de informatie dupa producerea sa.

$$U(x_i) = i(x_i)$$

i de la *information*

Incertitudinea/informatia unui eveniment depinde de probabilitatea de aparitie p_i a evenimentului:

$$U(x_i) = i(x_i) = F(p_i) = -\log(p_i)$$

3.1.2. Specificarea functiei F

Trei proprietati intuitive pentru F :

- F trebuie sa fie descrescatoare (incertitudinea este mai mica atunci cand probabilitatea de aparitie a evenimentului este mare).
- F trebuie sa fie aditiva (incertitudinea asupra a doua evenimente, rezultate din experimente independente, trebuie sa fie egala cu suma incertitudinilor asupra celor doua evenimente):

$$F(p_i \cdot q_j) = F(p_i) + F(q_j)$$

unde p_i si q_j sunt probabilitatile celor doua evenimente independente.

c) $F(1)=0$ (incertitudinea asupra unui eveniment sigur este nula).

Functia care indeplineste cerintele b) si c) este **logaritmul**; pentru a satisface si cerinta a), se ia **valoarea negativa a logaritmului**:

$$F(p_i) = -\log(p_i)$$

Deci, incertitudinea/informatia asupra unui eveniment care are probabilitatea p_i , se calculeaza cu:

$$U(x_i) = i(x_i) = -\log(p_i)$$

Observatie: expresia $-\log(p_i)$ este o masura **cantitativa**, nu calitativa, a incertitudinii/informatiei. Cand vorbim de informatie in sensul expresiei $-\log(p_i)$, vorbim, de fapt, de *cantitatea de informatie*.

Proprietati :

- informatia este totdeauna o cantitate pozitiva

3.1.3. Unitati de masura pentru informatie

a) BIT (BInary uniT)

Definitie : 1 bit este cantitatea de informatie care se obtine cand se realizeaza un eveniment cu probabilitatea 1/2.

$$1bit = -\log_2(1/2)$$

b) DIT (Decimal unit)

Definitie : 1 dit este cantitatea de informatie care se obtine cand se realizeaza un eveniment care are probabilitatea 1/10..

$$1dit = -\log_{10}(1/10)$$

c) NAT (Natural unit)

Definitie : 1 nat este cantitatea de informatie care se obtine cand se realizeaza un eveniment cu probabilitatea 1/e (numarul lui Euler e=2,7...).

$$1nat = -\ln(1/e)$$

Transformarea unitatilor :

$$\begin{aligned} 1dit &= 3,32bit \\ 1nat &= 1,44bit \end{aligned}$$

3.1.4. Informatia mutuala a doua evenimente

De ce este necesar studiul a doua evenimente? De exemplu, in transmisia semnalelor, pe canalul de comunicatie, de cele mai multe ori, apar perturbatii care modifica semnalul. De aceea, semnalul de la intrarea in canal si cel de la iesire se descriu prin doua v.a. diferite, X si Y . Daca puterea perturbatiilor este finita, atunci aceste v.a. **nu sunt independente**.

Fie x_i si y_j doua realizari particulare ale v.a. X si Y . Sa pp. ca numai y_j este observabil.

Informatia mutuala a celor doua evenimente reprezentate de x_i si y_j este:

$$i(x_i, y_j) = U(x_i) - U(x_i / y_j)$$

unde $U(x_i / y_j)$ este incertitudinea care ramane asupra producerii lui x_i dupa observarea lui y_j .

$$i(x_i, y_j) = -\log p(x_i) + \log p(x_i / y_j) = \log \frac{p(x_i, y_j)}{p(x_i)p(y_j)}$$

Observatie: informatia mutuala poate fi si negativa.

Exemplu:

$$p(x_i) = 1/2 \text{ (probabilitatea de a se obtine o fata impara la aruncarea cu zarul)}$$

$$p(x_i / y_j) = 1/3 \text{ (probabilitatea de a se obtine o fata impara cand stim ca fata obtinuta este >3)}$$

$$i(x_i, y_j) = 1 - 1.58 = -0.58$$

Informatia mutuala in diverse cazuri de dependenta intre X si Y :

a) X si Y sunt identice (maxim de dependenta):

$$p(x_i, y_j) = p(x_i) = p(y_j) \text{ si } i(x_i, y_j) = i(x_i)$$

b) X si Y sunt independente statistic

$$p(x_i, y_j) = p(x_i)p(y_j) \text{ si } i(x_i, y_j) = 0$$

c) X si Y sunt diferite, dar dependente statistic

$$p(x_i / y_j) < 1 \text{ deci } \log p(x_i / y_j) < 0 \Rightarrow i(x_i, y_j) < i(x_i)$$

3.2. Surse discrete de informatie

3.2.1. Definitii si notatii

Definitie :

Sursa discreta de informatie este un mecanism de generare a unui sir de v.a. discrete :

$$\dots, X_{k-1}, X_k, X_{k+1}, \dots,$$

unde k este, de cele mai multe ori, un indice de timp.

Sursa de informatie este definita printr-un **alfabet** $[X] = [x_1, x_2, \dots, x_N]$, care este multimea realizarilor particulare ale v.a. $\dots, X_{k-1}, X_k, X_{k+1}, \dots$ si un **set de probabilitati** $[P] = [p_1, p_2, \dots, p_N]$, unde $p_i = p(x_i)$ (setul de probabilitati poate varia in functie de k daca sursa este nestacionara).

$$\sum_i p_i = 1$$

Definitii :

Simbolul (sau **litera**) este elementul fundamental, ireductibil, care contine informatie. x_1, x_2, \dots, x_N sunt simboluri

Alfabetul este totalitatea simbolurilor diferite care pot fi generate de sursa. $[X]$ este alfabetul sursei

Cuvantul este o succesiune de simboluri (*Exemplu*: un byte este o succesiune de 8 simboluri binare).

Limba este totalitatea cuvintelor formate cu un alfabet (*Exemplu*: 256 de cuvinte binare de 8 biti).

Exemple de surse discrete:

1. Banda cu text de la TV este o sursa care emite litere: IN TARA AU FOST INUNDATII...
2. Un semafor este o sursa cu trei simboluri: rosu, galben, verde

3.2.2. Clasificarea surselor discrete

- a) **Din punctual de vedere al dependentei** dintre v.a X_k :
- surse fara memorie
 - surse cu memorie

Definitie: **Sursa fara memorie** (numita si simpla sau independenta) genereaza v.a. independente. Cu alte cuvinte, probabilitatea de a genera un anumit simbol x_i la momentul k nu depinde de simbolurile generate anterior.

$$p(X_k = x_i / X_{k-1}, X_{k-2}, \dots) = p(X_k = x_i)$$

Definitie: **Sursa cu memorie** genereaza v.a. dependente.

Definitie: **Dimensiunea memoriei** sursei este egala cu numarul de simboluri anterioare care conditioneaza probabilitatea de aparitie a unui nou simbol.

Exemplu: $p(X_k = x_i / X_{k-1})$ este o sursa cu memorie de lungime 1.

b) **Din punctul de vedere al stabilitatii setului de probabilitati**

- surse stationare
- surse nestationare

Definitie: o sursa stationara are un set de probabilitati care nu variaza in functie de k .

$$p(X_k = x_i) = p(X_{k+\tau} = x_i) \text{ oricare ar fi } k, \tau \text{ sau } i.$$

Un caz particular al surselor stationare este **sursa ergodica**. Pentru a defini sursa ergodica, ne bazam pe notiunea de **sir tipic**.

Definitie: **Sir tipic**

Fie un sir de simboluri generat de sursa, suficient de lung a.i. sa putem estima probabilitatile simbolurilor folosind definitia probabilitatii ca raport intre numarul de aparitii ale fiecarui simbol si numarul total de simboluri din sir.

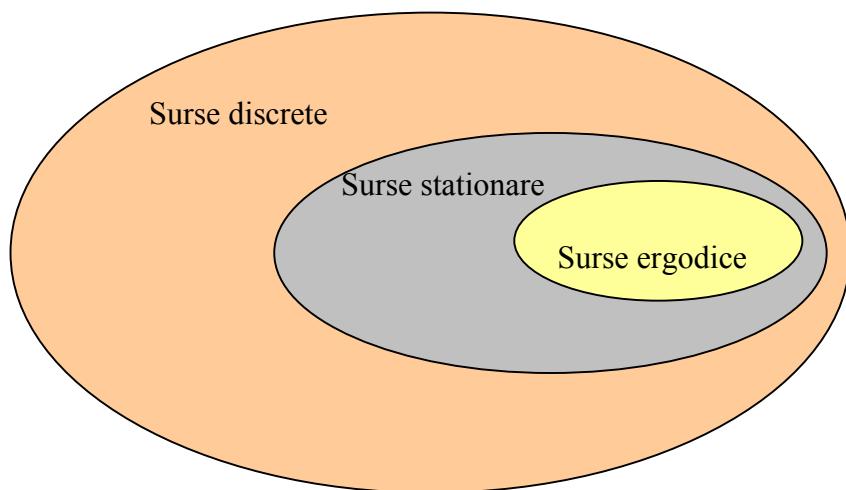
Daca intr-un sir, probabilitatile astfel estimate sunt egale cu probabilitatile din setul sursei, atunci **sirul este tipic**.

Altfel spus, daca n este lungimea sirului tipic considerat si n_i este numarul de simboluri x_i din sir, atunci $p_i = \frac{n_i}{n}$ oricare ar fi i .

Definitie: O sursa ergodica este o sursa care genereaza numai siruri tipice.

Observatii:

- in cazul surselor ergodice, probabilitatile simbolurilor se pot estima pe un sir emis de sursa (suficient de lung pentru a nu avea erori de estimare mari)
- Definitiile *stationaritatii* si *ergodicitatii* de mai sus sunt valabile pentru sursa fara memorie. In cazul sursei cu memorie, ele se enunta inlocuind notiunea de *simbol* cu cea de *stare* (definitia starii este data in subcapitolul de Surse Markov).
- o sursa ergodica este totdeauna si stationara, reciproca nu este adevarata



3.3. Surse Markov

Sursa Markov este un model matematic des folosit in practica pentru a descrie sursele discrete de informatie, cu memorie. Exista diverse definitii pentru sursa Markov.

3.3.1. Definitii si notatii

Definitia I : Sursa Markov este o sursa discreta, cu memorie de lungime constanta.

Definitie : Ordinul sursei Markov este dat de lungimea memoriei.

Definitie : Starea sursei Markov la un momentul k este data de sirul de simboluri de lungime egala cu ordinul sursei, care conditioneaza aparitia simbolului de la momentul k .

Exemplu : Sursa Markov binara de ordinul 2

Alfabetul : $[X] = [0,1]$

Probabilitatile de aparitie a simbolurilor sunt probabilitati conditionate, de forma
 $p(X_k = x_i / X_{k-1}, X_{k-2})$

$$\begin{array}{cccc} p(0/0,0) & p(0/0,1) & p(0/1,0) & p(0/1,1) \\ p(1/0,0) & p(1/0,1) & p(1/1,0) & p(1/1,1) \end{array}$$

Multimea starilor $[S] = [00 \ 01 \ 10 \ 11]$

Multimea starilor are N^R , unde N este dimensiunea alfabetului, iar R este ordinul sursei.

Definitie: Probabilitatea ca sursa Markov sa fie intr-o anumita stare este egala cu probabilitatea de aparitie a sirului de simboluri care constituie starea.

Definitia II: O sursa Markov se defineste prin urmatoarele marimi:

Alfabetul simbolurilor: $[X] = [x_1, x_2, \dots, x_N]$

Setul de probabilitati ale simbolurilor: $[P(X)] = [p_1, p_2, \dots, p_N]$ cu $\sum_i p_i = 1$

Multimea starilor: $[S_k] = [s_1, s_2, \dots, s_{N^k}]$

Setul de probabilitati ale starilor: $[P(S_k)] = [q_1, q_2, \dots, q_{N^k}]$ unde $q_i = p(s_i)$ si

$$\sum_i q_i = 1$$

Relatia dintre probabilitatile simbolurilor si probabilitatile starilor este (T. probabilitati totale) :

$$p(x_i) = \sum_j p(x_i / s_j) p(s_j) \Leftrightarrow p_i = \sum_j p(x_i / s_j) q_j$$

Fiecare simbol nou generat constituie, impreuna cu cele anterioare, o noua stare :

Exemplu : Sursa Markov binara de ordinul 2

$$p(1/0,0) \Leftrightarrow p(1,0/0,0)$$

Probabilitatea ca sursa sa genereze simbolul 1 cand se afla in starea 0,0 este totuna cu probabilitatea ca sursa sa treaca din starea 0,0 in starea 1,0.

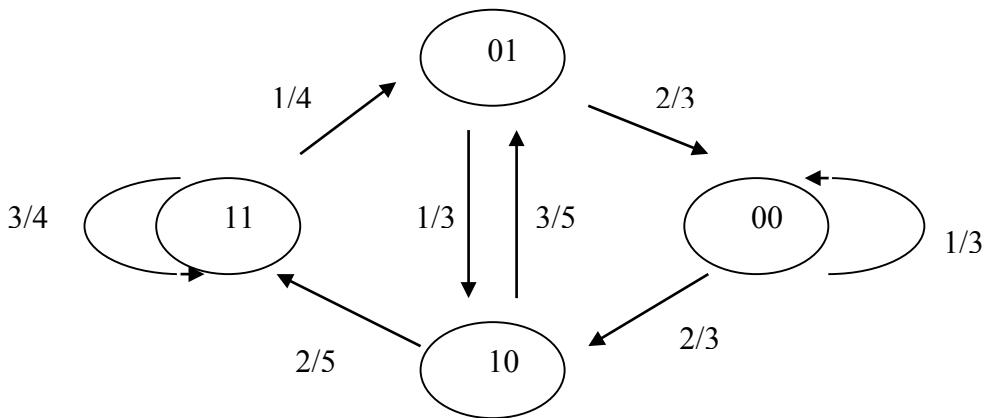
Definitia III : O sursa Markov este o sursa cu memorie la care probabilitatea de aparitie a unei stari nu depinde decat de starea anterioara.

3.3.2. Descrierea surselor Markov prin diagrame de stare

Exemplu : Sursa binara Markov de ordinul 2

$$[S] = [s_1, s_2, s_3, s_4] = [00 \quad 01 \quad 10 \quad 11]$$

$$\begin{aligned} p(0/0,0) &= p(0,0/0,0) = 1/3 & p(1/0,0) &= p(1,0/0,0) = 2/3 \\ p(0/0,1) &= p(0,0/0,1) = 2/3 & p(1/0,1) &= p(1,0/0,1) = 1/3 \\ p(0/1,0) &= p(0,1/1,0) = 3/5 & p(1/1,0) &= p(1,1/1,0) = 2/5 \\ p(0/1,1) &= p(0,1/1,1) = 1/4 & p(1/1,1) &= p(1,1/1,1) = 3/4 \end{aligned}$$



2.3.3. Descrierea surselor Markov prin matricea de tranzitie si prin vectorul probabilitatilor starilor

Definitie : Matricea de tranzitie are ca elemente probabilitatile conditionate ale sursei Markov.

$$T = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N^k} \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ p_{N^k,1} & p_{N^k,2} & \cdots & p_{N^k,N^k} \end{bmatrix}$$

unde $p_{i,j}$ este probabilitatea ca sursa sa treaca din starea s_i in starea s_j .

Proprietate: suma elementelor de pe orice linie este egală cu 1, de aceea spunem că T este o matrice stoastica.

Definitie : Vectorul probabilitatilor starilor este constituit din probabilitatile tuturor starilor:

$$P(S) = [q_1 \quad \dots \quad q_{N^k}]$$

Daca $P(S_k)$ este vectorul probabilitatilor starilor la momentul k si $P(S_{k-1})$ acelasi vector inaintea ultimei tranzitii, atunci:

$$P(S_k) = P(S_{k-1})T \quad (\text{conform Teoremei probabilitatii totale})$$

Prin tranzitivitate:

$$P(S_k) = P(S_{k-1})T = P(S_{k-2})T^2 = \dots = P(S_0)T^k$$

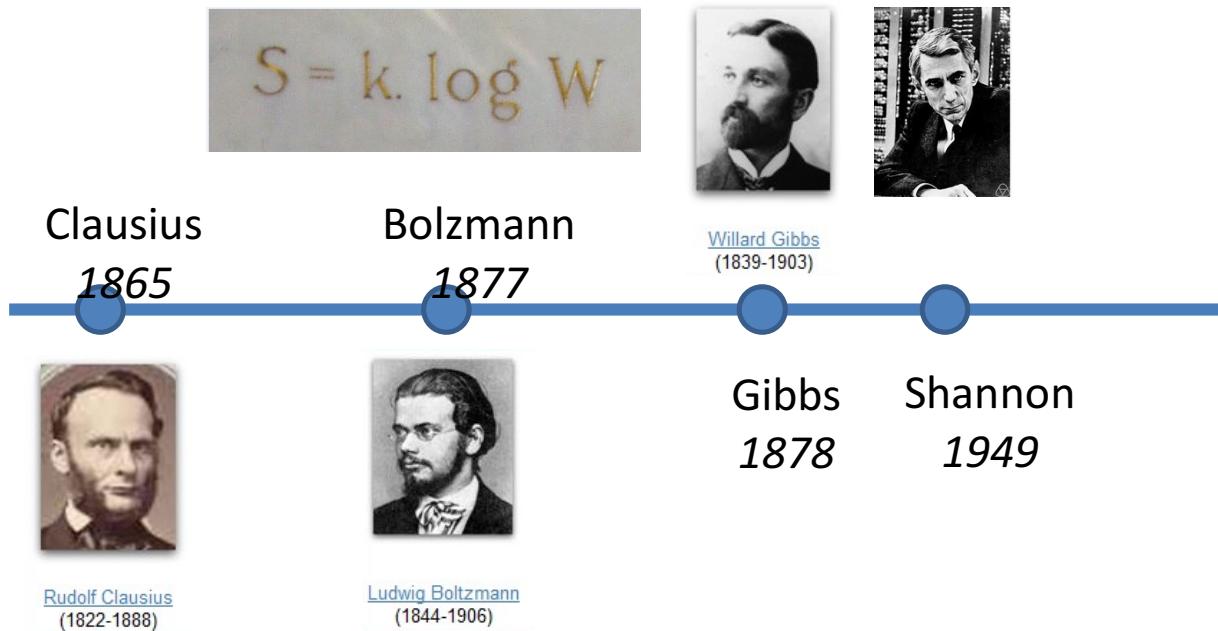
unde $P(S_0)$ este vectorul probabilitatilor in starea initiala a sursei

Definitie : Sursa Markov este **regulata** daca, atunci cand $n \rightarrow \infty$, $P(S_n)$ tinde sa fie constant. In acest caz, $P(S_n)$ se numeste **distributie de echilibru** sau asimptotica a starilor sursei Markov.

Exemplu : sursa Markov regulata (binara de ordinul 1).

$$P(S_0) = [1/3 \quad 2/3] \quad T = \begin{bmatrix} 1/4 & 3/4 \\ 1/2 & 1/2 \end{bmatrix}$$

3. ENTROPIA SURSELOR DISCRETE DE INFORMATIE



Definitie : Entropia unei surse discrete de informatie este cantitatea de informatie, medie pe simbol, generata de sursa.

4.1. Entropia sursei fara memorie

4.1.1. Expresia entropiei

Entropia unei surse fara memorie se calculeaza cu urmatoarea expresie :

$$H(X) = -\sum_i p_i \log(p_i)$$

Justificare: Fie $S : X_1, X_2, \dots, X_n$ un sir tipic de lungime n , generat de sursa. Din numararea simbolurilor de acelasi fel rezulta valorile n_1, n_2, \dots, n_N ($\sum_i n_i = n$). Sirul fiind tipic, pentru $n \gg 1$, numarul de aparitii ale unui simbol este aproximativ $n_i \approx np_i$. Deci, probabilitatea estimata de aparitie a sirului este: $p(S) = (p_1^{p_1} p_2^{p_2} \dots p_n^{p_n})^n$ si, in consecinta, informatia sirului este:

$$i(S) = -\log p(S) = -n \sum_i p_i \log(p_i)$$

iar entropia $H(X) = \frac{i(S)}{n} = -\sum_i p_i \log(p_i)$

Observatie : Aceasta expresia a entropiei este valabila si pentru sursele neergodice sau sursele nestationare. In aceste cazuri, probabilitatile se estimeaza pe un set de siruri generate de sursa.

Unitatea de masura pentru entropie : bit/simbol.

4.1.2. Proprietatile entropiei

- a) Entropia este totdeauna mai mare sau egala cu zero
- b) Continuitate: $H(X)$ este continua in raport cu variabilele p_i
- c) Simetrie: $H(X)$ este simetrica in raport cu variabilele p_i
- d) $H(X)$ este maxima cand simbolurile sursei sunt echiprobabile
- e) Aditivitate:
 - e1) compunerea simbolurilor descreste entropia
 - e2) scindarea simbolurilor creste entropia

Justificarea proprietatii d): Demonstratia se face folosind metoda multiplicatorului lui Lagrange

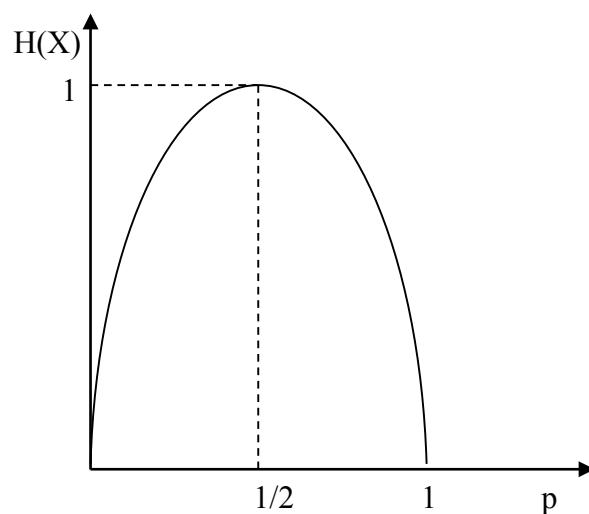
4.1.3. Entropia sursei binare

Fie alfabetul $[X] = [x_1, x_2]$ cu probabilitatile $[P] = [p \ 1-p]$

Entropia $H(X) = -p \log(p) - (1-p) \log(1-p)$

Pentru $p = 0$ sau $p = 1$, $H(X) = 0 \text{ bit/simb}$

Pentru $p = 1/2$, entropia este maxima $H(X) = 1 \text{ bit/simb}$



4.2. Entropia sursei Markov

Fie sursa Markov de ordin k, cu alfabetul :

$$[X] = [x_1, x_2, \dots, x_N]$$

si alfabetul starilor :

$$[S_k] = [s_1, s_2, \dots, s_{N^k}]$$

Definitie : Entropia sursei Markov este informatia medie pe stare, generata de sursa:

$$H(S_k) = \sum_j p(s_j) H(S_k | s_j)$$

unde $H(S_k | s_j)$ este informatia medie cand sursa se afla in starea particulara s_j :

$$H(S_k | s_j) = -\sum_i p(s_i | s_j) \log p(s_i | s_j)$$

Proprietate : Entropia sursei Markov este mai mica decat entropia unei surse fara memorie care ar genera aceleasi simboluri (dependenta de trecut diminueaza cantitatea medie de informatie pe simbol):

$$H(S_k) < H(S_0)$$

Justificare:

Demonstratia se bazeaza pe *Inegalitatea fundamentala*, enuntata mai jos. Pentru demonstratie, consideram seturile de probabilitati $p(x_i, s_j)$ si $p(x_i)p(s_j)$

Inegalitatea fundamentala (lema):

Fie $[P] = [p_1, p_2, \dots, p_N]$ cu $\sum_i p_i = 1$ si
 $[Q] = [q_1, q_2, \dots, q_N]$ cu $\sum_i q_i = 1$
doua seturi de probabilitati.

$$\text{Atunci } \sum_i p_i \log_2 \frac{q_i}{p_i} \leq 0$$

Demonstratie lema (indicatie): se porneste de la $\log_2 x \leq x - 1$ si se noteaza

$$\frac{q_i}{p_i} = x.$$

Definitie: Marimea $\sum_i p_i \log_2 \frac{p_i}{q_i}$ se numeste **entropie relativa** a doua surse de

informatie sau **distanta Kullback-Liebler**. Entropia relativa este o marime nenegativa; ea ia valoarea zero cand cele doua distributii sunt egale (se foloseste pentru a masura similaritatea a doua distributii).

4.3. Reducerea memoriei sursei Markov prin decorelare

Definitie : Decorelarea este operatia prin care un semnal numeric, modelat printr-o sursa de informatie cu memorie, este transformat intr-o sursa fara memorie (de fapt, cu memorie de lungime redusa).

Cea mai simpla metoda de decorelare este DPCM (Differential Pulse Code Modulation)

4.3.1. Cazul semnalelor 1D

Fie un semnalul numeric format din urmatoarele esantioane: $x_1, x_2, \dots, x_n, \dots$

Semanalul decorelat se obtine calculand diferența intre simbolurile consecutive:

$$x_1, x_2 - x_1, \dots, x_n - x_{n-1}, \dots$$

4.3.2. Cazul semnalelor 2D

Fie imaginea constituita din pixelii:

$$\begin{matrix} i_{1,1} & \dots & i_{1,j-1} & i_{1,j} & \dots \\ \dots & \dots & \dots & \dots & \dots \\ i_{i-1,1} & \dots & i_{i-1,j-1} & i_{i-1,j} & \dots \\ i_{i,1} & \dots & i_{i,j-1} & i_{i,j} & \dots \\ \dots & \dots & \dots & \dots & \dots \end{matrix}$$

Imaginea decorelata este constituita din pixelii diferență $d_{i,j} = i_{i-1,j} - i_{i-1,j-1} + i_{i,j-1}$:

$$\begin{matrix} i_{1,1} & \dots & i_{1,j-1} & i_{1,j} & \dots \\ \dots & \dots & \dots & \dots & \dots \\ i_{i-1,1} & \dots & d_{i-1,j-1} & d_{i-1,j} & \dots \\ i_{i,1} & \dots & d_{i,j-1} & d_{i,j} & \dots \\ \dots & \dots & \dots & \dots & \dots \end{matrix}$$

4.4. Debit, redundanta, redundanta relativa

Definitie : Debitul de informatie al unei surse este cantitatea medie de informatie generata pe secunda de sursa.

$$H_t(X) = \frac{H(X)}{\tau} \text{ unde } \tau \text{ este durata unui simbol}$$

Unitatea de masura pentru debit este *bit/sec*.

Definitie : Redundanta unei surse de informatie este:

$$R(X) = H_{\max}(X) - H(X)$$

Unde $H_{\max}(X)$ este entropia maxima a sursei (entropia in cazul simbolurilor echiprobabile) si $H(X)$ este entropia sursei.

Unitatea uzuala de masura este *bit/simbol*.

Definitie : Redundanta relativa a unei surse este

$$\rho(X) = \frac{H_{\max}(X) - H(X)}{H_{\max}(X)} \quad \rho(X) \in [0 \quad 1]$$

Redundanta relativa este adimensională.

4.5. Entropia conjugata a doua surse de informatie

Fie doua surse de informatie:

$$\begin{aligned} [X] &= [x_1, x_2, \dots, x_N] \\ [P] &= [p_1, p_2, \dots, p_N] \text{ cu } \sum_i p_i = 1 \\ \text{si} \end{aligned}$$

$$\begin{aligned} [Y] &= [y_1, y_2, \dots, y_M] \\ [Q] &= [q_1, q_2, \dots, q_M] \text{ cu } \sum_i q_i = 1 \end{aligned}$$

Definitie : Entropia conjugata (sau compusa) a surselor X si Y este

$$H(X, Y) = - \sum_i \sum_j p(x_i, y_j) \log p(x_i, y_j)$$

Observatii:

- Entropia conjugata este totdeauna pozitiva
- Unitatea uzuala de masura pentru entropia conjugata este *bit/simbol*.

Cazuri particulare :

1. Daca sursele de informatie sunt independente statistic :

$$H(X, Y) = H(X) + H(Y)$$

Demonstratia se bazeaza pe definitia v.a. independente: $p(x_i, y_j) = p(x_i)p(y_j)$

2. Daca sursele sunt identice:

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

3. Daca sursele sunt dependente statistic:

$$H(X, Y) \leq H(X) + H(Y)$$

Demonstratia se face folosind **inegalitatea fundamentală**, in cazul seturilor de probabilitati $p(x_i, y_j)$ si $p(x_i)p(y_j)$.

4.6. Informatia mutuala a doua surse

Definitie : Informatia mutuala a doua surse X si Y este media informatiilor mutuale a perechilor de simboluri (x_i, y_j) generate de surse:

$$I(X, Y) = \sum_i \sum_j p(x_i, y_j) \log \frac{p(x_i, y_j)}{p(x_i)p(y_j)}$$

Unitatea de masura pentru $I(X, Y)$ este *bit/simbol*.

Cazuri particulare :

1. Daca X si Y sunt independente:

$$I(X, Y) = 0$$

Demonstratia se bazeaza pe definitia v.a. independente: $p(x_i, y_j) = p(x_i)p(y_j)$.

2. Daca X si Y sunt identice:

$$I(X, Y) = H(X) = H(Y)$$

3. Daca X si Y sunt dependente statistic:

$$I(X, Y) \leq H(X) \text{ si } I(X, Y) \leq H(Y)$$

Proprietati:

1. $I(X, Y) = H(X) + H(Y) - H(X, Y)$

Justificare: Se calculeaza expresia din dreapta, scriindu-i pe $H(X)$ si $H(Y)$ ca functii de probabilitatile ambelor v.a. De exemplu, $p(x_i) = \sum_j p(x_i, y_j)$, conform Teoremei probabilitatii totale.

2. Informatia mutuala este o marime nenegativa: $I(X, Y) \geq 0$.

Justificare: Rezulta din proprietatea entropiei conjugate $H(X, Y) \leq H(X) + H(Y)$

Observatie: Desi informatia mutuala a doua simboluri poate fi si negativa, informatia mutuala a doua surse este totdeauna nenegativa.

4.7. Entropia conditionata a sursei de informatie

Definitie : Entropia sursei X , conditionata de sursa Y , este cantitatea medie de incertitudine care ramane asupra lui X , cand se cunoaste Y .

$$H(X|Y) = -\sum_i \sum_j p(x_i, y_j) \log p(x_i|y_j) = -\sum_i \sum_j p(y_j) p(x_i|y_j) \log p(x_i|y_j)$$

Observatie: $H(X|y_j) = -\sum_i p(x_i|y_j) \log p(x_i|y_j)$ este incertitudinea medie asupra lui X , cand Y a generat simbolul y_j . In medie, cand se cunoaste y_j , incertitudinea este:

$$H(X|Y) = \sum_j p(y_j) H(X|y_j)$$

Cazuri particulare:

1. Daca X si Y sunt independente:

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

Demonstratia se bazeaza pe definitia v.a. independente: $p(x_i, y_j) = p(x_i)p(y_j)$.

2. Daca X si Y sunt identice:

$$H(X|Y) = 0$$

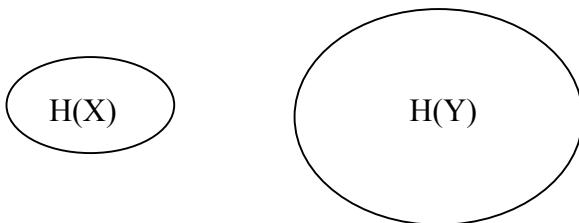
3. Daca X si Y sunt dependente statistic:

$$H(X/Y) \leq H(X)$$

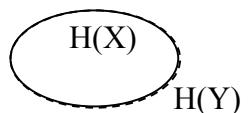
4.8. Relatii intre entropii (Diagrame Venn)

4.8.1. Reprezentarea entropiilor prin Diagrame Venn :

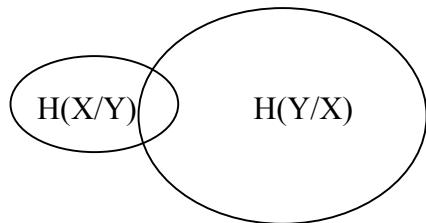
Sursele X si Y sunt independent



Sursele X si Y sunt identice



Sursele X si Y sunt dependente statistic



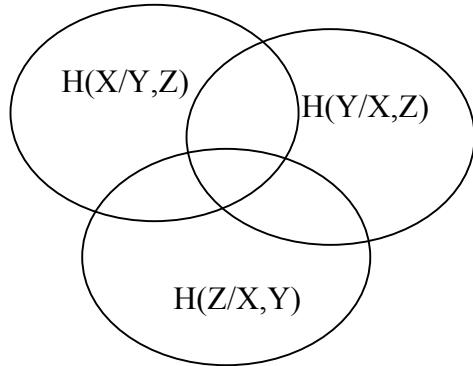
4.8.2. Relatii intre entropii

$$H(X,Y) = H(Y|X) + H(X) \quad \text{si} \quad H(X,Y) = H(X|Y) + H(Y)$$

$$H(X/Y) \leq H(X) \leq H(X,Y) \leq H(X) + H(Y)$$

4.9. Generalizare (cazul a n surse)

Diagrama Venn pentru 3 surse de informatie :



a) $H(X,Y,Z) = H(X) + H(Y|X) + H(Z|X,Y)$ (se deduce din Diagrama Venn)

unde $H(Z|X,Y) = -\sum_i \sum_j \sum_k p(x_i, y_j, z_k) \log p(z_k | x_i, y_j)$

b) $0 \leq H(Z|X,Y) \leq H(Z|X) \leq H(Z)$

Pentru n surse, prin analogie cu relatiile anterioare, putem scrie:

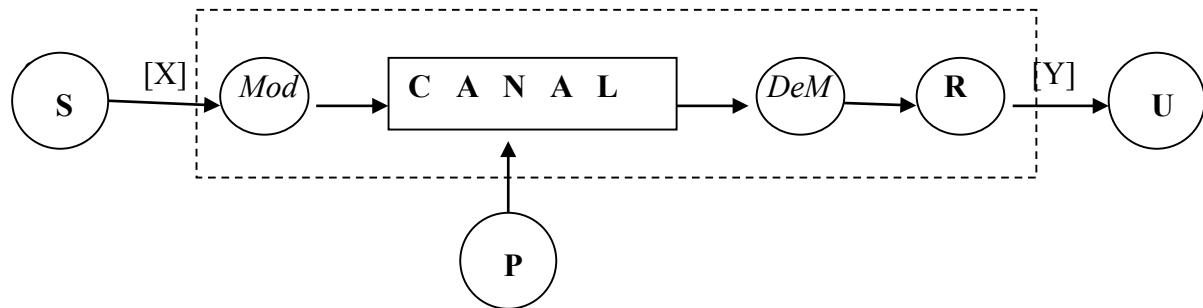
a) $H(X_1, \dots, X_n) = H(X_1) + H(X_2|X_1) + \dots + H(X_n|X_1, \dots, X_{n-1})$

Daca sursele sunt independente, atunci: $H(X_1, \dots, X_n) = \sum_i H(X_i)$

b) $0 \leq H(X_n|X_1, \dots, X_{n-1}) \leq H(X_n|X_1, \dots, X_{n-2}) \leq \dots \leq H(X_n|X_1) \leq H(X_n)$

5. CANALE DE TRANSMITERE A INFORMATIEI

Definitie : Un canal de transmitere a informatiei este constituit din mediul de transmitere si echipamentele care fac posibile transmiterea informatiei de la sursa la utilizator.



Mediul de transmisie : fire de cupru, fibrele optice, atmosfera, etc.

5.1. Clasificari ale canalelor

a) Dupa domeniul de valori al v.a. X si Y de la intrarea, respectiv iesirea canalului :

- continuu/continuu
- discret/continuu
- continuu/discret
- discret/discret

b) Dupa evolutia in timp a v.a. X si Y :

- continuu in timp
- discret in timp

c) Dupa redundanta transmisiei :

- canal fara memorie
- canal cu memorie

d) Dupa statistica trasmisiei :

- stationar
- nestationar

5.2. Canale discrete de transmitere a informatiei

Aceasta sectiune priveste canalele discrete/discrete, fara memorie si stationare. Notiunile prezentate nu depind de tipul de continuitatea in timp

5.2.1. Marimi caracteristice

Fie X , sursa de informatie care genereaza la intrarea in canal:

$$\begin{bmatrix} X \end{bmatrix} = \begin{bmatrix} x_1, \dots, x_N \end{bmatrix}$$

$$\begin{bmatrix} P \end{bmatrix} = \begin{bmatrix} p_1, \dots, p_N \end{bmatrix}$$

si Y , sursa de informatie care modeleaza iesirea din canal (sursa de informatie pentru utilizator):

$$\begin{bmatrix} Y \end{bmatrix} = \begin{bmatrix} y_1, \dots, y_M \end{bmatrix}$$

$$\begin{bmatrix} Q \end{bmatrix} = \begin{bmatrix} q_1, \dots, q_M \end{bmatrix}$$

Din cauza perturbatiilor de pe canal, X si Y sunt, in general, diferite.

Spatiul produs:

$$\begin{bmatrix} X, Y \end{bmatrix} = \begin{bmatrix} (x_1, y_1) & (x_1, y_2) & \dots & (x_1, y_M) \\ (x_2, y_1) & (x_2, y_2) & \dots & (x_2, y_M) \\ \dots & \dots & \dots & \dots \\ (x_N, y_1) & (x_N, y_2) & \dots & (x_N, y_M) \end{bmatrix}$$

Matricea probabilitatilor corespunzatoare spatiului produs:

$$\begin{bmatrix} P(X, Y) \end{bmatrix} = \begin{bmatrix} p(x_1, y_1) & p(x_1, y_2) & \dots & p(x_1, y_M) \\ p(x_2, y_1) & p(x_2, y_2) & \dots & p(x_2, y_M) \\ \dots & \dots & \dots & \dots \\ p(x_N, y_1) & p(x_N, y_2) & \dots & p(x_N, y_M) \end{bmatrix}$$

Matricea de zgomot a canalului:

$$\begin{bmatrix} P(Y|X) \end{bmatrix} = \begin{bmatrix} p(y_1|x_1) & p(y_2|x_1) & \dots & p(y_M|x_1) \\ p(y_1|x_2) & p(y_2|x_2) & \dots & p(y_M|x_2) \\ \dots & \dots & \dots & \dots \\ p(y_1|x_N) & p(y_2|x_N) & \dots & p(y_M|x_N) \end{bmatrix}$$

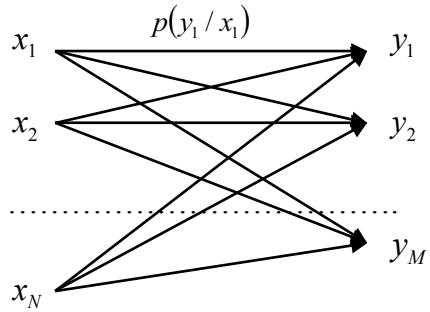
Matricea de zgomot este stohastica:

$$\sum_i p(y_i|x_j) = 1 \text{ (suma elementelor de pe orice linie este 1).}$$

Canalele studiate sunt stationare, deci $p(y_j/x_i) = ct.$ in timp.

Canalele sunt fara memorie, deci probabilitatea de aparitie a lui y_j nu depinde decat de simbolul generat simultan la intrarea in canal, simbolul x_i pentru $p(y_j/x_i)$.

5.2.2. Reprezentarea grafica a transmisiei prin canalele discrete



5.2.3. Entropii caracteristice

Entropia la intrarea in canal: $H(X) = -\sum_i p(x_i) \log p(x_i)$

Entropia la iesirea din canal: $H(Y) = -\sum_i p(y_i) \log p(y_i)$

Entropia reunita a intrarii si iesirii: $H(X, Y) = -\sum_i \sum_j p(x_i, y_j) \log p(x_i, y_j)$

Echivocatia: $H(X|Y) = -\sum_i \sum_j p(x_i, y_j) \log p(x_i|y_j)$

Definitie: Echivocatia este cantitatea medie de incertitudinea care ramane asupra simbolurilor de la intrarea in canal, atunci cand se cunosc simbolurile de la iesire.

Eroarea medie: $H(Y|X) = -\sum_i \sum_j p(x_i, y_j) \log p(y_j|x_i)$

Definitie: Eroarea medie este cantitate medie de informatie eronata, la iesirea din canal.

Informatia medie transmisa prin canal: $I(X, Y) = \sum_i \sum_j p(x_i, y_j) \log \frac{p(x_i, y_j)}{p(x_i)p(y_j)}$

Definitie: Informatia medie este cantitatea medie de informatie care se transmite corect prin canal.

Cazuri particulare :

- a) Canale cu perturbatii infinite (X si Y sunt independente)

$$H(X, Y) = H(X) + H(Y)$$

$H(X|Y) = H(X)$ (la iesire, nu aflam nimic despre X ; incertitudinea asupra lui X ramane la fel de mare)

$$H(Y|X) = H(Y) \text{ (toata informatia de la iesire este eronata)}$$

$$I(X, Y) = 0 \text{ (informatia medie transmisa prin canal este nula)}$$

- b) Canale fara perturbatii (sursele X si Y sunt identice)

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

$H(X|Y) = 0$ (cunoscand iesirea din canal, nu mai exista nicio incertitudine asupra lui X)

$$H(Y|X) = 0 \text{ (nu exista erori la iesirea din canal)}$$

$$I(X, Y) = H(X) \text{ (informatia de la intrare se transmite integral prin canal)}$$

- c) Canale cu perturbatii finite (X si Y sunt diferite, dar dependente statistic)

$$H(X, Y) < H(X) + H(Y)$$

$H(X|Y) < H(X)$ (cunoscand iesirea din canal, incertitudine asupra lui X devine mai mica)

$$H(Y|X) < H(Y) \text{ (o parte a informatiei de la iesirea din canal este corecta)}$$

$$I(X, Y) < H(X) \text{ (informatia de la intrare se transmite partial prin canal)}$$

5.3. Capacitatea canalului discret

Definitie : Capacitatea unui canal este informatia medie *maxima*, care se poate transmite prin canal :

$$C = \max_{[P(X)]} I(X, Y)$$

Observatie: maximul se ia dupa probabilitatile sursei de la intrarea in canal, pentru ca numai aceste probabilitati pot fi controlate intr-o transmisie prin un canal cu perturbatii.

Unitatea de masura pentru capacitate este bit/simbol.

Definitie : Redundanta canalului este : $R = C - I(X, Y)$ [bit/simbol].

Definitie : Redundanta relativa a canalului este : $\rho_C = 1 - \frac{I(X, Y)}{C} \in [0,1]$.

Definitie : Randamentul sau eficienta canalului arata cat de mica este cantitatea medie de informatie transmisa prin canal, in raport cu capacitatea canalului.

$$\eta_C = \frac{I(X, Y)}{C} \in [0,1]$$

Observatie: redundanta relativa si randamentul sunt marimi complementare: $\rho_C = 1 - \eta_C$

Definitie : Debitul de informatie prin canal este : $I_t(X, Y) = \frac{I(X, Y)}{\tau}$ [bit/sec], unde τ este durata unui simbol.

Observatie: **Debitul maxim** de informatie prin canal este: $C_t = \frac{C}{\tau}$.

Proprietati :

a) Capacitatea canalului este o marime nenegativa :

$$C \geq 0 \text{ (deoarece } I(X, Y) \geq 0\text{)}$$

b) $C \leq H(X)$ (deoarece $I(X, Y) \leq H(X)$)

c) Capacitatea este o functie continua in raport probabilitatile $[P(X)]$.

5.4. Calculul capacitati canalului discret

Date initiale : Matricea de zgomot $[P(Y/X)]$.

Etape:

- 1) Aplicand *Metoda multiplicatorului lui Lagrange*, se calculeaza probabilitatile p_i^{\max} , care maximizeaza functia $I(X, Y) = H(Y) - H(Y/X)$.
- 2) Capacitatea se obtine calculand $I(X, Y) = H(Y) - H(Y/X)$ pentru probabilitatile obtinute.

Rezolvare:

Se construieste functia:

$$\Phi = H(Y) - H(Y/X) + \lambda \left(\sum_i p_i - 1 \right)$$

Pentru a pune in evidenta probabilitatile p_i in expresia lui $H(Y)$, probabilitatile $[Q]$ se scriu:

$$q_j = \sum_i p(x_i, y_j) = \sum_i p(y_j | x_i) p(x_i) = \sum_i p(y_j | x_i) p_i$$

Se calculeaza derivatele partiale ale lui Φ in raport cu p_i :

- derivata lui $H(Y)$ in raport cu p_i :

$$\begin{aligned} \frac{\partial H(Y)}{\partial p_i} &= \sum_j \frac{\partial H(Y)}{\partial q_j} \frac{\partial q_j}{\partial p_i} = - \sum_j \left(\log q_j + \frac{1}{\log e} \right) p(y_j / x_i) = \\ &= - \frac{1}{\log e} \sum_j p(y_j / x_i) - \sum_j p(y_j / x_i) \log q_j = - \frac{1}{\log e} - \sum_j p(y_j / x_i) \log q_j \end{aligned}$$

- derivata lui $H(Y/X)$ in raport cu p_i :

$$\frac{\partial H(Y/X)}{\partial p_i} = - \frac{\partial \sum_j \sum_i p(y_j / x_i) p_i \log p(y_j / x_i)}{\partial p_i} = - \sum_j p(y_j / x_i) \log p(y_j / x_i)$$

- derivata termenului in λ :

$$\frac{\partial \lambda \left(\sum_i p_i - 1 \right)}{\partial p_i} = \lambda$$

Se egaleaza derivatele partiale ale lui Φ cu zero; din rezolvarea sistemului, rezulta probabilitatile p_i^{\max} , care maximizeaza Φ si, deci, informatia transmisa prin canal:

$$- \frac{1}{\log e} - \sum_j p(y_j / x_i) \log q_j + \sum_j p(y_j / x_i) \log p(y_j / x_i) + \lambda = 0 \quad \text{pentru } i = 1, N$$

Grupand termenii cu sume si constantele, se obtin ecuatiile:

$$\sum_j p(y_j / x_i) \log \frac{p(y_j / x_i)}{q_j} = ct \quad \text{pentru } i = 1, N$$

Compleinand aceste ecuatii cu:

$$q_j = \sum_i p(y_j / x_i) p_i \quad \text{pentru } j = 1, M$$

si

$$\sum_i p_i = 1$$

se obtine un sistem cu $N + M + 1$ ecuatii si acelasi numar de necunoscute (p_i , q_j si λ), din care se pot obtine probabilitatile p_i^{\max} si q_j^{\max} , care maximizeaza informatia transmisa prin canal.

Capacitatea canalului se calculeaza cu relatia:

$$C = \sum_i \sum_j p(y_j / x_i) p_i^{\max} \log \frac{p(y_j / x_i)}{q_j^{\max}}$$

Observatii:

- acest sistem nu are, in general, o solutie analitica; cand nu exista o solutie analitica, capacitatea se calculeaza cu metode numerice (algoritmullui Frank-Wolfe, care este bazat pe metoda gradientului, sau algoritmul iterativ al lui Arimoto si Blahut)
- daca alfabetele surselor de la intrarea si de la iesirea din canal au acelasi numar de simboluri si, daca, determinantul matricii de zgomot este diferit de zero, atunci sistemul are solutie analitica

5.5. Modele de canale discrete

Aceasta sectiune cuprinde patru cazuri particulare de canale (modele), pentru care capacitatea se poate calcula analitic.

5.5.1. Canalul uniform fata de intrare

Definitie: Fiecare linie a matricii de zgomot a canalului uniform fata de intrare este o permutare a altiei linii (pe fiecare linie gasim aceleasi probabilitati, diferit ordonate).

Exemplu:

$$P(Y / X) = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 2 & 3 & 6 \\ \frac{1}{6} & \frac{1}{3} & \frac{1}{2} \end{bmatrix}$$

Proprietati:

- Eroarea medie nu depinde de probabilitatile simbolurilor de la intrarea in canal:

$$\begin{aligned}
H(Y/X) &= -\sum_{i,j} p(x_i, y_j) \log p(y_j/x_i) = \\
&= -\sum_{i,j} p(y_j/x_i) p(x_i) \log p(y_j/x_i) = \\
&= -\sum_i p(x_i) \sum_j p(y_j/x_i) \log p(y_j/x_i) = -ct \sum_i p(x_i) = -ct
\end{aligned}$$

- b) Eroarea medie la iesirea din canal se poate calcula considerand numai probabilitatile de pe o linie a matricii de zgromot:

$$\begin{aligned}
H(Y/X) &= -\sum_{i,j} p(x_i, y_j) \log p(y_j/x_i) = \\
&= -\sum_i p(x_i) \sum_j p(y_j/x_i) \log p(y_j/x_i) = \\
&= -\sum_j p(y_j/x_i) \log p(y_j/x_i)
\end{aligned}$$

- c) Capacitatea canalului este:

$$C = \max_{[P]} I(X, Y) = \max_{[P]} [H(Y) - H(Y/X)] = \max_{[P]} H(Y) - H(Y/X)$$

5.5.2. Canalul uniform fata de iesire

Definitie: Fiecare coloana a matricii de zgromot a canalului uniform fata de iesire este o permutare a altrei coloane (pe fiecare coloana gasim aceleasi probabilitati, diferit ordonate).

Exemplu:

$$P(Y/X) = \begin{bmatrix} 0,5 & 0,5 \\ 0,3 & 0,7 \\ 0,7 & 0,3 \end{bmatrix}$$

Proprietate:

- a) Daca simbolurile de la intrarea in canal sunt echiprobabile, atunci si cele de la iesire sunt echiprobabile:

$$p(y_j) = \sum_i p(y_j/x_i) p(x_i) = \frac{1}{N} \sum_i p(y_j/x_i) = \frac{1}{N} ct.$$

5.5.3. Canalul simetric

Definitie: Canalul simetric este canalul uniform atat fata de intrare cat si fata de iesire.

Exemplu:

$$P(Y/X) = \begin{bmatrix} 0,3 & 0,2 & 0,5 \\ 0,5 & 0,3 & 0,2 \\ 0,2 & 0,5 & 0,3 \end{bmatrix}$$

Proprietati:

- Capacitatea canalului se obtine pentru simboluri echiprobabile la intrarea in canal si este:

$C = \log_2 M - H(Y/X)$ unde M este numarul de simboluri ale sursei de la iesirea din canal (simbolurile de la iesire sunt echiprobabile, daca si cele de la intrare sunt echiprobabile).

5.5.4. Canalul slab simetric

Definitie: Canalul slab simetric este uniform fata de intrare si are suma probabilitatilor de pe fiecare coloana constanta.

Exemplu:

$$P(Y/X) = \begin{bmatrix} \frac{1}{3} & \frac{1}{6} & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & \frac{1}{6} \\ \frac{1}{3} & \frac{1}{6} & \frac{1}{2} \end{bmatrix}$$

Proprietati:

- Daca simbolurile de la intrarea in canal sunt echiprobabile, atunci si cele de la iesire sunt echiprobabile:

$$p(y_j) = \sum_i p(y_j/x_i)p(x_i) = \frac{1}{N} \sum_i p(y_j/x_i) = \frac{1}{N} ct.$$

- Capacitatea canalului se obtine pentru simboluri echiprobabile la intrarea in canal si este:

$$C = \log_2 M - H(Y/X)$$

Observatie: Uniformitatea fata de iesire nu este indispensabila pentru a putea avea o expresie analitica pentru capacitatea canalului. Aceasta conditie poate fi relaxata la conditia ca suma probabilitatilor de pe coloane sa fie constanta.

5.6. Exemple de canale discrete

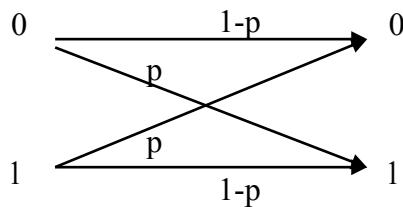
5.6.1. Canalul binar simetric

Matrice de zgomot:

$$P(Y/X) = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}$$

p este probabilitatea de transmisie eronata a unui simbol.

Reprezentare grafica:



Calculul capacitatii:

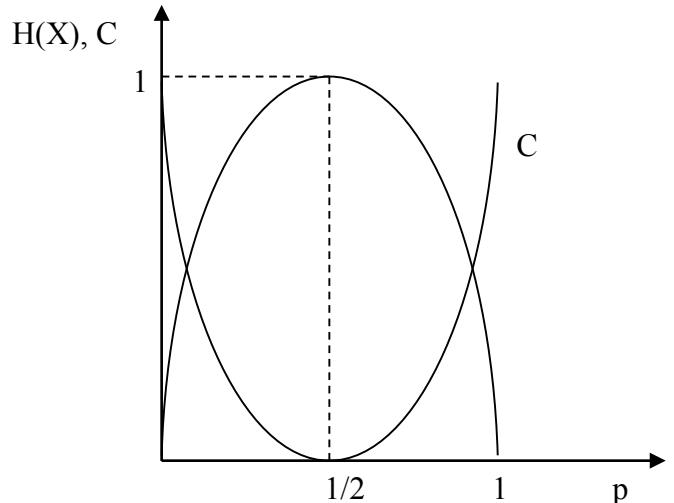
$$C = \log 2 - H(Y/X) = 1 - H(Y/X)$$

unde

$$\begin{aligned} H(Y/X) &= -\sum_{j=1}^2 p(y_j/x_i) \log p(y_j/x_i) = \\ &= -p \log p - (1-p) \log(1-p) \end{aligned}$$

deci

$$C = 1 + p \log p + (1-p) \log(1-p)$$



Cazuri particulare:

a) Canal fara perturbatii:

Matricea de zgomot: $P(Y/X) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Reprezentare grafica: $0 \xrightarrow{\hspace{2cm}} 0$

$1 \xrightarrow{\hspace{2cm}} 1$

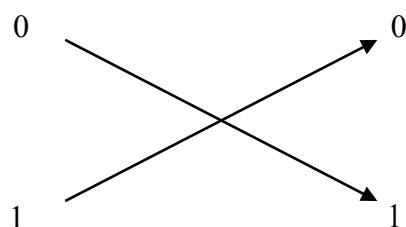
Capacitatea este maxima : $C = 1 \text{ bit/simbol}$

Observatie:

Celalalt punct de maxim al capacitatii corespunde canalului inversor:

$$P(Y/X) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$C = 1 \text{ bit/simbol}$



b) Canalul cu perturbatii infinite (foarte puternice)

Matricea de zgomot: $P(Y/X) = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}$

Capacitatea : $C = 0 \text{ bit/simbol}$

5.6.2. Canalul binar cu erori si anulari

Matrice de zgomot:

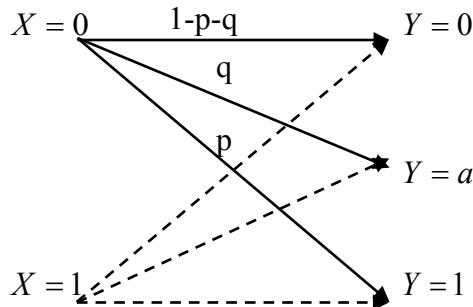
$$P(Y/X) = \begin{bmatrix} 1-p-q & p & q \\ p & 1-p-q & q \end{bmatrix}$$

Canalul este uniform doar fata de intrare.

p este probabilitatea de transmisie eronata.

q este probabilitatea ca simbolul receptionat sa fie anulat.

Reprezentare grafica:



Calculul capacitatii:

Canalul este uniform fata de intrare:

$$C = \max_{[P]} [H(Y)] - H(Y/X)$$

unde

$$\begin{aligned} H(Y/X) &= -\sum_{j=1}^2 p(y_j/x_i) \log p(y_j/x_i) = \\ &= -p \log p - q \log q - (1-p-q) \log(1-p-q) \end{aligned}$$

Calculul lui $\max_{[P]} [H(Y)]$:

- se noteaza $p(X=0) = x$ si $p(X=1) = 1-x$
- se exprima $p(Y=0) = \sum_{i=1}^2 p(Y=0/x_i)p(x_i)$, $p(Y=a) = \dots$ si $p(Y=1) = \dots$ ca functii de x
- se exprima $H(Y)$ ca functie de x , folosind probabilitatile calculate mai sus
- se rezolva ecuatia $\frac{\partial H(Y)}{\partial x} = 0$
- cu solutia ecuatiei de mai sus, se obtine $\max_{[P]} [H(Y)]$

Exercitiu:

Calculul capacitatii canalului binar cu erori si anulari.

$$Raspuns : C = 1 - q - (1-q) \log(1-q) + p \log p + (1-p-q) \log(1-p-q).$$

Observatie: Capacitatea canalului devine zero pentru $p = \frac{1-q}{2}$ (se rezolva ecuatia

$$\frac{\partial C}{\partial p} = 0 \text{ si se obtine solutia } p = \frac{1-q}{2}$$

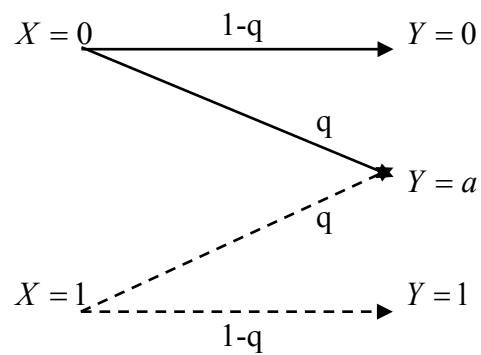
5.6.3. Canalul binar cu anulari

Este un caz particular al canalului binar cu erori și anulari ($p = 0$).

Matricea de zgromot:

$$P(Y/X) = \begin{bmatrix} 1-q & 0 & q \\ 0 & 1-q & q \end{bmatrix}$$

Reprezentarea grafică:



Capacitatea: $C = 1 - q$

6. SURSE DE INFORMATIE SI CANALE CONTINUE

6.1. Entropia sursei de informatie continue

Definitie : Sursa continua de informatie este un mecanism de generare a unui sir de v.a. continue ..., $X_{k-1}, X_k, X_{k+1}, \dots$, unde k este, de cele mai multe ori, un indice de timp.

X_k sunt v.a. continue, care iau valori in R .

X_k poate fi si complexa (de exemplu, cand prin Transformare Fourier, s-a trecut in domeniul frecventelor).

V. a. continue X_k sunt caracterizate de o densitatea de probabilitate $f(x)$

Definitie : Shannon a definit entropia unei surse de informatie continua ca fiind:

$$H(X) = - \int_R f(x) \log_2 f(x) dx$$

Observatii:

Aceasta marime, desi este o extindere a entropiei surselor discrete la cazul continuu, nu are nici aceeasi semnificatie, nici aceleasi proprietati. Ea este cunoscuta sub numele de **entropie diferentiala**.

Proprietatea de pozitivitate nu mai exista. $H(X)$ poate poata lua si valori negative (matematic, acest lucru se intampla pentru ca $f(x)$ poate fi > 1).

Exemplu: v.a. X cu distributie uniforma pe intervalul $[0 \quad 1/2]$

$$f(x) = \begin{cases} 2 & x \in [0 \quad 1/2] \\ 0 & \text{in_rest} \end{cases}$$

$$H(X) = - \int_0^{1/2} 2 \log_2 2 dx = - 2 \int_0^{1/2} dx = -1$$

6.1.1. Semnificatia entropiei diferențiale

Faptul ca $H(X)$ poate fi si negativa pune un semn de intrebare asupra semnificatiei acestei marimi, in cazul surselor de informatie continue.

Fie un semnal continuu, care este esantionat cu frecventa $2W$, unde W este frecventa maxima din spectrul semnalului (criteriul lui Nyquist). Aceasta ipoteza nu reduce generalitatea rezultatelor urmatoare deoarece un semnal continuu poate fi reconstruit identic din esantioanele sale daca acestea au o frecventa $\geq 2W$.

Semnalul esantionat, fiind continuu in amplitudine, poate fi modelat printr-un sir de v.a. continue X_k cu distributia $f(x)$. Altfel spus, el este o sursa de informatie continua, pe care o vom nota, in continuare, cu X . Pp. pentru simplitatea demonstratiei, ca realizarile particulare ale lui X_k sunt cuprinse in intervalul $[0 \quad Nq]$, unde q este un numar pozitiv, iar N un numar natural.

Prin cuantizare uniforma cu un pas de cuantizare egal cu q , toate valorile semnalului cuprinse intr-un anumit interval de decizie, avand latimea q , devin egale cu nivelul de cuantizare al intervalului (altfel spus, semnalul continuu este discretizat) :

daca $x_t \in [(n-1)q \quad nq]$ atunci $x_t \Rightarrow x_n = nq$ (cu x_t s-a notat realizarea particulara a lui X_t la momentul t).

Semnalul discretizat poate fi modelat de un sir de v.a. discrete $X_k^{(q)}$, altfel spus, sursa de informatie continua devine o sursa discreta $X^{(q)}$.

V.a. $X_k^{(q)}$ iau valori in multimea $[X] = [x_1, x_2, \dots, x_N]$ unde $x_n = nq$.

Multimea probabilitatilor sursei discrete este constituita din urmatoarele valori:

$$p(x_n) = \int_{(n-1)q}^{nq} f(x)dx \approx qf(nq) \quad (\text{aproximarea este valabila pentru pasi de quantizare foarte mici in comparatie cu domeniul de valori al v.a. continue})$$

Entropia sursei discrete este:

$$H(X^q) = -\sum_{n=1}^N p(x_n) \log p(x_n) = -\sum_{n=1}^N qf(nq) \log(qf(nq))$$

Prelucrand relatia entropiei, obtinem :

$$H(X^q) = -\log q \sum_{n=1}^N qf(nq) - \sum_{n=1}^N qf(nq) \log(f(nq))$$

La limita, cand cuanta q tinde catre zero :

$$f(nq) \rightarrow f(x) \text{ si } q \rightarrow dx$$

si relatia entropie devine:

$$H(X^q) = -\log q \int f(x)dx - \int f(x) \log f(x)dx = -\log q + H(X)$$

Concluzie:

- a) $H(X^q)$ este entropia unei surse de informatie discrete, deci are semnificatia unei informatii medii. La limita, cand $q \rightarrow 0$, sursa devine continua si $\lim_{q \rightarrow 0} H(X^{(q)})$ este informtia medie a sursei continue, ceea ce nu este acelasi lucru cu $H(X)$ din cauza termenului $-\log q$. Deci, **entropia diferentiala nu are semnificatia unei cantitati medii de informatie.**
- b) La limita, termenul $-\log q$ tinde catre infinit, ceea ce arata ca **informatia medie a sursei continue este infinita** (in timp ce entropia sa $H(X)$ este de cele mai multe ori o marime finita).

6.1.2. Inegalitatea fundamentala in cazul distributiilor continue

Fie $f(x)$ si $g(x)$ doua densitati de probabilitate.

Se poate arata, cu acelasi demers logic ca in cazul distributiilor discrete, ca:

$$\int_R f(x) \log \frac{g(x)}{f(x)} \leq 0$$

$\int_R f(x) \log \frac{f(x)}{g(x)}$ se numeste **entropie relativă** sau **distanța Kullback-Leibler** in cazul

distributiilor continue. Este o marime nenegativa; ia valoarea zero cand cele doua distributii sunt indentice.

Observatie:

1. In continuu, entropia diferentiala isi pastreaza semnificatia din cazul discret, unde ea reprezinta numarul mediu de extrabiti necesari in cazul in care desi distributia de probabilitati este $p(x_i)$, numarul de biti cu care se face codarea simbolurilor este $\log_2 q_i$. De aici provine si numele de **entropie diferentiala**.

Entropia diferentiala este o marime nenegativa; ia valoarea zero cand cele doua distributii sunt indentice.

6.1.3. Cazuri de entropie maxima

Maximul absolut al entropiei surselor continue este infinit. Ne intereseaza maximul in anumite conditii restrictive.

- a) V.a. ia valori intr-un domeniu finit $[a \quad b]$

Se cauta maximul lui $H(X) = -\int_a^b f(x) \log_2 f(x) dx$ cu restrictia $\int_a^b f(x) dx = 1$

Indicatie: Se foloseste metoda multiplicatorului lui Lagrange; se construieste functia $\Phi = H(x) + \lambda \left(\int_a^b f(x)dx - 1 \right)$ si se deriveaza in raport cu f .

Rezultat: distributia care maximizeaza entropia este **distributia uniforma**.

$$f(x) = \begin{cases} 1/(b-a) & x \in [a \quad b] \\ 0 & \text{in_rest} \end{cases}$$

$$H_{\max}(X) = \log(b-a)$$

b) V.a. ia numai valori pozitive si are media statistica m

Se cauta maximul lui $H(X) = - \int_0^\infty f(x) \log_2 f(x) dx$ cu restrictiile $\int_0^\infty f(x)dx = 1$ si media statistica m .

Indicatie: Se foloseste metoda multiplicatorului lui Lagrange; se construieste functia:

$$\Phi = H(x) + \lambda \left(\int_0^\infty f(x)dx - 1 \right) + \mu \left(\int_0^\infty xf(x)dx - m \right)$$

Rezultat: distributia care maximizeaza entropia este **distributia exponentiala**.

$$f(x) = \begin{cases} me^{-mx} & x \geq 0 \\ 0 & \text{in_rest} \end{cases}$$

$$H_{\max}(X) = \log m + \frac{m}{\log e}$$

c) V.a. ia valori pe R si are media statistica 0 si varianta σ^2 .

Se cauta maximul lui $H(X) = - \int_{-\infty}^\infty f(x) \log_2 f(x) dx$ cu restrictiile $\int_{-\infty}^\infty f(x)dx = 1$, media statistica $m = 0$ si varianta σ^2 .

Indicatie: Se foloseste metoda multiplicatorului lui Lagrange; se construieste functia:

$\Phi = H(x) + \lambda \left(\int_0^\infty f(x)dx - 1 \right) + \mu \left(\int_0^\infty xf(x)dx \right) + \eta \left(\int_0^\infty x^2 f(x)dx - \sigma^2 \right)$ si se deriveaza in raport cu f .

Rezultat: distributia care maximizeaza entropia este **distributia gaussiana**:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}$$

$$H_{\max}(X) = \log(\sigma\sqrt{2\pi}e)$$

6.1.4. Variatia entropiei cu schimbarea spatiului de reprezentare a semnalului

Fie un semnal continuu, modelat printr-un sir de v.a. continue X_1, \dots, X_N , dependente statistic (cazul majoritatii semnalelor intalnite in practica), unde, de multe ori, indicele este un indice de timp. Altfel spus, fie o sursa de informatie continua, cu memorie.

Printr-o transformare \mathcal{F} (de exemplu, Fourier), se trece din spatiul N-dimensional al esantioanelor temporale, intr-un alt spatiu spatiul N-dimensional al esantioanelor frecventiale, daca am aplicat Transformarea Fourier. In acest spatiu, semnalul este reprezentat prin sirul de v.a.: V_1, \dots, V_n .

$$V_1, \dots, V_n = \mathcal{F}([X_1, \dots, X_N])$$

Pp. ca densitatile de probabilitate conjugate ale celor doua siruri de v.a. sunt :

$$f(X_1, \dots, X_N) \text{ in spatiul esantioanelor temporale}$$

si

$$g(V_1, \dots, V_N) \text{ in spatiul esantioanelor frecventiale.}$$

Probabilitatile ca sirurile sa aiba realizari particulare foarte apropiate de sirurile de valori :

$$x_1, \dots, x_N \text{ si } v_1, \dots, v_N$$

sunt

$$f(x_1, \dots, x_N) dx_1 \dots dx_N \text{ si } g(v_1, \dots, v_N) dv_1 \dots dv_N$$

Variatiile $dx_1 \dots dx_N = dX$ determina variatiile $dv_1 \dots dv_N = dV$, deoarece la acestea din urma s-a ajuns prin relatia matematica a transformarii \mathcal{F} .

Se poate arata ca $\frac{dV}{dX} = \left| J\left(\frac{V}{X}\right) \right|$ unde cu $J\left(\frac{V}{X}\right)$ s-a notat jacobianul transformarii:

$$J\left(\frac{V}{X}\right) = \begin{bmatrix} \frac{dV_1}{dx_1} & \dots & \frac{dV_N}{dx_1} \\ \dots & \dots & \dots \\ \frac{dV_{N1}}{dx_N} & \dots & \frac{dV_N}{dx_N} \end{bmatrix}$$

Cum transformarea F este determinista, urmatoarea relatie este satisfacuta :

$$f(x_1, \dots, x_N) dx_1 \dots dx_N = g(v_1, \dots, v_N) dv_1 \dots dv_N$$

Impartind relatia prin $dx_1 \dots dx_N$, se obtine:

$$f(x_1, \dots, x_N) = g(v_1, \dots, v_N) \left| J\left(\frac{V}{X}\right) \right|$$

ceea ce conduce la urmatoarea relatie intre entropiile semnalului inainte si dupa transformare:

$$\begin{aligned} H(X) &= - \int_X f(x_1, \dots, x_N) \log f(x_1, \dots, x_N) dx_1 \dots dx_N = \\ &= - \int_X f(x_1, \dots, x_N) \log g(v_1, \dots, v_N) \left| J\left(\frac{V}{X}\right) \right| dx_1 \dots dx_N = \\ &= - \int_X f(x_1, \dots, x_N) \log \left| J\left(\frac{V}{X}\right) \right| dx_1 \dots dx_N - \int_V g(v_1, \dots, v_N) \log g(v_1, \dots, v_N) dv_1 \dots dv_N = \\ &= - \int_X f(x_1, \dots, x_N) \log \left| J\left(\frac{V}{X}\right) \right| dx_1 \dots dx_N + H(V) \end{aligned}$$

ceea ce arata ca, in general, entropia semnalului se schimba atunci cand se aplica o transformare.

Se poate arata insa ca, in cazul unei transformari ortogonale (Fourier, Cosinus, etc.) :

$$\left| J\left(\frac{V}{X}\right) \right| = 1$$

si atunci

$$H(X) = H(V)$$

Concluzie: O transformare ortogonală nu schimba entropia unui semnal.

EXEMPLU de aplicatie: Separarea surselor folosind Analiza in Componente Independente

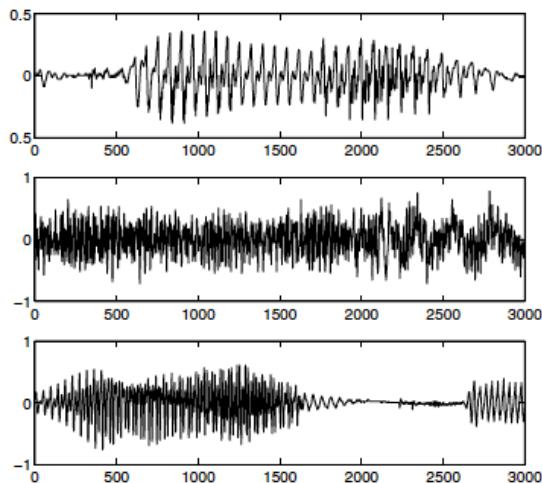


Fig. 7.1 The original audio signals.

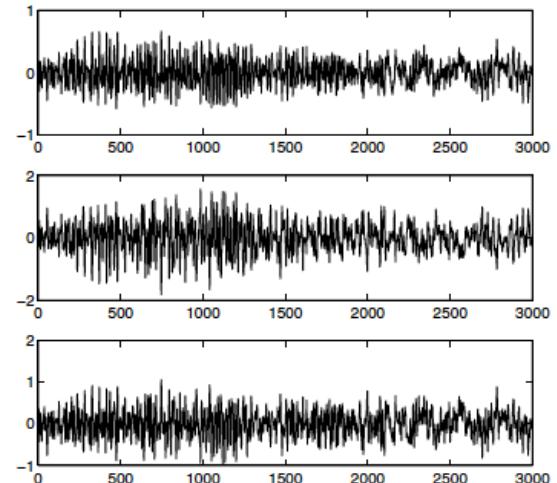


Fig. 7.2 The observed mixtures of the original signals in Fig. 7.1.

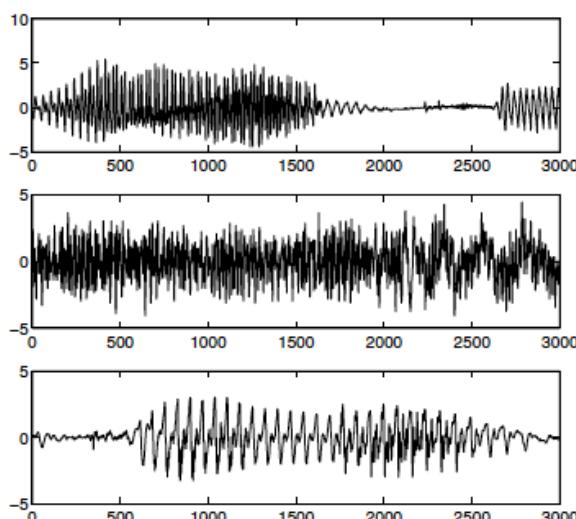
Sursele sunt semnale independente statistic. Amestecul se obtine prin combinatii liniare:

$$\begin{aligned}x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) + a_{13}s_3(t) \\x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t) + a_{23}s_3(t) \\x_3(t) &= a_{31}s_1(t) + a_{32}s_2(t) + a_{33}s_3(t)\end{aligned}$$

X este matricea combinatiilor, **S** este matricea surselor si **A** este matricea de amestec.

$$\mathbf{x} = \mathbf{As}$$

A si s sunt necunoscute, x este cunoscuta. Ecuatia este nedeterminata. Sistemul se poate rezolva adaugand ca data suplimentara, **independenta surselor**.



Entropia diferentiala:

$$H(\mathbf{y}) = - \int p_y(\boldsymbol{\eta}) \log p_y(\boldsymbol{\eta}) d\boldsymbol{\eta}$$

Negentropia:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y})$$

Independenta \Leftrightarrow maximizarea negentropiei.

Algoritmul FastICA maximizeaza cu o aproximatie a negentropiei

$$J(y) \approx \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} \text{kurt}(y)^2$$

Kurtosis este o statistica de ordin 4:

$$\text{kurt}(y) = E\{y^4\} - 3(E\{y^2\})^2$$

DEMO: http://cnl.salk.edu/~tewon/Blind/blind_audio.html

6.2. Canale continue de transmisie a informatiei

Prin un canal continuu, trec semnale continue in amplitudine. De aceea, intrarea si iesirea canalului sunt modelate prin doua surse continue de informatie.

In acest capitol, se studiaza canalele continue fara memorie (esantioanele semnalului continuu in amplitudine sunt independente) si stationare (distributia valorilor esantioanelor nu se schimba in timp).

Fie X sursa continua de la intrare, cu densitatea de probabilitate $f_X(x)$
 Y sursa continua de la iesire, cu densitatea de probabilitate $f_Y(y)$

6.2.1. Informatia medie in canalele continue

Pentru a deduce informatia medie transmisa prin canalul continuu, vom porni de la rezultatul obtinut pentru canalul discret si, prin trecere la limita, vom obtine informatia medie in canalul continuu.

Pp ca semnalul de la intrare este esantionat cu frecventa $2W$, unde W este frecventa maxima din spectrul semnalului (criteriul lui Nyquist). Aceasta ipoteza nu reduce generalitatea rezultatelor urmatoare deoarece un semnal continuu poate fi reconstruit identic din esantioanele sale daca acestea au o frecventa $\geq 2W$.

Pp., de asemenea, ca semnalul este cuantizat cu cuanta q . Rezultatul este un semnal discret care poate fi modelat printr-o sursa de informatie discreta avand alfabetul:

$$[X] = [x_1, x_2, \dots, x_N] \text{ unde } x_n = nq$$

si probabilitatile:

$$[P] = [p(x_1), p(x_2), \dots, p(x_N)] \text{ unde } p(x_n) \approx f_X(x_n)q$$

La ieșirea din canal, prin esantionare (sincrona cu intrarea) si cuantizare cu cuanta q , se obtine un semnal discret care poate fi modelat de o sursa de informatie discreta avand alfabetul:

$$[Y] = [y_1, y_2, \dots, y_M] \text{ unde } y_m = mq$$

si probabilitatile:

$$[Q] = [p(y_1), p(y_2), \dots, p(y_M)] \text{ unde } p(y_m) \approx f_Y(y_m)q$$

Informatia medie pe esantion transmisa prin canal este (cf. rezultatului obtinut la canalele discrete):

$$\begin{aligned} I(X, Y) &= \sum_i \sum_j p(x_i, y_j) \log \frac{p(x_i, y_j)}{p(x_i)p(y_j)} = \\ &= \sum_i \sum_j f_{X,Y}(x, y) q^2 \log \frac{f_{X,Y}(x, y)q^2}{f_X(x)qf_Y(y)q} \end{aligned}$$

unde $f_{X,Y}(x, y)$ este densitatea de probabilitate conjugata a v.a. x si y .

La limita, cand $q \rightarrow 0$, suma dubla se transforma intr-o integrala

$$I(X, Y) = \int \int f_{X,Y}(x, y) \log \frac{f_{X,Y}(x, y)}{f_X(x)f_Y(y)} dx dy$$

Prelucrand integrala dubla, se ajunge la o relatie similara cazului canalelor discrete:

$$\begin{aligned}
I(X, Y) &= - \int \int f_{X,Y}(x, y) \log f_Y(y) dx dy - \int \int f_{X,Y}(x, y) \log \frac{f_X(x)}{f_{X,Y}(x, y)} dx dy = \\
&= - \int \log f_Y(y) (\int f_{X,Y}(x, y) dx) dy + \int \int f_{X,Y}(x, y) \log f_{X,Y}(y/x) dx dy = \\
&= - \int f_Y(y) \log f_Y(y) dy + \int \int f_{X,Y}(x, y) \log f_{X,Y}(y/x) dx dy = H(Y) - H(Y/X)
\end{aligned}$$

unde, prin analogie cu eroarea medie din cazul discret, se defineste *entropia diferentiala conditionata*:

$$H(Y/X) = - \int \int f_{X,Y}(x, y) \log f_{X,Y}(y/x) dx dy$$

Observatie: Spre deosebire de entropie, care isi pierde semnificatia la trecerea de la discret la continuu, $I(X, Y)$ isi pastreaza semnificatia de cantitatea medie de informatie pe esantion.

Pe durata D a semnalului, daca esantioanele de semnal sunt independente, se transmite o cantitate de informatie egala cu $D \cdot 2W \cdot I(X, Y)$, unde $D \cdot 2W$ este numarul total de esantioane transmise.

6.2.2. Proprietatile informatiei mutuale in canalele continue

a) **Informatia mutuala este o marime nenegativa:**

$$I(X, Y) \geq 0$$

Justificare:

Ne bazam pe inegalitatea fundamentala, in cazul continuu. Considerand densitatile de probabilitate $f_{X,Y}(x, y)$ si $f_X(x)f_Y(y)$, se poate scrie urmatoarea inegalitate:

$$-I(X, Y) = \int \int f_{X,Y}(x, y) \log \frac{f_X(x)f_Y(y)}{f_{X,Y}(x, y)} dx dy \leq 0$$

b) **Informatia mutuala este, in general, o marime finita.**

c) Relatia $I(X, Y) \leq H(X)$ din cazul discret, **nu mai este valabila**, deoarece entropia in continuu nu mai are aceeasi semnificatie ca in discret (in unele cazuri, entropia poate fi chiar negativa).

d) **$I(X, Y)$ este invarianta la schimbarea coordonatelor**

Pp. ca esantioanele semnalelor de la intrarea si iesirea din canal, sunt transformate in esantioane de frecventa, prin aplicarea Transformarii Fourier:

$$X \xrightarrow{F} U \text{ si } Y \xrightarrow{F} V$$

Se poate demonstra ca:

$$I(X, Y) = I(U, V)$$

6.2.3. Capacitatea canalelor continue

Definitie : Capacitatea canalului continuu este data de maximul cantitatii de informatie care poate fi transmisa prin canal in unitatea de timp ($D = 1\text{sec.}$)

$$C = \max_{f_X(x)} [2 \cdot W \cdot I(X, Y)] = 2 \cdot W \cdot \max_{f_X(x)} [H(Y) - H(Y/X)]$$

unde W este banda canalului.

Pentru calculul capacitatii, se fac urmatoarele ipoteze:

a) Pp. ca avem urmatoarele limitari de putere pentru semnale si zgomotul din canal:

P_X este putere semnalului la intrarea in canal

P_Y este puterea semnalului la iesirea din canal

N este puterea zgomotului din canal

b) Pp. ca zgomotul este aditiv si independent de semnalul X , transmis prin canal. Daca ambele semnale sunt de medie nula, atunci:

$$P_Y = P_X + N$$

In cazul mediei nule, puterea este varianta semnalului. Se demonstreaza usor ca varianta sumei a doua semnale aleatorie independente este egala cu suma variantelor.

Incertitudinea medie asupra iesirii este data numai de zgomot:

$$H(Y | X) = H(Z)$$

unde Z este zgomotul. Daca zgomotul este Gaussian de medie nula si dispersie σ^2 , atunci:

$$H(Z) = \frac{1}{2} \log(2\pi e \sigma^2) = \frac{1}{2} \log(2\pi e N)$$

Deci, entropia conditionata nu depinde de distributia lui X , iar capacitatea devine:

$$C = 2 \cdot W \cdot \left[\max_{f_X(x)} H(Y) - \frac{1}{2} \log(2\pi e N) \right]$$

Semnalul de la iesire ia valori in R (din cauza zgomotului gaussian). Rezulta ca entropia este maxima atunci cand Y este gaussian. In acest caz, entropia sa este:

$$\max_{f_X(x)} H(Y) = \frac{1}{2} \log(2\pi e P_Y) = \frac{1}{2} \log(2\pi e (P_X + N))$$

Deci, capacitatea canalului este:

$$C = W \cdot \log\left(\frac{P_y}{N}\right) = W \cdot \log\left(1 + \frac{P_x}{N}\right)$$

ceea ce arata ca, in cazul canalului continuu, capacitatea creste cu banda si cu puterea semnalului de la intrare si descreste cu puterea zgomotului.

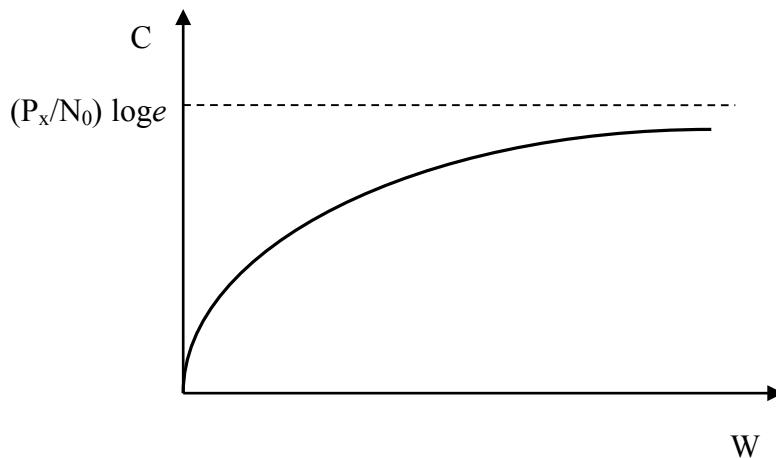
Daca zgomotul de pe canal este alb si de densitate spectrala de putere N_0 , atunci $N = WN_0$ si:

$$C = W \cdot \log\left(1 + \frac{P_x}{WN_0}\right)$$

Reprezentarea grafica a acestei relatii, arata o curba a capacitatii tinzand asymptotic spre:

$$C_\infty = \lim_{W \rightarrow \infty} W \cdot \log\left(1 + \frac{P_x}{WN_0}\right) = \frac{P_x}{N_0} \log e$$

Concluzie: Cresterea largimii de banda peste o anumita valoare nu mai este rationala deoarece capacitatea canalului nu mai creste decat foarte putin.



7. CODAREA DE SURSA

Locul codarii de sursa intr-o schema de transmisiune a datelor :



Rolul codarii de sursa :

- adaptarea alfabetului sursei la alfabetul canalului
- adapatarea statistica a sursei (simboluri echiprobabile pentru alfabetul de canal)

Observatii :

- codarea de sursa priveste sursele discrete de informatie;
- codarea de sursa nu rezolva problema erorilor cauzate de perturbatii;
- prin codare, sursa de informatie, numita si **sursa primara**, este transformata intr-o noua sursa de informatie, numita **sursa secundara**, care debiteaza informatie pe canal.

Doua exemple de codare:

Fie o sursa de informatie primara care genereaza simboluri din alfabetul :

$$[X] = [x_1, x_2, x_3, x_4] \text{ cu probabilitatile } [P] = \left[\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8} \right]$$

Simbolurile trebuie transmise pe un canal binar cu alfabetul $[C] = [0, 1]$. De aceea, ele trebuie transcrise in binar, inainte de transmisie. Transcrierea in binar - **codarea** - se poate face in multe feluri. De exemplu:

1) $x_1 \rightarrow 0 \ 0$
 $x_2 \rightarrow 0 \ 1$
 $x_3 \rightarrow 1 \ 0$
 $x_4 \rightarrow 1 \ 1$

2) $x_1 \rightarrow 0$
 $x_2 \rightarrow 1 \ 0$
 $x_3 \rightarrow 1 \ 1 \ 0$
 $x_4 \rightarrow 1 \ 1 \ 1$

Definitie : **Codarea** este operatia prin care fiecare simbol al sursei primare este inlocuit printr-o succesiune de simboluri ale alfabetului canalului. **Decodarea** este operatia inversa codarii.

Definitie : **Cuvantul de cod** este succesiunea finita de simboluri din alfabetul canalului, cu care este inlocuit un simbol al sursei primare.

Definitie : **Codul** este totalitatea cuvintelor de cod folosite in codarea unei surse.

Definitie : **Lungimea unui cuvant de cod** este egala cu numarul de simboluri din alfabetul canalului, care constituie cuvantul de cod.

Observatii :

- Codarea stabileste o corespondenta biunivoca intre simbolurile sursei primare si cuvintele codului.
- O succesiune de simboluri ale alfabetului canalului, care nu corespunde niciunui simbol al sursei, se numeste **cuvant fara sens**. Prin analogie, un cuvant de cod se mai numeste si **cuvant cu sens**.

Exemplele de mai sus cuprind un cod de lungime fixa (exemplul 1), care are toate cuvintele de aceeasi lungime, si un cod de lungime variabila (exemplul 2), care are cuvinte de lungime variabila. In acest caz, se defineste notiunea de **lungime medie a cuvinelor de cod**.

Definitie : **Lungime medie** a cuvintelor de cod se calculeaza cu expresia :

$$\bar{l} = \sum_{i=1}^N p(x_i)l_i$$

unde cu l_i s-a notat lungimea cuvintelor asociate simbolurilor x_i , iar cu p_i , probabilitatile simbolurilor x_i .

Exemplu: $\bar{l} = \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = \frac{14}{8} \approx 1,7$

Observatii :

- lungimea medie a cuvintelor de cod se numeste, pe scurt, **lungime a codului**;
- la codurile formate din cuvinte de lungime fixa, lungimea codului este egala cu lungimea oricarui cuvant de cod ($\bar{l} = l_i = l$).

Prin codarea cu cuvinte de lungime variabila, se poate realiza o compresie a datelor (reducere a volumului de date).

Definitie : **Raportul de compresie** obtinut prin codare cu un cod de lungime variabila \bar{l} se calculeaza cu expresia :

$$R = \frac{l}{\bar{l}}$$

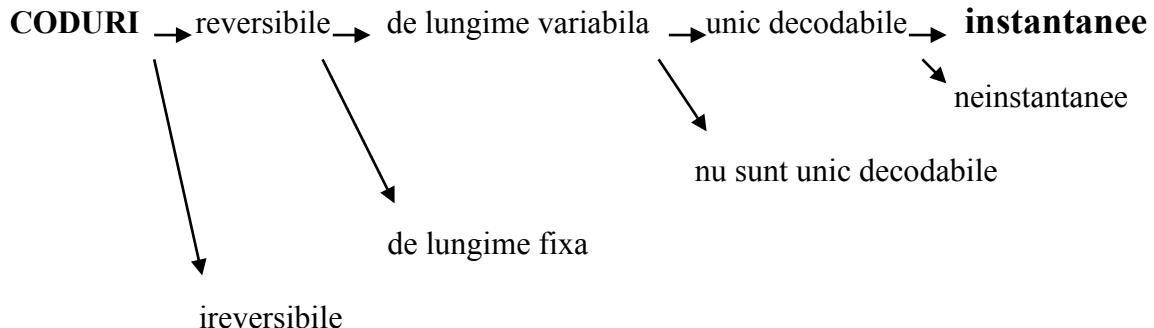
unde cu l s-a notat lungimea unui cod de lungime fixa, obtinut cu acelasi alfabet al canalului.

Exemplu : $R = \frac{2}{1,7} \approx 1,15$ (compresia care se obtine in cazul sursei din exemplul de mai sus, cand se utilizeaza codul de lungime variabila in locul celui de lungime fixa.)

Definitie : Rata de compresie este inversul raportului de compresie :

$$r = \frac{1}{R}$$

7.1. Clasificarea codurilor de sursa



7.1.1. Coduri ireversibile si coduri reversibile

Exemplu :

1) Cod binar ireversibil (la decodare, codul lui x_1 nu poate fi distins de cel al lui x_2 ; la fel pentru x_3 si x_4)

$$\begin{aligned} x_1 &\rightarrow 0 \\ x_2 &\rightarrow 0 \\ x_3 &\rightarrow 1 \\ x_4 &\rightarrow 1 \end{aligned}$$

2) Cod binar reversibil

$$\begin{aligned} x_1 &\rightarrow 0\ 0 \\ x_2 &\rightarrow 0\ 1 \\ x_3 &\rightarrow 1\ 0 \\ x_4 &\rightarrow 1\ 1 \end{aligned}$$

7.1.2. Coduri unic decodabile si coduri care nu sunt unic decodabile

Exemplu :

1) Cod care nu este unic decodabil :

$$\begin{aligned}x_1 &\rightarrow 0 \\x_2 &\rightarrow 1 \ 0 \\x_3 &\rightarrow 1 \ 1 \\x_4 &\rightarrow 1 \ 1 \ 0\end{aligned}$$

La decodare, grupul 1 1 0 poate fi interpretat fie ca simbolul x_4 , fie ca grupul de simboluri $x_3 x_1$.

2) Cod unic decodabil

$$\begin{aligned}x_1 &\rightarrow 0 \\x_2 &\rightarrow 1 \ 0 \\x_3 &\rightarrow 1 \ 1 \ 0 \\x_4 &\rightarrow 1 \ 1 \ 1 \ 0\end{aligned}$$

Orice cuvant de cod poate fi decodat intr-un singur simbol.

7.1.3. Coduri neinstanee si coduri instantanee

Exemplu :

1) Cod neinstantaneu :

$$\begin{aligned}x_1 &\rightarrow 0 \\x_2 &\rightarrow 0 \ 1 \\x_3 &\rightarrow 0 \ 1 \ 1 \\x_4 &\rightarrow 0 \ 1 \ 1 \ 1\end{aligned}$$

Trebuie asteptat primul simbol al urmatorului cuvat de cod pentru a face decodarea cuvantului receptionat (acest cod se mai numeste si cod cu separator).

2) Cod instantaneu

$$\begin{aligned}x_1 &\rightarrow 0 \\x_2 &\rightarrow 1 \ 0 \\x_3 &\rightarrow 1 \ 1 \ 0 \\x_4 &\rightarrow 1 \ 1 \ 1\end{aligned}$$

Decodarea se poate face la primirea ultimului simbol al cuvantului de cod.

Observatie:

- codurile instantanee sunt cele utilizate in practica compresiei.

7.2. Coduri instantanee

Definitie : Fie cuvantul de cod c , constituit din n simboluri ale alfabetului de canal :

$$c = [c_1 \dots c_n]$$

Sirul format din primele k simboluri, se numeste prefix al cuvantului.

Teorema : Conditia necesara si suficienta ca un cod sa fie instantaneu este ca niciun cuvant sa nu fie prefix al altui cuvant mai lung.

Observatii:

- spunem despre un cod instantaneu ca este un cod cu proprietatea de prefix;
- codurile instantanee se mai numesc si ireductibile.

7.3. Inegalitatea Kraft-McMillan

Teorema : Fie sursa primara de informatie cu alfabetul :

$$[X] = [x_1, \dots, x_N]$$

si alfabetul de canal $[C] = [c_1, \dots, c_D]$, cu simbolurile caruia se vor forma cuvinte de cod pentru sursa primara. **O conditie necesara si suficienta pentru a construi un cod instantaneu** (ireductibil) cu cuvinte de lungime l_1, \dots, l_N este :

$$\sum_{i=1}^N D^{-l_i} \leq 1 \quad (\text{Inegalitatea Kraft-McMillan})$$

Justificare: ne folosim de reprezentarea prin arbori a codurilor ireductibile

7.4. Limita inferioara a lui \bar{l}

Fie o sursa primara de informatie cu alfabetul :

$$[X] = [x_1, \dots, x_N] \text{ si probabilitatile } [P] = [p(x_1), \dots, p(x_N)]$$

Simbolurile sursei sunt codate cu un cod de lungime medie \bar{l} . Cuvintele de cod sunt constituite din simboluri ale alfabetului de canal $[C] = [c_1, \dots, c_D]$.

Daca $H(X)$ este entropia sursei, atunci fiecare simbol c_d poarta in medie o cantitate de informatie:

$$\frac{H(X)}{\bar{l}}$$

Aceasta cantitate, nu poate fi mai mare decat entropia maxima a sursei secundare $H_{\max}(C) = \log_2 D$:

$$\frac{H(X)}{\bar{l}} \leq \log_2 D \Rightarrow \bar{l} \geq \frac{H(X)}{\log_2 D}$$

Deci, limita inferioara pentru lungimea medie a oricarui cod instantaneu este:

$$\bar{l}_{\min} = \frac{H(X)}{\log_2 D}$$

Observatii:

- daca codarea se face cu alfabet binar, atunci limita inferioara pentru \bar{l} este chiar entropia sursei primare $H(X)$;
- aceasta relatie este, uneori, folosita ca definitie a entropiei.

A 2-a definicie a entropiei : Entropia unei surse este egala cu lungimea medie a unui cod binar minim, cu care sursa poate fi codata (nu totdeauna acest cod exista).

7.5 Coduri absolut optimale

In practica, ne intereseaza codurile cu \bar{l} cat mai mic.

Definitie: Codurile care au $\bar{l} = \bar{l}_{\min} = \frac{H(X)}{\log_2 D}$ se numesc *coduri absolut optimale*.

Conform Sectiunii 7.4, cantitatea medie de informatie transmisa fiecarui simbol de canal prin codare, altfel spus entropia sursei secundara $H(C)$, este invers proportionala cu \bar{l} :

$$H(C) = \frac{H(X)}{\bar{l}}$$

Aceasta relatie arata ca \bar{l} isi atinge minimul cand $H(C)$ este maxim, adica atunci cand, prin codare, simbolurile c_d ajung sa fie transmise echiprobabil:

$$p(c_1) = \dots = p(c_D) = \frac{1}{D}$$

Considerand ca nu exista dependenta statistica intre simbolurile c_d , care intra in componenta cuvintelor de cod, rezulta urmatoarele probabilitati pentru cuvintele de cod si, deci, pentru simbolurie sursei primare:

$$p(x_i) = \left(\frac{1}{D}\right)^{l_i} \text{ unde } l_i \text{ este lungimea cuvantului de cod pentru } x_i.$$

Cum $\sum_i p(x_i) = 1$, rezulta ca un cod absolut optimal indeplineste urmatoarea conditie:

$$\sum_{i=1}^N D^{-l_i} = 1$$

Observatii:

- egalitatea $\sum_{i=1}^N D^{-l_i} = 1$ este o conditie de existenta pentru codurile absolut optimale;
- in cazul codarii binare, conditia $p(x_i) = \left(\frac{1}{D}\right)^{l_i}$ se traduce prin a cere pentru simbolurile sursei primare probabilitati, care sunt puteri intregi negative ale lui 2, de exemplu: $[P] = \left[\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\right]$.
- codurile absolut optimale sunt un caz limita pentru Inegalitatea Kraft-McMillan.

7.6. Coduri optimale

Asa cum am vazut la sectiunea anterioara, codarea unei surse de informatie cu un cod binar absolut optimal este posibila numai daca probabilitatile sursei satisfac conditia:

$$p(x_i) = \left(\frac{1}{D}\right)^{l_i} \Leftrightarrow l_i = -\frac{\log_2 p(x_i)}{\log_2 D}$$

De cele mai multe ori, $-\frac{\log_2 p(x_i)}{\log_2 D}$ este un numar zecimal oarecare. De aceea, se construiesc

cuvinte de cod cu lungimea minima posibila, adica $l_i = \left\lceil -\frac{\log_2 p(x_i)}{\log_2 D} \right\rceil$. Aceste cuvinte satisfac conditia:

$$l_i \leq -\frac{\log_2 p(x_i)}{\log_2 D} + 1 \quad \forall i$$

Amplificand inegalitatile cu $p(x_i)$ si insumandu-le dupa i , rezulta:

$$\sum_i p(x_i) l_i \leq -\frac{\sum_i p(x_i) \log_2 p(x_i)}{\log_2 D} + \sum_i p(x_i)$$

Deci

$\bar{l} \leq \frac{H(X)}{\log_2 D} + 1$ ceea ce arata ca se poate gasi un cod unic decodabil, care sa aibe lungimea mai mica decat limita superioara $\frac{H(X)}{\log_2 D} + 1$. In cazul codurilor binare, aceasta proprietate devine:

$$\bar{l} \leq H(X) + 1$$

Vom demonstra, in continuare, ca aceste coduri satisfac Inegalitatea Kraft-McMillan, deci ca ele sunt si instantanee.

Deoarece $l_i = \left\lceil -\frac{\log_2 p(x_i)}{\log_2 D} \right\rceil$, putem scrie:

$$-\frac{\log_2 p(x_i)}{\log_2 D} \leq l_i \Leftrightarrow p(x_i) \geq D^{-l_i} \quad \forall i$$

Insumand dupa i , rezulta :

$$\sum_i p(x_i) \geq \sum_i D^{-l_i} \Leftrightarrow \sum_i D^{-l_i} \leq 1$$

Deci, aceste coduri satisfac Inegalitatea Kraft-McMillan, care este conditia necesara si suficienta pentru a avea un cod ireductibil.

Definitie: Codurile constituie din cuvinte de lungime $l_i = \lceil -\log_2 p(x_i) \rceil$ sunt **coduri optimale**.

7.7. Capacitatea, eficienta si redundanta codurilor

Definitie : Capacitatea unui cod este maximul cantitatii medii de informatie ce poate fi transmisa de simbolurile din alfabetul canalului :

$$C = H_{\max}(C) = \log D$$

Definitie : Eficienta unui cod se defineste prin :

$$\eta = \frac{\bar{l}_{\min}}{\bar{l}} \leq 1$$

$$\eta = \frac{\frac{H(X)}{\log D}}{\frac{\bar{l}}{\log D}} = \frac{\frac{H(X)}{\bar{l}}}{\frac{H(C)}{\log D}} = \frac{H(X)}{\bar{l} \cdot H(C)}$$

Definitie : Redundanta unui cod se defineste prin :

$$\rho = 1 - \eta = 1 - \frac{H(C)}{\log D} \in [0,1]$$

Observatie: Capacitatea, eficienta si redundanta codului sunt marimi similare celor prezentate la Capitolul de canale discrete. Expresiile sunt diferite pentru ca, in cazul canalelor, se ia in considerare prezenta perturbatiilor. Codarea de sursa – numita si codarea canalelor fara perturbatii – urmareste numai maximizarea cantitatii de informatie transmisa de simbolurile alfabetului de canal.

7.8. Extensia unei surse de informatie

Fie o sursa de informatie cu alfabetul :

$$[X] = [x_1, \dots, x_N] \text{ si probabilitatile } [P] = [p(x_1), \dots, p(x_N)]$$

Presupunem ca sursa X genereaza urmatorul sir de v.a., care iau valori in alfabetul sursei:

$$X_0, X_1, X_2, X_3, \dots, X_{2n}, X_{2n+1}, \dots$$

Definitie : Extensia de ordin 2 a sursei X , este o sursa notata X^2 , care genereaza sirul:

$$Z_0, Z_1, \dots, Z_n, \dots$$

unde v.a. Z_n sunt siruri de doua v.a. consecutive ale sirului $X_0, X_1, X_2, X_3, \dots, X_{2n}, X_{2n+1}, \dots$
Mai precis: $Z_0 = (X_0, X_1)$, $Z_1 = (X_2, X_3)$, ..., $Z_n = (X_{2n}, X_{2n+1})$

Observatii:

- extensia de ordin m se noteaza cu X^m si este o sursa ale carei simboluri sunt siruri de lungime m
- alfabetul extensiei X^m este constituit din N^m simboluri (siruri de lungime m).

Teorema : Entropia extensiei X^m , este de m ori mai mare decat entropia sursei fara memorie X :

$$H(X^m) = mH(X)$$

7.9. Prima Teorema a lui Shannon

Conform rezultatelor din Sectiunile 7.4 si 7.6, lungimea unui cod optimal pentru codarea unei surse de informatie fara memorie X , satisface urmatoarele inegalitati :

$$\frac{H(X)}{\log_2 D} \leq \bar{l} \leq \frac{H(X)}{\log_2 D} + 1$$

Aceasta dubla inegalitate este valabila si pentru extensia X^m , care este tot o sursa fara memorie :

$$\frac{H(X^m)}{\log_2 D} \leq \bar{l}^{(m)} \leq \frac{H(X^m)}{\log_2 D} + 1$$

unde $\bar{l}^{(m)}$ este lungimea medie a cuvintelor de cod pentru simbolurile sursei extinse, care sunt siruri de m simboluri ale sursei initiale. Deci, $\bar{l}^{(m)} = m\bar{l}$, unde \bar{l} este lungimea medie, care revine fiecarui simbol al sursei neextinse.

Aplicand rezultatul Sectiunii 7.8, dubla inegalitate devine:

$$\frac{H(X)}{\log_2 D} \leq \bar{l} \leq \frac{H(X)}{\log_2 D} + \frac{1}{m}$$

ceea ce reprezinta expresia matematica a Primei teoreme a lui Shannon

Prima teorema a lui Shannon sau Teorema codarii canalelor fara zgromot: Codand siruri de simboluri suficient de lungi, ne putem apropi la oricat de mult de *codarea absolut optimala*.

Observatie:

- sursa trebuie sa fie fara memorie.

7.10. Algoritmi de codare entropica

7.10.1. Codul Shannon-Fano

Exemplul 1: fie sirul de simboluri $s_2, s_4, s_3, s_2, s_4, s_5, s_4, s_5, s_1, s_4, s_4, s_4, \dots$ generat de o sursa de informatie cu alfabetul $X = [s_1, s_2, s_3, s_4, s_5]$ si probabilitatile $P[X] = [0.15; 0.25; 0.05; 0.35; 0.2]$

Obtinerea cuvintelor de cod:

1. Se scriu simbolurile in ordinea descrescatoare a probabilitatilor;
2. Sirul simbolurilor se imparte in doua subsiruri, cu probabilitati cat mai apropiate (probabilitatea unui subsir este egala cu suma probabilitatilor simbolurilor care il compun);
3. Se atribuie “0” subsirului superior si “1” subsirului inferior;
4. Pentru fiecare subsir, se reiau pasii 2 si 3;
5. Obtinerea cartii de cod se incheie atunci cand pentru toate simbolurile s-au obtinut cuvinte de cod distincte.

0.35	s_4	0 0
<hr/>		
0.25	s_2	0 1
<hr/>		
0.2	s_5	1 0
<hr/>		
0.15	s_1	1 1 0
<hr/>		
0.05	s_3	1 1 1

Cartea de cod:

Simbol	Cuvant de cod
s_1	110
s_2	01
s_3	111
s_4	00
s_5	10

Semnal codat: 01 00 111 01 00 10 00 00 10 110 ... (22 bits)

Lungimea medie a cuvintelor de cod : $\bar{l} = 0.2 \cdot 3 + 0.25 \cdot 2 + 0.05 \cdot 3 + 0.35 \cdot 2 + 0.2 \cdot 2 = 2.2$

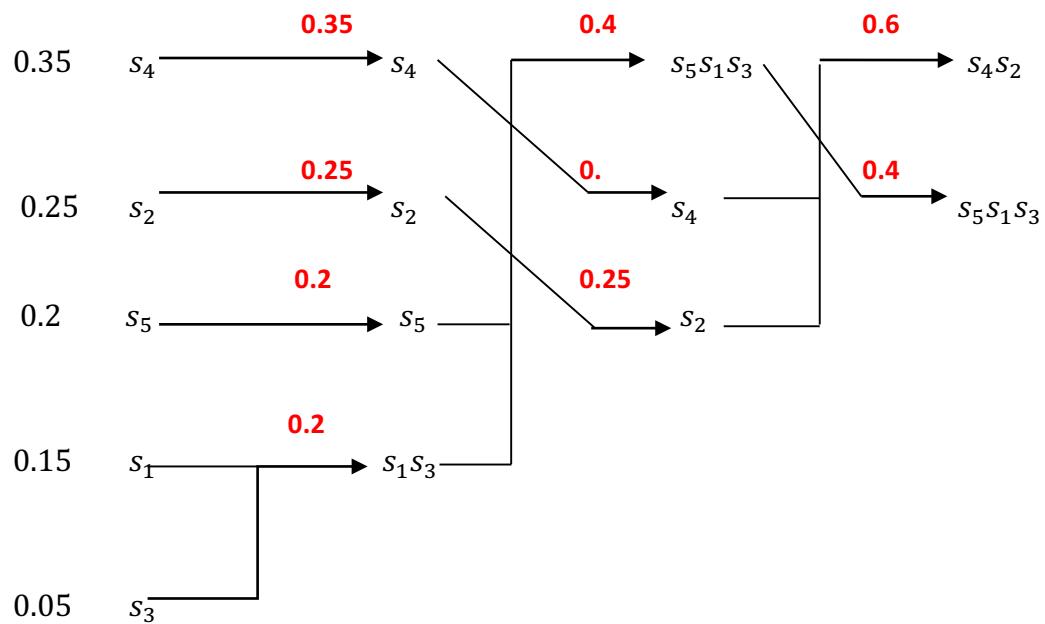
Entropia sursei: $H=2.12$ bits

$$\bar{l} > H$$

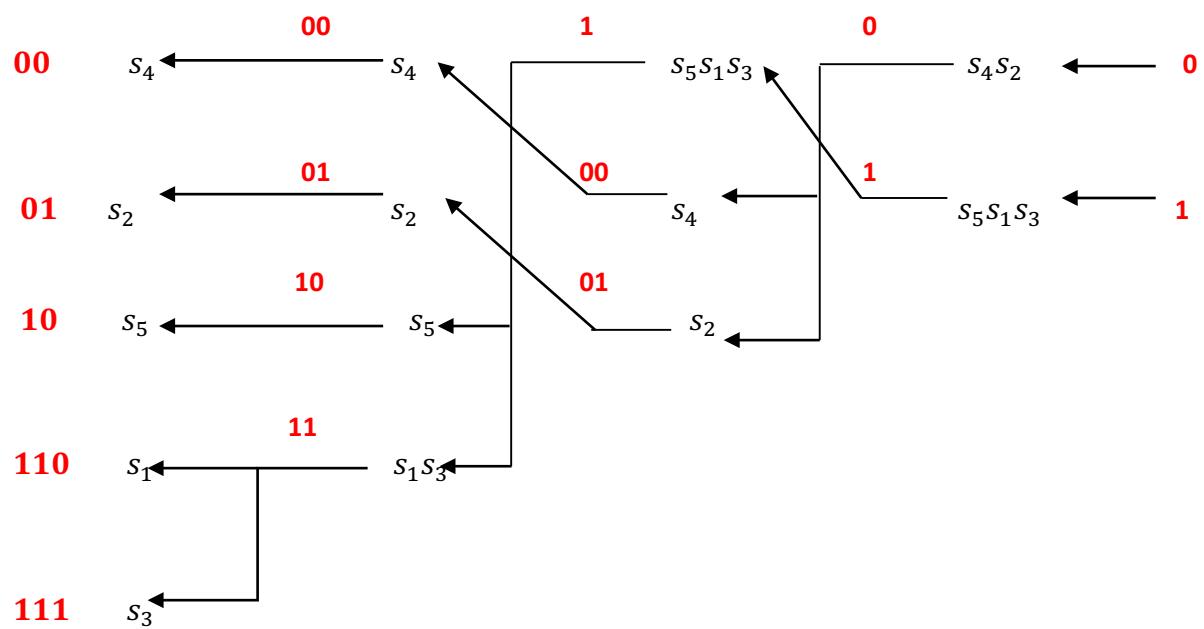
7.10.2. Codul Huffman

Pentru aceeasi sursa de informatie cu alfabetul $X = [s_1, s_2, s_3, s_4, s_5]$ si setul de probabilitati $P[X] = [0.15; 0.25; 0.05; 0.35; 0.2]$.

Reducerea sursei:



Obtinerea cuvintelor de cod:



Cartea de cod:

Simbol	Cuvantul de cod
s_1	110
s_2	01
s_3	111
s_4	00
s_5	10

Observatii:

- Ambele coduri au proprietatea de prefix si sunt coduri instantanee (cuvantul poate fi decodat odata cu receptionarea ultimului simbol)
- Pentru acest exemplu simplu, lungimea medie a cuvintelor de cod este aceeasi pentru ambele coduri; in general, codul Huffman, conduce la lungimi mai mici ale cuvintelor de cod.

Doua moduri de a scurta lungimea medie a cuvintelor de cod:

- Prin codare de siruri de simboluri (conform Teoremei I a lui Shannon)
- Prin codare contextuala (daca sursa de informatie are memorie)

Exemplul 2 : $s_9, s_3, s_5, s_3, s_5, s_5, s_5, s_3, s_5, s_5, \dots$

Semnalul de mai sus este generat de o sursa de informatie cu alfabetul $X = [s_1, s_2, s_3]$ si setul de probabilitati $P[X] = [0.3; 0.1; 0.6]$.

Cartea de cod pentru codare simbol cu simbol:

Simbol	Cuvant de cod
s_1	10
s_2	11
s_3	0

Lungimea medie a cuvintelor de cod: $\bar{l} = 0.2 \cdot 2 + 0.2 \cdot 2 + 0.6 \cdot 1 = 1.4$ bits

CODARE DE SIRURI DE SIMBOLURI

Cartea de cod pentru siruri de 2 simboluri (extensia de ordin 2 a sursei) :

Sir	Probabilitate	Cuvint de cod
s_3s_3	0.36	1
s_3s_2	0.18	000
s_2s_3	0.18	001
s_2s_2	0.09	0100
s_3s_1	0.06	0110
s_1s_3	0.06	0111
s_1s_2	0.03	01010
s_2s_3	0.03	010110
s_1s_1	0.01	010111

Lungimea medie pe simbol, a cuvintelor de cod: $\bar{l} = \frac{\sum l_i p_i}{2} = 1.33$ bits

CODAREA CONTEXTUALA

Cartile de cod pentru codarea contextuala:

Context s_1 :

Probabilitati estimate: $[X|s_1] = \left[\frac{0}{3}; \frac{0}{3}; \frac{1}{3} \right]$

Cartea de cod:

Simbol	Cuvant de cod
s_1	-
s_2	-
s_3	0

Lungimea medie a cuvintelor de cod: $\bar{l} = 1$ bit

Context s_2 :

Probabilitati estimate: $[X|s_2] = \left[\frac{1}{1}; \frac{0}{1}; \frac{0}{1} \right]$

Cartea de cod:

Simbol	Cuvant de cod
s_1	0
s_2	-
s_3	-

Lungimea medie a cuvintelor de cod: $\bar{l} = 1$ bit

Context s_3 :

Probabilitati estimate: $[X|s_3] = \left[\frac{2}{5}; \frac{0}{5}; \frac{3}{5} \right]$

Cartea de cod:

Simbol	Cuvant de cod
s_1	0
s_2	-
s_3	1

Lungimea medie a cuvintelor de cod: $\bar{l} = 1$ bit

Lungimea medie a cuvintelor de cod in codarea contextuala:

$$\bar{l} = p(s_1) \cdot 1 + p(s_2) \cdot 1 + p(s_3) \cdot 1 = 1 \text{ bits}$$

Observatii:

- Ambele tehnici se folosesc simultan pentru a obtine o compresie cat mai buna;
- Codarea de siruri urmareste atingerea limitei inferioare (entropia sursei) a lungimii medii a cuvintelor de cod, conform Teoremei I a lui Shannon;
- Codarea contextuala se face cand sursa de informatie are memorie.

7.10.3. Codul aritmetic

Arithmetic encoding is applied to strings of symbols. The entire file is encoded by a unique binary codeword.

Algorithm in two stages:

- 1) Partition of [0, 1) interval, according to symbols statistics,
- 2) Generation of the codeword.

Example: Suppose a source with the alphabet [A, B, C] and the probabilities [0.5; 0.3; 0.2]]. The source generates the following string of symbols that must be encoded: AAAAABBBC.

ENCODING

Stage 1) Partition of [0, 1) interval

Given the interval [0.0, 1.0) the distribution of probabilities can be visualized such as each symbol "occupies" a range directly proportional to its own probability. Thus, A occupies [0; 0.5), B occupies [0.5; 0.8) and C occupies the rest of [0.8; 1). At this step, the current interval is [0:0; 1:0).

Afterwards, since the first symbol of the message is A, the current interval becomes [0.0, 0.5) with the endpoints calculated as follows:

$$\begin{aligned} \text{LOW}_1 &= 0.0 \\ \text{HIGH}_1 &= 0.0 + 0.5 \times (1 - 0) = 0.5 \end{aligned}$$

The second symbol to be encoded is also A. Consequently, the current interval becomes [0.0; 0.25) with the endpoints calculated as follows:

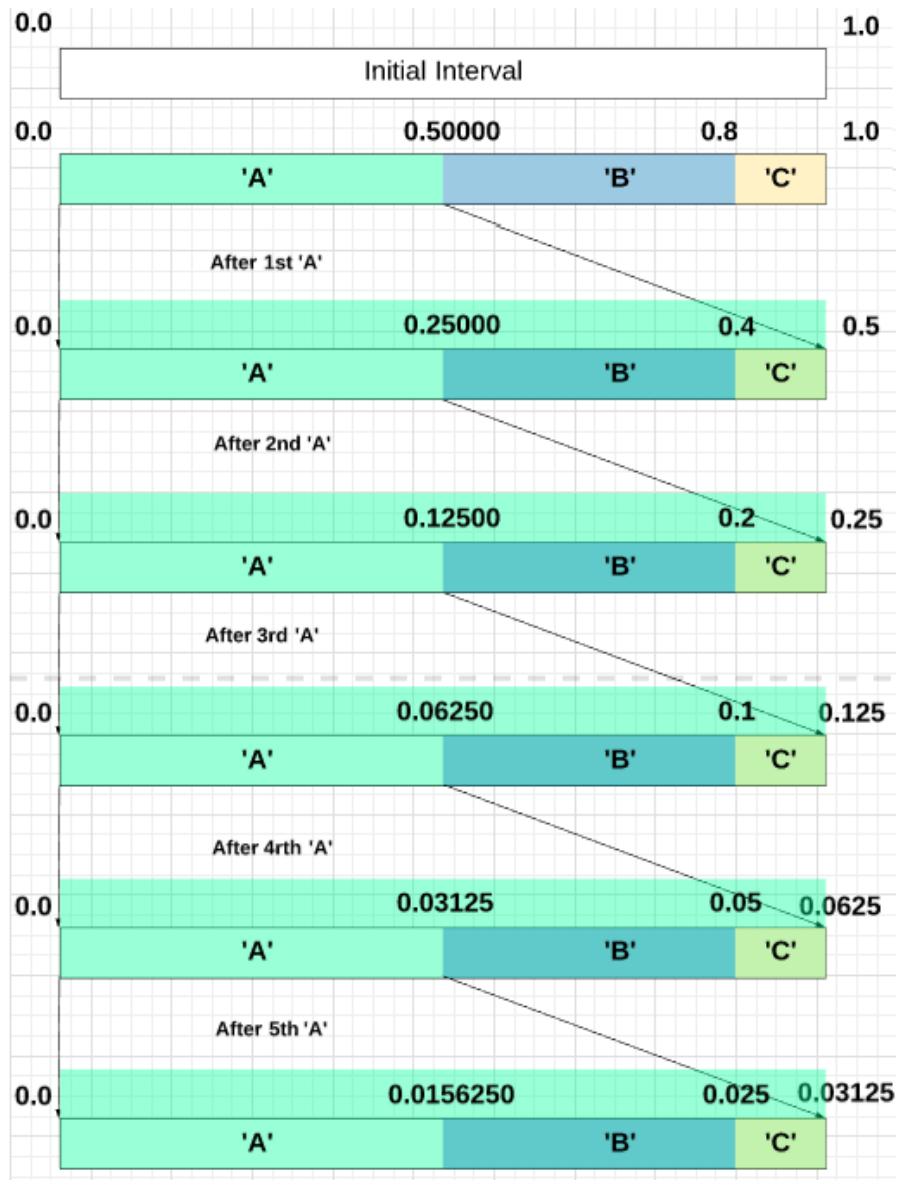
$$\begin{aligned} \text{LOW}_2 &= 0.0 \\ \text{HIGH}_2 &= 0.0 + 0.5 \times (0.5 - 0) = 0.25 \end{aligned}$$

To generalize, when the n-th successive symbol is processed, the LOW_n and HIGH_n endpoints of the current interval are:

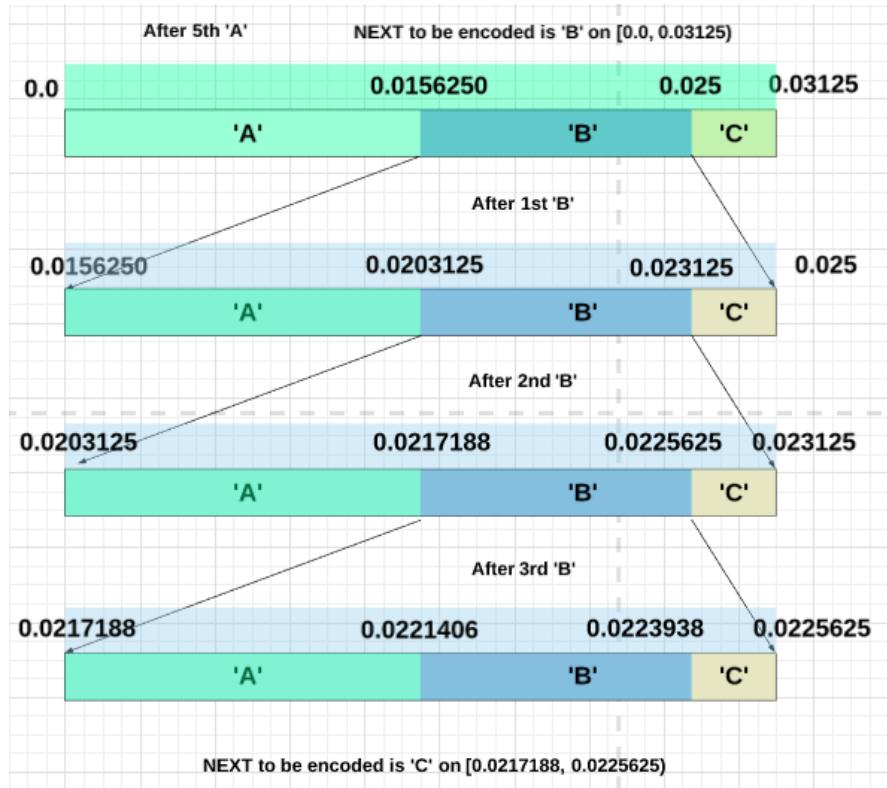
$$\begin{aligned} \text{LOW}_n &= \text{LOW}_{n-1} \\ \text{HIGH}_n &= \text{LOW}_{n-1} + p \times (\text{HIGH}_{n-1} - \text{LOW}_{n-1}) \end{aligned}$$

where p is the probability of the symbol to encode.

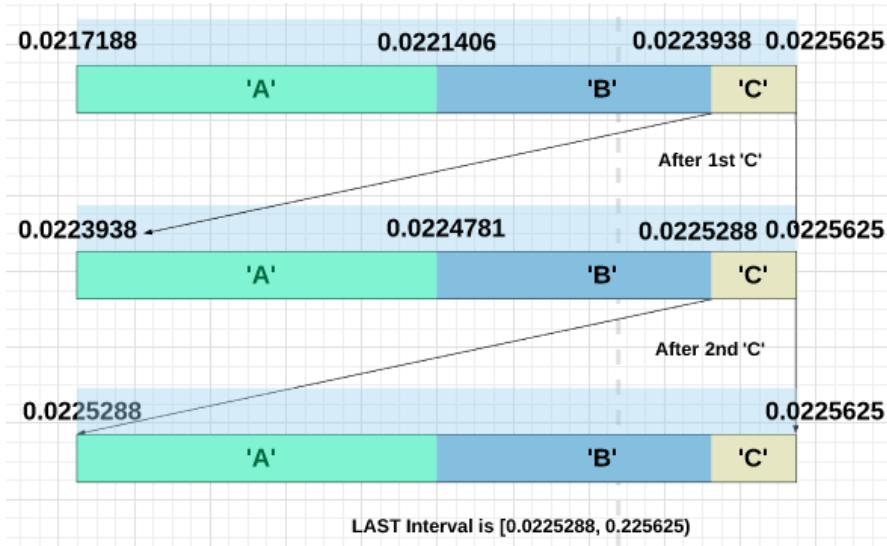
The current interval develops and changes with each new symbol as depicted in the figures below.



Current interval development for the sequence AAAAA.



Current interval development for the sequence BBB.



Current interval development for the sequence CC.

After the last C in the message, the current interval becomes [0:0225288; 0:0225625]. The width of interval is the mere probability of the message, estimated by supposing that the symbols are statistically independent and their probabilities do not change along the message.

Stage 2) Generation of the codeword

We choose a number \hat{c} from the interval [0.0225288; 0.0225625] to encode the entire sequence of symbols. This number, converted into binary, is the message **codeword**.

The length of the codeword is given by the message probability ($HIGH_n - LOW_n$)

$$L_n = \lceil -\log_2(HIGH_n - LOW_n) \rceil$$

The representation on a limited number of bit shifts the chosen number to the left. A choice that guarantees that \hat{c} remains inside the interval is:

$$\hat{c} = 2^{-L_n} \lfloor 2^{L_n} LOW_n + 1 \rfloor > LOW_n$$

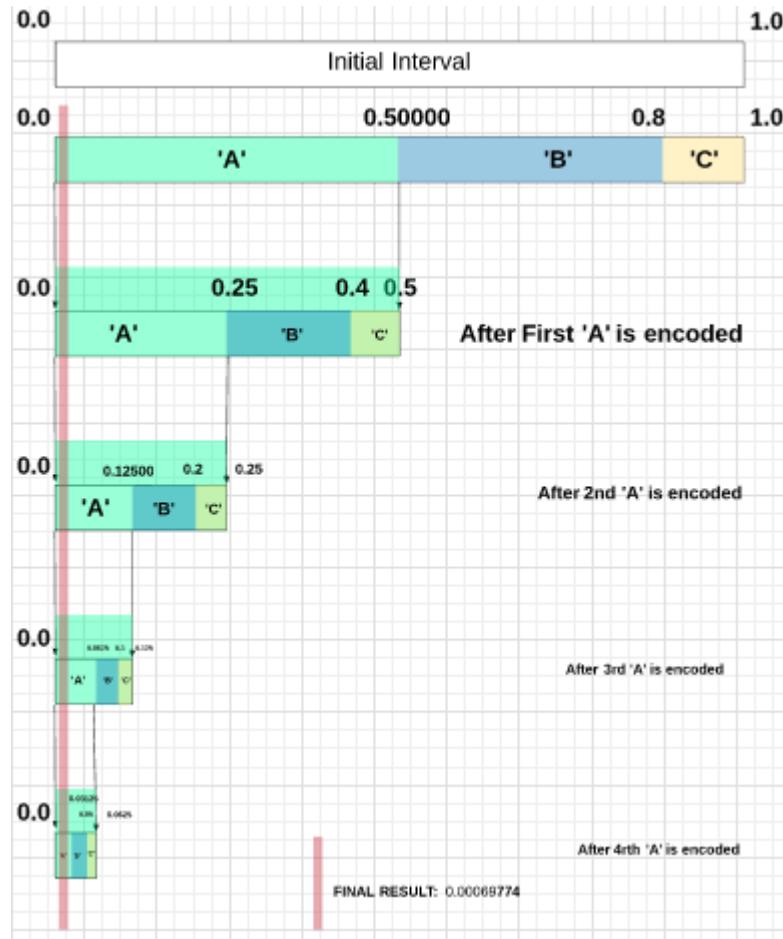
In our example, $L = 15$, $\hat{c}=0.02255249023$ and the codeword is 000001011100010. Due to the positioning of \hat{c} the decoder will be able to perform the inverse steps correctly in order to find out the message.

DECODING

The codeword is converted into a fractional number and the decoder identifies the interval that originated it. In order to do this, the decoder must rebuild the interval partition.

Suppose the fractional number is 0.02255249023. The decoder proceeds as follows:

- **At the first step**, the number is compared with the partition of the initial interval [0; 1); since the number is included in [0; 0.5), the decoder decides that the first symbol of the message is A.
- **At the second step**, the decoder considers only the interval [0; 0.5) and proceeds to its partition; since the number is lower than 0.25, the next decoded symbol is also A.
- **Next steps**: the decoder considers the last identified interval, proceeds to its partition and identifies the next symbol by looking for the interval that includes the number.
- **Stop condition**: the algorithm stops when the last identified interval is narrower than 2^{-L_n}



Decoding of the first 4 symbols of the message: AAAA

Remarks:

1. The decoder must be aware of symbols probabilities, otherwise it cannot rebuild the partition of $[0, 1)$ interval.
2. The arithmetic coding is an entropic coding since the length of the codeword depend on the string probability.
3. The arithmetic coding can adapt to string non-stationarity. The encoding starts with a set of predefined probabilities and then the probabilities are adjusted on-fly, according with the last encoded symbols.

7.11 Codarea cu harta de pozitii

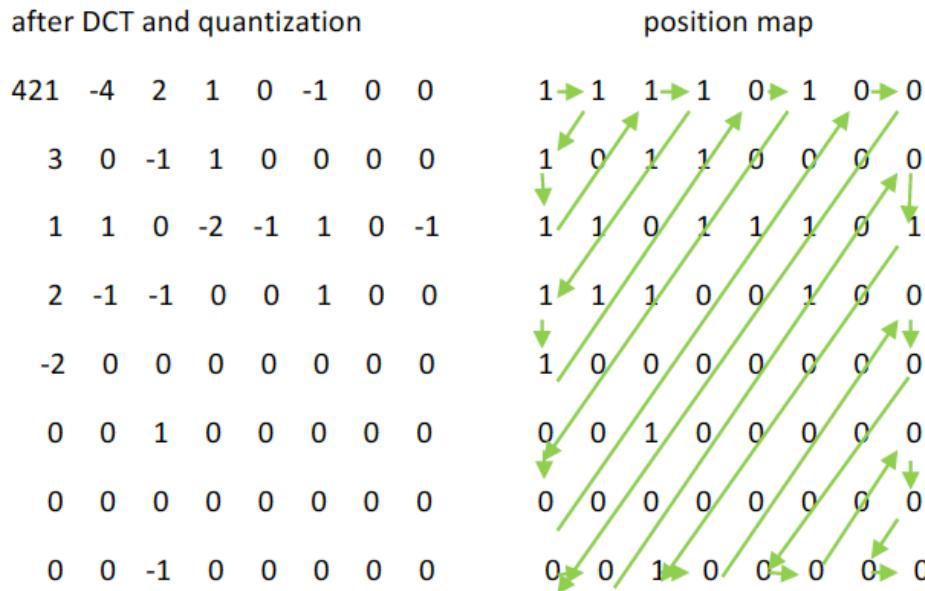
The pixels of the decorrelated image can be serialized and encoded entirely. Since the number of zero pixels can be very high after decorrelation and quantization, sometimes it is preferable to encode only nonzero pixels and their position (the position map). RLC is the technique currently used for this purpose.

RLC (Run Length Coding) for position map encoding

The quantization of decorrelated image creates many zero coefficients. In a symbol by symbol encoding, each zero would need at least one bit in order to be encoded (they represent the majority, as the coefficient distribution shows).

In order to save bits, only non-zero coefficients are encoded and their position is marked on a map that is a binary image ('0' indicates the position of zero coefficients). This map is serialized and RLC + Huffman encoded.

Example: 8x8 image block of Lena,



Serialized position map (JPEG utilizes a zig-zag scan):

1 111011111101010110000001000010010000001001000010000000000000000

RLC consists in specifying the length of alternating runs of '1' and '0'.

1111 0 1111111 0 1 0 1 0 11 000000 1 0000 1 00 1 000000 1 00 1 0000 1 00000000 0000000

RLC code: 4 1 7 1 1 1 1 2 6 1 4 1 2 1 6 1 2 4 1 7 0 7 0 1 (for maximum 7 bits runs).

Huffman code of RLC string:

- 0 (p=2/25) -> 0000
- 1 (p=12/25) -> 1
- 2 (p=3/25) -> 001
- 4 (p=3/25) -> 010
- 6 (p=2/25) -> 0001
- 7 (p=3/25) -> 011

Length of the RLC+Huffman encoded map= $2 \times 4 + 12 \times 1 + 3 \times 3 + 3 \times 3 + 2 \times 4 + 3 \times 3 = 52$

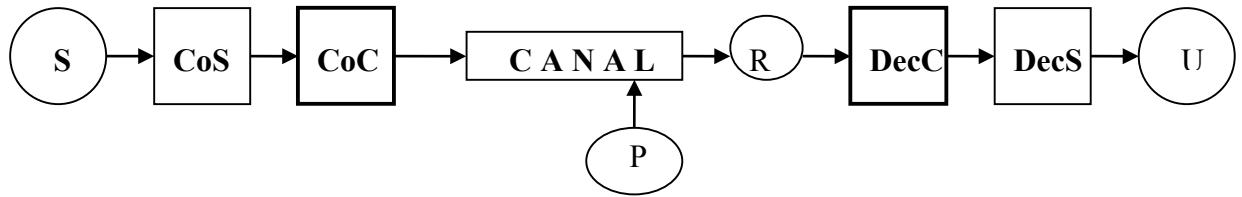
Remarks:

- By using a position map, only 21 from 64 coefficients need to be encoded: -4, -2, -1, 1, 2, 3, 421

- Their Huffman encoding gives 48 bits
- The position map must also be transmitted, otherwise the decoding is not possible. It means a total of $48+52 = 100$ bits
- By RLC+Huffman encoding of the map, there are saved $64-52=12$ bits for a single 8x8 block
- If the 8x8 block were encoded entirely (without considering a position map), the resulted code would have been 109 bit (longer by 9%)

8. CODAREA DE CANAL

Locul *codarii de canal* intr-o schema de transmisiune a datelor :



Rolul codarii de canal : La trecerea prin canal, se produc modificari *aleatoare* ale informatiei din cauza perturbatiilor. De aceea, la iesirea din canal, informatia nu poate fi reconstituita fidel. Putem construi totusi, un *Codor de canal* care sa reduca probabilitatea de eroare printr-o codare adevarata a sirului de simboluri, inainte ca acestea sa fie transmise prin canal. La iesirea din canal, *Decodorul de canal*, face operatia inversa pentru a reconstituui sirul de simboluri si, in cazul in care sunt erori, le poate detecta sau corecta intre anumite limite.

Observatii :

- Codarea de canal nu elimina erorile, ci doar reduce probabilitatea lor de aparitie.

8.1. Probabilitatea de eroare la receptie (receptorul cu rata minima de eroare)

In cazul particular al unui canal binar, probabilitatea ca un “0” receptionat sa fie interpretat (decodat) gresit este:

$$1 - p(x=0/y=0)$$

In cazul general, deoarece intr-o transmisie printr-un canal cu zgomot, in mod normal probabilitatea de transmisie corecta $p(y_j/x_j)$ este mai mare decat probabilitatea de transmisie eronata si $p(x=0)=p(x=1)=1/2$ (datorita adaptarii statistice facute de codarea de sursa), putem spune ca $p(x_j/y_j)$ este probabilitatea care minimizeaza probabilitatea de decodare gresita a lui y_j , egala cu $1 - p(x_j/y_j)$.

Deci, pentru a minimiza numarul de erori la receptie (cand nu exista alte mijloace de reducere a erorilor) decodorul trebuie construit astfel incat y_j sa fie decodat in simbolul x_i cel mai probabil, adica in simbolul pentru care $p(x_i/y_j)$ este maxima.

In acest caz, in medie, probabilitatea de eroare la decodare va fi:

$$P(E) = \sum_j (1 - p(x_j/y_j)) p(y_j)$$

Observatii:

- decodorul care lucreaza pe acest principiu se numeste *Decodor cu rata minima de eroare*;
- aceasta probabilitate poate fi calculata daca se cunosc matricea de zgomot a canalului si probabilitatile simbolurilor la intrarea in canal (simbolurile sunt practic echiprobabile daca s-a facut, in prealabil, o codare de sursa):

$$P(E) = \sum_j (1 - p(x_j / y_j)) p(y_j) = \sum_j p(y_j) - \sum_j p(y_j / x_j) p(x_j) = 1 - \sum_j p(y_j / x_j) p(x_j)$$

Exemplul: Canalul binar simetric

Fie canalul cu matricea de zgomot: $P(Y/X) = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}$ unde $p = 0,2$ este probabilitatea de transmisie eronata a unui simbol. Daca inainte s-a facut o codare de sursa, atunci simbolurile de la intrare sunt practic echiprobabile :

$$p(x_1) \approx p(x_2) \approx \frac{1}{2}$$

Conform rezultatului de mai sus, probabilitatea de eroare a *Decodorului cu rata minima de eroare* va fi :

$$P(E) = 1 - \sum_j p(y_j / x_j) p(x_j) = 1 - 2(1-p)\frac{1}{2} = p = 0,2$$

Observatie: Probabilitatea $P(E)$ este mare, de aceea trebuie redusa prin codare de canal.

8.2. O metoda simpla: codarea de canal prin repetarea simbolurilor

O metoda simpla de codare de canal este *prin repetarea simbolurilor*. Ea consta din a transmite fiecare simbol de un numar impar de ori. Decodarea se face prin logica majoritara.

Exemplul:

- a) Codarea unui sir binar prin repetare de trei ori a fiecarui simbol (transmisia se face prin canalul din exemplul anterior)

Codarea : 0 -> 000
 1-> 111

Decodarea : 000->0 111->1
 001->0 110->1
 010->0 101->1
 100->0 011->1

$$p(y_{decodat} = 0 / x = 0) = p(000/x = 0) + p(001/x = 0) + p(010/x = 0) + p(100/x = 0) = \\ = (1-p)^3 + 3(1-p)^2 p = (1-p)^2 (1+2p)$$

$$p(y_{decodat} = 1 / x = 1) = \dots = (1-p)^2 (1+2p)$$

Rezulta :

$$P(E) = 1 - \sum_j p(y_j/x_j)p(x_j) = 1 - 2(1-p)^2(1+2p)\frac{1}{2} = p^2(3-2p) \approx 0,1$$

Observatii:

- probabilitatea de eroare $P(E)$ a scazut la jumata;
- se transmit de trei ori mai multe simboluri, deci debitul sursei (nr de simboluri pe secunda) trebuie micsorat astfel incat capacitatea canalului sa nu fie depasita.

b) Codarea prin repetarea de cinci ori a fiecarui simbol:

$$p(y_{decodat} = 0 / x = 0) = C_5^0(1-p)^5 + C_5^1 p(1-p)^4 + C_5^2 p^2(1-p)^3 = (1-p)^3(1+3p+6p^2)$$

$$P(E) = 1 - (1-p)^3(1+3p+6p^2) \approx 0,05$$

Observatie :

- probabilitatea de eroare $P(E)$ a scazut si mai mult, dar debitul de informatie R al sursei trebuie sa fie cel mult o cincime din capacitatea canalului C_τ (biti/sec) :

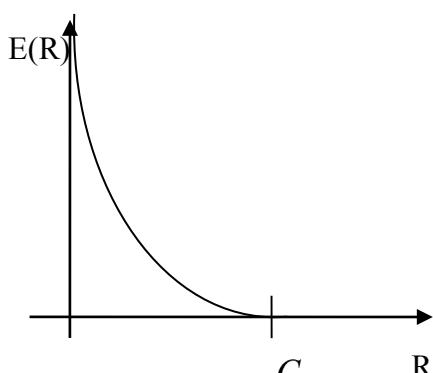
$$R \leq \frac{C_\tau}{5}$$

8.3. Teorema a 2-a a lui Shannon

Teorema: Daca avem o sursa cu un debit de informatie R si un canal cu perturbatii, cu o capacitate de transmisie $C_\tau > R$, exista un cod cu cuvinte de lungime n , astfel incat probabilitatea de eroare sa fie :

$$P(E) \leq 2^{-nE(R)}$$

unde $E(R)$ este o functie nenegativa numita *exponentul erorii*.



Observatii:

- Teorema a 2-a a lui Shannon este cunoscuta si sub numele de *Teorema codarii canalelor cu perturbatii*;
- C , trebuie intelese ca debitul maxim de informatie prin canal (cantitate maxima de informatie transmis ape secunda);
- Teorema a 2-a arata ca pe un canal se poate face o transmisie cu probabilitate de eroare $P(E)$ oricat de mica, daca debitul R se diminueaza suficient de mult;
- Intr-o aplicatie practica, daca se impune $P(E)$, cunoscand functia $E(R)$, se poate determina debitul maxim R al sursei sau, daca se impune R , se poate afla $P(E)$ minim, cu care se poate face transmisia pe canal.
- Teorema a 2-a a lui Shannon este o teorema de existenta; ea nu da solutii pentru constructia codurilor de canal.

8.4. Spatiul cuvintelor

In exemplul de la Sectiunea 8.2, fiecare simbol al sursei binare era codat printr-un cuvant de lungime 3, obtinut prin repetarea simbolului. Se obtinea astfel, o carte de cod constituuta din doua cuvinte :

Codarea : 0 -> 000
 1-> 111

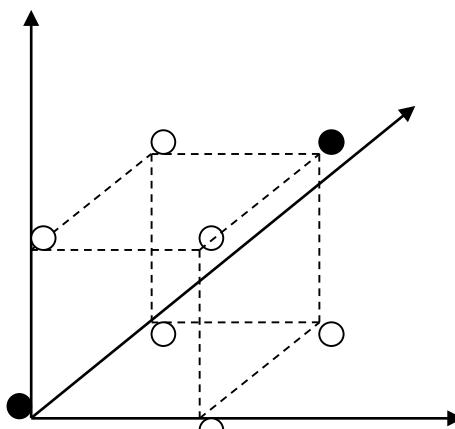
La decodare, din cauza perturbatiilor, poate fi receptionat orice cuvant de lungime 3:

Decodarea : 000->0 111->1
 001->0 110->1
 010->0 101->1
 100->0 011->1

Definitie: Cuvintele emise de codor se numesc **cuvinte cu sens**, iar restul cuvintelor de aceeasi lungime se numesc **cuvinte fara sens**. Impreuna, ele constituie multimea cuvintelor de lungime n ($n = 3$ in exemplu).

8.5. Reprezentarea grafica a cuvintelor

In exemplul mentionat, s-au folosit cuvinte de lungime 3. Intr-un spatiu 3D, aceste cuvinte pot fi reprezentate prin puncte :



Observatii :

- cuvintele cu sens sunt marcate cu negru;
- schimbarea unui bit intr-un cuvant este echivalent cu deplasarea pe una din laturile cubului, spre unul dintre cuvintele vecine;
- pentru a trece de la un cuvant cu sens la celalalt, trebuie facuti minim 3 pasi;
- decodorul cu logica majoritara din exemplu decodeaza cuvintele fara sens cautand cuvantul cu sens cel mai apropiat.

8.6. Distanța Hamming

Definitie: **Distanța Hamming** dintre două cuvinte este egală cu suma bitilor prin care cuvintele difera.

$$d_H(000,111) = 3$$

Observatie : În reprezentarea grafică, distanța Hamming este numărul minim de pași necesari pentru a trece de la un cuvant la celalalt.

R.W. Hamming (1915-1998) a lucrat la Los Alamos între 1944 și 1946 și apoi la Bell Labs și Univ. Princeton.

8.7. Erori detectabile și erori corectabile

Codurile de canal pot fi folosite pentru:

- corectarea de erori (cuvintele fara sens sunt detectate și corectate) ;
- detectarea de erori (cuvintele fara sens sunt detectate și rejectate, iar decodorul cere retransmisia cuvantului)

Codul din exemplul de la Secțiunea 8.2. poate corecta o singură eroare (numai cuvintele fara sens care difera printr-un singur bit de un cuvant cu sens sunt corectate). Dacă apar două erori, cuvantul este decodat gresit.

Cu același cod, dacă se dorește doar detectarea cuvantului fara sens, atunci pot fi detectate două erori. Spunem că avem *un cod corector de o eroare și detector de două erori*.

8.8. Specificarea cuvintelor cu sens

Cuvintele cu sens trebuie alese astfel încât *distanța Hamming minima dintre ele să fie cât mai mare*.

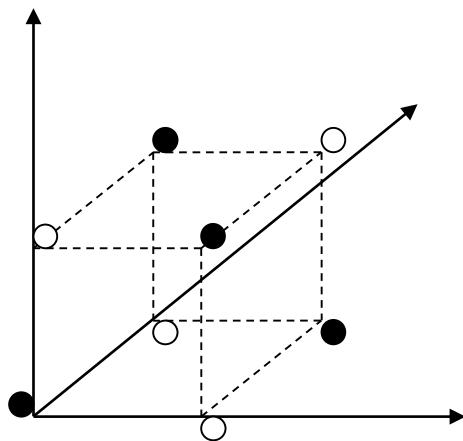
Dacă $d_{H\min} = 2e+1$, codul este corector de e erori și detector de $2e$ erori.

Dacă $d_{H\min} = 2e$, codul este corector de $e-1$ erori și detector de $2e-1$ erori.

Exemplu: Codare prin adaugarea bitului de paritate (cuvinte de lungime 3)

Codarea :

00	->	000
01	->	011
10	->	101
11	->	110



Observatii:

- distanta minima dintre cuvinte este $d_{H\min} = 2$
- este, deci, un cod care poate detecta o eroare, dar nu poate corecta
- de fapt, cu bitul de paritate se poate detecta orice numar impar de erori.

Exercitiu: Cate erori poate corecta/detecta urmatorul cod:

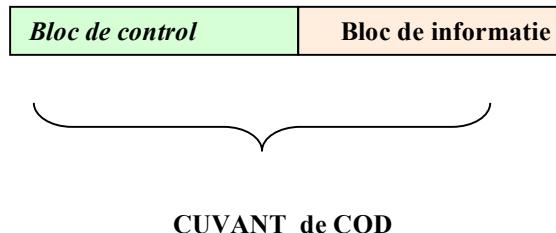
00000, 00111, 11001, 11110

9. CODURI BLOC

Clasificare codurilor corectoare/detectoare de erori :

- coduri **bloc** : - coduri **grup**
 - coduri **ciclice**
- coduri **convolutionale**

Codurile bloc se obtin taind sirul de simboluri ce urmeaza sa fie codat in blocuri de lungime fixa, numite **blocuri de informatie**, la care se adauga simboluri de control, calculate pe baza simbolurilor de informatie. Simbolurile de control constituie **blocul de control**.



Coduri bloc :

- **sistematice** (simbolurile de control sunt grupate la inceputul sau sfarsitul cuvantului)
- **nesistematice** (simbolurile de control sunt inserate in blocul de informatie)

La codarea cu **coduri convolutionale**, sirul de simboluri de informatie se prelucreaza continuu.

9.1. Coduri grup

Formalism matematic :

$$\begin{aligned} \text{blocul de informatie sau mesajul : } i &= [i_1 \quad \dots \quad i_k] \\ \text{blocul de control : } c &= [c_1 \quad \dots \quad c_m] \\ \text{cuvantul de cod : } v &= [c_1 \quad \dots \quad c_m \quad i_1 \quad \dots \quad i_k] = [v_1 \quad \dots \quad v_n] \\ \text{cuvantul de eroare : } \varepsilon &= [\varepsilon_1 \quad \dots \quad \varepsilon_n] \\ \text{cuvantul de cod eronat : } v' &= v \oplus \varepsilon \end{aligned}$$

Lungimea cuvantului de cod este $n = m + k$.

Observatii :

- cuvintele corecte sunt cuvintele de cod ; ele se mai numesc si *cuvinte cu sens*;
- cuvintele eronate se mai numesc si *cuvinte fara sens*;
- cuvantul de cod este un vector de dimensiune n
- elementele vectorilor sunt numere binare;
- cuvintele de cod apartin unui spatiu vectorial, care are o *structura de grup* in raport cu operatiile de adunare si inmultire modulo 2 (aceasta proprietate da numele de *coduri grup*):

+	0	1
0	0	1
1	1	0

X	0	1
0	0	0
1	0	1

9.1.1. Codarea

Pentru a intelege mecanismul codarii, trebuie cunoscut, mai intai, principiul corectiei/detectiei de erori. Corectia sau detectia erorilor se fac cu ajutorul **corectorilor**.

Definitie : **Corectorul** este un vector, notat cu $z = [z_1 \dots z_m]$, care se obtine pe baza simbolurilor cuvantului receptionat v' :

$$H(v') = z$$

unde H este un operator liniar.

Observatii :

- a) prin conventie, daca v' este corect, atunci corectorul este nul ; daca v' este un cuvant eronat, atunci $z \neq 0$;
- b) pentru corectie, intre multimea cuvintelor fara sens si multimea coreectorilor trebuie sa existe o *corespondenta biunivoca*;
- c) pentru detectia de erori, este suficientea conditia $. z \neq 0$

Observatia b) ne da urmatoarea regula de calcul pentru lungimea blocului de control al unui cod corector de e erori :

$$\sum_{i=1}^e C_n^i \leq 2^m - 1$$

Aceasta inegalitate traduce conditia « *Numarul total de configuratii posibile de erori trebuie sa fie mai mic sau egal cu numarul de corectori nenuli.* »

Definitie : codurile pentru care inegalitatea de mai sus devine egalitate, se numesc **coduri perfecte** sau **coduri de redundanta minima**.

In cazul particular al codurilor corectoarea de o singura eroare, aceasta conditie devine :

$$k + m \leq 2^m - 1$$

Elementele corectorului se obtin prin rezolvarea urmatorului sistem de ecuatii liniare :

$$\begin{cases} h_{11}v_1 + h_{12}v_2 + \dots + h_{1n}v_n = z_1 \\ \dots \\ h_{m1}v_1 + h_{m2}v_2 + \dots + h_{mn}v_n = z_m \end{cases}$$

care se poate scrie sub forma matriceala

$$H \cdot (v^t)^\tau = z$$

unde H este o matrice cu dimensiunea $m \times n$. H se numeste **matrice de control**.

a) Codarea cu matricea de control H

Deoarece, prin conventie, corectorul $z = 0$ corespunde cuvintelor corecte, rezulta ca putem construi cuvintele de cod rezolvand ecuatia :

$$H \cdot v^\tau = 0$$

care este echivalenta cu un sistem de m ecuatii liniare, suficiente pentru a determina cele m simboluri de control necunoscute din componenta cuvantului de cod.

Observatie:

- matricea H este predefinita
- continutul sau depinde de capacitatea de corectie/detectie a codului
- matricea H nu este unica

Forma canonica a lui H :

$$H = \begin{bmatrix} 1 & 0 & \dots & 0 & q_{11} & \dots & q_{1k} \\ 0 & 1 & \dots & 0 & q_{21} & \dots & q_{2k} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & q_{m1} & \dots & q_{mk} \end{bmatrix} = [I_m Q]$$

este utila in obtinerea codurilor sistematice, la care simbolurile de control sunt grupate la inceputul cuvantului de cod (in plus, fiecare ecuatie liniara are ca necunoscuta un singur simbol de control, ceea ce usureaza rezolvarea sistemului).

$$[I_m \quad Q] \cdot \begin{bmatrix} c^\tau \\ i^\tau \end{bmatrix} = I_m c^\tau \oplus Q i^\tau = 0 \quad \Rightarrow \quad c^\tau = Q i^\tau$$

b) Codarea cu matricea generatoare G

O cale alternativa de a obtine cuvintele de cod este prin intermediul unei matrici G, numita **matricea generatoare**. In acest caz, cuvintele se obtin rezolvand ecuatia:

$$v = iG$$

unde i este blocul simbolurilor de informatie.

Observatii:

- G are dimensiunea $k \times n$;
- Codul obtinut este nesistemtic;
- Intre G si H exista urmatoarea relatie: $H \cdot G^\tau = 0$

$$\begin{aligned} \text{Demonstratie :} \quad & \text{deoarece } H \cdot v^\tau = 0 \\ & H \cdot (iG)^\tau = 0 \\ & H \cdot G^\tau i^\tau = 0 \quad \forall i \\ \text{rezulta} \quad & H \cdot G^\tau = 0 \end{aligned}$$

- forma canonica a matricii generatoare este $G = [Q^\tau \quad I_k]$
- Demonstratie: $[I_m \quad Q] \cdot \begin{bmatrix} Q^\tau \\ I_k \end{bmatrix} = Q \oplus Q = 0$
- cu forma canonica a lui G se obtine un cod sistematic cu simbolurile de control grupate la inceputul cuvantului; blocul de control se calculeaza cu relatia $c = iQ^\tau$.
- Demonstratie: $v = i \cdot [Q^\tau \quad I_k] = [iQ^\tau \quad iI_k] = [c \quad i]$

9.1.2. Decodarea

La receptie, decodarea cuvintelor se face conform urmatoarelor etape:

a) calculul corectorului

- in cazul codului nesistematic :

$$z = Hv'$$

- in cazul codului sistematic :

$$z = Hv' = H \begin{bmatrix} c' & i' \end{bmatrix}^\tau = [I_m \quad Q] \cdot \begin{bmatrix} c' \\ i' \end{bmatrix} = I_m c' + Q i' = c' + c''$$

b) identificare erorii ε (este o etapa care difera de la cod la cod)

c) corectia cuvantului:

$$v = v' \oplus \varepsilon$$

Observatii :

- c'' este blocul de corectie recalculat pe baza simbolurilor de informatie receptionate (care pot fi gresite)

- c' este blocul de corectie receptionat

Deci, corectorul est suma dintre blocul de control receptionat si blocul de control recalculat pe baza simbolurilor de informatie receptionate.

9.1.3. Relatii intre coloanele matricii de control H

Fie vectorul eroare

$$\varepsilon = [\varepsilon_1 \quad \dots \quad \varepsilon_n]$$

si cuvantul eronat

$$v' = v \oplus \varepsilon$$

Corectorul calculat la receptie este :

$$z = Hv' = H(v^\tau \oplus \varepsilon^\tau) = H\varepsilon^\tau$$

Daca notam h_1, h_2, \dots, h_n , coloanele matricii de control, atunci corectorul este:

$$z = \sum_i h_i \varepsilon_i$$

Acest mod de a exprima corectorul pune in evidenta urmatoarele proprietati ale matricei de control (proprietati folosite in definirea sa):

a) cazul codurilor *corectoare de e erori*

- sumele oricaror e coloane ale matricii de control trebuie sa fie diferite intre ele (deoarece intre coretori si vectorii de eroare trebuie sa existe o corespondenta biunivoca). In cazul particular al codurilor *corectoare de o singura eroare*, conditia devine “coloanele lui H trebuie sa fie diferite intre ele”.

- ponderea cuvintelor de cod este minim $2e+1$ (exceptie cuvantul constituit numai din ‘0’)

Definitie : **Ponderea** unui cuvant de cod este data de numarul de simboluri ‘1’ din componenta cuvantului.

Demonstratie: fie doua cuvinte de cod, v si w ; distanta minima dintre cuvinte trebuie sa fie cel putin $2e+1$ (codul este corector de e erori), deci:

$$d(v, w) = v \oplus w$$

Dar suma $v \oplus w$ este un tot un cuvant de cod, deoarece $H(v \oplus w)^T = 0$, deci ponderea sa minima trebuie sa fie $d(v, w)$, deci $2e+1$.

b) cazul codurilor *detectoare de e erori*

- sumele oricaror e coloane ale matricii de control trebuie sa fie nenule; spre deosebire de cazul codurilor corectoare de erori, sumele pot fi identice pentru erori diferite.
- In cazul particular al codurilor detectoare de o eroare, matricea H trebuie sa aibe toate coloanele nenule. Cea mai simpla matrice, care indeplineste aceasta conditie este

$$H = [1 \ 1 \ \dots \ 1]$$

Cu aceasta matrice se obtine binecunoscutul *bit de paritate*, care este un cod detector de o eroare, de fapt, detector de un numar impar de erori (suma modulo 2 a unui nr impar de ‘1’ este ‘1’). Blocul de control – bitul de paritate - are lungime $m = 1$.

Observatii :

- bordand cu o linie de ‘1’ matricea H a unui cod corector de e erori, acesta capata si proprietate de *cod detector de un numar impar de erori*

$$H' = \begin{bmatrix} 0 & h_1 & \dots & h_n \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

- cu aceasta matrice extinsa, se calculeaza un simbol de control suplimentar, respectiv *bit de paritate*, care apare la inceputul cuvantului de cod.

9.2. Codul Hamming grup corector de o eroare

Matricea de control are pe coloane, reprezentarea binara a numarului de ordine al coloanei.

Exemplu: pentru $k = 4$ si $e = 1$, rezulta $m = 3$, deci $n = 7$.

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

Cuvantul de cod este nesistematic, el are urmatoarea structura :

$$v = [c_1 \ c_2 \ i_3 \ c_4 \ i_5 \ i_6 \ i_7]$$

Observatii :

- bitul cel mai putin semnificativ al coloanei este bitul de jos ;
- codul este nesistematic, deoarece simbolurile de control nu sunt grupate;
- simbolurile de control sunt plasate in cuvant, in dreptul coloanelor lui H, care contin un singur 1; aceasta asezare usureaza calculul simbolurilor de control ;

Pozitia, pe care apare eroarea, se calculeaza convertind in zecimal corectorul obtinut.

Justificare: fie vectorul de eroare $\varepsilon = [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0]$. Corectorul, care se obtine cand un cuvant este eronat pe pozitia a 3, este :

$$z = \sum_i h_i \varepsilon_i = h_3$$

9.3. Coduri ciclice

Codurile ciclice fac parte din categoria **codurilor bloc**. Pentru reprezentarea cuvintelor de cod, se folosesc aceleasi notatii ca la codurile grup, cu exceptia faptului ca numerotarea elementelor vectorilor incepe de la 0 (justificare in sectiunea urmatoare):

blocul de informatie : $i = [i_0 \ \dots \ i_{k-1}]$

blocul de control : $c = [c_0 \ \dots \ c_{m-1}]$

cuvantul de cod : $v = [c_0 \ \dots \ c_{m-1} \ i_0 \ \dots \ i_{k-1}] = [v_0 \ \dots \ v_{n-1}]$

cuvantul de eroare : $\varepsilon = [\varepsilon_0 \ \dots \ \varepsilon_{n-1}]$

cuvantul de cod eronat : $v' = v \oplus \varepsilon$

Elementele vectorilor $\in GF(2)$. Cu GF , notam un camp Galois.

Lungimea blocului de control se calculeaza la fel ca in cazul codurilor grup:

$$\sum_{i=1}^e C_n^i \leq 2^m - 1$$

unde e este numarul maxim de erori pe care codul le poate corecta.

9.3.1. Reprezentarea cuvintelor de cod ca polinoame

In studiul codurilor ciclice, un cuvant de lungime n se reprezinta printr-un polinom de grad $n-1$:

$$v = [v_0 \quad \dots \quad v_{n-1}] \Leftrightarrow v(x) = v_0 + v_1 x + \dots + v_{n-1} x^{n-1}$$

Observatie :

- la codurile ciclice, numerotarea elementelor binare, care constituie cuvantul, se face cu indici cuprinsi intre 0 si $n-1$ pentru a avea o notatie congruenta cu gradul polinomului; aceasta schimbare de notatie nu modifica lungimile blocurilor de informatie si control.

Blocurile de informatie si de control sunt polinoame de grad $k-1$ si, respectiv, $m-1$:

$$i(x) = i_0 + i_1 x + \dots + i_{k-1} x^{k-1}$$

$$c(x) = c_0 + c_1 x + \dots + c_{m-1} x^{m-1}$$

9.3.2. Spatiul cuvintelor

Cuvintele unui cod ciclic de lungime n sunt clase de resturi modulo $p(x) = x^n + 1$:

Clasa de resturi 0 :	$v(x) = 0$	$v(x) = p(x)$
Clasa de resturi 1 :	$v(x) = 1$	$v(x) = p(x) + 1$
Clasa de resturi $1+x$:	$v(x) = 1+x$	$v(x) = p(x) + 1+x$
.....		
$1+x+\dots+x^{n-1}$		

Observatie :

- sunt 2^n clase de resturi
- dintre ele, prin codare, se aleg 2^k cuvinte cu sens

Clasele de resturi modulo $p(x) = x^n + 1$ constituie o algebra.

9.3.3. Inmultirea claselor de resturi modulo $p(x) = x^n + 1$

Exemplu: Fie polinomul $p(x) = x^3 + 1$ si doua dintre clasele sale de resturi:

$$h(x) = h_0 + h_1 x$$

$$v(x) = v_0 + v_1 x + v_2 x^2$$

Produsul celor doua clase este tot o clasa de resturi modulo $p(x) = x^3 + 1$:

$$h(x)v(x) = h_0 v_0 + (h_0 v_1 + h_1 v_0)x + (h_0 v_2 + h_1 v_1)x^2 + h_1 v_2 x^3$$

Deoarece $p(x)$ face parte din clasa de resturi 0, rezulta ca:

$$x^3 + 1 = 0 \quad x^3 = 1$$

Deci

$$h(x)v(x) = h_0 v_0 + h_1 v_2 + (h_0 v_1 + h_1 v_0)x + (h_0 v_2 + h_1 v_1)x^2$$

Observatie:

- conditia $h(x)v(x) = 0$ este echivalenta cu

$$h_0 v_0 + h_1 v_2 = 0$$

$$h_0 v_1 + h_1 v_0 = 0$$

$$h_0 v_2 + h_1 v_1 = 0$$

unde fiecare ecuatie contine produsul scalar dintre $v = [v_0 \quad v_1 \quad v_2]$ si o permutare circulara a lui $h = [h_0 \quad h_1 \quad 0]$.

- setul de ecuatii se poate scrie sub forma matriceala astfel:

$$\begin{bmatrix} h_0 & 0 & h_1 \\ h_1 & h_0 & 0 \\ 0 & h_1 & h_0 \end{bmatrix} \cdot \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix} = 0 \quad H \cdot v^\tau = 0$$

sau

Regasim aceeasi relatie ca la codarea cu coduri grup dar o structura particulara pentru matricea H (liniile matricii sunt permute).

9.3.4. Specificarea cuvintelor cu sens folosind un polinom generator

Cuvintele cu sens se aleg astfel incat sa fie multiplii unui polinom $g(x)$, de grad m , numit *polinom generator* :

$$g(x) = g_0 + g_1x + \dots + g_mx^m$$

Polinomul $g(x)$ trebuie sa fie un divizor al lui $p(x)$:

$$p(x) = g(x)h(x)$$

9.3.5. Codarea

a) Obtinerea codului nesistematic

$$v(x) = i(x)g(x)$$

Observatii :

- gradul lui $v(x)$ este $k - 1 + m = n - 1$

b) Obtinerea codului sistematic

Polinomul care reprezinta cuvantul cu sens trebuie sa fie de forma :

$$v(x) = c(x) + x^m i(x)$$

Calculul polinomului de control $c(x)$ se bazeaza pe faptul ca orice cuvant cu sens este multiplu al lui $g(x)$:

$$\text{rest} \frac{v(x)}{g(x)} = 0 \Leftrightarrow \text{rest} \frac{c(x) + x^m i(x)}{g(x)} = c(x) + \text{rest} \frac{x^m i(x)}{g(x)} = 0$$

Deci :

$$c(x) = \text{rest} \frac{x^m i(x)}{g(x)}$$

9.3.6. Decodarea

Fie cuvantul receptionat :

$$v'(x) = v(x) + \varepsilon(x)$$

unde $\varepsilon(x)$ este un polinom de grad $n - 1$, care descrie eronarea :

$$\varepsilon(x) = \varepsilon_0 + \dots + \varepsilon_{n-1}x^{n-1}$$

Corectarea se face, ca si in cazul codurilor grup, calculand un *corector*. La codurile ciclice, corectorul este un polinom $z(x)$, de grad $m - 1$.

a) Etapele decodarii la codurile nesistematice

- calculul corectorului:

$$z(x) = \text{rest} \frac{v'(x)}{g(x)} = \text{rest} \frac{\varepsilon(x)}{g(x)}$$

- identificarea pozitiilor eronate se face prin cautare intr-un tabel predefinit, in care pe o coloana avem toate polinoamele $\varepsilon(x)$ posibile si pe cealalta, corectorii $z(x)$ corespunzatori. Acest tabel se numeste *Tablou al claselor de resturi*.

- corectarea cuvantului:

$$v(x) = v'(x) + \varepsilon(x)$$

- extragerea blocului de informatie

$$i(x) = \frac{v(x)}{g(x)}$$

b) Etapele decodarii la codurile sistematice

- cuvintele de cod sistematice fiind de forma $v(x) = c(x) + x^m i(x)$, calculul corectorului cu formula generala $z(x) = \text{rest} \frac{v'(x)}{g(x)}$ se reduce la urmatoarele doua operatii:

- se recalculeaza polinomul de control folosind simbolurile de informatie receptionate $i'(x)$:

$$c''(x) = \text{rest} \frac{x^m i'(x)}{g(x)}$$

- corectorul se obtine adunand $c''(x)$ cu blocul de control receptionat $c'(x)$:

$$z(x) = c''(x) + c'(x)$$

- se identifica pozitiilor eronate la fel ca la codurile nesistematice
- se corecteaza cuvantul:

$$v(x) = v'(x) + \varepsilon(x)$$

- se extrage blocul de informatie prin retinerea ultimilor k coeficienti ai lui $v(x)$

Observatii :

- la codurile sistematice, decodarea este mai simplu de implementat pentru ca se elimina impartirea din ultima etapa (de extragere a simbolurilor de informatie).
- in *Tabloul claselor de resturi*, fiecare rest corespunde unei singur polinom $\varepsilon(x)$, dar poate proveni din impartirea a 2^k cuvinte cu sens diferite, care au fost eronate:

	Cuvinte eroare	Cuvinte receptionate	Corectori
Cuvinte cu sens	0	$v_1(x) \dots v_{2^k}(x)$	0
Cuvinte fara sens	$\varepsilon_1(x)$	$\varepsilon_1(x) + v_1(x) \dots \varepsilon_1(x) + v_{2^k}(x)$	$rest \frac{\varepsilon_1(x)}{g(x)}$
	$\varepsilon_2(x)$	$\varepsilon_2(x) + v_1(x) \dots \varepsilon_2(x) + v_{2^k}(x)$	$rest \frac{\varepsilon_2(x)}{g(x)}$

Justificare pentru calculul corectorului la codurile sistematice:

$$z(x) = c'(x) + c''(x) = c'(x) + rest \frac{x^m i'(x)}{g(x)}$$

$c'(x)$ fiind de grad $< m$, relatia anterioara se poate scrie:

$$z(x) = rest \frac{c'(x) + x^m i'(x)}{g(x)} = rest \frac{v'(x)}{g(x)} = rest \frac{\varepsilon(x)}{g(x)}$$

ceea ce arata ca este vorba de acelasi corector ca si cazul codului nesistemtic.

9.3.7. Codarea folosind polinomul $h(x)$

Din conditia $p(x) = g(x)h(x)$, dat fiind $p(x)$ face parte din clasa de resturui zero, rezulta ca:

$$h(x)v(x) = h(x)i(x)g(x) = 0$$

Ceea ce arata ca relatia $h(x)v(x)=0$ este o modalitate alternativa de obtinere a cuvintelor cu sens.

Observatii:

- $h(x)$ este un polinom de grad k :

$$h(x) = h_0 + h_1x + \dots + h_kx^k$$

- in calculul produsului $h(x)v(x)$ trebuie tinut seama de faptul ca $h(x)$ si $v(x)$ sunt clase de resturi modulo $p(x)$ si ca rezultatul este tot o clasa de resturi modulo $p(x)$.

9.3.8. Codarea folosind calculul matriceal

- a) Codarea cu polinomul generator $g(x)$

Relatia $v(x)=i(x)g(x)$ se poate scrie :

$$v(x) = i_0g(x) + i_1xg(x) + \dots + i_{k-1}x^{k-1}g(x)$$

Daca notam

$$G = \begin{bmatrix} g(x) \\ xg(x) \\ \dots \\ \dots \\ x^{k-1}g(x) \end{bmatrix} = \begin{bmatrix} g_0 & g_1 & \dots & g_m & 0 & \dots & 0 \\ 0 & g_0 & g_1 & \dots & g_m & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & g_0 & g_1 & \dots & g_m \end{bmatrix}$$

atunci inmultirea de polinoame $v(x)=i(x)g(x)$ se poate scrie ca o inmultire de matrice:

$$v = i \cdot G$$

unde

$$v = [v_0 \quad v_1 \quad \dots \quad \dots \quad v_{n-1}]$$

$$i = [i_0 \quad i_1 \quad \dots \quad \dots \quad i_{k-1}]$$

b) Cealalta modalitate de obtinere a cuvintelor cu sens, respectiv $h(x)v(x)=0$, se poate scrie sub forma matriceala astfel (generalizarea exemplului din Sectiunea 8.2.3.):

$$\begin{bmatrix} h_0 & 0 & \dots & \dots & \dots & h_2 & h_1 \\ h_1 & h_0 & 0 & \dots & \dots & h_3 & h_2 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & h_k & \dots & \dots & h_0 & 0 \\ \dots & \dots & 0 & h_k & \dots & h_1 & h_0 \end{bmatrix} \cdot \begin{bmatrix} v_0 \\ v_1 \\ \dots \\ \dots \\ v_{n-1} \end{bmatrix} = 0$$

Daca v sunt cuvinte de cod sistematice, numai m elemente trebuie determinate (simbolurile de control). In acest caz, se lucreaza cu o matrice redusa la m linii:

$$H = \begin{bmatrix} h_k & h_{k-1} & \dots & \dots & \dots & 0 & 0 \\ 0 & h_k & h_{k-1} & \dots & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & h_k & \dots & \dots & h_0 & 0 \\ \dots & \dots & 0 & h_k & \dots & h_1 & h_0 \end{bmatrix}$$

Observatie:

- ecuatia matriceala este echivalenta celei de la codarea codurilor grup folosind matricea de control H

$$Hv^\tau = 0$$

- intre G si H , exista aceeasi relatie ca la codurile grup :

$$GH^\tau = HG^\tau = 0$$

9.3.9. Doua proprietati ale codurilor ciclice

Proprietatea 1: Orice permutare circulara a unui cuvant de cod este tot un cuvant de cod.

Justificare :

Fie cuvantul de cod $v = [v_0 \dots v_{n-1}]$ reprezentat prin $v(x) = v_0 + v_1x + \dots + v_{n-1}x^{n-1}$. Produsul $xv(x)$ corespunde permutarii circulara dreapta, cu o pozitie, a cuvantului v :

$xv(x) = v_0x + v_1x^2 + \dots + v_{n-1}x^n = v_{n-1} + v_0x + v_1x^2 + \dots + v_{n-2}x^{n-1}$, deoarece in clasele de resturi modulo $p(x)$, avem $x^n = 1$.

Din $h(x)v(x) = 0$ (conditie indeplinita de toate cuvintele de cod), rezulta ca $h(x)xv(x) = 0$, ceea ce arata ca o permutarea dreapta a lui v este tot un cuvant de cod. Iterand, orice permutare circulara este tot un cuvant de cod.

Proprietatea 2: Orice combinatie liniara de cuvinte de cod este tot un cuvant de cod.

Justificare:

Fie doua cuvinte de cod reprezentate prin polinoamele $v(x)$ si $u(x)$. Cuvantul $w(x) = av(x) + bu(x)$ - cu a si $b \in GF(2)$ - este un cuvant de cod deoarece

$$h(x)w(x) = h(x)[av(x) + bu(x)] = ah(x)v(x) + bh(x)u(x) = 0$$

Observatie:

- numele de *coduri ciclice* este dat de Proprietatea 1.

9.3.10. Exemplu: Codarea si decodarea folosind un cod ciclic corector de o eroare, pentru blocuri de informatie de lungime $k = 4$.

Calculul lungimii blocului de control :

Din conditia $m + k \leq 2^m - 1$, rezulta $m = 3$ si $n = 7$

Codare cu cod sistematic obtinut cu un polinom generator $g(x)$

Alegerea polinomului generator :

Divizorii lui $p(x) = x^7 + 1$: $p(x) = (x+1)(x^3 + x + 1)(x^3 + x^2 + 1)$

Dintre divizorii lui $p(x)$, se alege un polinom $g(x)$ de grad $m = 3$. De exemplu, $g(x) = 1 + x^2 + x^3$

Calculul blocului de control :

Pentru fiecare polinom de informatie, se calculeaza polinomul de control cu relatia $c(x) = \text{rest} \frac{x^3 i(x)}{g(x)}$. De exemplu, pentru $i = [0 \ 0 \ 1 \ 0]$, rezulta $i(x) = x^2$ si $c(x) = 1 + x$, deci $c = [1 \ 1 \ 0]$

Asamblarea cuvantului de cod:

$$v(x) = c(x) + x^3 i(x) = 1 + x + x^5$$

$$\text{deci } v = [1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0]$$

Constituirea cartii de cod:

Se construieste un tabel cu 2^4 linii (numarul de blocuri de informatie posibile) si doua coloane : bloc de informatie si cuvant de cod corespunzator.

Constituirea Tabloului claselor de resturi pentru corectia erorilor:

Se construieste un tabel cu $n+1 = 8$ linii (numarul de corectori distincti) si doua coloane: eroarea $\varepsilon(x)$ si corectorul $z(x)$. Corectorul se obtine cu relatia $z(x) = \text{rest} \frac{\varepsilon(x)}{g(x)}$.

Decodarea cuvintelor receptionate, de exemplu a cuvantului $v' = [0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1]$:

- se recalculeaza polinomul de corectie folosind polinomul de informatie receptionat
$$c''(x) = \text{rest} \frac{x^3 i'(x)}{g(x)} = \text{rest} \frac{x^3 (1+x^3)}{g(x)} = 1+x$$
- se calculeaza corectorul $z(x) = c''(x) + c'(x) = 1+x + x + x^2 = 1+x^2$
- se identifica $\varepsilon(x)$ prin cautare in *Tabloul claselor de resturi pentru corectia erorilor*
- se corecteaza cuvantul: $v(x) = v'(x) + \varepsilon(x)$

Alternativa 2: Codare folosind polinomul de control $h(x) = (x+1)(x^3 + x + 1)$

Alternativa 3: Codare cu cod nesistematic folosind polinomul generator $g(x)$

Alternativa 4: Codare folosind matricea generatoare G

Alternativa 5: Codare folosind matricea de control H

10. CODURI CORECTOARE DE ERORI MULTIPLE

Ce sunt erorile multiple si cum apar?

In stocarea datelor pe diverse suporturi, pot aparea erori din cauza neomogenitatii mediului de stocare sau, in timp, din cauza uzurii. Aceste erori nu sunt izolate, ci apar grupate in *trenuri de erori* (error burst). Acelasi efect apare si in transmisiile radio, de exemplu, din cauza interferentelor.

Exemplu de eroare multipla, care apare sub forma unui tren de lungime 7:

...00001011001000000... (cu 1, sunt marcati bitii eronati)

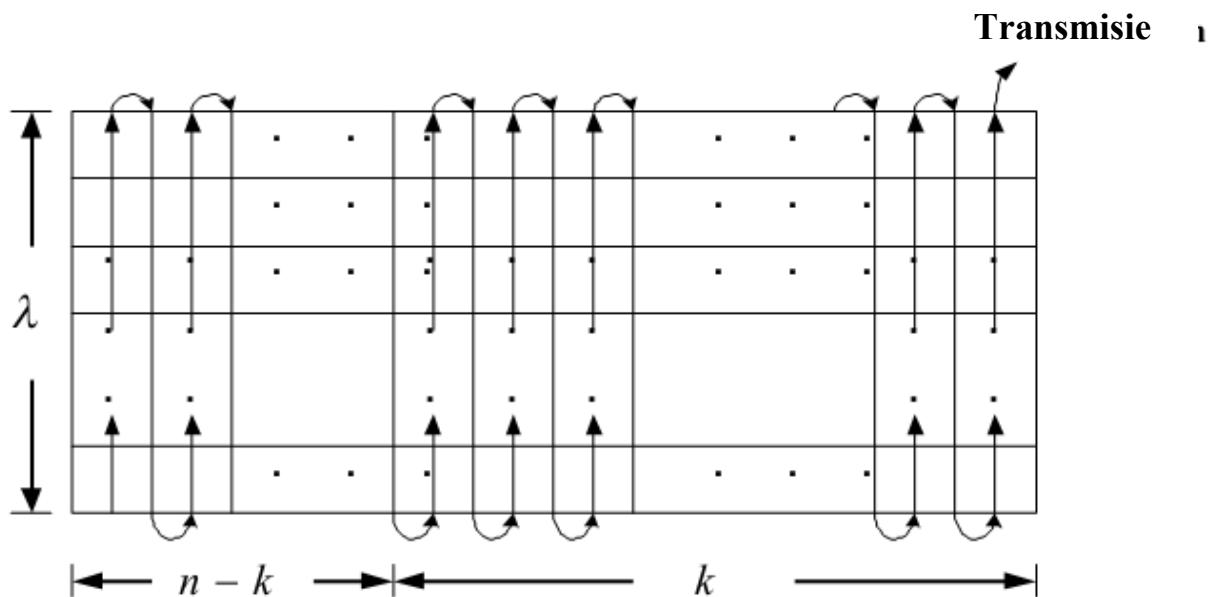
Cateva coduri si tehnici de corectie a erorilor multiple

- intreteserea
- codurile Reed-Solomon
- codurile convolutionale
- alte solutii : coduri Fire, coduri concatenate, coduri in cascada etc.

10.1. Intreteserea

Intreteserea este o tehnica care permite corectarea erorilor multiple folosind coduri bloc corectoare de o singura eroare.

Exemplu: corectarea unui tren de lungime λ , folosind un cod bloc corector de o eroare (n este lungimea cuvintelor de cod, k este lungimea blocului de informatie)



Zona intretesuta: λ cuvinte de cod de lungime n .

10.2. Coduri Reed-Solomon

Aplicatii:

- Stocare (CD, DVD, coduri de bare etc)
- Comunicatii fara fir (telefoane celulare)
- Comunicatii prin satelit
- Televiziune digitala
- Modemuri de mare viteza (ADSL, xDSL etc.)

Alfabetul codurilor Reed-Solomon are un numar mare de simboluri (>2). Alfabetul este un camp Galois.

Campul este o structura algebraica in care operatiile de adunare si inmultire satisfac regulile obisnuite (asociativitate, comutativitate, existenta unui element neutru, existenta unui element simetric, distributivitate).

Exemple de campuri: multimea numerelor rationale, multimea numerelor reale, multimea numerelor complexe.

Campul Galois sau **campul finit** este un camp cu un numar finit de elemente.

Exemplu de camp Galois : multimea claselor de resturi modulo p , unde p este un numar prim.

Conditie de existenta pentru campurile Galois (Evariste Galois 1811-1832) : numarul de elemente ale campului trebuie sa fie egal cu puterea unui numar prim $q = p^r$. Pentru orice $q = p^r$, exista un camp Galois si acesta este unic.

Constructia campului Galois $GF(q)$:

- Fie F_0 campul Galois constituit din clasele de resturi modulo numarul prim p .
- se alege un polinom ireductibil de grad r in F_0 : $f(X) = X^r + c_{r-1}X^{r-1} + \dots + c_0$
- Elementele lui $GF(q)$ sunt toate expresiile:
 $x_0 + x_1a + x_2a^2 + \dots + x_{r-1}a^{r-1}$, unde a satisface $f(a) = 0$ si $x_0, x_1, x_2, \dots, x_{r-1} \in F_0$.

Exemplu :

Constructia unui camp Galois de ordinul $9=3^2$ folosind polinomul $f(X) = X^2 + 1$, care este ireductibil in campul Galois al claselor de resturi modulo 3. Elementele campului sunt polinoame de forma $x_0 + x_1a$, unde $a^2 = 2$, unde $x_0, x_1 = 0, 1, 2$.

Adunarea si inmultirea elementelor $2 + a$ si $2 + 2a$:

$$(2 + a) + (2 + 2a) = 4 + 3a = 1 + 0 = 1$$
$$(2 + a)(2 + 2a) = 4 + 6a + 2a^2 = 1 + 0 + 1 = 2$$

Observatie:

Pentru $r = 1$, elementele campului Galois sunt resturile modulo p , respectiv valorile $0, 1, 2, \dots, p - 1$.

Parametrii codurilor Reed-Solomon :

q este dimensiunea alfabetului Σ_q ; trebuie sa fie puterea unui numar prim ($q = p^r$)

n este lungimea cuvantului de cod ($n < q$)

k este lungimea blocului de informatie (mesajul)

Observatie:

- codurile Reed-Solomon sunt coduri bloc, ca si codurile Hamming grup si codurile ciclice. Diferenta este ca simbolurile nu mai sunt binare.

Codarea mesajului:

1. Se genereaza alfabetul Σ_q ;
2. Se iau n simboluri distincte $\alpha_1, \alpha_2, \dots, \alpha_n$ ale alfabetului ;
3. Fie mesajul i_0, i_1, \dots, i_{k-1} constituit din simboluri ale alfabetului Σ_q . Se construieste polinomul $I(x) = i_0 + i_1x + \dots + i_{k-1}x^{k-1}$.
4. Cuvantul de cod asociat mesajului se obtine calculand polinomul in cele n valori selectate :

$$[I(\alpha_1) \quad I(\alpha_2) \quad \dots \quad I(\alpha_n)]$$

Proprietati ale codurilor Reed-Solomon

1. Codurile Reed-Solomon sunt coduri liniare, adica orice combinatie liniara de cuvinte de cod este tot un cuvant de cod.
2. Distația minima (in nr de simboluri) dintre oricare două cuvinte de cod este $n - k + 1$.

Justificare : două polinoame $I_1(x)$ și $I_2(x)$ pot coincide în maxim $k-1$ puncte și să ramane în același timp distincte (din rezolvarea ecuației $I_1(x) = I_2(x)$). Rezulta că există minim $n-(k-1)=n-k+1$ valori $I_1(\alpha)$ care nu coincid cu $I_2(\alpha)$. Numărul maxim al erorilor care pot fi corectate este deci $\left\lfloor \frac{n-k+1}{2} \right\rfloor$ (multiplicat cu numărul de biti cu care sunt reprezentate simbolurile alfabetului).

Teorema : pentru orice q , putere a unui numar prim, și orice $k \leq n \leq q$, există un cod Reed-Solomon cu parametrii $[n, k, n - k + 1]_q$.

Exemplu :

O secventa de bytes este impartita in blocuri de cate 240 de bytes. Fiecare bloc este codat folosind un alfabet Σ_{256} (256 este puterea unui numar prim : 2^8). Se construiesc cuvinte de lungime 256, considerand toate elementele alfabetului. Distanta minima dintre cuvinte este de $256-240+1=17$ simboluri. Simbolurile fiind bytes, rezulta ca se pot corecta trepturi de maxim $\left\lfloor \frac{256-240+1}{2} \right\rfloor \times 8 = 64$ de erori.

Codurile Reed-Solomon in contextul codurilor ciclice

Codurile Reed-Solomon pot fi construite in doua moduri:

- Evaluand polinomul $I(x)$ in n puncte, ca mai sus ; aceasta este modul in care au fost propuse de Reed si Solomon in articolul lor din 1960, *Polynomial Codes over Certain Finite Fields*.
- Folosind teoria codurilor ciclice : codurile Reed-Solomon fac parte din clasa mai larga a codurilor ciclice, corectoare de erori multiple, numite coduri BCH (Bose-Chaudhuri-Hocquenghem)

In contextul teoriei codurilor ciclice, cuvintele de cod Reed-Solomon se obtin prin aceeasi procedura ca si codurile Hamming ciclice corectoare de o eroare, cu deosebirea principala ca simbolurile nu mai sunt binare ci apartin unui camp Galois $GF(q)$.

Codarea mesajului folosind teoria codurilor ciclice :

Fie mesajul i_0, i_1, \dots, i_{k-1} , unde i_i sunt simboluri ale alfabetului Σ_q . Cu aceste simboluri, se construieste polinomul :

$$i(x) = i_0 + i_1x + \dots + i_{k-1}x^{k-1}$$

Cuvintele de cod Reed-Solomon se obtin prin inmultirea polinomului reprezentand blocul de informatie cu un polinom generator $g(x)$:

$$v(x) = i(x)g(x)$$

Corectia cuvantului de cod :

Fie $v'(x)$ cuvantul de cod receptionat la iesirea din canalul cu perturbatii. Intre $v'(x)$ si $v(x)$, exista urmatoarea relatie:

$$v'(x) = v(x) + \varepsilon(x)$$

unde $\varepsilon(x)$ este un polinom de acelasi grad ca si $v(x)$ si cu coeficienti in Σ_q :

$$\varepsilon(x) = e_0 + e_1x + \dots + e_{n-1}x^{n-1}$$

Coefficientii nuli ai lui $\varepsilon(x)$ corespund simbolurilor corecte din $v'(x)$, iar cei nenuli simbolurilor eronate. Cum, in cazul codurilor Reed-Solomon, simbolurile nu sunt binare, pentru corectie nu mai este suficiente aflarea pozitiei erorilor, ca la codurile Hamming ciclice, ci trebuie calculata si valoarea erorii, care poate fi orice element din Σ_q .

Pentru corectia erorilor, se calculeaza un sindrom, pe baza caruia se determina pozitia erorilor si valoarea lor.

Calculul corectorulu (sindromului)

Fie $\beta_1, \beta_2, \dots, \beta_{n-k-1}$ radacinile polinomului generator $g(x)$.

Sindromul este dat de setul de valori $v'(\beta_j)$ cu $j = 1, \dots, n - k$.

Daca $v'(x)$ este corect, atunci toate valorile sindromului vor fi nule pentru ca β_j sunt radacini ale polinomului generator si $v(x) = i(x)g(x)$.

In caz contrar, rezulta ca $v'(x)$ este eronat si trebuie facuta corectia erorilor.

Din sistemul de ecuatii $v'(\beta_j) = \varepsilon(\beta_j)$ se pot afla afila coefficientii e_j nenuli si

pozitiile lor (polinomul eroare $\varepsilon(x)$ are cel mult $\left\lfloor \frac{n-k+1}{2} \right\rfloor$ coefficienti nenuli

pentru ca acesta este capacitatea de corectie a codului reed-Solomon). Cele

$n - k$ ecuatii sunt suficiente pentru a afla aceste doua categorii de necunoscute,

cand numarul de erori este cel mult $\left\lfloor \frac{n-k+1}{2} \right\rfloor$.

10.3. Coduri convolutionale

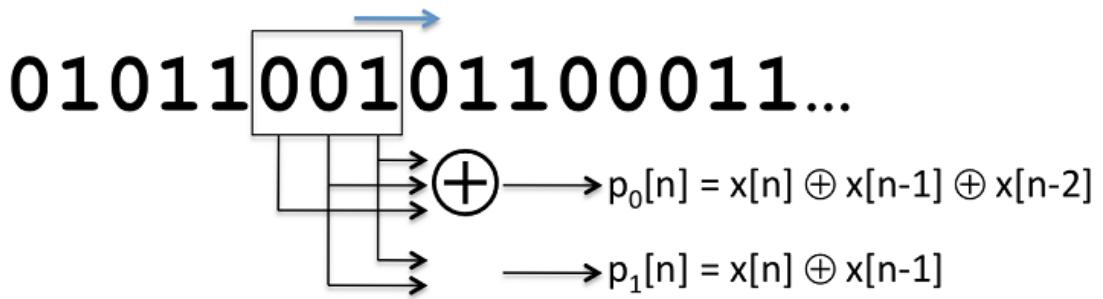
10.3.1. Codarea

Codarea consta, ca si in cazul codurilor bloc, in calcularea si transmiterea unor simboluri de control.

Sunt doua deosebiri importante fata de codarea cu coduri bloc :

- Mesajul (sirul de simboluri de informatie) nu se partitioneaza in blocuri de informatie, ci se foloseste o fereastra care gliseaza de-a lungul sirului, pentru a calcula simbolurile de control la un moment dat ;
- Pe canal, se transmit numai simbolurile de control, nu si blocul de informatie ca in cazul codurilor bloc.

Exemplu :



In acest exemplu, fereastra are dimensiunea 3. Pentru fiecare pozitie a sa, se calculeaza 2 biti de control, folosind relatiile din figura. Sirul codat va fi : 1 1 0 1 0 0 0 1 1 0 1....

Parametrii codurilor convolutionale :

- *Lungime de constrangere* este data de dimensiunea ferestrei (3 in exemplu) ;
- *Rata codului* este $1/r$, unde r este numarul simbolurilor de control calculate pentru fiecare pozitie a ferestrei.

Numarul simbolurilor de control, care depind de un anumit simbol de informatie, este egal cu *lungimea de constrangere*. Cu cat acest numar este mai mare, cu atat capacitatea de corectie a erorilor este mai ridicata, dar si decodarea devine mai complicata.

Capacitatea de corectie creste cu r . Dezavantajul, in acest caz, este marirea sirului codat, deci ocuparea canalului.

Calculul simbolurilor de control, adica a sirului codat, este descris matematic printr-o *operatie de convolutie*, intre sirul de simboluri de informatie (mesaj) $x(n)$ si un set de polinoame generatoare G_i :

$$c_i(n) = \sum_{j=0}^{k-1} G_i(j)x(n-j)$$

In exemplul de mai sus, codarea se face cu doua polinoame generatoare : G_1 cu coeficientii (1,1,1) si G_2 cu coeficientii (1,1,0). Numarul de polinoame generatoare este egal cu numarul de biti de control r , generati pentru fiecare pozitie a ferestrei de lungime k .

Polinoame generatoare pentru $r=2$ si diferite lungimi de constrangere k :

Constraint length	G_1	G_2
3	110	111
4	1101	1110
5	11010	11101
6	110101	111011
7	110101	110101
8	110111	1110011
9	110111	111001101
10	110111001	1110011001

Capacitatea de corectie

In cazul codurilor convolutionale, nu mai vorbim de *cuvinte de cod* ci de *secvente codate*.

Distanta libera (free distance) d este Distanta Hamming minima dintre toate secventele codate, de aceeasi lungime. Numarul de erori care pot fi corectate se determina din relatia:

$$d=2e+1$$

Daca d este par, atunci e se obtine prin rotunjirea inferioara a rezultatului.

Condurile convolutionale pot corecta si trenuri de erori, nu numai erori disparate. Capacitatea de corectie se mai exprima si ca lungimea maxima a trenului de erori, aparut la inceputul codarii, ce poate fi corectat.

Exemplu :

Cod convolutional cu $k=3$ si $r=2$.

De la stanga la dreapta, in tabel : mesajele posibile (msg), secventele codate corespunzatoare (X_{mit}), secventa receptionata ($Rcvd$), distantele Hamming (d) dintre secventa receptionata si secventele valide (rezultate din codarea mesajelor).

Mesajul se extinde cu cate doua zerouri la stanga si la dreapta, inainte de a incepe codarea.

Secventa receptionata se decodeaza in mesajul corespunzator distantei Hamming celei mai scurte. In acest exemplu, se corecteaza doua erori consecutive.

Msg	Xmit *	Rcvd	d
0000	000000000000		7
0001	000000111110		8
0010	000011110000		8
0011	000011010110		4
0100	001111100000		6
0101	001111011110		5
0110	001101001000		7
0111	001100100110		6
1000	111110000000	111011000110	4
1001	111110111110		5
1010	111101111000		7
1011	111101000110		2
1100	110001100000		5
1101	110001011110		4
1110	110010011000		6
1111	110010100110		3

10.3.2. Decodarea

Exista mai multi algoritmi de decodare (gasire a celei mai apropiate secvente valide) a codurilor convolutionale.

Pentru valori relativ mici ale lui *lungimii de constrangere*, **algoritmul Viterbi de decodare** este universal folosit deoarece permite gasirea secventei celei mai probabile si se preteaza la calculul paralel. Codurile decodabile Viterbi, concatenate cu coduri Reed-Solomon de lungime mare, duc la probabilitati de eroare foarte scazute. Aceasta solutia fost folosita de programul spatial Voyager.

Cand lungimea de constrangere este mare, sunt mai potriviti algoritmii de decodare secventiale. Dintre acestia, cel mai cunoscut este *algoritmul lui Fano*. Spre deosebire de decodarea Viterbi, algoritmii secventiali nu ajung la secventa optimala, dar au avantajul unei complexitatii care creste incet cu lungimea de constrangere (complexitatea algoritmului Viterbi creste exponential cu lungimea de constrangere). Acesti algoritmi au fost au fost folositi in programul spatial Pioneer, de la inceputul anilor 70.

BIBLIOGRAFIE in LIMBA ROMANA

1. Al. Spătaru, *Teoria Transmisiunii Informației*, Editura Didactică și Pedagogică, București, 1983.
2. Al. Spătaru, *Fondaments de la Theorie de la Transmission de l'Information*, Presses Polytechnique Romandes, 1987.
3. A.T. Murgan, *Principiile Teoriei Informației în Ingineria Informației și a Comunicațiilor*, Editura Academiei Romane, București, 1998.
4. Valeriu Munteanu, *Teoria Transmiterii Informației*, Editura “Gh. Asachi”, Iași, 2001.
5. A.T. Murgan, Iulia Spanu, Inge Gavat, I. Sztojanov, V.E. Neagoe, Adriana Vlad, *Teoria Transmisiunii Informației. Probleme*, Editura Didactică și Pedagogică, București, 1983.

BIBLIOGRAFIE in LIMBI STRAINE

1. F. Auger, *Introduction a la théorie du signal et de l'information*, Edition Technip, Paris, 1999.
2. A. Papoulis, *Probability, Random Variables and Stochastic Processes*, McGraw Hill, 1987.
3. Thomas M. Cover, Joy A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 1991.
4. Stanford university
<https://web.stanford.edu/~montanar/RESEARCH/BOOK/partA.pdf>
<https://ee.stanford.edu/~gray/it.pdf>