Curs TTI 2003 (sem. vara)

Prof. Dr. ing. Inge Gavat

I Masura informatiei in sisteme discrete

1. **Formularea problemei**
2. **Cantitatea de informatie in cazul discret**
   - **Informatia proprie**
   - **Informatia mutuala**
1. **Entropia informationala**

II Surse discrete de informatie

1. **Definitii si terminologie**
2. **Tipuri de surse discrete**
   - Sursa discreta cu memorie
   - Sursa Markov
   - Sursa stationara
   - Sursa ergodica
   - Sursa cu debit controlabil
   - Sursa cu debit necontrolabil
3. **Parametrii surselor discrete**
   - Entropia surselor discrete
     - Entropia sursei discrete, fara memorie, ergodica
     - Entropia sursei extinse
     - Entropia sursei discrete, cu memorie, ergodica, de tip Markov
       - Sursa Markov ergodica
       - Entropia unei surse cu memorie
       - Entropia unei surse stationare:
       - Entropia unei stari *Sj* este:
       - Entropia pentru sursa Markov ergodica, unifilara
   - Debitul de informatie
   - Cantitatea de decizie a sursei
   - Redundanta sursei
   - Eficienta sursei
4. **Exemple de surse discrete si entropiile lor**

III Canale discrete

1. **Entropia la intrarea si iesirea din canal**
   - Entropia campului reunit intrare – iesire
2. **Entropii conditionate**
   - Relatiile entropiilor conditionate cu entropiile proprii
3. **Transinformatia**

- Capacitatea canalului

4. **Parmetrii canalului discret**
5. **Modele de canale discrete**
   - Canal binar simetric
   - Canal binar cu anulari
   - Canal binar cu erori si anulari
   - Canal uniform
   - Canal dublu uniform:
   - Canal simetric:
6. **Capacitatea canalelor discrete**
   - Canal discret general, fara memorie
   - Canal binar general (n = m = 2)
     - Canal binar simetric
7. **Exemple de canale discrete**
8. **Canale discrete cu constrangeri**
   - Caracterizarea canalelor cu constrangeri

IV Masura informatiei in sisteme continue

1. **Transinformatia in canale continue**
2. **Capacitatea canalului continuu**
3. **Variatia entropiei cu schimbarea coordonatelor**

V Receptoare de simboluri discrete

1. **Matricea strategiei de decizie a receptorului**
2. **Matricea de tranzitie a canalului echivalent**
3. **Criteriul riscului minim**
   - Criteriul lui Bayes in cazul binar
     - Criteriul probabilitatii *a posteriori* maxime,
     - Criteriul plauzabilitatii maxime,
4. **Criteriul minimax**

VI Codarea de sursa (pentru canale fara perturbatii)

1. **Obiectivul codarii**
2. **Tipuri de coduri de sursa**
   - **Coduri unic decodabile**
   - **Coduri separabile**
   - **Coduri instantanee**
3. **Reprezentarea codurilor**
4. **Eficienta codarii**
   - Lungimea medie a cuvantului de cod
     - Limita inferioara a lungimii medii:
5. **Parametrii codului**

- Capacitatea codului
- Eficienta codului
- Redundanta codului

6. **Coduri absolut optimale si optimale**
   - Coduri absolut optimale
   - Coduri optimale
   - Teorema I-a a lui Shannon
   - Procedee de codare optimala
     - Principii generale:
     - Codare simbol cu simbol (n = 1)
     - Procedeul Shannon – Fano:
     - Procedeul de codare Huffman

VII Codarea pentru canalele cu perturbatii

1. **Obiectivul codarii**
2. **Categorii de coduri**
   - Coduri bloc:
     - Coduri grup:
     - Coduri ciclice:
     - Coduri convolutionale (recurente):
3. **Teorema a II-a a lui Shannon:**
4. **Coduri grup binare**
     - Cuvintele ca elemente ale claselor “alaturate”
5. **Distanta Hamming**
   - Decizia pe baza distantei minime
   - Regiuni de decizie
6. **Cuvantul eroare**
   - Erori
     - Erori individuale
     - Erori in pachete

VIII Mecanismul de detectie si de corectie a erorilor

1. **Corectori**
2. **Conditia de corectia a erorilor**
3. **Mecanismul de detectie sau corectia erorilor**
4. **Matricea de corectia a erorilor**
5. **Codarea codurilor grup cu ajutorul matricei de control H**
   - Coduri sistematice
   - Relatii intre coloanele matricei **H** in cazul corectiei erorilor
     - Relatii intre coloanele matricei H in cazul detectiei erorilor
   - Codarea codurilor grup cu ajutorul matricei generatoare **G**

- Formarea corectorilor la codarea cu matricea G
- Codul Hamming grup corector de o eroare
  - Codarea codului Hamming
  - Decodarea codului Hamming

IX Coduri ciclice

1. **Codarea cuvintelor de cod ca elemente in multimea cuvintelor cu sens generata de $g(x)$**
2. **Codarea cuvintelor de cod cu ajutorul polinomului de control $h(x)$**
3. **Decodarea codurilor ciclice**

Anexa A1: modul de generare a cuvintelor

X Circuite de codare si decodare a codurilor ciclice

1. Circuite de divizare prin polinomul $g(x)$, pentru codarea si decodarea codurilor ciclice.

- Codor cu circuite de divizare
- Decodor cu circuite de divizare

2. Registre de deplasare cu reactie pe baza polinomului $g(x)$, pentru codarea si decodarea codurilor ciclice

- Codor cu registru de deplasare cu reactie
- Decodor cu registru de deplasare cu reactie

XI Coduri convolutionale

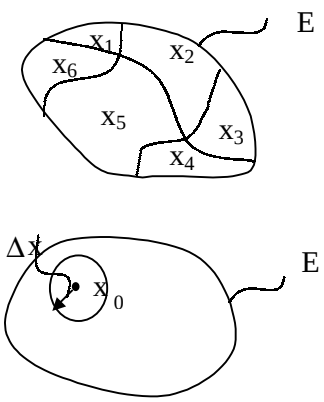
1. **Codarea**
   - Reprezentarea grafica a codurilor convolutionale
     - Diagrama de stari
     - Diagrama trellis (reprezentarea evolutiei starilor in timp)
2. **Decodarea**
     - algoritmul Viterbi

Anexa A2: Unele aplicatii ale teoriei codurilor

I Masura informatiei in sisteme discrete (Shannon,1950)

1. Formularea problemei

- Daca se urmareste un experiment care poate conduce la mai multe rezultate, asupra realizarii fiecaruia din rezultate planeaza o anumita incertitudine; incertitudinea este eliminata in momentul aflarii rezultatului; exemple de experimente: aruncarea zarului, masurarea unei tensiuni.
- Primirea unei informatii este echivalenta cu eliminarea unei incertitudini;
- Obtinerea informatiei este legata de caracterul intamplator (aleator, stochastic) al fenomenului sau experimentului observat si de aceea unui eveniment (rezultat) i se asociaza o masura probabilistica;



- Daca rezultatele sunt discrete, ca in figura, fiecarui eveniment x_i i se poate asocia probabilitatea $p(x_i)$;
- Daca rezultatele sunt continue se poate asocia o densitate de probabilitate $p(x)$ unui punct din spatiul E al esantioanelor astfel incat $p(x)dx = \lim_{\Delta x \to 0} P[x_0 - \Delta x \le X \le x_0 + \Delta x]$;

- Daca informatia asupra realizarii unui eveniment x_i rezulta chiar din observarea acelui eveniment, se obtine **informatia proprie** $i(x_i)$; Exemplu: in experimentul cu aruncarea zarului se afla (vede) nemijlocit fata care a aparut;
- Daca informatia asupra realizarii unui eveniment x_i rezulta din observarea altui eveniment y_i, legat de x_i ,se obtine **infomatia mutuala** $i(x_i,y_j)$; Exemplu: se afla ce fata a aparut din comentariile cuiva care a vazut-o nemijlocit.
- In teoria lui Shannon se iau in consideratie numai aspectele cantitative ale informatiei, nu si cele calitative, legate de sensul (semantica) mesajului.

2. **Cantitatea de informatie in cazul discret**

- **Informatia proprie; unitati de masura**

$[\mathbf{X}]=[x_1, x_2 ... x_n] \qquad \bigcup_{i=1}^{n} x_i = E \qquad x_i \cap x_j = \phi \quad \forall i \neq j$

$[\mathbf{P}_x]=[p(x_1), p(x_2), ... p(x_n)] \qquad \sum_{i=1}^{n} p(x_i) = 1$

$U(x_i)$ = incertitudinea ce planeaza asupra realizarii x_i ; se mai numeste si incertitudine *a priori.*

$U(x_i) = F[p(x_i)]$

$i(x_i) =$ informatia proprie, rezultata prin realizarea si observarea x_i

$i(x_i) = \quad U(x_i) = F[p(x_i)]$

Criterii de alegere a *F:*

$i(x_i) \geq 0$

$i(x_i) \uparrow$ *incertitudinea* $\uparrow$

$i(x_i)$ sa aiba proprietatea de aditivitate:

$\mathbf{x}_i = [x_{i1}, x_{i2}]$ independente

$i(x_i) = i(x_{i1}) + i(x_{i2})$

$U(x_i) = U(x_{i1}) + U(x_{i2})$

$F[p(x_i)] = F[p(x_{i1})] + F[p(x_{i2})]$

Pentru evenimentele x_{i1} si x_{i2} independente: $p(x_i) = p(x_{i1}) \cdot p(x_{i2})$

$F[p(x_{i1}) \cdot p(x_{i2})] = F[p(x_{i1})] + F[p(x_{i2})]$

$F(p) = -\lambda \log_B p$

$i(x_i) = -\lambda \log_B p(x_i)$

- **Unitati de masura a informatiei**:
  - bit, pentru B = 2

$[\mathbf{X}]=[x_1, x_2]$

$\mathbf{P}_x = [p(x_1), p(x_2)] = [1/2, 1/2]$

$i(x_1) = i(x_2) = -\lambda \log_2 (1/2) = 1 bit, \; pentru \lambda = 1$

bitul este cantitatea de informatie ce se obtine prin realizarea (alegerea) unuia din doua evenimente echiprobabile (decizie binara). Multipli: Byte, Kbit, Mbit, Kbyte, Mbyte.

- nit, pentru B = e

nitul este cantitatea de informatie ce se obtine prin realizarea unuia din e evenimente echiprobabile.

$$1nit = \log_2 e = 1{,}44bit$$

- dit, pentru B = 10

ditul este cantitatea de informatie ce se obtine prin realizarea unuia din 10 evenimente echiprobabile.

$$1dit = \log_2 10 = 3{,}32bit$$

- **Informatia mutuala**

$$[\mathbf{X}] = [x_1, x_2 ... x_n] \qquad \bigcup_{i=1}^{n} x_i = E \qquad x_i \cap x_j = \phi \forall i \neq j$$

$$[\mathbf{P}_x] = [p(x_1), p(x_2), ... p(x_n)] \quad \sum_{i=1}^{n} p(x_i) = 1$$

$$[\mathbf{Y}] = [y_1, ..., y_m] \qquad \bigcup_{j=1}^{m} y_j = E \qquad y_j \cap y_k = \phi, \forall j \neq k$$

$$[\mathbf{P}_y] = [p(y_1), ..., p(y_m)] \quad unde \sum_{j=1}^{m} p(y_j) = 1$$

Se defineste, plecand de la probabilitatea conditionata $p(x_i / y_j)$, informatia conditionata $i(x_i / y_j) = -\log p(x_i / y_j)$, care reprezinta incertitudinea cu privire la realizarea x_i prin observarea y_j si se mai numeste incertitudine *a posteriori.* Informatia mutuala este incertitudinea ramasa dupa realizarea x_i si observarea y_j:

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

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

daca $x_i = y_j$:

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

daca x_i este independent de y_j :

$$i(x_i, y_j) = -\log p(x_i) + \log p(x_i / y_j) = 0$$

3. Entropia informationala

Entropia informationala se defineste ca valoarea medie pe un simbol a informatiei (incertitudinii *a priori*) continuta in mesaj.

$$H(X) = \sum_{i=1}^{n} p(x_i) i(x_i) = -\sum_{i=1}^{n} p(x_i) \log p(x_i)$$

Proprietatile entropiei sunt:

- Este o functie continua in raport cu fiecare $p(x_i) = p_i$.
- Este invarianta la orice permutatie a probabilitatilor p_i.
- Este o functie nenegativa marginita superior de valoarea "log *n*", care se obtine cand evenimentele sunt echiprobabile: $p_1=p_2=....p_n=1/n$; atunci $H(p_1,p_2,....p_n) = \log n$.
- Valoarea ei nu se modifica daca la multimea evenimentelor posibile se adauga evenimentul imposibil ($p_{n+1} = 0$).
- Valoarea ei creste prin descompunerea unui eveniment in mai multe evenimente independente.

Exemplul 1: se dau doua evenimente x_1 si x_2 pentru care:

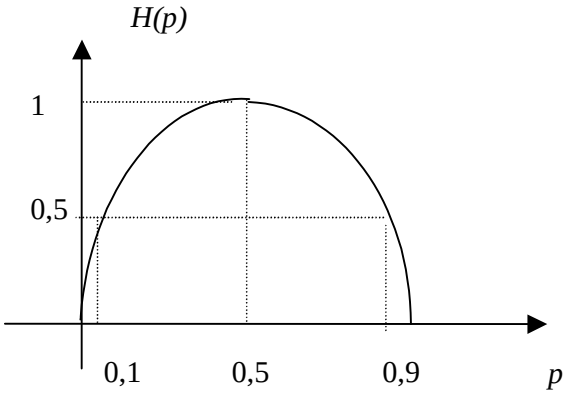
$$[\mathbf{X}] = [x_1, x_2]$$

$$[\mathbf{P}_x] = [1-p, p]$$

$$H(X) = H(p) = -(1-p)\mathbf{log}(1-p) - p\,\mathbf{log}\,p$$

$$H(0) = H(1) = 0$$

$$H_{\mathbf{max}}(p) = H(1/2) = 1$$



Exemplul 2: Se considera un camp de 8 evenimente echiprobabile; Entropia evenimentului aleator este maxima: H_M = - 8.(1/8)log(1/8) = log 8 = 3 bit/eveniment iar numarul minim de decizii binare (bit) necesare pentru detectia unui anumit eveniment este 3.

II Surse discrete de informatie

1. Definitii si terminologie

- Sursa de informatie este un dispozitiv (obiect, proces) care emite mesaje (sunete, imagini) si despre care se dispune de cunostinte partiale.
- Sursa discreta de informatie este sursa care emite mesaje la momente discrete de timp, fiecare mesaj fiind reprezentat printr-un numar finit de simboluri.

Simbol: elementul fundamental ireductibil care contine o informatie;

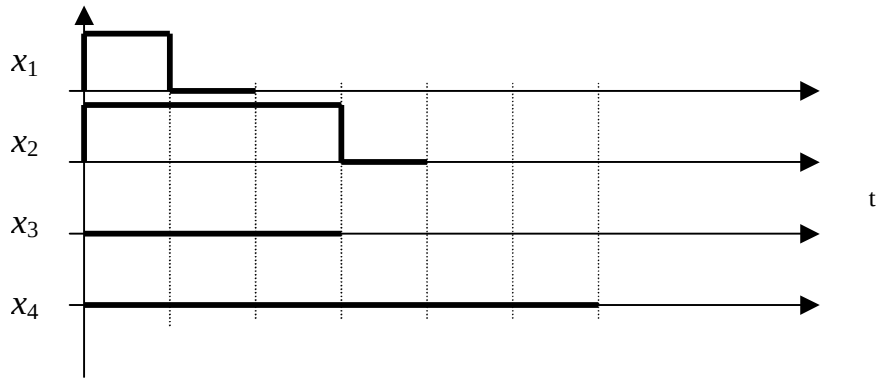
Alfabet: totalitatea simbolurilor;

Cuvant: succesiunea finita de simboluri care reprezinta un mesaj (o semnificatie);

Limba: totalitatea cuvintelor formate cu un anumit alfabet;

Codare (decodare): stabilirea unei anumite corespondente intre o limba si alta limba;

Exemple: a) Codul Morse, care utilizeaza patru simboluri: x_1(punct), x_2 (linie), x_3 (spatiul dintre litere), x_4 (spatiul dintre cuvinte) ca in figura de mai jos:



b) Semnal cuantizat: limba este alcatuita din totalitatea nivelelor posibile.

2. Tipuri de surse discrete

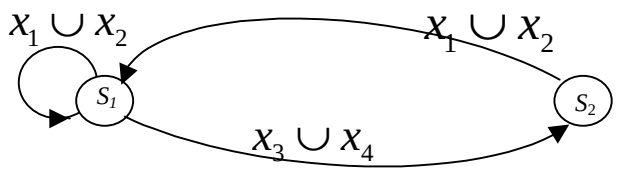
- Sursa discreta fara memorie (SDFM): sursa la care probabilitatea de aparitie a unui simbol nu depinde de simbolurile precedente:

$p\ (x_{i\ n} / x_{j\ n-1}, x_{k\ n-2}, \ldots) = p\ (x_{i\ n})$ unde $x_{i\ n}$ este simbolul x_i la momentul n.

Exemplu: o succesiune de date binare, privite ca independente.

- Sursa extinsa: SDFM care emite blocuri de simboluri (situatia reala); astfel, daca pentru emisia individuala a simbolurilor, modelul este:
  $[\mathbf{X}] = [x_1, x_2, \ldots x_D]$, alfabetul finit al sursei
  $[\mathbf{P}] = [p_1, p_2, \ldots p_D]$, distributia probabilitatilor,
  $[\boldsymbol{\tau}] = [\tau_1, \tau_2, \ldots \tau_D]$, duratele simbolurilor
  pentru emisia a doua simboluri grupate (extensie de ordinul 2), modelul este:
  $[\mathbf{X}^2] = [\sigma_1, \sigma_2, \ldots \sigma_{D\times D}]$ unde $\sigma_1 \rightarrow x_1x_1$, $\sigma_2 \rightarrow x_1x_2, \ldots \sigma_D \rightarrow x_1x_D, \ldots \sigma_k \rightarrow x_ix_j, \ldots \sigma_{D\times D} \rightarrow x_Dx_D$
  $[\mathbf{P}^2] = [p(\sigma_1), p(\sigma_2), \ldots p(\sigma_k), \ldots p(\sigma_{D\times D})]$, unde $p(\sigma_k) = p_i \cdot p_j$ (evenimente independente)
  Exemplu: pentru un alfabet al sursei compus din doua simboluri (D =2),
  $[\mathbf{X}] = [0, 1]$, $[\mathbf{P}] = [p_1, p_2]$,
  iar extensia de ordin 2 (grup de 2 simboluri):
  $[\mathbf{X}^2] = [\sigma_1 \rightarrow 00, \sigma_2 \rightarrow 01, \sigma_3 \rightarrow 10, \sigma_4 \rightarrow 11]$,
  $[\mathbf{P}^2] = [p(\sigma_1) = p_1^2, p(\sigma_2) = p_1p_2, p(\sigma_3) = p_2p_1, p(\sigma_4) = p_2^2]$

- Sursa discreta cu memorie: este o sursa cu constrangeri privind succesiunea simbolurilor
  - Sursa cu constrangeri fixe (deterministe): este o sursa la care anumite succesiuni de simboluri sunt interzise
    Exemplu: sursa generatoare de cod Morse, cu doua stari, S_1 si S_2; in starea S_1 poate fi generat orice simbol, dar in starea S_2 se poate genera numai punct sau linie, fiind interzisa (constransa) generarea intervalelor.



  - Sursa cu constrangeri probabilistice: sursa la care probabilitatea de aparitie a unui simbol la un moment dat depinde de simbolurile anterioare. Numarul de simboluri

anterioare asupra carora se extinde memoria sursei reprezinta ordinul sursei (memoriei).

Pentru sursa de ordin 1: $p\,(x_{i\,n} / x_{j\,n-1}, x_{k\,n-2}, \ldots) = p\,(x_{i\,n} / x_{j\,n-1})$

Pentru sursa de ordin 2: $p\,(x_{i\,n} / x_{j\,n-1}, x_{k\,n-2}, \ldots) = p\,(x_{i\,n} / x_{j\,n-1}, x_{k\,n-2})$

Exemplu: vorbirea.

- Sursa Markov: sursa discreta cu memorie de ordinul 1 (emiterea unui simbol este conditionata numai de starea precedenta), care poate fi modelata de un lant Markov finit si omogen.
  - Lant Markov finit: un proces aleator discret

{ ...,S_{-2}, S_{-1}, S_0, S_1, S_2,...}, unde elementele S_i, numite stari, sunt variabile aleatoare discrete care iau valori in alfabetul starilor $\{s_0, s_1, \ldots, s_{r-1}\}$ unde $r \leq D$ (numarul de simboluri ale alfabetului sursei), iar dependenta satisface conditia Markov:

$p\,(S_1 = s_j / S_0, S_{-1}, \ldots) = p(S_1 = s_j / S_0) = p(S_1 / S_0)$

  - Lant Markov omogen: lantul Markov la care probabilitatea de trecere dintr-o stare in alta nu depinde de timp.
  - Matricea de tranzitia starilor:

In fiecare moment lantul Markov trece intr-o noua stare, sursa funizand un simbol din alfabetul sursei. Probabilitatile de trecere dintr-o stare S_i in alta stare S_j, independente de momentul de timp, sunt:

$p_{ij} = p(S_1 = s_j / S_0 = s_i) = p(s_j / s_i)$, constituind matricea de tranzitie **T**:

$$\mathbf{T} = \begin{vmatrix} p_{11} & p_{12} & \cdot & \cdot & \cdot & p_{1r} \\ p_{21} & p_{22} & \cdot & \cdot & \cdot & p_{2r} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ p_{r1} & p_{r2} & \cdot & \cdot & \cdot & p_{rr} \end{vmatrix}$$

daca notam cu $\mathbf{P}_n$ distributia de probabilitati a starilor la momentul n si cu $\mathbf{P}_{n-1}$ distributia de probabilitati a starilor la momentul n -1:

$$\mathbf{P_n} = \begin{pmatrix} p(s_{1n}) \\ . \\ . \\ p(s_{rn}) \end{pmatrix} \qquad \mathbf{P_{n-1}} = \begin{pmatrix} p(s_{1n-1}) \\ . \\ . \\ p(s_{rn-1}) \end{pmatrix}$$

si tinem cont de relatia: $p(s_{jn}) = \sum_{i=1}^{r} p(s_{in-1}) \cdot p_{ij}$,

are loc relatia: $\mathbf{P_n = T^T P_{n-1}}$ **;** daca se noteaza cu $\mathbf{P_0}$ distributia initiala de probabilitati, se obtine:

$$\mathbf{P_n = (T^T)^n P_0}$$

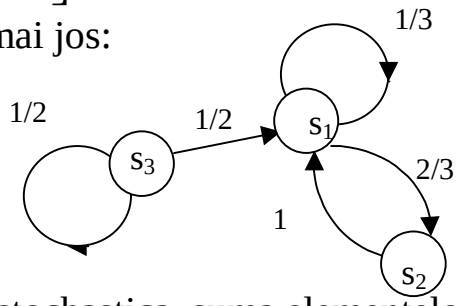
Pentru surse Markov stationare se obtine:

$$\lim_{\mathbf{n}\to\infty} \left(\mathbf{T}^T\right)^{\mathbf{n}} \to \mathbf{I}$$

aceste surse pot fi reprezentate prin grafe; daca ***T*** se ia de forma:

$$\mathbf{T} = \begin{bmatrix} 1/3 & 2/3 & 0 \\ 1 & 0 & 0 \\ 1/2 & 0 & 1/2 \end{bmatrix}$$

rezulta graful de mai jos:



Matricea **T**, fiind stochastica, suma elementelor pe fiecare linie este 1.

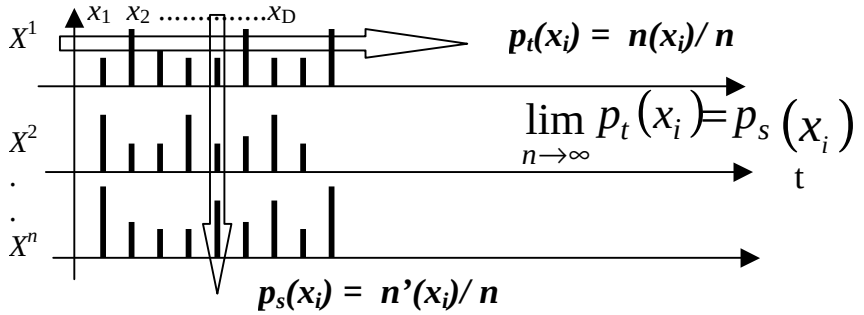
- Sursa stationara: sursa la care probabilitatea diferitelor simboluri nu depinde de originea timpului ci numai de pozitia lor relativa: $p(x_{in}) = p(x_{in+k})$ *pentru* $\forall k$
- Sursa ergodica: sursa la care este indeplinita conditia de stationaritate iar sirurile de simboluri pe care le furnizeaza sunt siruri tipice;

- Sirul tipic: sirul care contine $n_i = n\,p\,(x_i)$ simboluri x_i unde $i = 1, 2, ...D$ si in care pentru frecventele relative de aparitie n_i/n ale simbolurilor, exista relatia:

$$\lim_{n\to\infty} \frac{n_i}{n} = p(x_i).$$

  Multimea sirurilor tipice are o probabilitate care tinde spre 1 pe masura ce n creste.
- Sirul netipic: sirul care are o alta compozitie. Multimea sirurilor netipice are o probabilitate care tinde spre 0 pe masura ce n creste.

Sursa ergodica are proprietatea ca probabilitatile evaluate statistic (pentru *n surse identice* se numara, la un moment dat, sursele care dau simbolul x_i ; frecventa n_i/n tinde spre p_i cand $n \to \infty$) coincid cu probabilitatile evaluate prin mediere temporala, de-a lungul unui sir furnizat de o *singura* sursa. Prin urmare, sursa ergodica va furniza, dupa un timp $t \to \infty$ cu probabilitate 1, un sir tipic. Satisfacerea ipotezei ergodice conduce la importante simplificari in evaluarea experimentelor.



- <u>Sursa cu debit controlabil</u>: sursa care genereaza mesaje ca urmare a unei comenzi externe, neexistand constrangeri cu privire la momentul transmiterii mesajului.
  Exemple: telefon, fax, telegraf .
- <u>Sursa cu debit necontrolabil</u>: sursa care genereaza mesaje continuu, cu un debit fix care este proprietatea interna a sursei.
  Exemple:
  - generatorul de semnale;
  - generatorul de semnale esantionate ;
  - generatorul de tact al calculatorului.

3. Parametrii surselor discrete

- **Entropia surselor discrete**
  - Entropia sursei discrete, fara memorie, ergodica

Daca sursa are alfabetul si distributia de probabilitati:

$[\mathbf{X}] = [x_1, x_2, ...x_D]$

$[\mathbf{P}] = [p_1, p_2, ...p_D]$, *cu* $\sum_{i=1}^{D} p(x_i) = 1$,

generand sirul tipic de lungime N:

$[\mathbf{X}_N] = \lfloor x_{i_1} x_{i_2} \cdots x_{i_N} \rfloor$ *unde* $i_j = \overline{1, D}$,

toate sirurile tipice au aceeasi probabilitate:

$p(X_N) = p_1^{N_1} p_2^{N_2} \cdots p_D^{N_D}$, unde N_1, N_2,....N_D reprezinta numarul de simboluri x_1, x_2, ...x_D din sirul X_N cu $\sum_{i=1}^{D} N_i = N$,

pentru un N foarte mare se poate scrie:

$N_i = N p_i$

respectiv:

$p(X_N) = \left(p_1^{p_1} p_2^{p_2} \cdots p_D^{p_D}\right)^N$

Cantitatea de informatie ce se obtine cand se realizeaza un sir tipic X_N este:

$$I(X_N) = -\log p(X_N) = -N \sum_{i=1}^{D} p_i \log p_i$$

Se numeste entropie a sursei X informatia medie pe simbol:

$$H(X) = -\sum_{i=1}^{D} p_i \log p_i$$

Rezultatul se poate obtine si calculand direct cantitatea medie de informatie pe simbol, plecand de la informatia proprie a unui simbol $i(x_i) = -\log p_i$ si observand ca informatia totala obtinuta prin realizarea de N_i ori a simbolului x_i este $I(x_i) = N_i$ $i(x_i) = -N_i \log p_i$ iar media pe toate simbolurile este:

$$H(X) = \frac{1}{N} \sum_{i=1}^{D} N_i i(x_i) = -\sum_{i=1}^{D} p_i \log p_i$$

  - Entropia sursei extinse

Pentru sursa binara, avand $\{X, P\}$, entropia este:

$$H(X) = -p_1 \log p_1 - p_2 \log p_2$$

Pentru extensie de ordinul 2 (se emit grupuri de 2 simboluri) a sursei anterioare se obtine o sursa cu $\{X^2\ P^2\}$ la care entropia este:

$$H(X^2) = -p_1^2 \log p_1^2 - 2p_1p_2 \log p_1p_2 - p_2^2 \log p_2^2 =$$
$$-2p_1 \log p_1 - 2p_2 \log p_2 = 2H(X)$$

Pentru extensie de ordinul n se obtine o sursa cu $\{X^m\ P^m\}$ la care entropia este:

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

o Entropia sursei discrete, cu memorie, ergodica, de tip Markov

- Sursa Markov ergodica are proprietatea ca secventa distributiilor de probabilitate $\mathbf{P}_n$, pe toata multimea de stari, tinde catre o distributie de probabilitate limita **w** (distributia de echilibru), independenta de distributia initiala sau $\lim_{n\to\infty} \mathbf{P}_n = \mathbf{w}$.
- Entropia unei surse cu memorie este entropia unui simbol oarecare al sursei dupa observarea tuturor simbolurilor anterioare. Cantitatea medie de informatie prin emisia unui simbol:

$$H_\infty(X) = \lim_{n\to\infty} H(X_n / X_1, \cdots, X_{n-1})$$

- Entropia unei surse stationare:

$$H_\infty(X) = \lim_{n\to\infty} H(X_n / X_1, \cdots, X_{n-1}) =$$
$$\lim_{n\to\infty} \frac{1}{n} H(X_1, ..., X_n)$$

- Entropia unei stari S_j este:

$$H(S_j) = -\sum_{k=1}^{q} p_{jk} \log p_{jk}$$ unde q este numarul starilor in care se poate accede intr-un pas plecand din starea s_j

- Entropia pentru sursa Markov ergodica, unifilara

Sursa Markov este unifilara daca toate simbolurile furnizate la parasirea unei stari s_j sunt distincte. Intre secventele de stari si secventele de simboluri exista o corespondenta univoca si daca sursa este stationara avem: $H_\infty(S) = H_\infty(X)$ deci:

$$H_\infty(X) = H_\infty(S) = \lim_{n\to\infty} H(S_n / S_0, \cdots, S_{n-1}) = \lim_{n\to\infty} H(S_n / S_{n-1}) =$$

$$= \lim_{n\to\infty} \sum_{j=1}^{r} p(S_{n-1} = s_j) H(S_n / S_{n-1} = s_j) = \sum_{j=1}^{r} w_j H(S_j) =$$

$$= -\sum_{j=1}^{r}\sum_{k=1}^{q} w_j \cdot p_{jk} \cdot \log p_{jk}$$

Daca nr. starilor corespunde cu nr. simbolurilor si din starea j putem ajunge in oricare stare intr-un pas:

$$H(X) = -\sum_{j=1}^{D}\sum_{k=1}^{D} p_k \cdot p_{jk} \log p_{jk}$$

- **Debitul de informatie**

  Daca durata medie a unui simbol este $\bar{\tau} = \sum_{i-1}^{D} \tau_i \cdot p(x_i)$, debitul de informatie al sursei cu emisie individuala a simbolurilor este $H_t = H(\bar{\tau})^{-1}$ si se exprima in bit/s. Debitul sursei extinse de ordinul m este :

  $$H_t^m = H^m(\overline{\tau_m})^{-1} = mH(m\bar{\tau})^{-1} = H(\bar{\tau})^{-1}$$

  $H_t = H_t^m$ se mai noteaza cu R_b, adica rata de bit, spre deosebire de rata de baud (rata de transmisie a unui grup de biti) care se noteaza cu $R_B = R_b/m$.

- **Cantitatea de decizie a sursei** este: valoarea maxima a entropiei sursei
  $H_{\max}(X) = \log D$, D fiind numarul simbolurilor din alfabetul sursei.

- **Redundanta sursei**
  - Redundanta absoluta este: $R_s = H_{\max}(X) - H(X)$
  - Redundanta relativa este: $\rho_s = 1 - \dfrac{H(X)}{H_{\max}(X)}$

- **Eficienta sursei**

  $$\eta_s = \frac{H(X)}{H_{\max}(X)} = 1 - \rho_s$$

  Aceste marimi arata cat este de indepartata entropia sursei de valoarea ei maxima.

4. Exemple de surse discrete si entropiile lor

- **sursa cu alfabet latin** cu D = 27 simboluri
  - aproximatie de rang 0: SDFM cu simboluri echiprobabile

    $H_0(X) = \log D = 4{,}75$ *bit/simbol*

    nu se poate identifica o limba.
  - aproximatie de rang 1: SDFM cu simboluri cu o distributie de probabilitati data (engleza)

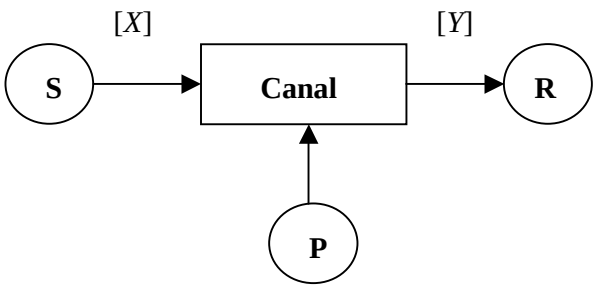
    $H_1(X) = -\sum_{i=1}^{D} p_i \log p_i = 4{,}03\ bit/simbol$
  - aproximatie de rang 2: sursa Markov (engleza)

    $H_2(X) = 3{,}32\ bit/simbol$
- **imagine TV alb / negru singulara** cu N = 500 linii × 600 coloane = 300.000 pixeli, fiecare avand *m* nivele de gri
  - $H_0(X) = \log D = \log m^N = N \log m$

    pentru m = 10, $H_0(X) \approx 10^6\ bit/imagine$

    pentru m = 100, $H_0(X) \approx 2 \cdot 10^6\ bit/imagine$
  - $H_2(X) \approx \frac{1}{2} H_0(X)$

III Canale discrete de transmiterea a informatiei



Campul de intrare: $\mathbf{[X]} = [x_1, x_2, ..., x_n]$

$\mathbf{[P_x]} = [p(x_1), p(x_2), ... p(x_n)]$

Campul de iesire: $\mathbf{[Y]} = [y_1, y_2, ..., y_m]$

$\mathbf{[P_y]} = [p(y_1), p(y_2), ..., p(y_m)]$

Canal discret este acel canal pentru care *n* si *m* sunt finite.

Canal continuu este acel canal pentru care n si m sunt infinite.

Canal discret/continuu: acel canal pentru care n este finit si m este infinit.

Canal continuu/discret: acel canal pentru care n este infinit si m este finit.

Canal discret in timp: la care timpul este discret $t = kT_{es}$.

Canal continuu in timp: la care timpul este continuu.

Canal cu memorie: in care transformarea $x_i \to y_j$ depinde de transformarile anterioare.

Canal fara memorie: in care transformarea $x_i \to y_j$ nu depinde de transformarile anterioare.

Canal stationar: in care $p(x_{i1}t_1) = p(x_{i1}t_2) \quad \forall i = \overline{1,n}$ (distributia de probabilitati nu depinde de origina timpului)

1. Entropia la intrarea si iesirea din canal; Entropia ansamblului intrare – iesire

$\mathbf{[X]} = [x_1, x_2, ..., x_n]$

$\mathbf{[P_x]} = [p(x_1), p(x_2), ... p(x_n)] \quad \sum_{i=1}^{n} p(x_i) = 1$

$[\mathbf{Y}] = [y_1, y_2, ..., y_m]$

$[\mathbf{P}_y] = [p(y_1), p(y_2), ..., p(y_m)]$ $\quad \sum_{j=1}^{m} p(y_j) = 1$

Entropia campului de la intrarea canalului:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i) \text{ (bit/simbol)}$$

Entropia campului de la iesirea canalului:

$$H(Y) = -\sum_{j=1}^{m} p(y_j) \log p(y_j) \quad \text{(bit/simbol)}$$

Se defineste campul reunit (campul produs) intrare – iesire:

$$[\mathbf{X} \cdot \mathbf{Y}] = \begin{pmatrix} x_1y_1 & x_1y_2 & \cdots & x_1y_m \\ x_2y_1 & x_2y_2 & \cdots & x_2y_m \\ \cdots & \cdots & \cdots & \cdots \\ x_ny_1 & x_ny_2 & \cdots & x_ny_m \end{pmatrix}$$

$x_1y_1 \rightarrow x_1 \cap y_1$ (s-a transmis x_1 si s-a receptionat y_1)

$$[P(X,Y] = \begin{bmatrix} p(x_1,y_1) & p(x_1,y_2) & \cdots & p(x_1,y_m) \\ p(x_2,y_1) & p(x_2,y_2) & \cdots & p(x_2,y_m) \\ \cdots & \cdots & \cdots & \cdots \\ p(x_n,y_1) & p(x_n,y_2) & \cdots & p(x_n,y_m) \end{bmatrix}$$

din aceasta matrice pot fi deduse probabilitatile:

$p(x_i) = P\{x_iy_1 \cup x_iy_2 \cup \mathrm{K} \cup x_iy_m\}$ de unde

$p(x_i) = \sum_{j=1}^{m} p(x_iy_j) \quad \forall i = \overline{1,n}$ sau:

$p(y_j) = P\{x_1y_j \cup x_2y_j \cup \mathrm{K} \cup x_ny_j\}$ de unde

$p(y_j) = \sum_{i=1}^{n} p(x_iy_j) \quad \forall j = \overline{1,m}$

$\sum_{i=1}^{n} \sum_{j=1}^{m} p(x_i, y_j) = 1$

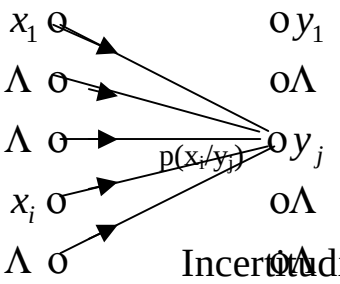
Entropia campului reunit intrare – iesire:

$$H(X,Y) = -\sum_{i=1}^{n} \sum_{j=1}^{m} p(x_i, y_j) \log p(x_iy_j)$$

2. Entropii conditionate

$H(X/Y)$ este incertitudinea medie asupra X dupa observarea lui Y
$H(Y/X)$ este incertitudinea medie asupra Y dupa observarea lui X

- $H(X/Y)$ sau **echivocatie**, este o masura a echivocului ce exista asupra campului de la intrare cand se cunoaste campul de iesire.



daca la iesirea canalului se observa y_j, exista o incertitudine asupra simbolului transmis la intrarea in canal x_i : $U(x_i/x_j) = -\log p(x_i/y_j)$

Incertitudinea medie asupra lui X cand s-a realizat y_j(entropia asociata cu receptionarea simbolului y_j) este:

$$H(X/y_j) = -\sum_{i=1}^{n} p(x_i/y_j)\log p(x_i/y_j) \qquad \forall j = \overline{1,m}$$

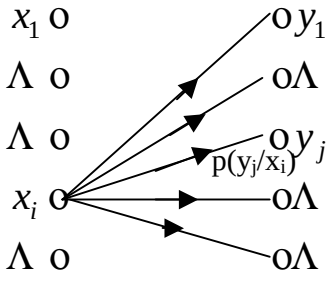
$$H(X/Y) = \sum_{j=1}^{m} p(y_j)\cdot H(X/y_j) = -\sum_{j=1}^{m}\sum_{i=1}^{n} p(y_j)\cdot p(x_i/y_j)\log p(x_i/y_j)$$

$$H(X/Y) = -\sum_{j=1}^{m}\sum_{i=1}^{n} p(x_i,y_j)\log p(x_i/y_j)$$

$$\mathbf{P}[X/Y] = \begin{vmatrix} p(x_1/y_1) & \dots & p(x_n/y_1) \\ \dots & \dots & \dots \\ p(x_1/y_m) & \dots & p(x_n/y_m) \end{vmatrix}$$

$$\sum_{i=1}^{n} p(x_i/y_j) = 1 \; pentru \; \forall j = \overline{1,m}$$

- $H(Y/X)$ sau **eroare medie**, este o masura a incertitudinii campului de iesire cand se cunoaste campul de intrare



$$U(y_j/x_i) = -\log p(y_j/x_i)$$

$$H(Y/x_i) = -\sum_{j=1}^{m} p(y_j/x_i)\log p(y_j/x_i)$$

$$H(Y/X)=\sum_{i=1}^{n} p(x_i)\cdot H(Y/x_i)=-\sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i)\cdot p(y_j/x_i)\log p(y_j/x_i)=$$

$$=-\sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i,y_j)\log p(y_j/x_i)$$

Matricea de zgomot a canalului:

$$\mathbf{P}[Y/X]=\begin{bmatrix} p(y_1/x_1) & \dots & p(y_m/x_1) \\ \dots & \dots & \dots \\ p(y_1/x_n) & \dots & p(y_m/x_n) \end{bmatrix}$$

Pentru canal fara zgomot: $m = n$

$p(y_j/x_i)=1 \; pentru \; \forall j=i$ si $p(y_j/x_i)=0 \; pentru \; \forall j\neq i$

In toate situatiile: $\sum_{j=1}^{m} p(y_j/x_i)=1 \; pentru \; \forall i=\overline{1,n}$

Relatiile entropiilor conditionate cu entropiile proprii

In general, $H(X)\geq H(X/Y)$

In cazuri extreme:

- canal neperturbat: $H(X)>H(X/Y)=H(Y/X)=0$
- canal foarte perturbat: $H(X)=H(X/Y)$ si $H(Y)=H(Y/X)$

Relatii intre entropii:

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

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

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

Ultima relatie rezulta astfel:

deoarece

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

$$H(X,Y)=-\sum_{i=1}^{n}\sum_{j=1}^{m}p(x_i,y_j)\log p(x_i y_j)=$$

$$-\sum_{i=1}^{n}\sum_{j=1}^{m}p(y_j/x_i)\cdot p(x_i)\cdot\log\left[p(x_i)\cdot p(y_j/x_i)\right]=-\sum_{i=1}^{n}p(x_i)\log p(x_i)-$$

$$\sum_{i=1}^{n}p(x_i)\cdot\sum_{j=1}^{m}p(y_j/x_i)\cdot\log p(y_j/x_i)=H(X)+H(Y/X)$$

Deasemenea:

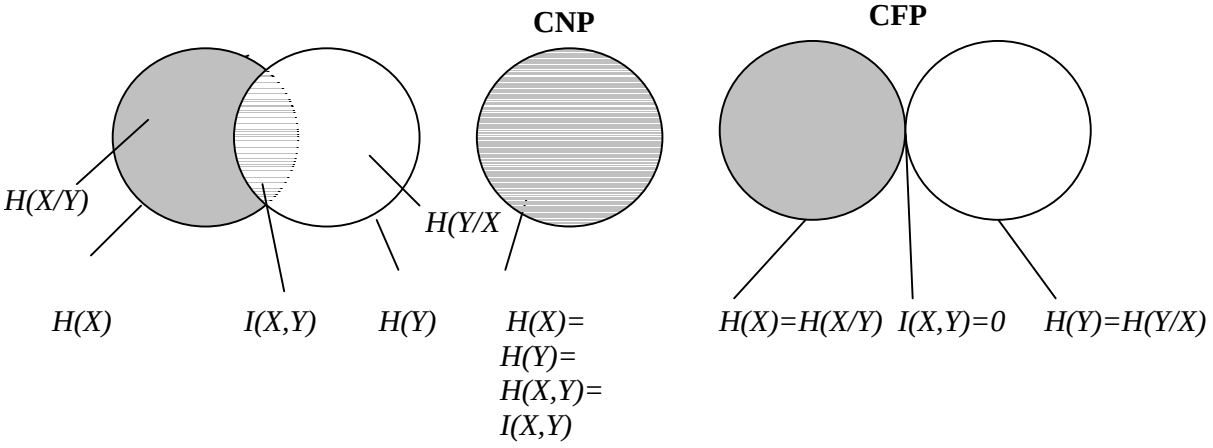
$$H(X,Y)=H(Y)+H(X/Y)$$

pentru canal fara perturbatii echivocatia si eroarea medie sunt nule:

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

pentru canal foarte perturbat:

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



2. Transinformatia; Capacitatea canalului

$I(X,Y)$ este valoarea medie a informatiei mutuale;daca $i(x_i,y_j)$ este informatia mutuala obtinuta asupra x_i cand se receptioneaza y_j, $I(X,Y)$ este informatia asupra alfabetului de la intrare cand se receptioneaza alfabetul de la iesire:

$$I(X,Y)=\sum_{i=1}^{n}\sum_{j=1}^{m}p(x_i,y_j)\cdot i(x_i,y_j)=\sum_{i=1}^{n}\sum_{j=1}^{m}p(x_i y_j)\log\frac{p(x_i,y_j)}{p(x_i)\,p(y_j)}=$$

$$\sum_{i=1}^{n}\sum_{j=1}^{m}p(x_i,y_j)\log p(x_i,y_j)-\sum_{i=1}^{n}\sum_{j=1}^{m}p(x_i,y_j)\log p(x_i)-\sum_{i=1}^{n}\sum_{j=1}^{m}p(x_i,y_j)\log p(y_j)=$$

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

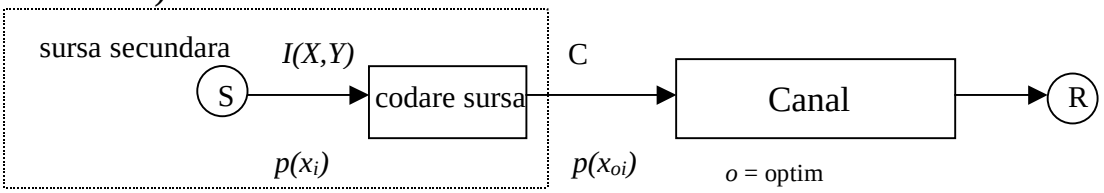
$$I(X,Y)\geq 0$$

Capacitatea canalului

Capacitatea canalului este prin definitie valoarea maxima a transinformatiei:

$$C \overset{def}{=} \max\{I(X,Y)\} = \max[H(X) - H(X/Y)] = \max[H(Y) - H(Y/X)]$$

Maximalizara se poate face in raport cu $P(X)$: pentru transmiterea prin canal a transinformatiei maxime este necesara **adaptarea statistica** a sursei la canal (trnsformarea sursei primare in sursa secundara specificata de probabilitatile care detemina valoarea maxima din relatia de mai sus)



4. Parmetrii canalului discret

- Capacitatea canalului: $C = \max I(X,Y)$ (bit/simbol)
- Capacitate de informare: $C_t = \frac{C}{\bar{\tau}} = \frac{\max I(X,Y)}{\bar{\tau}}$ (bit/sec)

  unde $\bar{\tau} = \sum \tau_i p_{oi}$ unde p_{oi} $\rightarrow$setul de probabilitati optime

- Redundanta absoluta: $R_c \overset{def}{=} C - I(X,Y)$ (bit/simbol)
- Redundanta relativa: $\rho_c = \frac{R_c}{C} = 1 - \frac{I(X,Y)}{C}$ (%)
- Eficienta canalului: $\eta_c = \frac{I(X,Y)}{C}$ (%)

Cand $I(X,Y) \rightarrow C \Rightarrow \rho_c \rightarrow 0,\ \eta_c \rightarrow 100\%$

Caz optim: $\rho_c = 0, \quad \eta_c = 100\%$

5. Modele de canale discrete

- Canal uniform fata de intrare: daca in fiecare linie a matricei sale de zgomot se foloseste aceeasi multime de probabilitati conditionate, ordinea de scriere putand diferi de la o linie la alta.
- Canal uniform fata de iesire: daca in fiecare coloana a matricei sale de zgomot se foloseste aceeasi multime de probabilitati conditionate, ordinea de scriere putand diferi de la o linie la alta.
- Canal simetric: Canalul uniform atat fata de intrare, cat si fata de iesire, caracterizat de o matrice de zgomot patrata. Pentru ordinul n:

$$\mathbf{P}(Y/X) = \begin{bmatrix} 1-p & \frac{p}{n-1} & \dots & \frac{p}{n-1} \\ \frac{p}{n-1} & 1-p & \dots & \frac{p}{n-1} \\ \dots & \dots & \dots & \dots \\ \frac{p}{n-1} & \frac{p}{n-1} & \dots & 1-p \end{bmatrix} \text{ unde } 0 \le p \le 1$$

Canalul fiind uniform fata de intrare, eroarea medie este o constanta:

$$H(Y/X) = -\sum_{k=1}^{n}\sum_{j=1}^{n} p(x_k)\,p(y_j/x_k)\log p(y_j/x_k) =$$

$$= -[(1-p)\log(1-p) + (n-1)\frac{p}{n-1}\log\frac{p}{n-1}]\cdot\sum_{k=1}^{n} p(x_k) =$$

$$= (1-p)\log(1-p) - p\log p + p\log(n-1) = const.$$

Capacitatea canalului este:

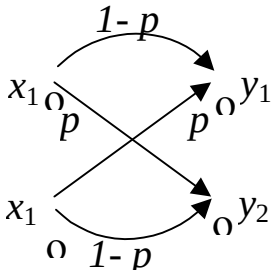
$C = \max[H(Y) - H(Y/X) = \max[H(Y)] - H(Y/X)$

Canalul fiind uniform fata de intrare si de iesire, daca se ia $p(x_1) = p(x_2) = ... = p(x_n) = 1/n$ avem si $p(y_1) = p(y_2) = ... = p(y_n) = 1/n$ in care caz avem valoarea maxima a entropiei: $\max[H(Y)] = \log n$ si capacitatea este:

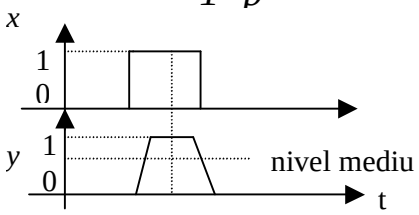
$C_{sim} = \log n + (1-p)\log(1-p) + p\log p - p\log(n-1)$

- Canal binar simetric: cu doua simboluri la intrare si iesire, avand matricea de zgomot si graful (n = 2):

$$P[Y/X]=\begin{bmatrix} p(y_1/x_1) & p(y_2/x_1) \\ p(y_1/x_2) & p(y_2/x_2) \end{bmatrix}$$



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



Capacitatea canal: $C = \max I(X,Y) = \max[H(Y) - H(Y/X)]$

Eroarea medie: $H(Y/X) = -\sum_{i=1}^{2}\sum_{j=1}^{2} p(x_i)\,p(y_j/x_i)\log p(y_j/x_i)$

Efectuand sumele, se observa ca $H(Y/X)$ nu depinde de $p(x_i)$:

$H(Y/X) = -[p(x_1)+p(x_2)]\cdot[p\log p+(1-p)\log(1-p)] =$

$= -[p\log p+(1-p)\log(1-p)]$ deci depinde numai de zgomot.

$C = \max H(Y) + [p\log p+(1-p)\log(1-p)] =$

$= 1+[p\log p+(1-p)\log(1-p)] = 1-H(p)$ deoarece $H(Y) = 1$

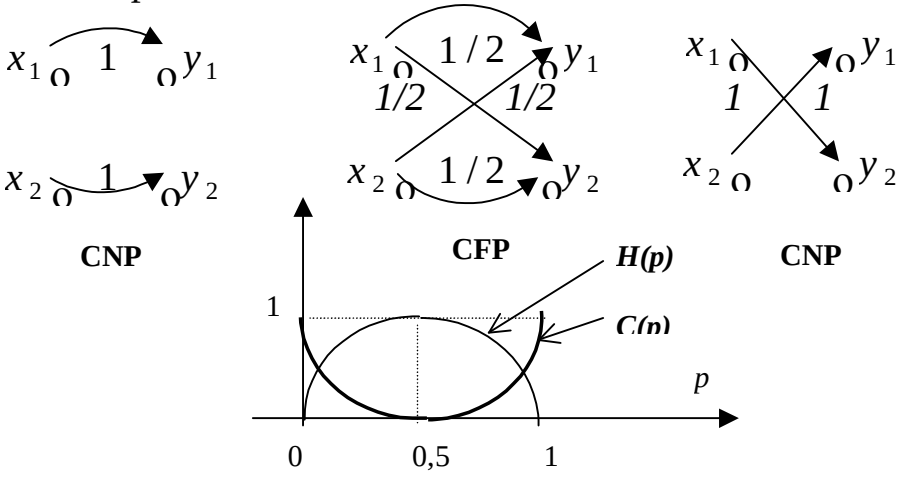
pentru $p\ (y_1) = p(y_2) = 1/2$ si (simetric) $p(x_1) = p(x_2) = 1/2$

Obs.: Expresia lui C rezulta si din C_{sim} pentru $n = 2$

Fara perturbatii: $p = 0 \Rightarrow C = 1$

Puternic perturbat (y_1 poate proveni cu aceeasi probabilitate din x_1 sau din x_2): $p = 1/2 \Rightarrow C = 0$ (nu se transmite informatie).

Inversor: $p = 1 \Rightarrow C = 1$

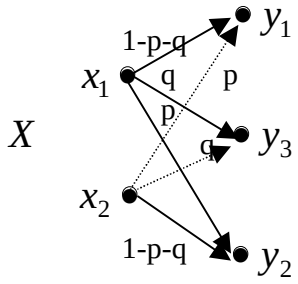


- Canal binar cu erori si anulari

  Canalul este uniform fata de intrare si are dimensiunile alfabetelor n = 2 si m = 3.

  Graful si matricea de zgomot sunt:

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



  Capacitatea canalului:

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

  pentru $p(x_1)=p(x_2)=1/2$;

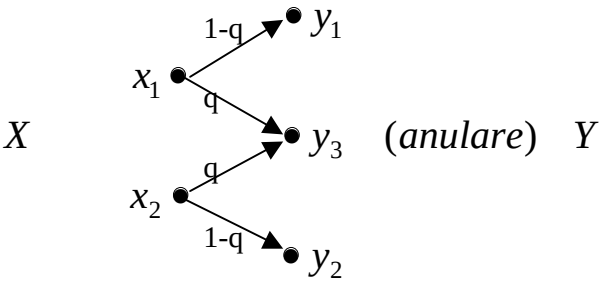
  daca se mareste rata de anulare, $p/q\to 0$, probabilitatea deciziei incorecte se reduce mult si canalul devine canal binar cu anulari.

- Canal binar cu anulari

  Daca q este probabilitatea de anulare a fiecarui simbol de intrare,

  Matricea de tranzitie si graful:

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



  Particularizand expresia capacitatii canalului cu erori si anulari pentru $p/q \to 0$ rezulta capacitatea canalului cu anulari:

$$C=1-q$$

6. Capacitatea canalelor discrete.

- **<u>Canal discret general, fara memorie</u>**; prin definitie:

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

unde:

$$\sum_{i=1}^{n} p(x_i) - 1 = 0, \quad p(x_i) \geq 0, \quad \forall i$$

Trebuie sa se determine maximul functiei:

$$\Phi = I(X,Y) - \lambda[\sum_{i=1}^{n} p(x_i) - 1]$$

Pentru determinarea probabilitatilor optime $p(x_i)$ pentru care Φ este maxim, se rezolva sistemul de ecuatii:

$$\frac{\partial \Phi}{\partial p(x_i)} = 0$$

$$\sum_{i=1}^{n} p(x_i) = 1$$

care, dupa efectuarea derivarii, se poate pune sub forma:

$$\sum_{j=1}^{m}[C + \ln p_0(y_j)]p(y_j / x_i) = -H(Y/x_i) \qquad \forall i = \overline{1,n}$$

$$p_0(y_j) = \sum_{i=1}^{n} p_0(x_i) \cdot p(y_j x_i) \qquad \forall j = \overline{1,m}$$

$$\sum_{j=1}^{m} p(y_j) = 1$$

rezultand un sistem de n + m +1 ecuatii cu n + m +1 necunoscute:

$p(x_1),..., p(x_n), \quad p(y_1),..., p(y_m) \quad si \quad C$

O solutie mai simpla a acestui sistem se obtine pentru n = m, cand:

$$\mathbf{P}(Y/X) = \begin{bmatrix} p(y_1/x_1) & \cdots & p(y_n/x_1) \\ p(y_1/x_n) & \cdots & p(y_n/x_n) \end{bmatrix} = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ p_{n1} & \cdots & p_{nn} \end{bmatrix}$$

$[\mathbf{P}(Y/X)]^{-1} = \mathbf{Q} = \begin{bmatrix} q_{11} & \cdots & q_{n1} \\ q_{1n} & \cdots & q_{nn} \end{bmatrix}$ ale carei elemente sunt date de:

$q_{ji} = \frac{P_{ij}}{|P(Y/X)|}$, unde P_{ij} este cofactorul elementului $p(y_j / x_i)$.

Rezulta:

$$C = \log \sum_{j=1}^{n} 2^{-\sum_{i=1}^{n} q_{ji} \cdot H(Y/x_i)};$$

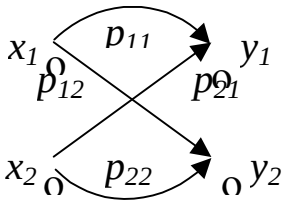
$$p_0(y_j) = 2^{-C} \cdot 2^{-\sum_{i=1}^{n} q_{ji} \cdot H(Y/x_i)};$$

$$p_0(x_i) = 2^{-C} \cdot \sum_{j=1}^{n} q_{ij} \cdot 2^{-\sum_{k=1}^{n} q_{jk} \cdot H(Y/x_k)};$$

$$H(Y/x_i) = -\sum_{j=1}^{n} p(y_j/x_i) \log p(y_j/x_i)$$

- Canal binar general (n = m = 2);

Graful este:



Matricea de zgomot este:

$$\mathbf{P}(Y/X) = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

Capacitatea canalului este:

$$C = \log[2^{-q_{11}H(Y/x_1)-q_{12}H(Y/x_2)} + 2^{-q_{21}H(Y/x_1)-q_{22}H(Y/x_2)}] = \log[2^{Q_1} + 2^{Q_2}]$$

$$p_0(x_1) = 2^{-C} \cdot [q_{11}2^{Q_1} + q_{12}2^{Q_2}]$$

$$p_0(x_2) = 2^{-C} \cdot [q_{21}2^{Q_1} + q_{22}2^{Q_2}]$$

$$Q_1 = -q_{11}H(Y/x_1) - q_{12}H(Y/x_2)$$

$$Q_2 = -q_{21}H(Y/x_1) - q_{22}H(Y/x_2)$$

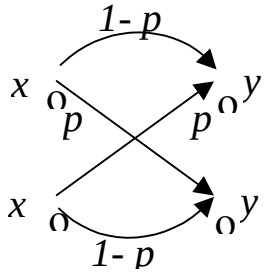
Capacitatea canalului are cea mai mare valoare daca:
$p_{11} = p_{22} = 1$ ceea ce implica $p_{12} = p_{21} = 0$ (canal fara zgomot) sau daca:
$p_{11} = p_{22} = 0$ ceea ce implica $p_{12} = p_{21} = 1$ (canal fara zgomot,inversor)

$$q_{11} = \frac{p_{22}}{\Delta}; q_{12} = \frac{-p_{21}}{\Delta}; q_{21} = \frac{-p_{12}}{\Delta}; q_{22} = \frac{p_{11}}{\Delta};$$

$$\Delta = p_{11}p_{22} - p_{12}p_{21}$$

- Canal binar simetric:

Graful este:



Matricea de zgomot :

$$\mathbf{[P(}Y / X\mathbf{)]} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix};$$

$$\mathbf{[Q]} = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} = \begin{bmatrix} \frac{1-p}{1-2p} & \frac{-p}{1-2p} \\ \frac{-p}{1-2p} & \frac{1-p}{1-2p} \end{bmatrix}$$

unde se vede ca **:**

$q_{11} = q_{22} \quad q_{12} = q_{21}$**;**

$H(Y / x_1) = H(Y / x_2) = -p \log p - (1-p) \log(1-p);$

Capacitatea canalului

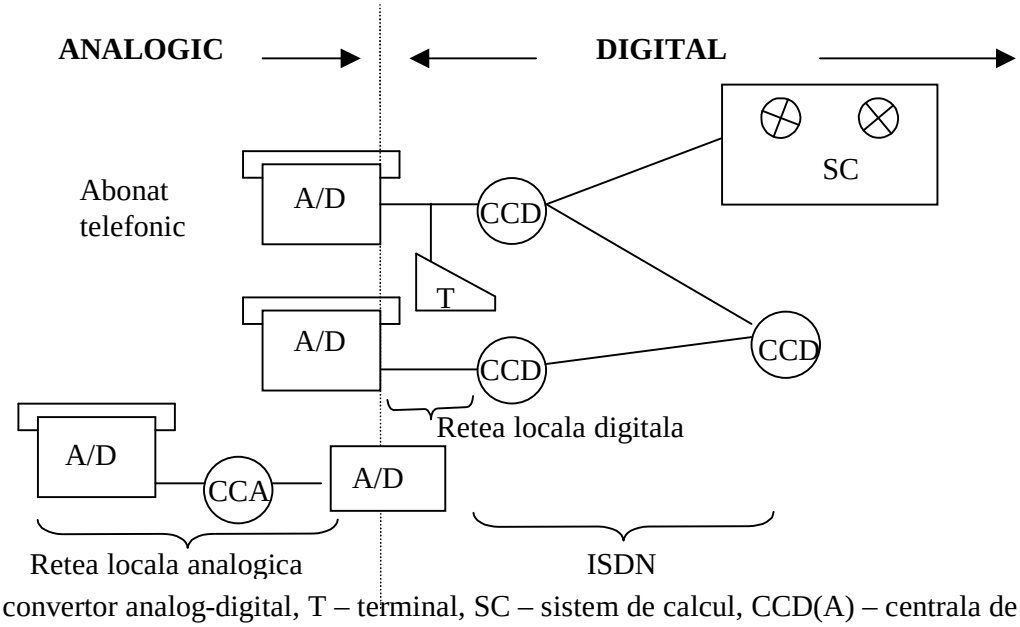
$$C = \log(2^{Q_1} + 2^{Q_2}) = \log[2 \cdot 2^{-H(Y/x_1)}] = 1 - H(Y / x_1) =$$

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

deoarece:

$$Q_1 = -q_{11} H(Y / x_1) - q_{12} H(Y / x_2) =$$
$$= -H(Y / x_1)[q_{11} + q_{12}] = -H(Y / x_1)$$
$$Q_2 = -q_{21} H(Y / x_1) - q_{22} H(Y / x_2) =$$
$$= -H(Y / x_2)[q_{21} + q_{22}] = -H(Y / x_2)$$

7. **Exemple de canale discrete**
   - *Canale ISDN* - Integrated Services Digital Networks – Retele digitale cu servicii integrate (integreaza transmisiile telefonice si transmisiile de date, vezi figura de mai jos).



A/D – convertor analog-digital, T – terminal, SC – sistem de calcul, CCD(A) – centrala de comutatie digitala (analogica)

Interfata ISDN suporta o retea de 144 Kbit/s care muliplexeaza mai multe canale:

A - 4kHz, canal telefonic analogic,
2×B - 64 Kbit/s, canale digitale pentru voce si date cu comutatie in pachete,
C - 8 sau 16 Kbit/s, canal digital pentru date de mica viteza,
D - 16 Kbit/s, canal digital pentru control si semnalizare,
E - 64 Kbit/s, canal digital pentru semnalizari interne,
H - 384 Kbit/s (H0), 1536 Kbit/s (H11), 1920 Kbit/s (H12), canale digitale de mare viteza.

- *Retele* LAN - Local Area Network – Retele locale (de calculatoare) cu:
  a) cu cablu coaxial, 1 – 10 Mbit/s
  b) cu fibra optica, > 100 Mbit/s

8. Canale discrete cu constrangeri

Prin definitie, un canal cu constrangeri nu accepta anumite secvente in sirul simbolurilor de intrare.

Exemple:

- Canal binar cu constrangeri RL (Run Length – Lungime de executie), in care se limiteaza numarul de simboluri identice care se pot succeda intr-o secventa: aceasta restrictie asigura un prag minim al frecventei tranzitiilor intre sirurile de 0 si 1, conditie care permite sincronizarea receptorului cu emitatorul pe baza informatiei extrase din aceste tranzitii.
- Canal binar cu constrangeri RLL (Run Length Limited):
  - Constrangere RLL (x,t) – impune ca intr-o succesiune de simboluri care dureaza un timp t sa nu existe mai multe simboluri succesive x (0 sau 1);
  - Constrangere RLL (r, s) – impune ca dupa un simbol 1 sau 0 sa urmeze cel putin r simboluri opuse dar nu mai mult de s asemenea simboluri.

Caracterizarea canalelor cu constrangeri

- Pentru canalul binar cu constrangeri se poate defini un set al *starilor permise*: $S = \{S_0, S_1, ..., S_{N-1}\}$, la fiecare impuls de tact al canalului va fi transmis unul din simbolurile permise, in acel moment starea schimbandu-se, noua stare depinde de starea veche si de simbolul transmis.
- Succesiunea starilor canalului poate fi reprezentata prin:
  - diagrame de stare: graf orientat, pe fiecare arc de tranzitie inscriindu-se simbolul transmis care face trecerea intre stari.
  - diagrame trellis: in care apar tranzitiile dintr-o stare in alta pe parcursul mai multor impulsuri de tact (diagrama are si axa timpului).

  Exemplu: Canal RLL (1,2) care admite 4 stari: S_0 = 0, S_1 = 1, S_2 = 00, S_3 = 11, fiecare stare exprimand cel mai recent sir de simboluri identice; tranzitia se face conform constrangerii impuse. Diagrama de stare este:

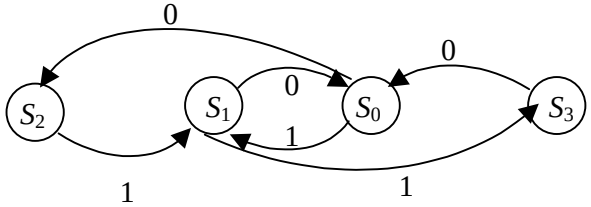
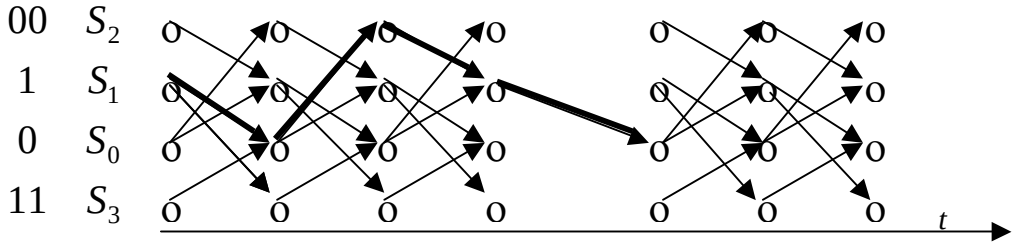


Diagrama trellis este (in care o succesiune posibila de tranzitii este reprezentata ingrosat):



- Matricea tranzitiilor de stare, unde elementul b_{ij} reprezinta numarul arcelor de trecere din starea S_i in starea S_j din diagrama de stare:

$$\mathbf{B} = \begin{bmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ b_{20} & b_{21} & b_{22} & b_{23} \\ b_{30} & b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

  Daca se extinde descrierea pentru un numar de 2 pasi, matricea de tranzitie se noteaza cu B^2 si are elementele $b_{ij}^{(2)}$ date de numarul arcelor distincte de la starea S_i la starea S_j si care au o lungime de 2 pasi. In general, matricea tranzitiilor de stare in m pasi este B^m care este chiar matricea B la puterea m.
- Matricea extinsa a tranzitiilor de stare $B(x)$, ia in consideratie duratele simbolurilor transmise pe fiecare cale dintre stari; elementul $b_{ij}(x)$ al acestei matrici este o functie de variabila fictiva x si este definit prin: $b_{ij}(x) = \sum_h x^{-t_{ij,h}}$ unde $t_{ij,h}$ reprezinta durata simbolului de intrare de pe pozitia h, car poate aparea in starea i, determinand o tranzitie in starea j. B se regaseste pentru $x = 1$ din $B(x)$. In exemplul anterior, simbolurile 0 si 1 au aceeasi durata, adica o perioada de tact si:

$$\mathbf{B}(x) = \begin{bmatrix} b_{00}(x) & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ b_{20} & b_{21} & b_{22} & b_{23} \\ b_{30} & b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} 0 & x^{-1} & x^{-1} & 0 \\ x^{-1} & 0 & 0 & x^{-1} \\ 0 & x^{-1} & 0 & 0 \\ x^{-1} & 0 & 0 & 0 \end{bmatrix}$$

- Capacitatea canalului cu constrangeri si simboluri de durate egale: $C = \log \lambda$ unde λ este cea mai mare valoare proprie (pozitiva) a matricei B. La determinarea lui λ pentru exemplul anterior, se poate scrie ecuatia:

$$\det[B - \lambda I] = 0 \text{ sau:}$$

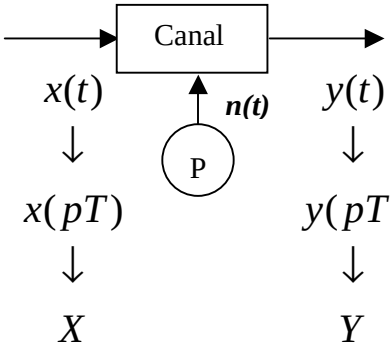
$$\det \begin{bmatrix} -\lambda & 1 & 1 & 0 \\ 1 & -\lambda & 0 & 1 \\ 0 & 1 & -\lambda & 0 \\ 1 & 0 & 0 & -\lambda \end{bmatrix} = 0$$

Ecuatia caracteristica este: $\lambda^4 - \lambda^2 - 2\lambda - 1 = 0$ si cea mai mare solutie este $\lambda_{\max} = 1{,}6$; capacitatea este:

$$C = \log \lambda_{\max} = 0{,}66 \quad \text{(bit/simbol)}$$

IV Masura informatiei in sisteme continue

1. Transinformatia in canale continue



Semnalul *x(t)* se cuantizeaza cu *n* nivele, marimea cuantei este $q_x = \Delta x$;

Semnalul *y(t)* se cuantizeaza cu *m* nivele, marimea cuantei este $q_y = \Delta y$; esantioanele observate la intrare si iesire sunt:

$$x_i = i\Delta x \qquad i = \overline{1,n}$$

$$y_j = j\Delta y \qquad j = \overline{1,m}$$

Transinformatia va fi:

$$I(X,Y) = \sum_{i=-n/2}^{n/2} \sum_{j=-m/2}^{m/2} p_{ij} \log \frac{p_{ij}}{p_i \cdot q_j} \quad unde:$$

$$p_i = P\{X = x_i = i\Delta x\}; \quad q_j = P\{Y = y_j = j\Delta y\}; \quad p_{ij} = P\{X = x_i \cap Y = y_j\}$$

$$pentru \quad \Delta x, \Delta y \to 0$$

$$p(x)dx = P[x < X < x + dx]$$

$$p(y)dy = P[y < Y < y + dy]$$

$$p(x,y)dxdy = P\begin{Bmatrix} x < X < x + dx \\ y < Y < y + dy \end{Bmatrix}$$

Setul de probabilitati p_i, q_j poate fi scris sub forma:

$$p_i \cong p(x_i)\Delta x; \quad q_j \cong q(y_j)\Delta y; \quad p_{ij} \cong p(x_i, y_j)\Delta x \Delta y$$

iar transinformatia devine:

$$I(X,Y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)} dxdy = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} p(x,y) \log p(x,y) dxdy$$

$$- \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} p(x,y) \log p(x) dxdy - \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} p(x,y) \log p(y) dxdy$$

Daca se tine cont ca a doua si a treia integrala dubla se pot pune sub forma:

$$\int_{-\infty}^{+\infty} \log p(x)dx \int_{-\infty}^{+\infty} p(x,y)dy = \int_{-\infty}^{+\infty} p(x)\log p(x)dx = H(X)$$

$$\int_{-\infty}^{+\infty} \log p(y)dy \int_{-\infty}^{+\infty} p(x,y)dx = \int_{-\infty}^{+\infty} p(y)\log p(y)dy = H(Y)$$

transinformatia devine:

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

$$= -H(X/Y) + H(X) = -H(Y/X) + H(Y)$$

unde s-au definit in mod analog:

$$H(X/Y) = -\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} p(x,y)\log p(x/y)dxdy$$

$$H(Y/X) = -\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} p(x,y)\log p(y/x)dxdy$$

Obs.: Reamintim ca transinformatia *I(X,Y)* se refera **la un singur esantion**, fiind informatia care se obtine despre x_i cand se observa y_j. Daca semnalul este format din *N* esantioane independente cuprinse intr-un interval de observare de durata D, transinformatia este:

$I_N(X,Y) = N \cdot I(X,Y)$ unde $N = \frac{D}{\Delta t} = \frac{D}{T_{es}} = \frac{D}{1/2f_{\max}} = 2f_{\max}D$ iar

$I_N(X,Y) = 2f_{\max}D \cdot I(X,Y)$, reprezentand valoarea totala a transinformatiei pe *N* esantioane.

2. <u>Capacitatea canalului continuu</u>

<u>Definitie:</u> Valoarea maxima a transinformatiei pe unitatea de timp:

$$C = \lim_{D\to\infty} \max \frac{1}{D} I_N(X,Y) = \max[2f_{\max} \cdot I(X,Y)] =$$

$$= \max_{P(X)}[2f_{\max} \cdot [H(Y) - H(Y/X)]$$

Presupunand canalul simetric, in cazul perturbatiilor aditive, pentru semnale: $y(t) = x(t) + n(t)$ unde *n(t)* este zgomotul in canal si pentru puteri: $P_y = P_x + P_n$. In cazul limitarii puterii P_x a semnalului, eroarea medie nu depinde decat de puterea P_n a zgomotului. Entropia conditionata *H(Y/X)* nu depinde de *P(X)* deci:

$\max I(X,Y) = \max H(Y) - H(Y/X)$.

Numarul de nivele de cuantizare ale semnalului de iesire este:

$m = \frac{\sqrt{P_y}}{\Delta y}$. Daca se considera aceste *m* nivele egal probabile,

$$\max H(Y) = \log m = \log \frac{\sqrt{P_y}}{\Delta y}$$

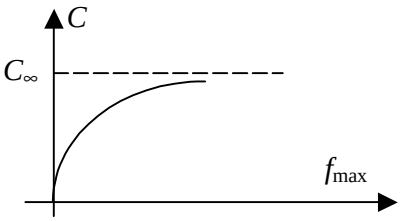
Pentru o valoare fixa a semnalului de intrare, incertitudinea asupra iesirii *H(Y/X)* este data numai de zgomot; daca se considera nivelele zgomotului cuantizat, numarul acestora este $K = \frac{\sqrt{P_n}}{\Delta y}$ iar daca acestea sunt egal probabile: $H(Y/X) = \log K$ deci:

$$\max I(X/Y) = \log \frac{\sqrt{P_y}}{\Delta y} - \log \frac{\sqrt{P_n}}{\Delta y} = \log \frac{\sqrt{P_y}}{\sqrt{P_n}}$$

$$C = 2 f_{\max} \log \frac{\sqrt{P_x + P_n}}{\sqrt{P_n}} = f_{\max} \log(1 + \frac{P_x}{P_n}) \cong f_{\max} \log \frac{P_x}{P_n} \quad pentru P_x >> P_n$$

Pentru canalul avand zgomot alb cu densitatea spectrala $N_0 = const, \quad P_n = f_{\max} N_0$ si

$$C = f_{\max} \log(1 + \frac{P_x}{f_{\max} N_0}) \cong f_{\max} \log \frac{P_x}{f_{\max} N_0}$$



$$C_\infty = \lim_{f_{\max} \to \infty} C = \frac{P_x}{N_0} \log e$$

Obs.: cresterea largimii de banda peste o anumita valoare nu este rationala deoarece nu conduce practic la o crestere semnificativa a capacitatii.

Exemple:

- Pentru audio (Hi Fi):
  $C = 20 \cdot 10^3 \cdot \log(10^3) \approx 200 \cdot 10^3 \quad (bit/s)$
- Pentru video (alb/negru):
  $C = 6 \cdot 10^6 \cdot \log(10^3) \approx 60 \cdot 10^6 \quad (bit/s)$

3. Variatia entropiei cu schimbarea coordonatelor

Daca se da un vector $\vec{x}$ intr-un spatiu $\vec{X}$ si o transformare biunivoca ψ a spatiului $\vec{X}$ in spatiul $\vec{U}$, are loc o transformare a (semnalului) *x* in (semnalul) *u*; se ridica problema cunoasterii entropiei $H(\vec{U})$ daca se cunoaste entropia $H(\vec{X})$. Se poate admite pentru transformari biunivoce relatia intre densitatile de probabilitate:

$p(\vec{x})dX = q(\vec{u})dU$ (sunt egale probabilitatile cu care varfurile vectorilor $\vec{x}, \vec{u}$ se pot gasi in domeniile *dX* respectiv *dU*);

$p(\vec{x}) = q(\vec{u})\left|\frac{dU}{dX}\right| = q(\vec{u})\left|J\left(\frac{U}{X}\right)\right|$ unde *J* este jacobianul transformarii ψ din spatiul $\vec{X}$ in spatiul $\vec{U}$. Generalizand expresia lui *H(X)* pentru un spatiu cu n dimensiuni:

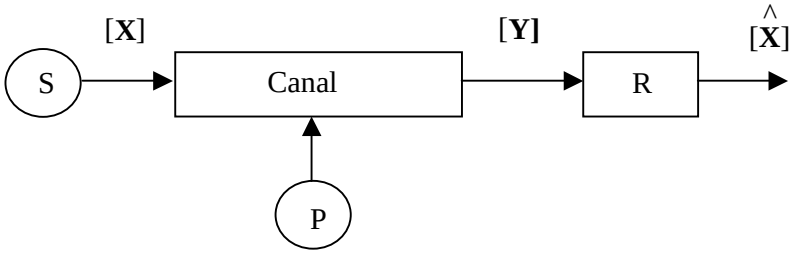
$$H(\vec{X}) = -\int_X p(\vec{x})\log p(\vec{x})dX = -\int_X p(\vec{x})\log\left[q(\vec{u})\left|J\left(\frac{U}{X}\right)\right|\right]dX =$$

$$= -\int_X p(\vec{x})\log q(\vec{u})dX - \int_X p(\vec{x})\log\left|J\left(\frac{U}{X}\right)\right|dX = H(U) - \int_X p(\vec{x})\log\left|J\left(\frac{U}{X}\right)\right|dX$$

deoarece: $\int_X p(\vec{x})\log q(\vec{u})dX = \int_U q(\vec{u})\log q(\vec{u})dU = -H(\vec{U})$

Obs.: in cazul continuu entropia depinde de sistemul de coordonate ales pentru reprezentarea semnalului. Daca transformarea este ortogonala $J\left|\left(\frac{U}{X}\right)\right| = 1$ si numai in acest caz $H\left(\vec{U}\right) = H\left(\vec{X}\right)$; aceasta proprietate a transformarilor ortogonale isi gaseste aplicatii in prelucrarea semnalelor, de exemplu la compresia de date.

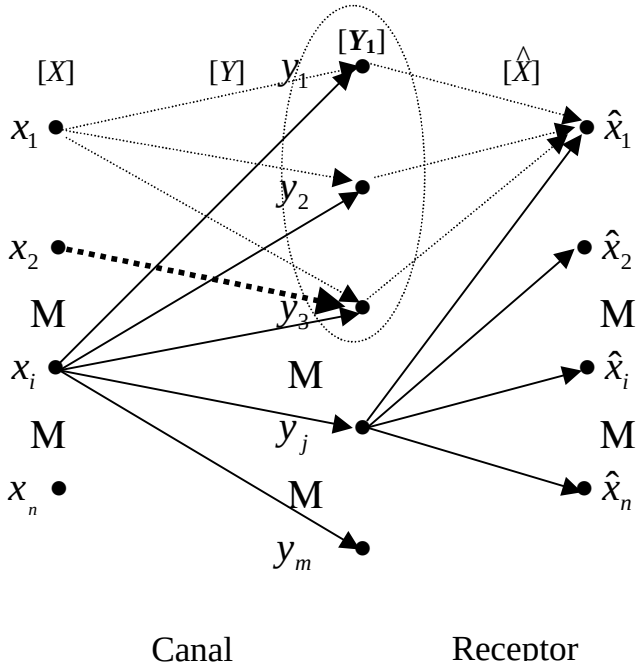
V Receptoare de simboluri discrete



1. Matricea strategiei de decizie a receptorului

Receptorul functioneaza ca estimator al variabilei X atasate sursei prin X estimat. El trebuie sa realizeze o regrupare a simbolurilor receptionate prin impartirea [Y] in submultimi disjuncte [Y_i] asa incat $y_j \in Y_i \rightarrow x = \hat{x}_i$

Graful de tranzitii al ansamblului canal – receptor (linie plina) este:



Problema de baza a receptorului este modul cum trebuie sa realizeze gruparile [Y_j] in cadrul multimii receptionate [Y], astfel incat probabilitatea de eroare sa fie minima (ex. de eroare: un element al [Y_1] poate proveni nu numai de la x_1(linie punctata) ci si, de exemplu, de la x_2 prin tranzitie parazita, reprezentata ingrosat). Pentru aceasta se introduce matricea strategiei de decizie S, care caracterizeaza receptorul, in timp ce canalul este caracterizat de matricea de zgomot T:

$$\mathbf{S} = \begin{bmatrix} p(\hat{x}_1 / y_1) & \dots & p(\hat{x}_n / y_1) \\ \dots & \dots & \dots \\ p(\hat{x}_1 / y_m) & \dots & p(\hat{x}_n / y_m) \end{bmatrix} \qquad \mathbf{T} = \begin{bmatrix} p(y_1 / x_1) & \dots & p(y_m / x_1) \\ \dots & \dots & \dots \\ p(y_1 / x_n) & \dots & p(y_m / x_n) \end{bmatrix}$$

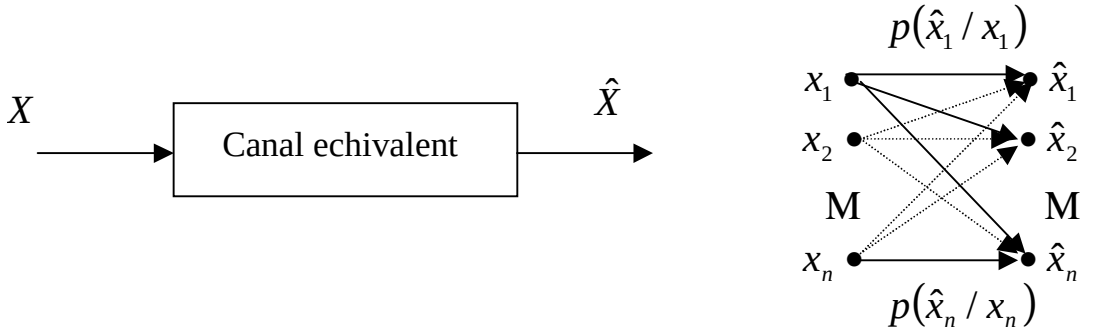
Obs.: Pe fiecare linie a matricei **S** (matrice stochastica) suma elementelor este 1. Daca un element $p(\hat{x}_i / y_j) = 1$ (celelalte fiind 0), semnificatia este ca receptia lui y_j conduce la decizia ca s-a transmis x_i.

2. Matricea de tranzitie a canalului echivalent

$$\mathbf{T_e} = \mathbf{T}\cdot\mathbf{S} = \begin{bmatrix} p(\hat{x}_1/x_1) & \dots & p(\hat{x}_n/x_1) \\ \dots & \dots & \dots \\ p(\hat{x}_1/x_n) & \dots & p(\hat{x}_n/x_n) \end{bmatrix}$$

elementul generic este:

$$p(\hat{x}_i/x_j) = \sum_{k=1}^{m} p(y_k/x_j)\,p(\hat{x}_i/y_k)$$



Strategiile pot fi: $\begin{cases} \text{det}\,er\,\text{min}\,iste & p(\hat{x}_j/x_i) = 0/1 \\ stohastice & p(\hat{x}_j/x_i) \in \overline{0,1} \end{cases}$

Ca strategii deterministe:

3. Criteriul riscului minim

In matricea $\mathbf{T}_e$ $\begin{cases} p(\hat{x}_j/x_i),\ i \neq j, & este\ probabilitatea\ deciziei\ eronate \\ p(\hat{x}_j/x_i),\ i = j, & este\ probabilitatea\ deciziei\ corecte \end{cases}$

Fiecarei decizii ii corespunde un cost $c_{ij} \geq 0$, care corespunde penalitatii deciziei $\hat{x}_j$ cand in realitate s-a transmis x_i

$$c_{ij} = 1 - \delta_{ij}, \quad unde\ \delta_{ij} = \begin{cases} 0, & i \neq j, \quad decizie\ eronata \\ 1, & i = j, \quad decizie\ corecta \end{cases}$$

Se defineste *riscul conditionat*

$$R(S, x_i) = \sum_{j=1}^{n} c_{ij}\,p(\hat{x}_j/x_i) \quad \forall i = \overline{1,n}$$

si *riscul* ca valoarea medie a costurilor:

$$R(S) = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\,p(\hat{x}_j, x_i) = \sum_i p(x_i)\cdot\sum_j p\left(\hat{x}_j/x_i\right) c_{ij} = \sum_i p(x_i)\cdot R(S, x_i)$$

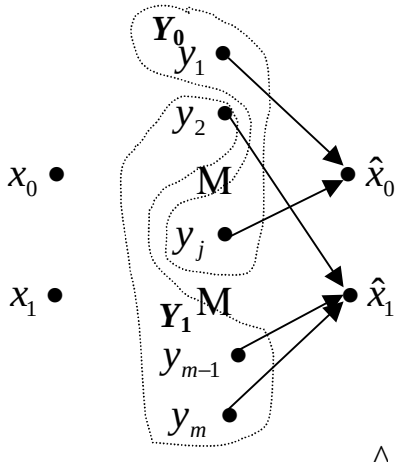
sau ca *probabilitatea de eroare*:

$$R(S) = \sum_{i=1}^{n=1}\sum_{j=1}^{n=1}(1-\delta_{ij})\,p(x_i)\,p\left(\hat{x}_j/x_i\right) = \sum_{i=1}^{n=1}\sum_{j=1}^{n=1}(1-\delta_{ij})\,p(x_i, \hat{x}_j) = 1 - \sum_{i=1}^{n=1} p(x_i, \hat{x}_i)$$

deoarece ultimul termen reprezinta probabilitatea deciziilor corecte.

Criteriul lui Bayes sau al riscului minim consta in alegerea strategiei de decizie care minimizeaza riscul.

- <u>Criteriul lui Bayes in cazul binar</u>: receptorul face o detectie binara, alegand ipoteza adevarata din doua posibile H_0 si H_1, pe baza unui criteriu de testare a ipotezelor. Graful de tranzitie si R devin:



$[X] \rightarrow [T] \quad [Y] \rightarrow [S] \quad [\hat{X}]$

$$R = \sum_{i=0}^{1}\sum_{j=0}^{1} p(x_i)\,p\left(\hat{x}_j / x_i\right)\cdot c_{ij} =$$

$$= p(x_0)\left[p\left(\hat{x}_0 / x_0\right)c_{00} + p\left(\hat{x}_1 / x_0\right)c_{01}\right] +$$

$$+ p(x_1)\left[p\left(\hat{x}_0 / x_1\right)c_{10} + p\left(\hat{x}_1 / x_1\right)c_{11}\right] =$$

$$= p(x_1)\cdot p\left(\hat{x}_0 / x_0\right)(c_{00} - c_{01}) + p(x_0)\cdot c_{01} +$$

$$+ p(x_1)\cdot p\left(\hat{x}_0 / x_1\right)(c_{10} - c_{11}) + p(x_1)\cdot c_{11}$$

$$R = p(x_0)c_{01} + p(x_1)c_{11} + [p(x_1)c_m \cdot p(\hat{x}_0 / x_1) - p(x_0)c_f \cdot p(\hat{x}_0 / x_0)] \quad \text{unde:}$$

$c_m = c_{10} - c_{11}; \quad c_f = c_{01} - c_{00};$ $\quad c_m =$ costul pierderii tintei

$c_f =$ costul alarmei false

Tinand cont ca:

$$p(\hat{x}_0 / x_1) = \sum_{y_k \in Y_0} p(y_k / x_1)$$

$$p(\hat{x}_0 / x_0) = \sum_{y_k \in Y_0} p(y_k / x_0)$$

rezulta:

$$R = const + \sum_{y_k \in Y_0} [p(x_1)c_m \cdot p\left(y_k / x_1\right) - p(x_0)c_f \cdot p\left(y_k / x_0\right)]$$

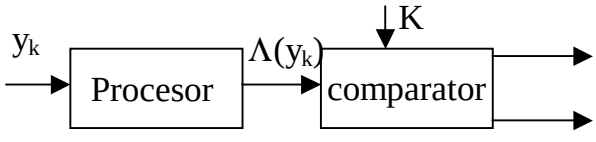
pentru ca riscul sa fie minim trebuie ca suma sa fie minima; aceasta se intampla daca in Y_0 sunt grupate y_k pentru care:

$c_f\, p(x_0)p\left(y_k / x_0\right) \geq c_m\, p(x_1)p\left(y_k / x_1\right)$ iar inY_1 sunt grupate y_k pentru care:

$c_f\, p(x_0)p\left(y_k / x_0\right) \leq c_m\, p(x_1)p\left(y_k / x_1\right)$; condensat, cele doua relatii devin:

$$\Lambda(y_k) = \frac{p(y_k / x_1)}{p(y_k / x_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{p(x_0)}{p(x_1)} \cdot \frac{c_f}{c_m} = K$$

unde H_1 este ipoteza ca y_k se atribuie lui Y_1 iar H_0 este ipoteza ca y_k se atribuie lui Y_0 ;



$y_k \rightarrow \hat{x}_0$

$y_k \rightarrow \hat{x}_1$

$\Lambda(y_k)$, K sunt: *raportul de plauzibilitate* respectiv *pragul testului*, la testarea ipotezelor H_0, H_1.

Pentru cazul particular $c_f = c_m = 1$ se obtin alte criterii de decizie:

- Criteriul probabilitatii *a posteriori* maxime,

  pentru $K = \frac{p(x_0)}{p(x_1)}$:

  $$\Lambda(y_k) \underset{H_0}{\overset{H_1}{\gtrless}} \frac{p(x_0)}{p(x_1)}$$ sau, functie de probabilitatile *a posteriori*:

$$\Lambda(y_k) = \frac{p(y_k / x_1)}{p(y_k / x_0)} = \frac{p(x_1, y_k)/\, p(x_1)}{p(x_0, y_k)/\, p(x_0)} = \frac{p(x_1 / y_k) \cdot p(y_k)/\, p(x_1)}{p(x_0, y_k) \cdot p(y_k)/\, p(x_0)} =$$

$$= \frac{p(x_1 / y_k) \cdot p(x_0)}{p(x_0 / y_k) \cdot p(x_1)} \quad ,$$

$$\frac{p(x_1 / y_k)}{p(x_0 / y_k)} \underset{H_0}{\overset{H_1}{\gtrless}} 1$$

- Criteriul plauzabilitatii maxime, $K = 1$ sau $p(x_0) = p(x_1)$:

  $$\Lambda(y_k) \underset{H_0}{\overset{H_1}{\gtrless}} 1$$

  Obs.: criteriul plauzabilitatii maxime asigura alegerea ipotezei H_i deci a x_i emis, care maximizeaza informatia mutuala raportata la y_j receptionat

  $$I(x_i, y_j) = \log \frac{p(x_i, y_j)}{p(x_i) \cdot p(y_j)} > \log \frac{p(x_k, y_j)}{p(x_k) \cdot p(y_j)}$$ sau

  $$\log \frac{p(x_i, y_j)}{p(x_k, y_j)} \cdot \frac{p(x_k)}{p(x_i)} \underset{H_0}{\overset{H_1}{\gtrless}} 0$$ sau :

  $$\frac{p(y_j / x_i)}{p(y_j / x_k)} \underset{H_0}{\overset{H_1}{\gtrless}} 1$$

  Se poate arata ca eroarea medie dupa decizie este mai mica decat inainte:

  $$H(\hat{X} / X) < H(Y / X)$$

Exista si alte criterii de decizie in afara celor bazate pe teoria lui Bayes (criteriul Neyman – Pearson).

4. Criteriul minimax

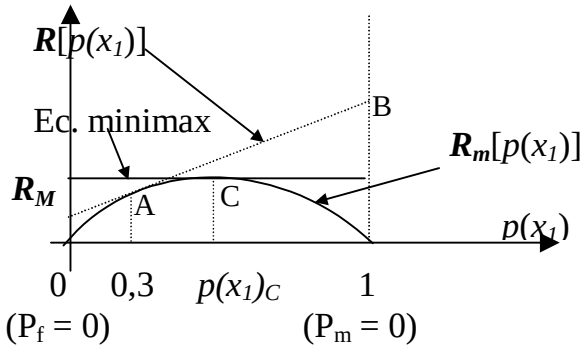
In criteriul lui Bayes, raportul de plauzabilitate Λ este comparat cu un prag K a carui valoare depinde de probabilitatile *a priori* $p(x_i)$; adesea $p(x_i)$ variaza. In acest caz se utiluzeaza criteriul minimax care isi conserva performantele pentru $p(x_i)$ variabil.

Se considera $p(x_1)$ ca fiind variabila, stiind ca $p(x_0) = 1 - p(x_1)$. Se ia $c_{00} = c_{11} = 0$ *si* $c_{01} = c_{10} = 1$ (obs. ideal). Riscul devine:

$$R\big(p(x_1), Y_0\big) = 1 - p(x_1) + \sum_{Y_0} [\, p(x_1)\, p(y_k / x_1) - \big(1 - p(x_1)\big) p(y_k / x_0 =$$

$$= R\big(p(x_1)\big) = P_f + (P_m - P_f), \quad cu \quad notatiile:$$

$$P_m = \sum_{Y_0} p(y_k / x_1), \quad P_f = \sum_{Y_1} p(y_k / x_0) = 1 - \sum_{Y_0} p(y_k / x_0)$$



In fig. s-a reprezentat $R[p(x_1), Y_0]$ unde pentru fiecare valoare particulara a lui $p(x_1) = p_1^*$ s-a fixat un Y_0 pentru care se asigura minimizarea riscului:

$$p(x_1) = p_1^* \Rightarrow R_m(p_1^*) = \min_{Y_0} R(p_1^*, Y_0)$$

Pentru a evita $R(p_1) > R_m(p_1)$ se alege Y_0 (si Y_1) asa ca $R(p_1)$ sa fie tangenta la $R_m(p_1)$ in punctul de risc bayesian maxim C, in care, punand conditia de max. pentru $R(p_1)$, avem :

$P_m - P_f = 0$ adica ecuatia minimax. Criteriul minimax asigura ca

$$R(p_1, Y_0) \le R_M \quad \text{unde:}$$

$$R_M = \max_{p_1} R_m(p_1) = \max_{p_1} \min_{Y_0} R(p_1, Y_0)$$

Strategii aleatoare:

- criteriul minimax (in alta interpretare): alegerea de catre receptor a celei mai defavorizante partitii (sau a simbolului) care maximizeaza riscul conditionat $R(S, x_i)$ si adoptarea unei strategii S care minimizeaza acest maximum:

$$x_i = x_i^*, \; S = S^* \quad \Rightarrow \quad R(S^*, x_i^*) = \min_S [\max_{x_i} R(S, x_i)]$$

Caracterul aleatoriu rezulta din faptul ca la acest criteriu y_k (receptionate) nu sunt atribuite univoc submultimilor Y_0 sau Y_1 ci sunt incluse cu o anumita probabilitate in una din ele respectiv (cu complementul acestei probabilitati) in cealalata.

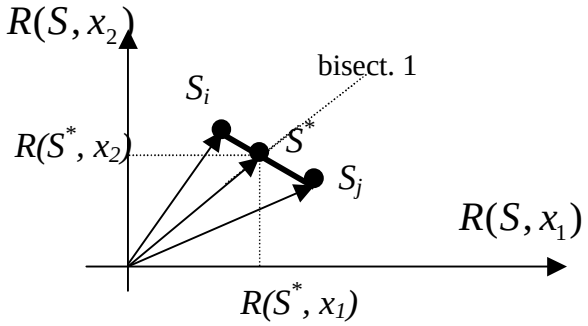
Strategia minimax se construieste astfel:

$$S^* = p_i S_i + p_j S_j$$

unde p_i este probabilitatea de utilizare a strategiei S_i si p_j este probabilitatea de utilizare a strategiei S_j iar $p_i + p_j = 1$;

Intr-o reprezentare vectoriala a strategiilor de decizie in spatiul riscului conditionat, care este un plan in cazul binar, cu axele de coordonate $R(S, x_1)$ si $R(S, x_2)$, S_i si S_j sunt varfurile unor vectori ale caror proiectii pe axe sunt riscurile $R(S_i, x_1)$ *si* $R(S_i, x_2)$ respectiv $R(S_j, x_1)$ si $R(S_j, x_2)$. Strategia minimax S^* corespunde punctului de intersectie a dreptei S_i S_j (care uneste varfurile vectorilor) cu prima bisectoare a planului, vectorul corespunzator respectand relatia:

$$R(S^*, x_1) = R(S^*, x_2)$$

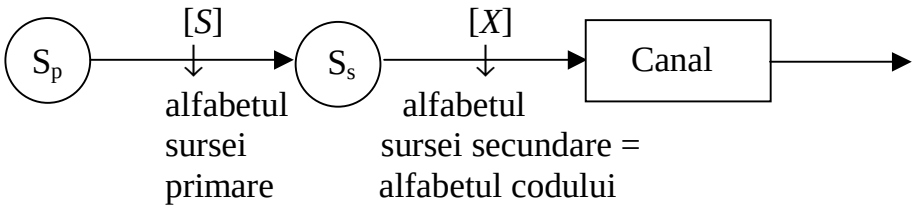


Exemplu: Dac avem doua strategii S_i si S_j si $p_i = 0,3$, $p_j = 0,7$ atunci $S^* = 0,7S_j + 0,3S_i$

$$\mathbf{S}_i = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}; \qquad \mathbf{S}_j = \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \rightarrow \qquad \mathbf{S}^* = \begin{bmatrix} 0 & 1 \\ 0,3 & 0,7 \\ 0,7 & 0,3 \\ 0,7 & 0,3 \\ 0,7 & 0,3 \end{bmatrix}$$

VI Codarea de sursa (pentru canale fara perturbatii)

1. <u>Obiectivul codarii</u>: reducerea redundantei sursei (sursa devine de entropie maxima). Dezavantaj: scade protectia la perturbatii.



$[S]=[s_1,...\, s_N]$ $\quad s_i \rightarrow$ *simbol al alfabetului sursei*

$[P]=[p(s_1),...,\, p(s_n)]$

$[X]=[x_1,...\, x_D]$ $\quad$ *este alfabetul(finit) al codului*

$[C]=[c_1,...,c_N]$ $\quad$ *este codul, alcatuit din cuvinte de cod:*

$c_i=[x_{i1}x_{i2}...x_{ik}] \quad x_{ik}\in X \quad \forall ik$

<u>Codarea</u> este stabilirea corespondentei: $\boxed{s_k \in S \quad \rightarrow \quad c_k \in C}$

2. <u>Tipuri de coduri de sursa</u>:

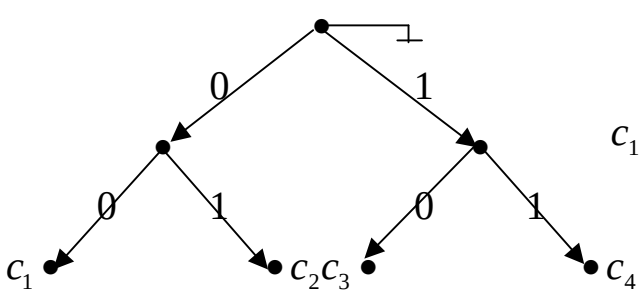
- <u>Coduri unic decodabile</u>: la care unei succesiuni de simboluri ale alfabetului codului (constituite in cuvinte $c_k \in C$), **î**i corespunde o singura succesiune de simboluri (cuvinte, semnificatii) ale sursei.

Exemple de coduri unic decodabile:

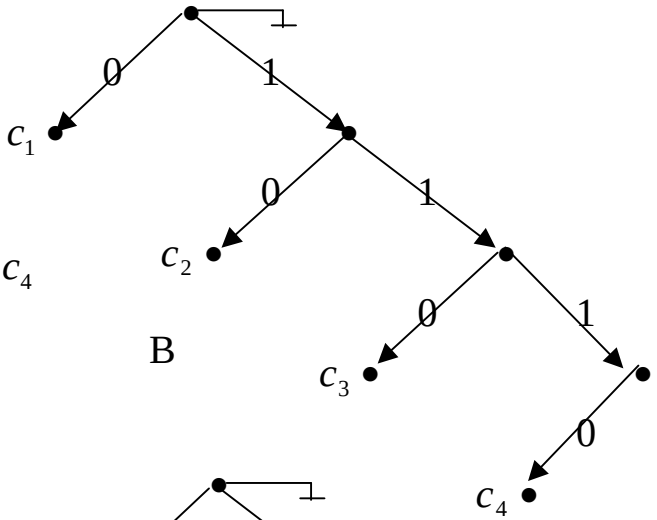
Mesaje s_k	Cod A	Cod B	Cod C	Cod D
s_1	00	0	0	0
s_2	01	10	01	10
s_3	10	110	011	110
s_4	11	1110	0111	111

- Coduri separabile: la care nu sunt necesare simboluri de demarcatie intre cuvinte; exemple: codurile A si D (la B “0” la sfarsit, la C “0” la inceput)
- Coduri instantanee: la care cuvintele se pot decoda imediat dupa receptionarea tuturor simbolurilor cuvantului fara a mai astepta simbolurile urmatoare (prin adaugarea unui simbol la cuvant nu se obtine un nou cuvant: sunt ireductibile); exemple: codurile A, B, D.

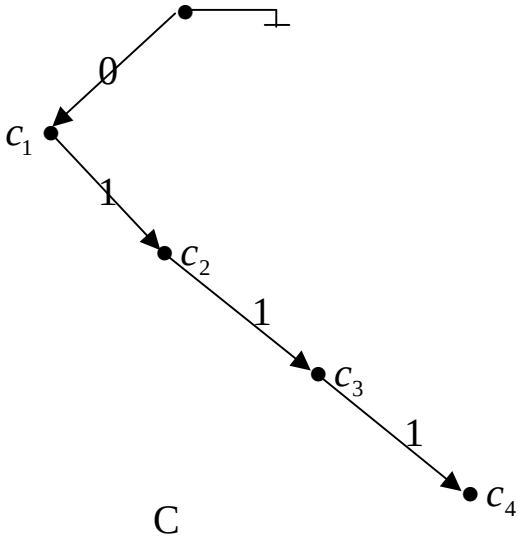
3. **Reprezentarea codurilor prin grafuri arbore binare** :



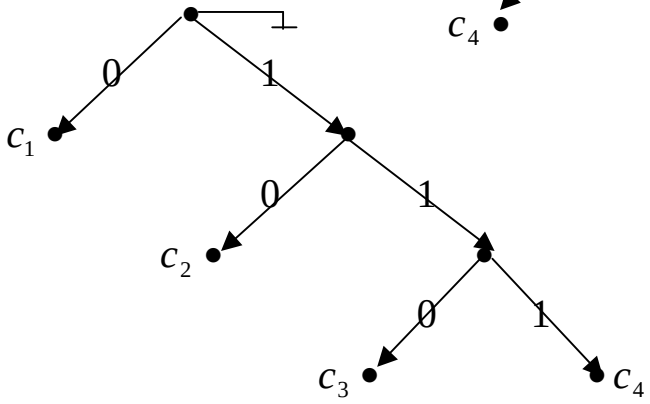
A



B



C



D

Obs.: la codul C, c_1, c_2, c_3 sunt prefixe pentru c_4

4. **Eficienta codarii**:

Pentru eficienta codarii se optimizeaza canalul in functie de un indicator de cost: durata medie de transmisie a cuvantului de cod (cost mediu):

$\overline{C} = \sum_{i=1}^{N} t_i p(c_i) = \sum t_i p(s_i)$ unde

$t_i = l_i \tau$ este durata de transmisie a cuvantului c_i,

$\tau =$ durata de transmisie a unui simbol;

l_i = lungimea cuvantului (numarul de simboluri din cuvant)

pentru $\tau = 1$ (aceeasi durata a simbolurilor): $t_i = l_i$

Se defineste:

- Lungimea medie a cuvantului de cod

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

Eficienta codarii: minimizarea $\overline{C} = \bar{l}$

- Limita inferioara a lungimii medii: pentru sursa caracterizata de :

$S = [s_1, ..., s_N]$

$P_S = [p(s_1), ..., p(s_N)]$

$C = [c_1, ..., c_N]$

$P_C = [p(c_1), ..., p(c_N)]$; $p(c_i) = p(s_i)$

$L = [l_1, ..., l_N]$

$X = [x_1, ..., x_D]$

$P_X = [p(x_1), ..., p(x_D)]$

Informatia medie pe simbol = informatia medie pe cuvant de cod, este entropia sursei:

$$H(S) = H(C) = -\sum_{i=1}^{N} [\log p(s_i)] p(s_i) \quad iar$$

$$H(X) = -\sum_{i=1}^{D} [\log p(x_i)] p(x_i)$$

este entropia alfabetului codului;

$H(S) = H(C) = \bar{l} H(X) \le \bar{l} \log D$ deoarece

$H(X) \le \log D = \max H(X)$ de unde rezulta limita inferioara a lungimii medii:

$$\boxed{\bar{l} \geq \frac{H(S)}{\log D} = \bar{l}_{\min}} \quad sau \quad \frac{H(S)}{\bar{l}} \leq \log D$$

adica: informatia medie pe simbolul alfabetului codului nu poate depasi entropia maxima a alfabetului codului (log D).

Conditia de eficienta maxima codarii: $H(C) = \bar{l}_{\min} \log D$

5. **Parametrii codului:**
   - **Capacitatea codului**:
     $C = \max H(X) = \log D$
   - **Eficienta codului**:
     $$\eta = \frac{\bar{l}_{\min}}{\bar{l}} = \frac{H(S)}{\bar{l} \log D} = \frac{H(X)}{\log D}$$
   - **Redundanta codului**:
     $$\rho = 1 - \eta = \frac{\bar{l} \log D - H(S)}{\bar{l} \log D} = \frac{\log D - H(X)}{\log D}$$

Coduri absolut optimale: $\bar{l} = \bar{l}_{\min}$ $(\eta = 1; \quad \rho = 0)$

Coduri optimale: la care, $\bar{l} \underset{\rightarrow}{>} \bar{l}_{\min}$ ($\eta \underset{\rightarrow}{<} 1; \quad \rho \underset{\rightarrow}{>} 0$)

Comparatie intre codurile A si D:

Pentru:

$[S] = [s_1, s_2, s_3, s_4]$

$[C] = [c_1, c_2, c_3, c_4]$

$[P_s] = \left[\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\right] = [P_c]$

$[X] = [1,0] \quad D = 2$

$H(S) = -\sum_{i=1}^{4} p(s_i) \log p(s_i) = \frac{7}{4}; \quad \bar{l}_{\min} = \frac{H(S)}{\log D} = \frac{7}{4}$

codul A:

$\bar{l} = \sum l_i p(s_i) = 2$

$\eta = \frac{l_{\min}}{\bar{l}} = \frac{7}{8} = 0{,}88$

codul B:

$\bar{l} = \sum l_i p(s_i) = \frac{7}{4}$

$\eta = \frac{l_{\min}}{\bar{l}} = 1$

6. Coduri absolut optimale si optimale

- Coduri absolut optimale; conditia de existenta a codurilor absolut optimale

$$H(S) = \bar{l}_{\min} \log D = [\sum_{i=1}^{N} l_i p(s_i)] \cdot \log D = \sum_{i=1}^{N} p(s_i) \log D^{l_i}; \textit{ deoarece}$$

$$H(S) = -\sum_{i=1}^{N} p(s_i) \log p(s_i) \quad \textit{rezulta}: l_i = -\frac{\log p(s_i)}{\log D} \quad \textit{si}$$

$$p(s_i) = p(c_i) = D^{-l_i} \quad \textit{si deoarece} \sum_{i=1}^{N} p(s_i) = 1 \textit{ se obtine}:$$

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

- Coduri optimale

  Daca $p(s_i) \neq D^{-l_i} \rightarrow \sum_{i=1}^{N} D^{-l_i} \leq 1$ rezultand *inegalitatea lui Mc Millan, care constituie teorema de existenta a codurilor optimale (instantanee, ireductibile)*

- Teorema I-a a lui Shannon

  Daca $p(s_i) \neq D^{-l_i} \rightarrow -\frac{\log p(s_i)}{\log D} \neq$ intreg; atunci l_i se poate alege ca sa satisfaca inegalitatea:

$$\frac{-\log p(s_i)}{\log D} \leq l_i < \frac{-\log p(s_i)}{\log D} + 1 \quad \textit{care prin inmultire cu } p_i \textit{ si sumare}$$

da:

$$\frac{H(S)}{\log D} \leq \bar{l} < \frac{H(S)}{\log D} + 1$$

relatie valabila pentru orice sursa fara memorie deci si pentru sursa extinsa [S^n] la care fiecare simbol este format dintr-un grup de *n* simboluri ale sursei [*S*]; in acest caz avem (sursa nu are memorie):

$H(S^n) = nH(S)$; in acest fel se face codarea pe grupe de *n* simboluri, lungimea medie a cuvantului de cod ce corespunde acestui grup de simboluri satisface relatia:

$$\frac{H(S^n)}{\log D} \le \bar{l}_n < \frac{H(S^n)}{\log D} + 1 \quad sau$$

$$\frac{H(S\ )}{\log D} \le \bar{l}_n / n < \frac{H(S\ )}{\log D} + 1/n \quad sau\ pentru\ n \to \infty$$

$$\lim_{n\to\infty} l_n \big/ n = \frac{H(S)}{\log D} = \bar{l}_{\min}$$

Teorema I-a a lui Shannon poate fi formulata astfel:

- Oricand se poate gasi un procedeu de codare absolut optimala a unei surse *S* astfel incat lungimea medie a cuvintelor de cod sa tinda catre $\bar{l}_{\min}$ si eficienta codului catre 1;
- Oricand se poate gasi un procedeu de codare a unei surse care sa apropie oricat de mult informatia medie pe simbol $H(S)/\bar{l}$ de capacitatea codului log *D* (codare absolut optimala).

In practica nu se poate ca *n* (numarul simbolurilor din grup)$\to\infty$; deaceea ne multumim cu un procedeu de codare care sa duca la o eficienta maxima, iar daca aceasta este subunitara, mai mica decat cea care se obtine pentru $n \to \infty$ am obtinut totusi o codare optimala. Adesea insa se se prefera din motive de simplitate cazul $n = 1$

- Procedee de codare optimala (compacta)
  - Principii generale:
    - Simbolurilor celor mai probabile ale sursei li se aloca cuvintele de cod cele mai scurte;
    - Lungimile l_i trebuie sa fie numere intregi;
    - Lungimile cuvintelor de cod sa satisfaca inegalitatea Mc Millan cat mai aproape de 1.

  - Codare simbol cu simbol ($n = 1$):
    - Procedeul Shannon – Fano:

Pentru codare cu *D* simboluri se fac urmatorii pasi:

a) Simbolurile sursei se aranjeaza in ordinea descrescatoare a probabilitatilor;

b) Se fac partitii succesive ale simbolurilor sursei in D submultimi de probabilitati (sume ale probabilitatilor simbolurilor grupate in submultime) cat mai apropiate; se atribuie fiecarei submultimi cat una din literele codului;
c) Partitiile se termina cand fiecare din submultimi contine cate un singur simbol al sursei;
d) Cuvantul de cod se obtine ca sir al literelor atribuite in partitiile succesive.

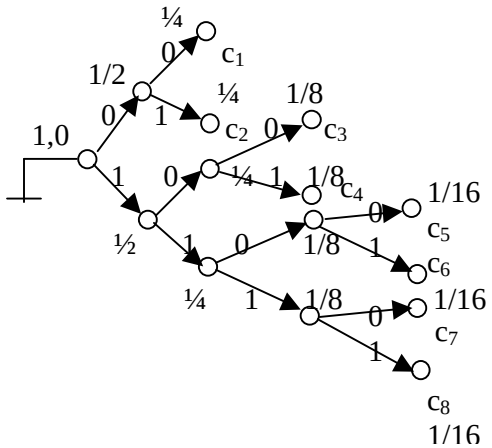
Exemplul 1[$p(s_i) = 2^{-l_i}$]: se considera sursa [S] = [s_1,…s_8] cu probabilitatile: [P] = [1/4, 1/4, 1/8, 1/8, 1/16, 1/16, 1/16, 1/16]. Se cere codarea compacta a sursei cu D = 2. Codarea se face ca mai jos:

s_i	$p(s_i)$	Partitii				c_i	l_i
s_1	1/4	0	0			00	2
s_2	1/4		1			01	2
s_3	1/8	1	0	0		100	3
s_4	1/8			1		101	3
s_5	1/16		1	0	0	1100	4
s_6	1/16				1	1101	4
s_7	1/16			1	0	1110	4
s_8	1/16				1	1111	4

Se calculeaza:

$\bar{l} = 2{,}75 = \bar{l}_{\min} = H(S); \quad \eta = 1; \quad \rho = 0$

Graful este:



Exemplul 2 [$p(s_i) \neq 2^{-l_i}$]: Se considera sursa S = [s_1,…, s_4] cu probabilitatile: [P] = [0,45 0,30 0,15 0,10]. Sa se codeze compact sursa cu D = 2

s_i	$p(s_i)$	partitii				c_i	l_i
s_1	0,45	0				0	1
s_2	0,30	1	0			10	2
s_3	0,15		1	0		110	3
s_4	0,10			1		111	3

Se calculeaza:

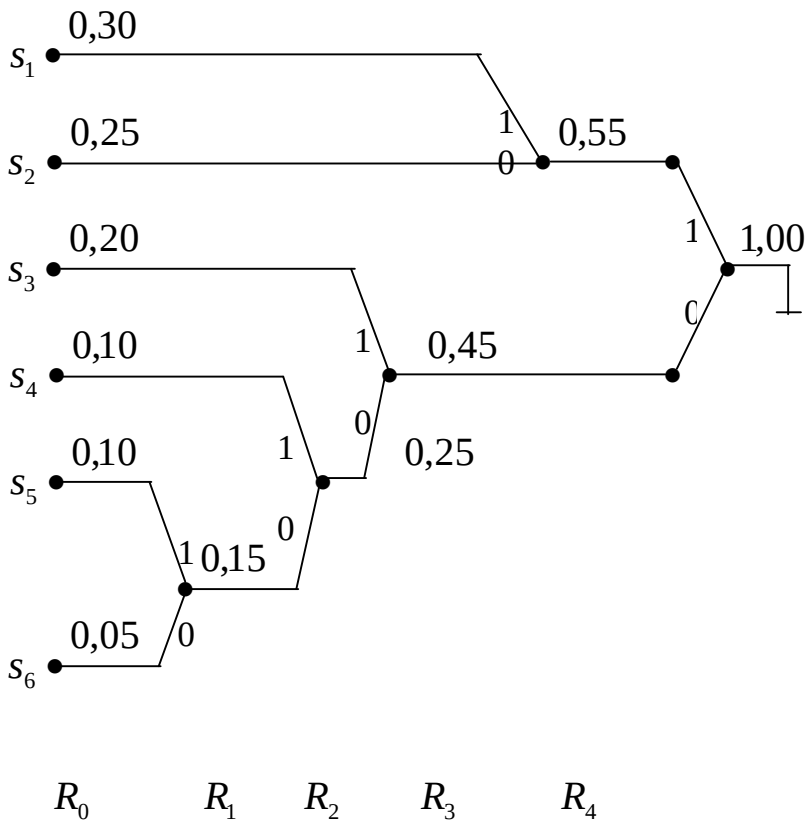
$H(S) = \bar{l}_{\min} = 1{,}782; \quad \bar{l} = 1{,}80; \quad \eta = 0{,}99$

- Procedeul de codare Huffman

Pentru codare cu *D* simboluri se fac urmatorii pasi:

a) Cele N simboluri ale sursei se aranjeaza in ordinea descrescatoare a probabilitatilor; se noteaza sursa primara cu R_0;
b) Se grupeaza ultimele *D* simboluri de cele mai mici probabilitati, intr-un simbol artificial r_1 cu probabilitatea $p(r_1)$ egala cu suma probabilitatilor simbolurilor grupate; se obtine sursa restransa de ordin 1 R_1;
c) Se asigneaza cate una din literele alfabetului codului celor *D* simboluri grupate;
d) Se repeta pasii precedenti pana cand se ajunge la o sursa restransa $R_{\text{N-D}}$ care furnizeaza doar *D* simboluri;
e) Cuvantul de cod complet, corespunzator unui simbol al sursei primare, este constituit din secventa literelor codului obtinuta prin parcurgerea surselor restranse in sensul opus restrangerii, pana la gasirea simbolului original; aceasta echivaleaza cu parcurgerea unui arbore de la un nod final la radacina.

Exemplu: Se considera sursa cu [*S*] = [s_1,…,s_6] si [*P*] = [0,3 0,25 0,20 0,10 0,10 0,05]. Se cere sa se codeze sursa cu *D* = 2



Cuvintele de cod:

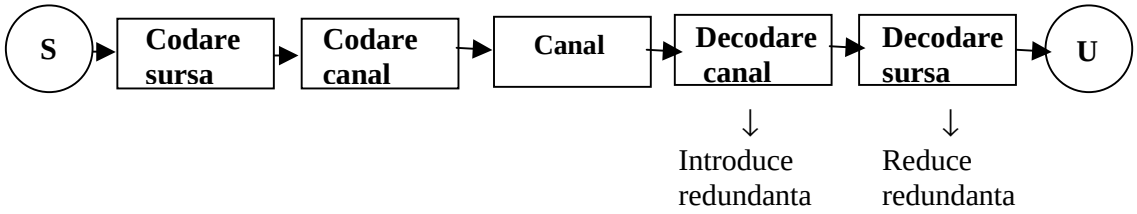
$c_1 = 11$; $c_2 = 10$; $c_3 = 01$; $c_4 = 001$; $c_5 = 0001$; $c_6 = 0000$

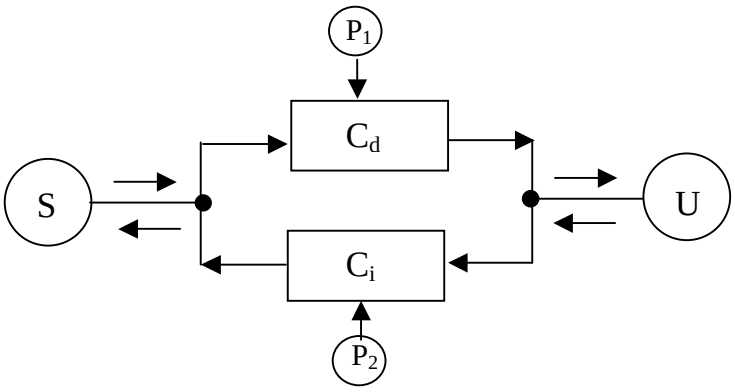
Parametrii codarii:

$H(S) = 2,3$; $\bar{l} = 2,4$; $\bar{l}_{\min} = 2,3$; $\eta = 0,96$

VII Codarea pentru canalele cu perturbatii

1. **<u>Obiectivul codarii</u>**: imbunatatirea stabilitatii la perturbatii; se face prin adaugare de redundanta, de fapt *simboluri de control* care permit detectia sau corectia erorilor. Locul codarii de canal intr-un sistem de comunicatii se arata mai jos:





Sistemul alaturat este un s*istem detector de erori, ARQ*, cu cerere automata de repetare (automatic repeat request), folosit la o sursa cu debit controlabil la care se controleaza oprirea si pornirea. Pentru detectia erorilor este necesar canalul de intoarcere C_i, de capacitate redusa, prin care se solicita sursei repetarea mesajului eronat.

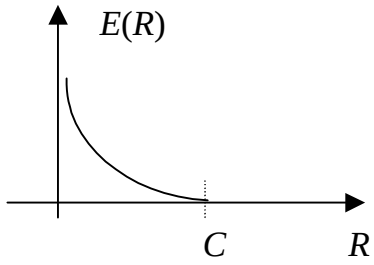
La canale cu perturbatii mari, pentru a limita repetarile, se foloseste si un *sistem automat de corectia erorilor*.

2. **<u>Categorii de coduri</u>**:
   - <u>Coduri bloc:</u> la care informatia este organizata in cuvinte (blocuri de *n* simboluri)
     - <u>Coduri grup:</u> la care cuvintele sunt privite ca vectori intr-un spatiu vectorial;
     - <u>Coduri ciclice:</u> la care cuvintele sunt privite ca elemente intr-o algebra.
   - <u>Coduri convolutionale (recurente):</u> la care prelucrarea simbolurilor generate de sursa se face continuu.

3. **<u>Teorema a II-a a lui Shannon</u>**: *daca avem o sursa cu debitul de informatie R (bit/s) si un canal de capacitate C (bit/s) si daca R < C, exista un cod cu cuvinte de lungime n astfel incat probabilitatea unei erori de decodare P_E sa fie:*

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

$E(R)$ este o functie nenegativa (vezi fig. alaturata), numita exponentul erorii.
Obs.: teorema afirma ca, *indiferent cat este perturbatia din canal, se pot face transmisii cu probabilitate de eroare oricat de mica. !!*



4. **<u>Coduri grup binare</u>**

Se reprezinta matriceal literele cuvantului;
w = [a_1, …,a_n], unde a_i sunt elementele unui camp cu doua elemente notate (0, 1). In spatiul vectorial regulile de operare sunt:

adunare modulo 2 si multiplicare:

⊕	0	1
0	0	1
1	1	0

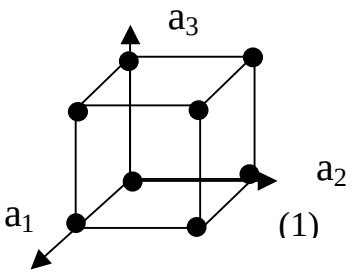
⊗	0	1
0	0	0
1	0	1

Simbolurile 0 / 1 pot insemna: absenta / prezenta semnal; sinusoida de frecventa f_1 / sinusoida de frecventa f_2 (inreg. pe banda magnetica); u = 0 V / u = 2,7 V (TTL), etc.
Cuvinte cu sens: care corespund cu mesajele generate de sursa (cuvinte de cod). Daca se noteaza cu W multimea cuvintelor, in numar de $N = 2^n$, cu V multimea cuvintelor cu sens, in numar $S = 2^k$ si cu F multimea cuvintelor fara sens, in numar $2^n / 2^k = 2^{n-k}$, avem urmatoarele situatii:

- $S = N = 2^n$ (toate cuvintele au sens), informatia medie pe cuvant este: $I = \log N = n$ iar informatia medie pe simbol este $i_n = I/n = 1$ (bit/simb)
- $S = 2^k$ unde $k < n$; atunci: $I = \log S = k$ si $i_k = k/n < i_n$, intervenind redundanta: cele $n - k$ simboluri redundante se

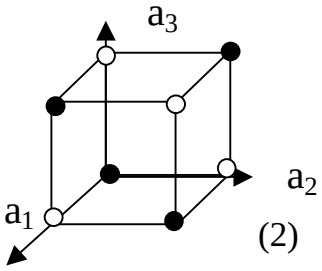
folosesc la detectia si corectia erorilor. Se impune deci ca F sa nu fie vida.

Obs.: se poate ilustra grafic (vezi fig. de mai jos) alegerea cuvintelor de cod pentru $n = 3$ (cu • s-a reprezentat cuvantul cu sens si cu o s-a reprezentat cel fara sens).



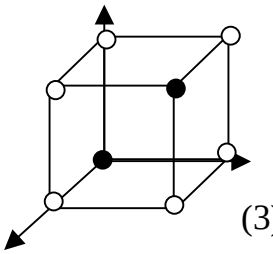
(1)

Toate cuvintele ce se pot forma ($N = 2^3$) sunt cu sens: orice cuvant de cod, prin eronare, se transforma in alt cuvant de cod: nu se poate face detectie sau corectie.



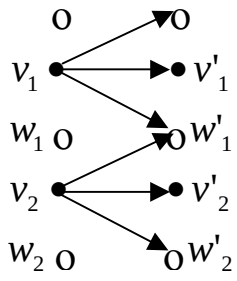
(2)

Se aleg $S = 2^2$ cuvinte de cod; aparitia <u>unei</u> erori conduce la un cuvant fara sens, modificandu-se doar <u>una</u> din coordonate: se poate face detectia <u>unei</u> erori (la a doua eroare cuvantul devine cuvant de cod) dar nu si corectia ei.


(3)

Se aleg $S = 2$ cuvinte de cod; se pot detecta <u>doua</u> erori sau corecta o eroare.

Graful de tranzitie pentru cuvintele de cod transmise pe un canal cu perturbatii este:



<u>Cuvintele ca elemente ale claselor “alaturate”</u>

Elementele multimii W se pot aranja in urmatorul tabel, astfel:

- in prima linie (clasa a I-a): elementele multimii V (cuvintele cu sens $v \in V$), incepand cu elementul 0 (matricea 0);
- in linia a doua (clasa a II-a) se incepe cu unul din cuvintele fara sens cu numarul cel mai mic de componente 1, notat cu $\boldsymbol{\varepsilon}_1$; restul elementelor se formeaza, adunand modulo 2 pe $\boldsymbol{\varepsilon}_1$ la elementele primei linii;

- in linia a treia (clasa a III-a) se incepe din nou cu unul din cuvintele fara sens cu numarul cel mai mic de componente 1, notat cu $\boldsymbol{\varepsilon}_2$; restul elementelor se formeaza, adunand modulo 2 pe $\boldsymbol{\varepsilon}_2$ la elementele primei linii;
- operatia se continua pana cand se epuizeaza elementele din *W*.

$$\begin{array}{ccccc} 0 & \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_{s-1} \\ \boldsymbol{\varepsilon}_1 & \mathbf{v}_1 \oplus \boldsymbol{\varepsilon}_1 & \mathbf{v}_2 \oplus \boldsymbol{\varepsilon}_1 & \dots & \mathbf{v}_{s-1} \oplus \boldsymbol{\varepsilon}_1 \\ \boldsymbol{\varepsilon}_2 & \mathbf{v}_1 \oplus \boldsymbol{\varepsilon}_2 & \mathbf{v}_2 \oplus \boldsymbol{\varepsilon}_2 & \dots & \mathbf{v}_{s-1} \oplus \boldsymbol{\varepsilon}_2 \\ \dots & \dots & \dots & \dots & \dots \end{array}$$

Dupa cum se stie, avem:

$$\mathbf{v}_i \oplus \boldsymbol{\varepsilon}_j = \mathbf{v}_i' \in W \quad \forall\, \mathbf{v}_i \in V \quad si \quad \boldsymbol{\varepsilon}_j \in W,$$

deci in tablou avem toate cuvintele din *W*, in prima linie fiind cele cu sens. Pe baza tabloului se face decodarea, recunoscand coloana in care figureaza cuvantul receptionat $\mathbf{v}_i$', primul element din coloana fiind cuvantul transmis de sursa, respectiv cuvantul in care erorile sunt corectate.

Exemplu: *W* are $2^4 = 16$ cuvinte in care $2^1 = 2$ sunt cuvinte cu sens. Clasele alaturate sunt:

[0000]	[1111]
[0001]	[1110]
[0010]	1101]
[0100]	[1011]
[1000]	[0111]
[0011]	[1100]
[0101]	[1010]
[1001]	[0110]

Obs.: - daca receptionam cuvantul: 1 0 0 0
decidem ca s-a transmis cuvantul: 0 0 0 0
- daca receptionam cuvantul: 1 0 1 1
decidem ca s-a transmis cuvantul: 1 1 1 1
- cuvintele din prima coloana (primele elemente in clasele alaturate) indica pozitia in care au intervenit erori: au 1 in pozitiile eronate restul fiind 0; acestea sunt *cuvintele eroare*. Cuvintele dintr-o coloana sunt mai "apropiate" de primul cuvant din coloana respectiva decat de oricare alt cuvant.

5. Distanta Hamming

Cuvintele de cod se noteaza cu $\mathbf{v}_i$ iar cuvintele receptionate cu $\mathbf{v}'_i$ (provin din $\mathbf{v}_i$ dar pot fi diferi de acestea datorita perturbatiilor),

$$\mathbf{v}_i = [a_{i1}\ a_{i2}\ \dots a_{in}]$$

$$\mathbf{v}'_i = [a'_{i1}\ a'_{i2} \dots a'_{in}]$$

$$daca \quad \mathbf{v}_i = \mathbf{v}'_i \ \ (a_{ij} = a'_{ij}) \rightarrow transmisie\ \ fara\ eroare$$

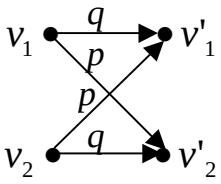
Daca avem doua cuvinte $\mathbf{v}_i$ si $\mathbf{v}_j$, prin definitie *functia distanta* este:

$$D(\mathbf{v}_i, \mathbf{v}_j) = \sum_{k=1}^{n} (a_{ik} \oplus a_{jk})$$

Obs.: distanta intre doua cuvinte de cod este egala cu nr. de simboluri (coordonate) prin care cele doua cuvinte se deosebesc. Pentru codurile din fig. de mai sus: $D = 1$, pentru (1); $D = 2$, pentru (2); $D = 3$, pentru (3).

- <u>Decizia pe baza distantei minime</u>
  Se presupune ca transmisia se face prin canal binar simetric fara memorie unde q este probabilitatea de transmisie corecta si p este probabilitatea de transmisie eronata

  Graful de tranzitie este:
  unde $p + q = 1$

  Daca se receptioneaza $\mathbf{v}'_i$, probabilitatea sa provina din $\mathbf{v}_i$ este:
  $$p(\mathbf{v}_i / \mathbf{v}_i') = p^{D(\mathbf{v}_i', \mathbf{v}_i)} q^{n - D(\mathbf{v}_i', \mathbf{v}_i)}$$
  iar probabiltatea sa provina din v_j este:
  $$p(\mathbf{v}_j / \mathbf{v}_i') = p^{D(\mathbf{v}_i', \mathbf{v}_j)} q^{n - D(\mathbf{v}_i', \mathbf{v}_j)}$$
  pentru p << 1 si q ≈ 1:
  $$p(\mathbf{v}_i / \mathbf{v}_i') \cong p^{D(\mathbf{v}_i', \mathbf{v}_i)}$$
  $$p(\mathbf{v}_j / \mathbf{v}_i') \cong p^{D(\mathbf{v}_i', \mathbf{v}_j)}$$
  daca $D(\mathbf{v}_i', \mathbf{v}_i) < D(\mathbf{v}_i', \mathbf{v}_j) \quad \forall\ j \neq i$ atunci *se poate decide pe baza relatiei:* $p(\mathbf{v}_i / \mathbf{v}_i') > p(\mathbf{v}_j / \mathbf{v}_i') \quad \forall\ j \neq i$ *ca s-a transmis* v_i *daca s-a receptionat* v_i*'* (decizie pe baza distantei minime sau pe baza probabilitatii conditionate maxime)

- <u>Regiuni de decizie</u>
  Se poate face o partitie a multimii W in submultimi disjuncteW_i in jurul punctelor $v_i \in W_i$ cu proprietatea ca toate punctele $v_i' \in W_i$ sunt mai aproape de punctul v_i decat de oricare punct v_j pentru $j \neq i$. Daca pentru un punct v_i*'* are loc relatia:
  $D(\mathbf{v}_i', \mathbf{v}_i) < D(\mathbf{v}_i', \mathbf{v}_j) \quad \forall\ j \neq i$,
  atunci
  $\mathbf{v}_i' \in W_i$

Multimea punctelor $\mathbf{v}_i$' care se gasesc la aceeasi distanta $D(\mathbf{v}_i', \mathbf{v}_i) = D(\mathbf{v}_i', \mathbf{v}_j)$ fata de $\mathbf{v}_i$ si $\mathbf{v}_j$ si care nu pot fi inglobate in una din W_i, se noteaza cu W_0.

In acest caz :

$W = V \cup F = W_0 \cup W_1 \cup,...,\cup W_k$

La receptionarea $\mathbf{v}_i$' pot apare urmatoarele situatii:

- $\mathbf{v}_i' \in F \Rightarrow \mathbf{v}_i'$ nu este cuvant de cod si se poate face detectia erorilor;
- $\mathbf{v}_i' \in W_i$ respectiv $D(\mathbf{v}_i', \mathbf{v}_i) < D(\mathbf{v}_i', \mathbf{v}_j) \quad \forall\ j \neq i$, se poate face corectia erorilor: se decide ca v_i' provine din v_i
- $\mathbf{v}_i' \in W_0$ respectiv $D(\mathbf{v}_i', \mathbf{v}_i) = D(\mathbf{v}_i', \mathbf{v}_j)$, erorile nu pot fi corectate dar pot fi detectate deoarece $W_0 \in F \quad si \quad \mathbf{v}_i' \in F$.

Obs.: *Posibilitatile de detectie si corectie ale unui cod depind de distanta minima intre doua cuvinte de cod* (toate elementele $\mathbf{v}_i' \in W_i$ se decodifica in $\mathbf{v}_i$ daca $\mathbf{v}_i \in W_i$ se gaseste la distanta mare de de cel mai apropiat cuvant cu sens $\mathbf{v}_j \in W_j$). Astfel,

- pentru detectia a e_d erori, distanta minima intre cuvintele de cod trebuie sa fie: $D_{min} = e_d + 1$
- pentru corectia a e_c erori, $D_{min} = 2e_c + 1$

6. Cuvantul eroare

Se defineste un cuvant eroare ε care are simboluri din acelasi alfabet cu cuvintele de cod si cu aceeasi lungime n, generat de perturbatiile din canal. Cuvantul eroare poate fi scris in forma:

$$\varepsilon = [\varepsilon_1\ \varepsilon_2\ ...\varepsilon_n]$$

Cuvantul eronat se obtine cu relatia:

$$\mathbf{v}_i' = \mathbf{v}_i \oplus \boldsymbol{\varepsilon} \text{ (transformarea directa)}$$

Exemplu (caz binar): pentru $\mathbf{v}_i = [100]$ si o perturbatie $\boldsymbol{\varepsilon} = [010]$ cuvantul eronat va fi: $\mathbf{v}_i' = [110]$

Cuvantul corectat se obtine cu relatia:

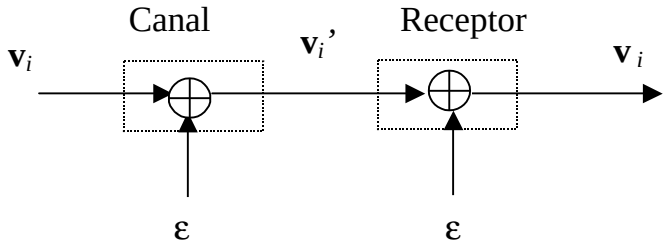
$$\mathbf{v}_i = \mathbf{v}_i' \oplus \boldsymbol{\varepsilon} \text{ (transformarea inversa)}$$

cu conditia cunoasterii lui $\boldsymbol{\varepsilon}$.

Cuvantul eroare mai poate fi scris si sub forma:

$$\varepsilon = [...\alpha_{i_1} ...\alpha_{i_2} ...]$$

unde α_{i_k} sunt simboluri 1, celelalte pana la n fiind 0 iar i_k sunt numere de la 1 la n, care arata pozitia in care apare eroarea.
Modelul de introducere si corectare a erorilor este dat in fig.:

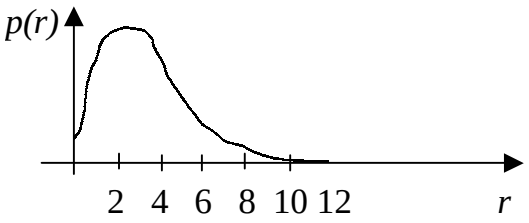


- Erori
  - Erori individuale: ele apar ca rezultat al unor perturbatii accidentale mai scurte decat durata simbolului si afecteaza in mod individual (izolat) fiecare simbol;

Probabilitatea de aparitie a unor r simboluri eronate intr-un cuvant de n simboluri este data de o lege de distributie binomiala de forma:

$p(r) = C_n^r p^r (1-p)^{n-r}$ iar probabilitatea de aparitie a $e \le r$ erori este data de functia de repartitie corespunzatoare:

$$P(r \le e) = \sum_{r=0}^{e} p(r) = \sum_{r=0}^{e} C_n^r p^r (1-p)^{n-r}$$



Pentru un caz particular (canal foarte perturbat) $p(r)$ este reprezentat alaturat: se vede ca scade rapid peste o valoare a lui r

Pentru un canal obisnuit $p \approx 10^{-3}...10^{-7}$ iar pentru canale profesionale $p \approx 10^{-10}$

  - Erori in pachete: ele apar pentru perturbatii care dureaza mai mult ca simbolul si de aceea apar grupate; ele apar deobicei ca rezultat al unor intreruperi in canalul de transmisie sau defectiuni in suportul de stocare a informatiei.

Definitii:

Pachete de erori: o succesiune de simboluri (corecte sau eronate), in care primul si ultimul simbol sunt eronate si in care simbolurile corecte succesive apar in grupuri de $s < r$ simboluri;

Lungimea *l* a unui pachet de erori: numarul total de simboluri (eronate sau neeronate) care formeaza pachetul de erori;

Densitatea *D* a erorilor: raportul dintre numarul de simboluri eronate din pachet si numarul total de simboluri;

Cuvantul eroare care contine un pachet de erori de lungime *l* va avea *l* simboluri 0 si 1, din care primul si ultimul (simboluri eronate) sunt 1. El poate fi scris sub forma:

$\varepsilon = [...\alpha_i \varepsilon_{i+1} ... \varepsilon_{i+l-2} \alpha_{i+l-1} ...]$ unde:

α_i este simbolul 1 plasat in pozitia *i*;

ε_{i+k} ese simbolul 0 sau1 plasat in pozitia $i+k$, cu restrictia ca nu pot sa apara succesiv un numar de “0” mai mare sau egal cu “*r*”.

VIII Mecanismul de detectie si de corectie a erorilor

1. **Corectori:** elemente $z \in Z$ destinate sa indice pozitiile din cuvantul cod in care s-au introdus erori: se stabileste o corespondenta univoca intre multimea *W* si multimea *Z*, definind operatorul $\mathcal{H}$ asa incat $\mathcal{H}\{\mathbf{v_i}'\} = \mathbf{z} \quad \mathbf{v_i}' \in \mathbf{W} \quad \mathbf{z} \in \mathbf{Z}$. Se impune conditia:
   $\mathcal{H}\{v_i'\} = 0$ *daca* $v_i' = v_i$ *adica* v_i' *este cuvant de cod (transmisie fara erori)*
   $\mathcal{H}\{v_i'\} = z \neq 0$ *in caz contrar*
2. **Conditia de corectia a erorilor:** pentru fiecare cuvant eroare generat de perturbatiile din canal sa existe un singur corector distinct diferit de zero sau: *corespondenta intre elementele multimii E a tuturor cuvintelor eroare si cele ale multimii Z a corectorilor trebuie sa fie biunivoca.*
   Aceasta corespondenta se poate stabili, definind un alt operator $\mathcal{D}$:
   $\mathcal{D}\{\mathbf{z}\} = \boldsymbol{\varepsilon} \quad \mathcal{D}^{-1}\{\boldsymbol{\varepsilon}\} = \mathbf{z}$
3. **Mecanismul de detectie sau corectia erorilor** este:
   - Daca $\mathcal{H}\{\mathbf{v_i}'\} \neq 0$, se spune ca s-a facut detectia erorilor;
   - Daca $\mathcal{H}\{\mathbf{v_i}'\} = \mathbf{z}$ **cunoscut**, se procedeaza astfel:
     $\mathcal{D}\{z\} = \varepsilon$, *obtinand cuvantul eroare si*
     $v_i = v_i' \oplus \varepsilon$, *obtinand cuvantul corectat (cuvantul de cod transmis)*
4. **Matricea de corectie a erorilor:**
   Se considera o transformare liniara univoca de la *W* la *Z*, definita de ecuatiile:

$$h_{11}a_1' + h_{12}a_2' + \ldots + h_{1n}a_n' = z_1$$

.....

.....

$$h_{m1}a_1' + h_{m2}a_2' + \ldots + h_{mn}a_n' = z_m$$

unde: a_i' sunt simbolurile cuvantului receptionat $\mathbf{v_i}' = [a_1' a_2' \ldots a_n']$

z_i sunt componentele corectorului $\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \ldots \\ z_m \end{bmatrix}$

h_{ij} sunt parametrii care determina transformarea $\mathcal{H}$; ei pot fi ordonati intr-o matrice, H, care in continuare se va numi (din motive ce vor fi explicate mai departe) matricea de control:

$$\mathbf{H}=\begin{bmatrix} h_{11} & h_{12} & \dots & h_{1n} \\ h_{21} & h_{21} & \dots & h_{2n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ h_{m1} & h_{m2} & \dots & h_{mn} \end{bmatrix}$$

Ecuatiile mai pot fi scrise astfel:

$\mathbf{Hv'^{T}}=\mathbf{z}$

sau, daca $\mathbf{v'}=\mathbf{v}$, $\mathbf{Hv'^{T}}=\mathbf{Hv^{T}}=0$.

5. Codarea codurilor grup cu ajutorul matricei de control H

In figura de mai jos este data schema codarii pe baza matricii **H.**

Relatia de codare este $\mathbf{Hv^{T}}=0$ si este un sistem de *m* ecuatii cu *m* necunoscute: simbolurile de control.

- Coduri sistematice:

Convenim in cele ce urmeaza ca in cuvantul de cod primele *m* simboluri sa fie simboluri redundante care servesc detectiei sau corectiei de erori (*simboluri de control*), iar ultimele *k* simboluri sa fie simboluri generate de sursa (*simboluri de informatie*); ca urmare cuvantul de cod are forma:

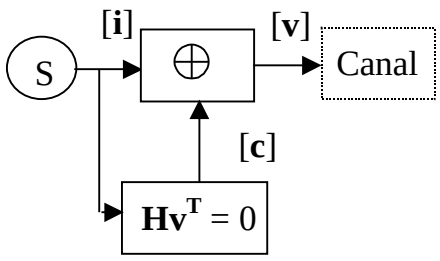
$\mathbf{v_i}'=[a_1 \dots a_m a_{m+1} \dots a_n]$; daca se noteaza:

$\mathbf{c}=[a_1 \dots a_m]$ si $\mathbf{i}=[a_{m+1}\ a_{m+2} \dots a_{m+k}]$

unde $m+k=n$

avem:

$\mathbf{v}=[\mathbf{ci}]$



Pentru a determina cele *m* simboluri de control in functie de cele *k* simboluri de informatie, se pune conditia ca **v** sa fie cuvant de cod (corectorul corespunzator sa fie nul):

$\mathbf{Hv^{T}}=[\mathbf{I_m Q}]\begin{bmatrix}\mathbf{c^{T}}\\ \mathbf{i^{T}}\end{bmatrix}=0$ respectiv $\mathbf{c^{T}}+\mathbf{Qi^{T}}=0$ sau $\mathbf{c^{T}}=\mathbf{Qi^{T}}$ sau

$$\begin{bmatrix} q_{11} & q_{12} & \dots & q_{1k} \\ q_{21} & q_{22} & \dots & q_{2k} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ q_{m1} & q_{m2} & \dots & q_{mk} \end{bmatrix}\begin{bmatrix} a_{m+1} \\ a_{m+2} \\ . \\ . \\ a_{m+k} \end{bmatrix}=\begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ a_m \end{bmatrix}$$

de unde:

$$a_j=\sum_{i=1}^{k} q_{ji} a_{m+1} \qquad j=\overline{1,m}$$

In general, codurile sistematice pot avea simbolurile de control si de informatie grupate atat pe pozitiile initiale si respectiv finale (ca mai sus) cat si invers. Ele au avantajul ca operatia de corectie a erorilor e simpla ca si separarea simbolurilor de informatie.

- Relatii intre coloanele matricei **H** in cazul corectiei erorilor
  Pentru corectia a *e* erori din cuvantul **v':**
  $\mathbf{v}' = \mathbf{v} \oplus \boldsymbol{\varepsilon}$
  $\boldsymbol{\varepsilon} = \lfloor ... \alpha_{i_1} .... \alpha_{i_e} ... \rfloor$

conditiile pentru **H** cand trebuie corectate cele *e* erori (simboluri 1) se determina astfel:

$\mathbf{z} = \mathbf{H}\mathbf{v}'^{\mathbf{T}} = \mathbf{H}\mathbf{v}^{\mathbf{T}} + \mathbf{H}\boldsymbol{\varepsilon}^{\mathbf{T}} = \mathbf{H}\boldsymbol{\varepsilon}^{\mathbf{T}}$ sau

$\mathbf{z} = [\mathbf{h}_1\ \mathbf{h}_2 ... \mathbf{h}_\mathbf{n}] \begin{bmatrix} ... \\ \alpha_{\mathbf{i}_1} \\ ... \\ \alpha_{\mathbf{i}_e} \\ ... \end{bmatrix}$, unde $\mathbf{h_i} = \begin{bmatrix} h_{1i} \\ h_{2i} \\ ... \\ h_{mi} \end{bmatrix}$ sunt coloanele lui **H,** sau

$\mathbf{z} = \lfloor .. \alpha_{\mathbf{i}_1} \mathbf{h}_{\mathbf{i}_1} + ... + \alpha_{\mathbf{i}_e} \mathbf{h}_{\mathbf{i}_e} + ... \rfloor = \lfloor \mathbf{h}_{\mathbf{i}_1} + ... + \mathbf{h}_{\mathbf{i}_e} \rfloor$ deoarece $\alpha_{\mathbf{i}_k} = 1$

Pentru a corecta e erori indiferent de pozitiile in care intervin este necesar ca sumele modulo 2 a cate e coloane oarecare din matricea **H** *sa fie distincte: in acest fel se obtin corectori distincti pentru fiecare cuvant eroare care contine e erori.*

Daca coloanele lui **H** sunt astfel alese incat :

$\mathbf{h}_{\mathbf{i}_1} + ... + \mathbf{h}_{\mathbf{i}_e} \neq \mathbf{h}_{\mathbf{j}_1} + ... + \mathbf{h}_{\mathbf{j}_e} \quad \forall\, \mathbf{i_k}\ \mathbf{distincte} = \overline{1,\mathbf{n}}\ \mathbf{si}\ \forall \mathbf{j_k}\ \mathbf{distincte} = \overline{1,\mathbf{n}}$

adunand in ambii membrii coloanele $\mathbf{h}_{\mathbf{j}_1} ... \mathbf{h}_{\mathbf{j}_e}$ se obtine:

$\mathbf{h}_{\mathbf{i}_1} + ... + \mathbf{h}_{\mathbf{i}_e} + \mathbf{h}_{\mathbf{j}_1} + ... + \mathbf{h}_{\mathbf{j}_e} \neq 0$ **sau**

$\mathbf{h}_{\mathbf{i}_1} + ... + \mathbf{h}_{\mathbf{i}2\mathbf{e}} \neq 0 \quad \forall\, \mathbf{i_k} = \overline{1,\mathbf{n}}$

Daca coloanele matricei **H** indeplinesc conditiile de mai sus, atunci exista posibilitatea de a se corecta *e* erori oarecare.

- Relatii intre coloanele matricei H in cazul detectiei erorilor
  Pentru detectia a *d* erori:
  $\boldsymbol{\varepsilon} = \lfloor .. \alpha_{\mathbf{i}_1} ... \alpha_{\mathbf{i}_\mathbf{d}} ... \rfloor$ din conditia:
  $\mathbf{H}'\boldsymbol{\varepsilon}^{\mathbf{T}} \neq 0$ rezulta

$$\left[\mathbf{h'_{1d}}+...+\mathbf{h'_{i_d}}\right]\begin{bmatrix}...\\ \alpha_{\mathbf{i_1}}\\ ...\\ \alpha_{\mathbf{i_d}}\\ ...\end{bmatrix}\neq 0 \quad \text{sau:}$$

$$\left[\mathbf{h'_{i1}}+...+\mathbf{h'_{i_d}}\right]\neq 0 \quad \forall \mathbf{i_k}\,\mathbf{distinct}=\overline{1,\mathbf{n}} \quad \text{adica:}$$

Pentru a detecta d erori indiferent de pozitiile in care intervin este necesar ca sumele modulo 2 a cate d coloane oarecare din matricea **H'** *sa fie diferite de zero fara a fi neaparat distincte. In cazul detectiei unei singure erori, toate coloanele lui* **H'** *trebuie sa fie diferite de zero dar nu neaparat distincte.*

Daca numarul erorilor care trebuie detectate este impar: d = 2p + 1 conditia de detectie devine:

$\mathbf{h'_{i_1}}+...+\mathbf{h'_{i_{2p+1}}}\neq 0$ care este satisfacuta daca se impune:

$$\mathbf{h'_{i_1}}=\begin{bmatrix}\mathbf{h_{i_1}}\\ 1\end{bmatrix},\quad \mathbf{h'_{i_2}}=\begin{bmatrix}\mathbf{h_{i_2}}\\ 1\end{bmatrix},... \quad \text{sau:}$$

$\mathbf{H'}=\begin{bmatrix}\mathbf{h_1} & \mathbf{h_2} & ... & \mathbf{h_n}\\ 1 & 1 & ... & 1\end{bmatrix}$ care daca se introduce in prima relatie se obtine (tinand cont ca suma modulo 2 a unui numar par de simboluri 1 este nula) :

$\begin{bmatrix}\mathbf{h_{i_1}}+...+\mathbf{h_{i_{2p}}}\\ 0\end{bmatrix}+\begin{bmatrix}\mathbf{h_{i_{2p+1}}}\\ 1\end{bmatrix}\neq 0$ relatie satisfacuta pentru orice valori ale coloanelor $\mathbf{h_k}$ in particular pentru valori nule. In acest caz:

$\mathbf{H'}=\left[1,1,...1\right]$ si in cuvantul de cod :

$\mathbf{v}=\left[a_1\ a_2...a_{n-1}a_n\right]$

ultimele k = n – 1 simboluri reprezinta simboluri de informatie iar primul simbol a_1 este simbolul de verificare a paritatii determinat de:

$\mathbf{H'}\,\mathbf{v^T}=0$ sau, tinand cont de $\mathbf{H'}=\left[1,1,...1\right]$,

$a_1=a_2+...+a_n$

Daca la receptie $\mathbf{H'}\,\mathbf{v'^T}\neq 0$ respectiv:

$a_1'+a'_2+...+a'_n\neq 0$ se poate afirma ca in transmisie s-a introdus un numar impar de erori.

Daca se introduce sau se adauga pe langa simbolurile de corectie un simbol de control al paritatii se creeaza posibilitatea sa se corecteze o eroare si sa se detecteze toate erorile duble. In acest caz matricea de control devine:

$$\mathbf{H}' = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & ... & \mathbf{h_n} \\ 1 & 1 & ... & 1 \end{bmatrix} \text{ sau } \mathbf{H}' = \begin{bmatrix} 0 & \mathbf{h}_1 & \mathbf{h}_2 & ... & \mathbf{h_n} \\ 1 & 1 & 1 & ... & 1 \end{bmatrix}$$

Corectorul este:

$$\mathbf{z} = \mathbf{H}' \mathbf{v}'^{\mathbf{T}} = \begin{bmatrix} \mathbf{z}_1 \\ z_2 \end{bmatrix}$$

Sau:

$$\begin{bmatrix} 0 & \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{...} & \mathbf{h_n} \\ 1 & 1 & 1 & \mathbf{...} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{a'}_0 \\ \mathbf{a'}_1 \\ \mathbf{...} \\ \mathbf{a'_n} \end{bmatrix} = \begin{bmatrix} 0 + \mathbf{a'}_1 \mathbf{h}_1 + ... + \mathbf{a'_n} \mathbf{h_n} \\ \mathbf{a'}_0 + \mathbf{a'}_1 + ... + \mathbf{a'_n} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \mathbf{z}$$

Se considera ca matricile coloana $\mathbf{h_k}$ din **H** indeplinesc conditiile necesare pentru corectia unei erori; atunci decidem urmatoarele:

- $\mathbf{z}_1 = 0$ si $z_2 = 0$ unde $\mathbf{z}_1$ are *m* elemente: nu avem erori;
- $\mathbf{z}_1 \neq 0$ si $z_2 = 1$: exista o eroare corectabila;
- $\mathbf{z}_1 = 0$ si $z_2 = 1$: a'_0 este eronat;
- $\mathbf{z}_1 \neq 0$ si $z_2 = 0$: exista doua erori necorectabile.

- **<u>Codarea codurilor grup cu ajutorul matricei generatoare G</u>**

Se defineste matricea generatoare cu relatia:

$\mathbf{v} = \mathbf{iG}$ sau tinand cont de relatia

$\mathbf{Hv}^{\mathbf{T}} = 0$

$\mathbf{H(iG)}^{\mathbf{T}} = \mathbf{HG}^{\mathbf{T}}\mathbf{i}^{\mathbf{T}} = 0 \quad \forall i$

$\mathbf{HG}^{\mathbf{T}} = 0$ care este legatura intre cele doua matrici

Daca $\mathbf{H} = [\mathbf{I_m Q}]$ si $\mathbf{G} = [\mathbf{Q}^{\mathbf{T}} \mathbf{I_k}]$, relatia de legatura este satisfacuta:

$$\mathbf{HG}^{\mathbf{T}} = [\mathbf{I_m Q}]\begin{bmatrix}\mathbf{Q}\\ \mathbf{I_k}\end{bmatrix} = [\mathbf{Q} + \mathbf{Q}] = 0$$

Daca se noteaza $\mathbf{P} = \mathbf{Q}^{\mathbf{T}}$, $\mathbf{G} = [\mathbf{PI_k}]$ avem:

$\mathbf{v} = \mathbf{iG} = [\mathbf{PI_k}] = [\mathbf{iP\ i}] = [\mathbf{ci}]$ deci

$\mathbf{c} = \mathbf{iP}$ sau

$$\mathbf{c} = [i_{m+1} \ldots i_{m+k}]\begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1m} \\ p_{21} & p_{22} & \cdots & p_{2m} \\ . & . & . & . \\ p_{k1} & p_{k2} & \cdots & p_{km} \end{bmatrix} = [c_1 \ldots . c_m]$$

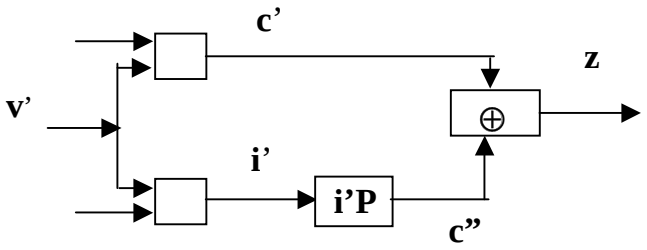
$$c_j = \sum_{i=1}^{k} i_{m+1} p_{ij}$$

- <u>Formarea corectorilor la codarea cu matricea **G**</u>

$\mathbf{z}^{\mathbf{T}} = \mathbf{v'H}^{\mathbf{T}} = [\mathbf{c'i'}][\mathbf{I_m}\ \mathbf{P}^{\mathbf{T}}]^{\mathbf{T}} = [\mathbf{c'} + \mathbf{i'P}]$

Cu notatia $\mathbf{c''} = \mathbf{i'P}$ (simboluri de control pe baza simbolurilor de informatie eronate) avem:

$\mathbf{z}^{\mathbf{T}} = \mathbf{c'} + \mathbf{c''}$



- Codul Hamming grup corector de o eroare

Acest cod este caracterizat de o matrice **H** in care coloana $\mathbf{h}_i$ este reprezentarea binara a numarului *i*:

$$\mathbf{H} = [\mathbf{h}_1\ \mathbf{h}_2 ... \mathbf{h}_n] = \begin{bmatrix} 0 & 0 & 0 & . & . & 1 \\ . & . & . & . & . & 1 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & . \\ 0 & 1 & 1 & . & 1 & . \\ 1 & 0 & 1 & . & 0 & 1 \end{bmatrix}$$

unde se remarca: $\mathbf{h}_i + \mathbf{h}_j \neq 0$ pentru orice $i \neq j$.

Cuvantul eroare, cu o singura eroare, este:

$\boldsymbol{\varepsilon} = [\ldots \alpha_i \ \ldots]$;

Cuvantul receptionat este:

$\mathbf{v'}_j = \mathbf{v}_j + \boldsymbol{\varepsilon}$;

Corectorul corespunzator este:

$\mathbf{z} = \mathbf{H}\mathbf{v'}_j^{\,T} = \mathbf{H}\boldsymbol{\varepsilon}^T$ sau:

$$\mathbf{z} = [\mathbf{h}_1\ \mathbf{h}_2\ \ldots \mathbf{h}_i\ \ldots \mathbf{h}_m] \begin{bmatrix} . \\ . \\ \alpha_i \\ . \\ . \end{bmatrix} = \mathbf{h}_i$$

adica corectorul este reprezentarea binara a numarului *i* care indica pozitia in care exista o eroare. In acest caz decodarea se reduce la o conversie de la binar la zecimal.

Codul Hamming care poate corecta toate erorile simple dar nu poate corecta nici o eroare dubla, se numeste *un cod perfect*.

- Codarea codului Hamming

Pentru simplificare, cele *m* pozitii ale simbolurilor de control se aleg sa corespunda coloanelor lui **H** cu o singura componenta diferita de zero:

pozitiile $2^0, 2^1, 2^2, \ldots, 2^{m-1}$ restul pozitiilor se repartizeaza simbolurilor de informatie (*codul nu este sistematic*). Daca se noteaza simbolurile de control cu c_i si cele de informatie cu i_j, cuvantul de cod se scrie:

$\mathbf{v} = [c_1\ c_2\ i_3\ c_4\ i_5\ \ldots\ i_n]$ iar c_i sunt date de relatia: $\mathbf{H}\mathbf{v}^T = 0$ sau de :

$$[\mathbf{h_1}\ \mathbf{h_2}\ \ldots\mathbf{h_n}]\begin{bmatrix} c_1 \\ c_2 \\ i_3 \\ \cdot \\ \cdot \\ i_n \end{bmatrix} = 0$$

sau:

$$c_1\begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 1 \end{bmatrix} + c_2\begin{bmatrix} 0 \\ \cdot \\ 1 \\ 0 \end{bmatrix} + i_3\begin{bmatrix} 0 \\ \cdot \\ 1 \\ 1 \end{bmatrix} + \ldots + i_n\begin{bmatrix} 1 \\ \cdot \\ \cdot \\ 1 \end{bmatrix} = 0$$ relatie echivalenta cu m

ecuatii cu m necunoscute cu care se determina c_i functie de i_j:

$$c_1 = i_3 + i_5 + \ldots + i_n$$
$$c_2 = i_3 + i_6 + \ldots + i_n$$
$$c_4 = i_5 + i_6 + \ldots + i_n$$
. . . . . .

- Decodarea codului Hamming

La receptie se calculeaza corectorul cu relatia:

$$\mathbf{z} = \mathbf{Hv'}^{T} = \begin{bmatrix} z_1 \\ \cdot \\ \cdot \\ z_m \end{bmatrix} = [\mathbf{h}_1 \ldots \mathbf{h}_n]\begin{bmatrix} c'_1 \\ c'_2 \\ \cdot \\ \cdot \\ i'_n \end{bmatrix}$$ sau, analog cazului precedent:

$$z_m = c'_1 + i'_3 + i'_5 + \ldots + i'_n$$
$$z_{m-1} = c'_2 + i'_3 + i'_6 + \ldots + i'_n$$
. . . . . .

numarul binar astfel calculat ($z_1\ z_2\ z_3 \ldots z_m$) este convertit binar – zecimal, obtinandu-se indicarea pozitiei erorii, permitand astfel corectia ei.

Pentru corectia unei singure erori, trebuie ca numarul corectorilor $2^m \geq n+1$ pentru a indica o eroare intr-unul din cele n simboluri ale cuvantului receptionat sau pentru a arata ca nu sunt erori. Deci relatia de mai sus sau relatia $2^m \geq k + m + 1$ determina numarul simbolurilor de control cand se da numarul k al simbolurilor de informatie, la corectia unei singure erori.

Exemplul 1: o sursa genereaza informatia, codata cu un cod Hamming corector de o eroare, sub forma de sir de mesaje independente dintr-un set de $N = 20$ mesaje. Se procedeaza astfel:

- $N = 20 \bullet 2^k$ deci $k = 5$ simboluri, relatia $2^m \geq k + m + 1$ este satisfacuta pentru $m = 4$
- Matricea **H** va avea dimensiunea m×n:

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

- Cuvantul de cod este:

$\mathbf{v} = [c_1\ c_2\ i_3\ c_4\ i_5\ i_6\ i_7\ c_8\ i_9]$

- Simbolurile de corectie sunt (din ecuatiile $\mathbf{Hv}^T = 0$):

$c_8 = i_9$
$c_4 = i_5 + i_6 + i_7$
$c_2 = i_3 + i_6 + i_7$
$c_1 = i_3 + i_5 + i_7 + i_9$

- Daca cuvantul eroare este:

$\boldsymbol{\varepsilon} = [0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$

cuvantul receptionat este:

$\mathbf{v}' = [c_1\ (c_2 + 1)\ i_3\ c_4\ i_5\ i_6\ i_7\ c_8\ i_9]$

corectorul este:

$z_4 = c_1 + i_3 + i_5 + i_7 + i_9 = 0$
$z_3 = (c_2 + 1) + i_3 + i_6 + i_7 = 1$
$z_2 = c_4 + i_5 + i_6 + i_7 = 0$
$z_1 = c_8 + i_9 = 0$

prin urmare trebuie sa se corecteaza pozitia 2.

$$\mathbf{z} = \mathbf{h_2} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Exemplul 2: se considera cazul $k = 4$ de unde rezulta $m = 3$ si $n = 7$. Matricea de control este:

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

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

$c_1 = i_3 + i_5 + i_7$

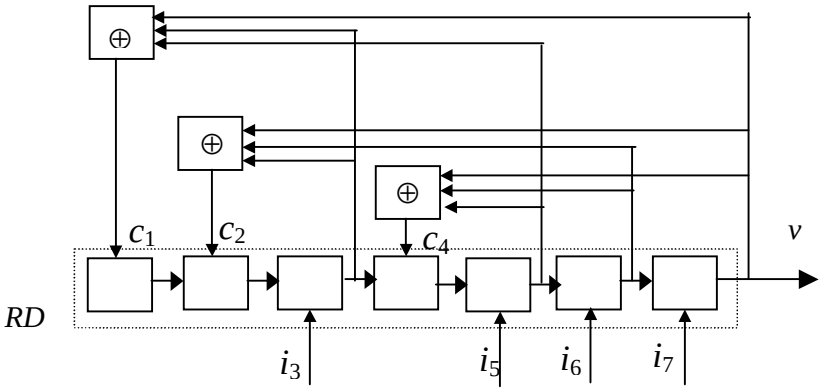
$c_2 = i_3 + i_6 + i_7$

$c_4 = i_5 + i_6 + i_7$

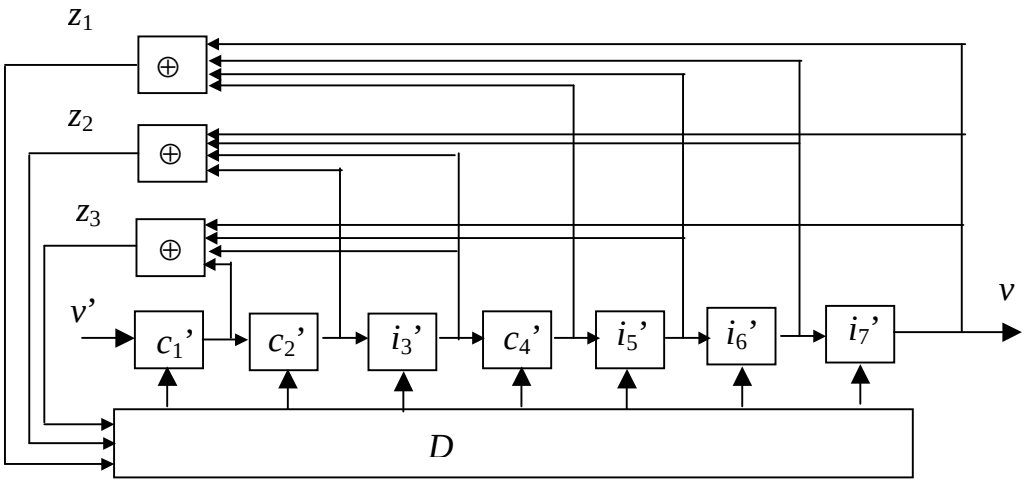
$z_1 = c_4' + i_5' + i_6' + i_7'$

$z_2 = c_2' + i_3' + i_6' + i_7'$

$z_3 = c_1' + i_3' + i_5' + i_7'$



In schema alaturata se da un codor pentru un cod Hamming corector de o eroare, conform cu exemplul 2: *RD* este registrul de deplasare in care se inscriu i_j si c_i calculate pe baza i_j, dupa care inscrierile se blocheaza si la iesire se emite **v** ca o succesiune i_7, i_6, i_5, c_4…, in ritmul de tact.



In schema de mai sus avem decodorul unde *D* este un convertor binar-zecimal care corecteaza automat eroarea la pozitia decodificata din (z_1,z_2,z_3).

IX Coduri ciclice

Sunt codurile bloc in care cele *n* simboluri din cuvant sunt considerate ca fiind coeficientii unui polinom de grad n –1, cuvantul fiind de forma:

$v(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{n-1}x^{n-1}$ cu $a_i \in \{0, 1\}$ sau

$\mathbf{v} = [a_0\ a_1 \ldots\ a_{n-1}]$

In cele ce urmeaza consideram ca multimea *tuturor* cuvintelor este generata[1] de un polinom *p(x)* de grad *n*, ales de forma :

$p(x) = x^n + 1$

iar multimea cuvintelor *cu sens* este generata de un polinom *g(x)*, numit si polinomul generator al codului, de grad *m* de forma:

$g(x) = g_0 + g_1x + \ldots + g_{m-1}x^{m-1} + x^m$

Se poate arata ca intre cele doua polinoame generatoare exista relatia:

$p(x) = g(x)\ h(x)$ unde *h(x)* este de forma:

$h(x) = h_0 + h_1x + \ldots + h_kx^k$

Proprietate[2]: La codul ciclic, daca **v** este un cuvant cu sens atunci si orice permutare ciclica a simbolurilor sale este un cuvant cu sens:

$[a_i\ a_{i+1} \ldots a_{n-1}\ a_0\ a_1 \ldots a_{i-1}]$

Obs.: Multimea cuvintelor generate de *p(x)* are 2^n elemente din care alegem 2^k carora le atribuim sens (acestea formeaza multimea cuvintelor cu sens generate de polinomul *g(x)* de grad *m*).

1. **<u>Codarea cuvintelor de cod ca elemente in multimea cuvintelor cu sens generata de *g(x)*</u>**

In acest caz *v(x)* trebuie sa fie un multiplu al lui *g(x):*

$v(x) = i(x)\ g(x)$ unde *i(x)* este polinomul simbolurilor de informatie:

$i(x) = a_m + a_{m-1}x + \ldots + a_{m+k-1}x^{k-1}$

- Pentru a obtine un cod sistematic se determina simbolurile de control, folosind polinomul generator *g(x)*, astfel:

1) se observa ca *v(x)* se poate scrie sub forma:

$v(x) = c(x) + x^m\ i(x)$ unde *c(x)* este polinomul simbolurilor de control:

$c(x) = a_0 + a_1x + \ldots + a_{m-1}x^{m-1}$

2) se imparte in ambele parti cu *g(x)*:

$\frac{v(x)}{g(x)} = \frac{c(x)}{g(x)} + \frac{x^m i(x)}{g(x)}$ iar ultimul termen se rescrie:

$\frac{x^m i(x)}{g(x)} = q(x) + \frac{r(x)}{g(x)}$ unde *r(x)* este restul impartirii, polinom de grad mai mic decat *m*. Ultima relatie se poate scrie si :

---

[1] vezi anexa A1- 1)

[2] vezi anexa A1- 2)

$\frac{r(x)}{g(x)} + \frac{x^m i(x)}{g(x)} = q(x)$ sau $r(x) + x^m i(x) = q(x)g(x)$ de unde:

$c(x) = r(x) = rest \frac{x^m i(x)}{g(x)}$

- alta metoda pentru determinarea simbolurilor de control se bazeaza pe introducerea matricii generatoare **G**. Se observa ca:

$v(x) = q(x)\, g(x)$ sau

$v(x) = q_0 g(x) + q_1 g(x) + \ldots + q_{k-1} x^{k-1} g(x)$ deci $v(x)$ se gaseste in liniile matricei generatoare **G,** avand k linii si n coloane:

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \ldots & g_m & 0 & \ldots & 0 \\ 0 & g_0 & \ldots & g_m & 0 & \ldots & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & \ldots & 0 & g_0 & g_1 & \ldots & g_m \end{bmatrix}$$

Unde coeficientii puterilor lui x absente au fost inlocuiti cu zero; cu matricea generatoare **G** se poate face codarea cu relatia:

$$\mathbf{v = i\, G}$$

cu care se obtin simbolurile de control rezultate cu metoda anterioara, cand s-a plecat de la relatia: $v(x) = q(x)\, g(x)$ daca $q(x) = i(x)$.

2. <u>Codarea cuvintelor de cod cu ajutorul polinomului de control *h(x)*</u>

$v(x) = q(x)\, g(x)$ sau, multiplicand cu $h(x)$:

$v(x)h(x) = q(x)\, g(x)h(x) = q(x)\, p(x)$

Deoarece $p(X) = g(X)\, h(X) = 0$ rezulta:

$h(X)v(X) = 0$;

se poate[3] arata ca relatia scalara de mai sus poate fi scrisa sub forma:

$\mathbf{Hv}^T = 0$, relatie cu care se poate face codarea.

unde:

$$\mathbf{H} = \begin{bmatrix} 0 & \ldots & & 0 & h_k & \ldots & & h_0 \\ 0 & \ldots & 0 & h_k & & \ldots & h_o & 0 \\ . & . & . & . & . & . & . & 0 \\ . & . & . & . & . & . & . & 0 \\ h_k & \ldots & & h_0 & 0 & \ldots & 0 & 0 \end{bmatrix}$$

$\mathbf{v} = [a_0\, a_1 \ldots a_{n-1}]$

De asemenea se poate arata ca:

$\mathbf{GH}^T = \mathbf{HG}^T$

[3] Vezi anexa A1- 3)

Exemplu: Se transmite un numar de 16 mesaje pe un canal cu perturbatii, utilzand un cod ciclic corector de o eroare.

Relatia $2^k \geq 16$ este satisfacuta de $k = 4$; relatia $2^m \geq m + k + 1$ de $m = 3$, $n = 7$. Gradul polinomului $g(x)$ va fi deci $m = 3$: $g(x) = x^3 + x^2 + 1$ iar gradul lui $p(x)$ va fi $n = 7$: $p(x) = x^7 + 1$.

$$h(x) = \frac{p(x)}{g(x)} = \frac{1+x^n}{g(x)} = \frac{1+x^7}{1+x^2+x^3} = x^4 + x^3 + x^2 + 1$$

deci matricile **G** (k×n) si **H** (m×n) vor fi:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

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

a) codarea cu **H** se face astfel:

$\mathbf{v} = [c_0\ c_1\ c_2\ i_3\ i_4\ i_5\ i_6]$

$\mathbf{H}\mathbf{v}^T = 0$

va da simbolurile de control functie de cele de informatie:

$c_2 + i_3 + i_4 + i_6 = 0 \quad c_2 = i_3 + i_4 + i_6$

$c_1 + c_2 + i_3 + i_5 = 0 \quad c_1 = i_4 + i_5 + i_6$

$c_0 + c_1 + c_2 + i_4 = 0 \quad c_0 = i_3 + i_4 + i_5$

Cele 16 mesaje emise de sursa sunt:

mesaj	i_3	i_4	i_5	i_6	c_0	c_1	c_2	i_3	i_4	i_5	i_6
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	0	0	1	0
3	0	0	1	1	1	0	1	0	0	1	1
…	..	..	..	..	..	..	..	..	..	..	..
15	1	1	1	1	1	1	1	1	1	1	1

b) codarea cu **G** se face astfel:

$$\mathbf{v = iG} = [i_3\ i_4\ i_5\ i_6] \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} =$$

$$= [i_3\ \ i_4\ \ i_3+i_5\ \ i_3+i_4+i_6\ \ i_4+i_5\ \ i_5+i_6\ \ i_6]$$

Cele 16 mesaje emise de sursa vor fi:

mesaj	i_3	i_4	i_5	i_6	i_3	i_4	i_3+i_5	$i_3+i_4+i_6$	i_4+i_5	i_5+i_6	i_6
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1	0	1	1
2	0	0	1	0	0	0	1	0	1	1	0
3	0	0	1	1	0	0	1	1	1	0	1
…	..	..	..	..	..	..	…	…	…	…	..
13	1	1	0	1	1	1	1	1	1	1	1
14	1	1	1	0	1	1	0	1	0	1	0
15	1	1	1	1	1	1	0	1	0	0	1

Obs.: spatiul cuvintelor cu sens este acelasi indiferent de modul de codare: cu **H** sau cu **G**, insa difera *corespondenta* intre secventa informationala de 4 bit ($i_3\ i_4\ i_5\ i_6$) si cuvintele cu sens de 7 bit. De ex. cuvantul [1 1 1 1 1 1 1] corespunde in primul caz secventei informationale (1 1 1 1) si in al doilea caz secventei (1 1 0 1), corespondente marcate in tabele cu linie ondulata.

3. <u>Decodarea codurilor ciclice</u>

a) se calculeaza pentru un cuvant receptionat:

$$z_i(x) = rest\frac{v'(x)}{g(x)}$$

Obs.: in acest caz s-a notat cu *i* faptul ca la transmisie a intervenit tipul de eroare (inca necunoscuta) ε_i.

b) se cauta intr-un tabel, construit in prealabil, corespondenta dintre corectorul calculat la pct. 1 $z_i(x)$ si cuvantul eroare respectiv $\varepsilon_i(x)$;

c) se calculeaza:

$v(x) = v'(x) + \varepsilon_i(x)$

Obs.: o metoda de stabilire a tabelului este sa se calculeze z_i, pe baza relatiei

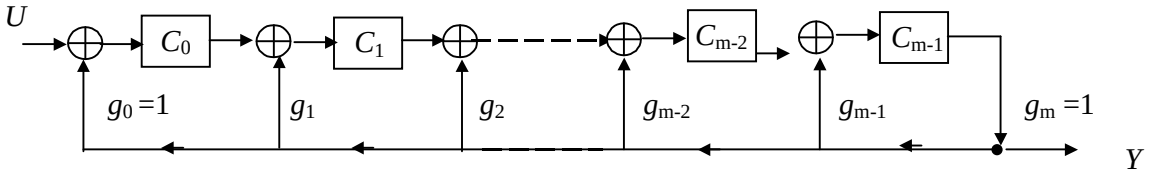
$$z_i = rest\,\frac{\varepsilon_i(x)}{g(x)}$$

pentru diversi ε_i. Se observa acum din nou ca exista un singur tip de corector z_i pentru toate cuvintele eronate cu ε_i. Tabelul poate fi stocat intr-o memorie ROM pentru calcul automat la receptie.

X Circuite de codare si decodare a codurilor ciclice

1. Circuite de divizare prin polinomul *g*(*x*), pentru codarea si decodarea codurilor ciclice.

Circuit de divizare a polinoamelor: in figura de mai jos se da schema circuitului de divizare a polinomului $u(x)$ prin $g(x)$, unde coeficientii restului $r(x)$ sunt stocati in celule iar catul este $y(x)$:



$u(x) = a_n x^n + \ldots + a_0$

$g(x) = g_m x^m + \ldots + g_0$

Se introduce operatorul de intarziere D care reprezinta intarzierea cu un tact introdusa de o celula binara; cu aceasta polinomul $u(x)$ se scrie ca secventa de date:

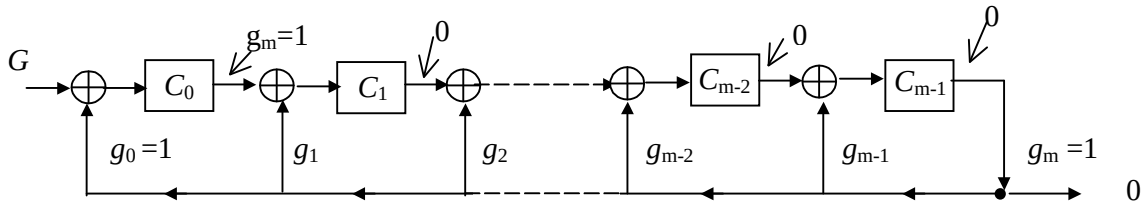
$U = a_n D^0 + \ldots + a_0 D^n$,

ceea ce inseamna ca coeficientii a_j se aplica la intrare in ordine descrescatoare; $D^0 = 1$ reprezinta operatorul de intarziere nula si arata ca initial a_n se gaseste la intrarea celulei C_0 iar termenul $a_0 D^n$ arata ca dupa n tacte a_0 ajunge la intrarea celulei C_0

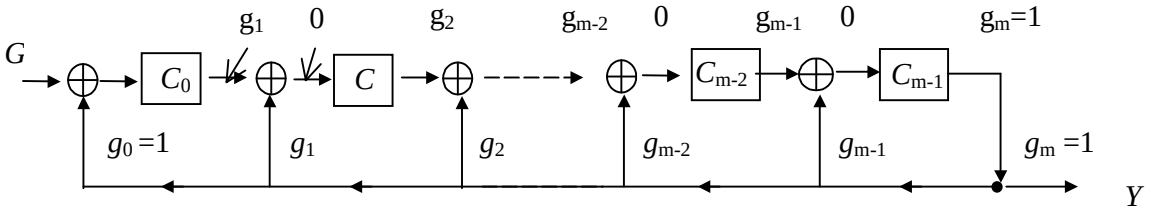
Functia de transfer a circuitului este :

$$T = \frac{Y}{U};$$

Pentru determinarea lui T, se ia un caz particular, presupunand ca la intrare se aplica chiar $g(x)$; dupa primul tact, pornind din starea initiala nula, circuitul se afla in urmatoarea situatie:



iar, dupa m tacte, circuitul va fi in urmatoarea situatie:



primul simbol aplicat la intrare g_m=1, ajungand la iesire deci in acest moment $Y = g_m D^m$ si la intrarea tuturor celulelor va fi simbolul 0; la tactul m+1 iesirile tuturor celulelor devin 0 deci $Y = g_m D^m \neq 0$ numai la tactul m, avand un singur termen. Deci in acest caz avem:

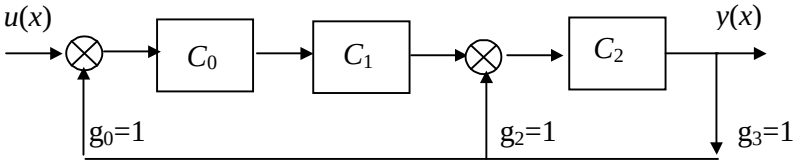
$$T = \frac{Y}{U} = \frac{g_m \mathbf{D}^m}{g_m \mathbf{D}^0 + g_{m-1}\mathbf{D}^1 + ... + g_0 \mathbf{D}^m} = \frac{1}{g_0 + ... + g_m \mathbf{D}^{-m}}$$

daca $D^{-1} \rightarrow x$, se poate afirma ca circuitul face o divizare cu $g(x)$.

In cazul general, cand polinomul de intrare este de grad n, catul divizarii se obtine la iesire dupa n tacte iar restul va apare stocat in registru la tactul n.

Exemplul 1: pentru n = 7, m = 3

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



a) Pentru:

$u(x) = x^7 + x^6 + x^5 + x^2$

$y(x) = x^4 + x^2$

$r(x) = 0$

Reguli de calcul pentru succesiunea starilor:

st C_0 = U + st C_2'; st C_1 = st C_0';

st C_2 = st C_1' + st C_2'; Y = st C_2'.

(st C_j' este starea precedenta a celulei C_j)

succesiunea starilor este data in tabelul urmator:

tact	IN (U)	C_0	C_1	C_2	OUT (Y)
	0	0	0	0	0
1	a_7= 1	1	0	0	0
2	a_6= 1	1	1	0	0
3	a_5= 1	1	1	1	0
4	a_4= 0	1	1	0	1 ($\leftarrow x^4$)
5	a_3= 0	0	1	1	0 ($\leftarrow x^3$)
6	a_2 = 1	0	0	0	1 ($\leftarrow x^2$)
7	a_1 = 0	0	0	0	0 ($\leftarrow x^1$)
8	a_0 = 0	0 ($\leftarrow x^0$)	0 ($\leftarrow x^1$)	0 ($\leftarrow x^2$)	0 ($\leftarrow x^0$)

b) Pentru

$u(x) = x^7 + x^6 + x^5 + x + 1$

$y(x) = x^4 + x^2$

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

succesiunea starilor este data in tabelul urmator:

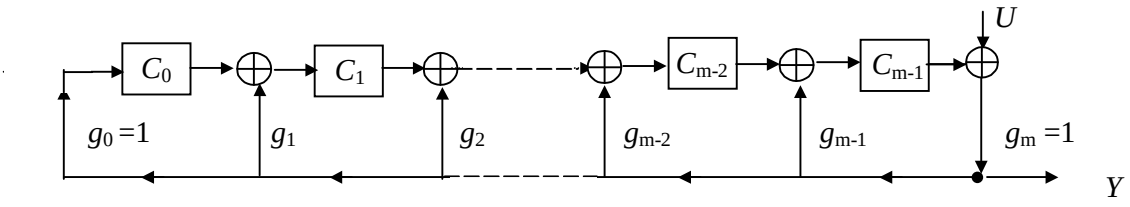
tact	IN (U)	C_0	C_1	C_2	OUT (Y)
	0	0	0	0	0
1	$a_7 = 1$	1	0	0	0
2	$a_6 = 1$	1	1	0	0
3	$a_5 = 1$	1	1	1	0
4	$a_4 = 0$	1	1	0	1 ($\leftarrow x^4$)
5	$a_3 = 0$	0	1	1	0 ($\leftarrow x^3$)
6	$a_2 = 0$	1	0	0	1 ($\leftarrow x^2$)
7	$a_1 = 1$	1	1	0	0 ($\leftarrow x^1$)
8	$a_0 = 1$	1 ($\leftarrow x^0$)	1 ($\leftarrow x^1$)	1 ($\leftarrow x^2$)	0 ($\leftarrow x^0$)

- Codor cu circuite de divizare

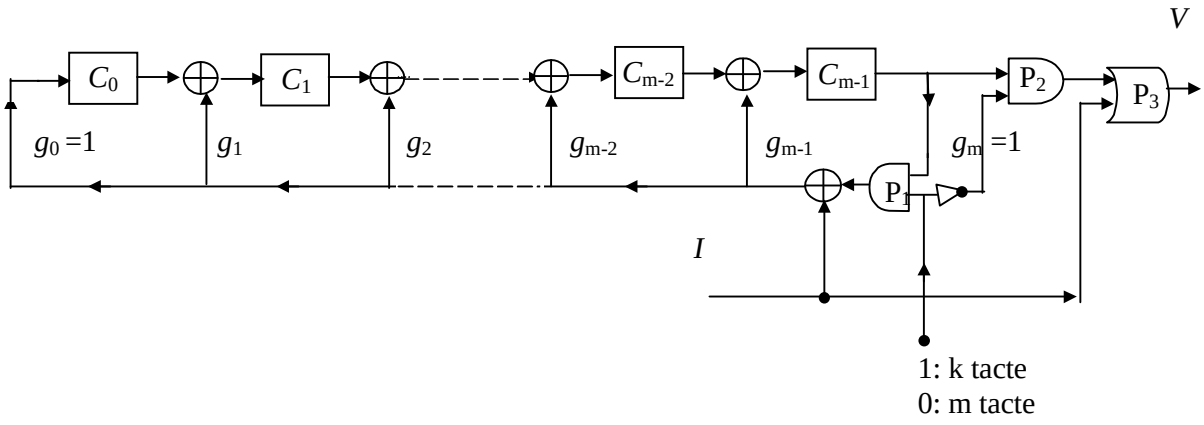
Relatia de codare pentru coduri ciclice este:

$$v(x) = rest\frac{x^m i(x)}{g(x)} + x^m i(x)$$

Se observa ca, daca $u(x) = i(x)$ se introduce ca in schema de divizare care urmeaza, aceasta divizeaza polinomul $x^m i(x)$ prin $g(x)$:

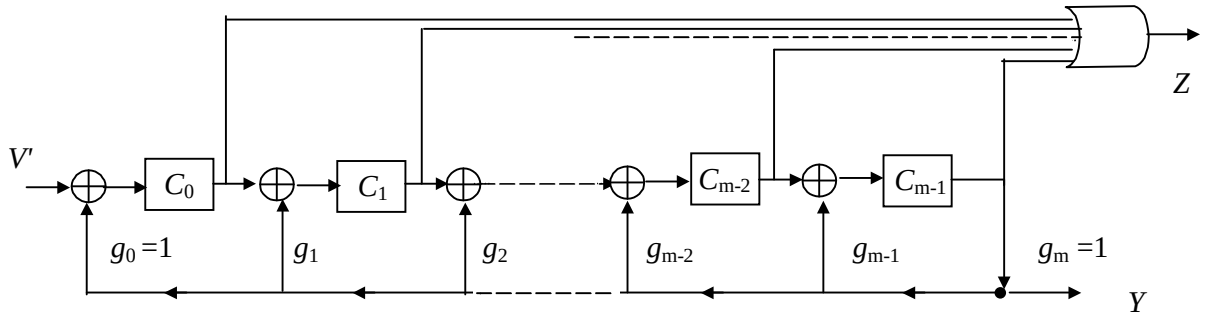


Circuitul de codare a codului ciclic este urmatorul:



Poarta P_1 este deschisa pe durata primelor k tacte, permitand pe de o parte transmisia spre iesire prin P_3 a celor *k* simboluri de informatie, pe de alta parte introducerea acestora in circuitul de divizare pentru calculul restului; restul, care constituie simbolurile de control, apare in registru la tactul *k*+1, cand P_2 este deschisa si P_1 este inchisa; P_2 ramane deschisa timp de *m* tacte pentru a permite evacuarea simbolurilor de control, care sunt plasate cu ajutorul lui P_3 in continuarea simbolurilor de informatie pentru completarea cuvantului de cod. Dupa evacuarea simbolurilor de control registrul revine in stare initiala nula, pregatit pentru codarea cuvantului urmator.

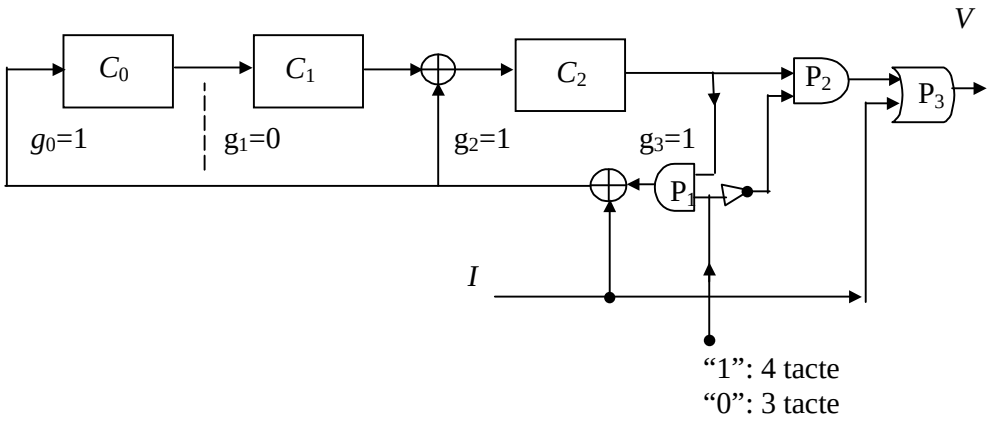
- Decodor cu circuite de divizare
  Decodarea presupune calculul corectorului $z(x) = rest(v'(x)/g(x))$ cu ajutorul urmatorului circuit:



Pentru detectia erorilor trebuie verificat daca $z(x)$ este sau nu nul: daca la iesirea portii SAU apare "1" atunci $z(x) \neq 0$, detectandu-se erori.

Exemplul 2: circuit codor pentru $g(x) = x^3 + x^2 + 1$

Regulile de functionare:

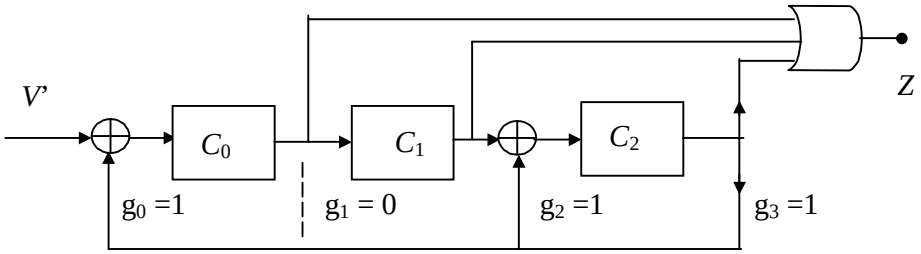


$C_2 = I + C_1'$ $+C_2'$;
$C_1 = C_0'$;
$C_0 = I + C_2$
$V = I$ pt. Tact 1..4
$V = C_2'$ pt. Tact 5..7

Succesiunea starilor este data in urmatorul tabel:

Tact	IN(I)	C_0	C_1	C_2	OUT(V)
	0	0	0	0	0
1	1	1	0	1	$1 = i_6$
2	0	1	1	1	$0 = i_5$
3	1	0	1	1	$1 = i_4$
4	0	1	0	0	$0 = i_3$
5	0	0	1	0	$0 = C_2$
6	0	0	0	1	$0 = C_1$
7	0	0	0	0	$1 = C_0$

Exemplul 3: Circuit decodor pentru $g(x) = x^3 + x^2 + 1$



Regulile de functionare:
$C_2 = C_1' + C_2'$
$C_1 = C_0'$
$C_0 = I + C_2'$

Succesiunea starilor este data in tabelele urmatoare, pentru doua cazuri:

a) se receptioneaza un cuvant de cod: $v'(x) = x^6 + x^4 + 1$;
$z(x) = r(x) = 0$ (vezi starea la tactul 7)

Tact	I	C_0	C_1	C_2
	0	0	0	0
1	1	1	0	1
2	0	1	1	1
3	1	0	1	1
4	0	1	0	0
5	0	0	1	0
6	0	0	0	1
7	1	0	0	0

b) se receptioneaza un cuvant eronat: $v'(x) = x^6 + 1$;
$z(x) = r(x) = x^2 + x + 1$ (vezi starea la tactul 7)

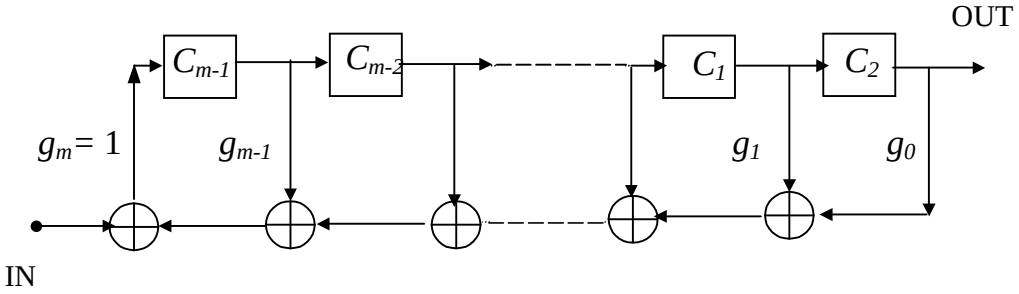
Tact	I	C_0	C_1	C_2
	0	0	0	0
1	1	1	0	0
2	0	0	1	0
3	0	0	0	1
4	0	1	0	1
5	0	1	1	1
6	0	1	1	0
7	1	1	1	1

2. Registre de deplasare cu reactie pe baza polinomului $g(x)$, pentru codarea si decodarea codurilor ciclice

Registre de deplasare, cu circuit de reactie lineara.

Fie un asemenea circuit de polinom caracteristic:

$$g(x) = g_0 + g_1 x + \ldots + g_{m-1} x^{m-1} + x^m$$



Starea registrului la momentul i este notata cu vectorul $\mathbf{S}(i)$:

$$\mathbf{S}(i) = \begin{bmatrix} s_0(i) \\ s_1(i) \\ \ldots \\ s_{m-1}(i) \end{bmatrix}$$

unde $s_j(i)$ = starea celulei C_j la momentul i este variabila de stare a circuitului.

Pentru regimul liber (intrare nula):

$$\mathbf{S}(i+1) = \mathbf{T}\mathbf{S}(i)$$

unde T este matricea de tranzitie a circuitului (matricea caracteristica):

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & 1 \\ g_0 & g_1 & g_2 & \ldots & g_{m-1} \end{bmatrix}$$

Se vede ca daca $\mathbf{S}(0) = 0$, $\mathbf{S}(1) = \mathbf{S}(2) = \ldots = 0$. Daca $\mathbf{S}(0) \neq 0$, dupa un numar finit de tranzitii (impulsuri de tact) circuitul revine in starea initiala, fiind un sistem determinist.

Conditia necesara si suficienta ca registrul cu reactie lineara conform $g(x)$ sa parcurga in regim liber toate starile nenule distincte este ca $g(x)$ sa fie primitiv (cel mai mic intreg pozitiv n pentru care $g(x)$ divide pe $x^n + 1$ este $n = 2^m - 1$).

Pentru regimul fortat (la momentele $t = 1, 2, \ldots, n$ se aplica la intrare corespunzator simbolurile succesive $a_{n-1}, \ldots, a_0$. Cu notatia:

$\mathbf{U}^T = [0\ 0\ \mathbf{...}\ 0\ 1]$

vom avea succesiv, corespunzator tactului:

$\mathbf{S(1)} = \mathbf{S(0)} + a_{n-1}\mathbf{U}$

$\mathbf{S(2)} = \mathbf{TS(1)} + a_{n-2}\mathbf{U} = \mathbf{T^2S(0)} + \mathbf{T^{-1}U} + a_{n-2}\mathbf{U}$

.....

$\mathbf{S(n)} = \mathbf{TS(n-1)} + a_{\mathbf{0}}\mathbf{U} = \mathbf{T^nS(0)} + a_{n-1}\mathbf{T^{n-1}U} + \mathbf{...} + a_{\mathbf{1}}\mathbf{T^1U} + a_{\mathbf{0}}\mathbf{T^0U}$

Daca se noteaza cuvantul **v** cu:

$\mathbf{v} = [a_0\ a_1\ \mathbf{...}\ a_{n-1}]$

si matricea H cu:

$\mathbf{H} = [\mathbf{T^0U}\quad \mathbf{T^1U}\quad \mathbf{T^2U}\quad \mathbf{...}\quad \mathbf{T}^{n-1}\mathbf{U}]$

atunci se obtine:

$\mathbf{S}(n) = \mathbf{T}^n\mathbf{S}(0) + \mathbf{Hv}^T$

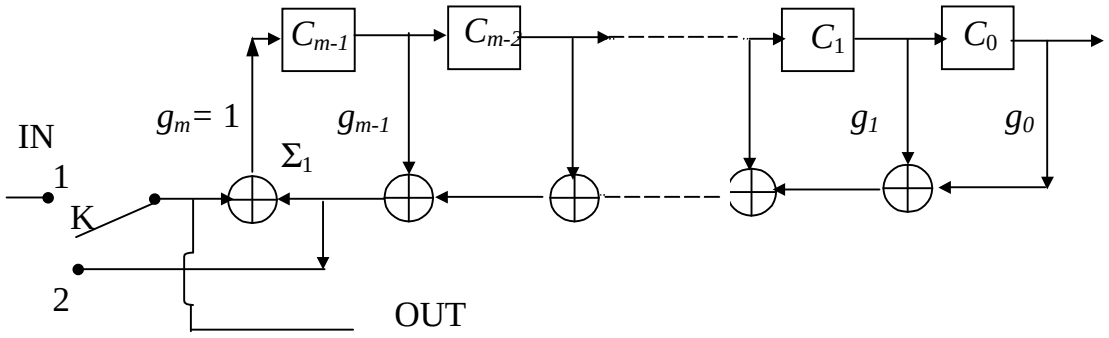
daca **S**(0) = 0, atunci starea la momentul *n* este:

$\mathbf{S}(n) = \mathbf{Hv}^T$

ceea ce reprezinta expresia corectorului: un asemenea circuit poate servi la codare sau la detectia / corectia erorilor.

- Codor cu registru de deplasare cu reactie

Cu registrul de mai sus se poate face un codor ciclic ca in figura de mai jos:



In primele k tacte, comutatorul K este pe pozitia 1 si se introduc succesiv cele k simboluri de informatie, a_{n-1}, a_{n-2}, …,a_{n-k}, acestea fiind transmise si la iesire. Se trece K pe pozitia 2 si circuitul genereaza in urmatoarele m tacte simbolurile de control, rezultand cuvintele de cod ciclic:

Daca **S**(0) = 0, pe pozitia 1, in primele k tacte avem succesiv starile :

$$\mathbf{S}(1) = \mathbf{T}\mathbf{S}(0) + a_{n-1}\mathbf{U}$$

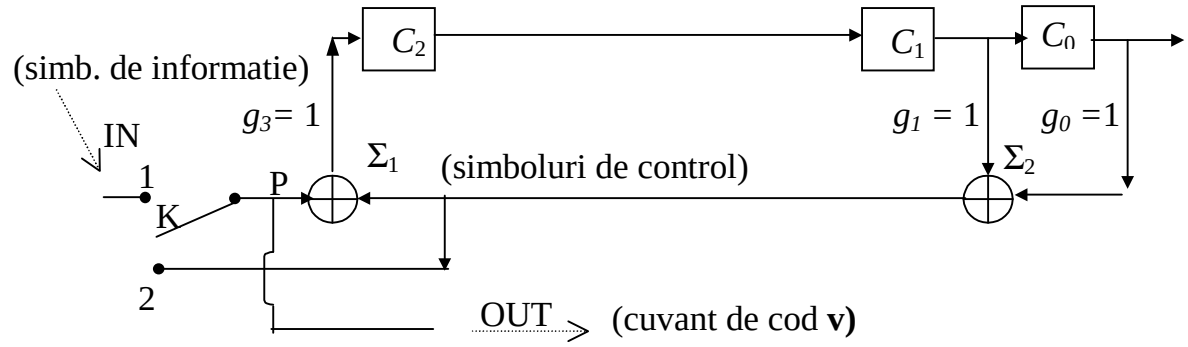
$$\mathbf{S}(2) = \mathbf{T}\mathbf{S}(1) + a_{n-2}\mathbf{U} = \mathbf{T}^2\mathbf{S}(0) + \mathbf{T}^{-1}\mathbf{U} + a_{n-2}\mathbf{U}$$

.....

$$\mathbf{S}(k) = \mathbf{T}\mathbf{S}(k-1) + a_{n-k}\mathbf{U} = \mathbf{T}^k\mathbf{S}(0) + a_{n-1}\mathbf{T}^{k-1}\mathbf{U} + \mathbf{...} + a_{n-k}\mathbf{T}^0\mathbf{U}$$

Daca se trece acum K pe pozitia 2, la urmatorul impuls de tact, ambele intrari ale sumatorului Σ_1 primesc acelasi simbol deci in celula C_{m-1} se introduce un 0 si starea ei la momentul k+1 va fi 0 deci $\mathbf{S}(k+1) = \mathbf{S}(k+2) = \ldots = \mathbf{S}(n) = 0$ deci (vezi mai sus): $\mathbf{H}\mathbf{v}^T = 0$, adica simbolurile generate constituie un cuvant de cod **v**. Deoarece $\mathbf{S}(n) = 0$, codorul este pregatit pentru generarea unui nou cuvant de cod, aceasta devenind starea initiala.

Exemplul 1: Pentru n = 7, k = 4, m = 3 si $g(x) = 1 + x + x^3$, schema codorului, cu sumatoare modulo 2 externe, se da mai jos:



In conformitate cu cazul general descris anterior codarea decurge astfel:

1. K pe pozitia 1: Prin IN, se introduce in registru, si simultan in canal (OUT), secventa informationala ($i_3\ i_2\ i_1\ i_0$);
2. K pe pozitia 2: Primul simbol de control c_2 se formeaza in punctul P, fiind produs la iesirea sumatorului Σ_2, si este livrat la iesire spre canal; in acelasi timp el se aduna cu el insusi in Σ_1, celula C_2 incarcandu-se cu 0;
3. Se repeta pasul anterior, obtinindu-se simbolurile de control c_1 si c_0; cele trei simboluri de control se alatura celor 4 simboluri din secventa informationala, pe canal transmitandu-se cuvantul de cod complet; registrul se reinitializeaza si este pregatit pentru un nou mesaj.

Daca i = (1001):

$i(x)=1+x^3$;

$c(x)=rest\dfrac{x^3 i(x)}{g(x)}=x+x^2$;

$v(x)=x+x^2+x^3+x^6$;

$\mathbf{v}=[0111001]$

In tabelul urmator se prezinta evolutia starilor registrului de deplasare cu reactie din exemplul 1:

K	tact	IN	C_2	C_1	C_0	OUT
			0	0	0	
1	1	i_3= 1	1	0	0	$i_3 = a_6 = 1$
1	2	i_2= 0	0	1	0	$i_2 = a_5 = 0$
1	3	i_1= 0	1	0	1	$i_1 = a_4 = 0$
1	4	i_0= 1	0	1	0	$i_0 = a_3 = 1$
2	5	*	0	0	1	$c_2 = a_2 = 1(1+0)$
2	6	*	0	0	0	$c_1 = a_1 = 1(0+1)$
2	7	*	0	0	0	$c_0 = a_0 = 0(0+0)$

$\neq$ rest

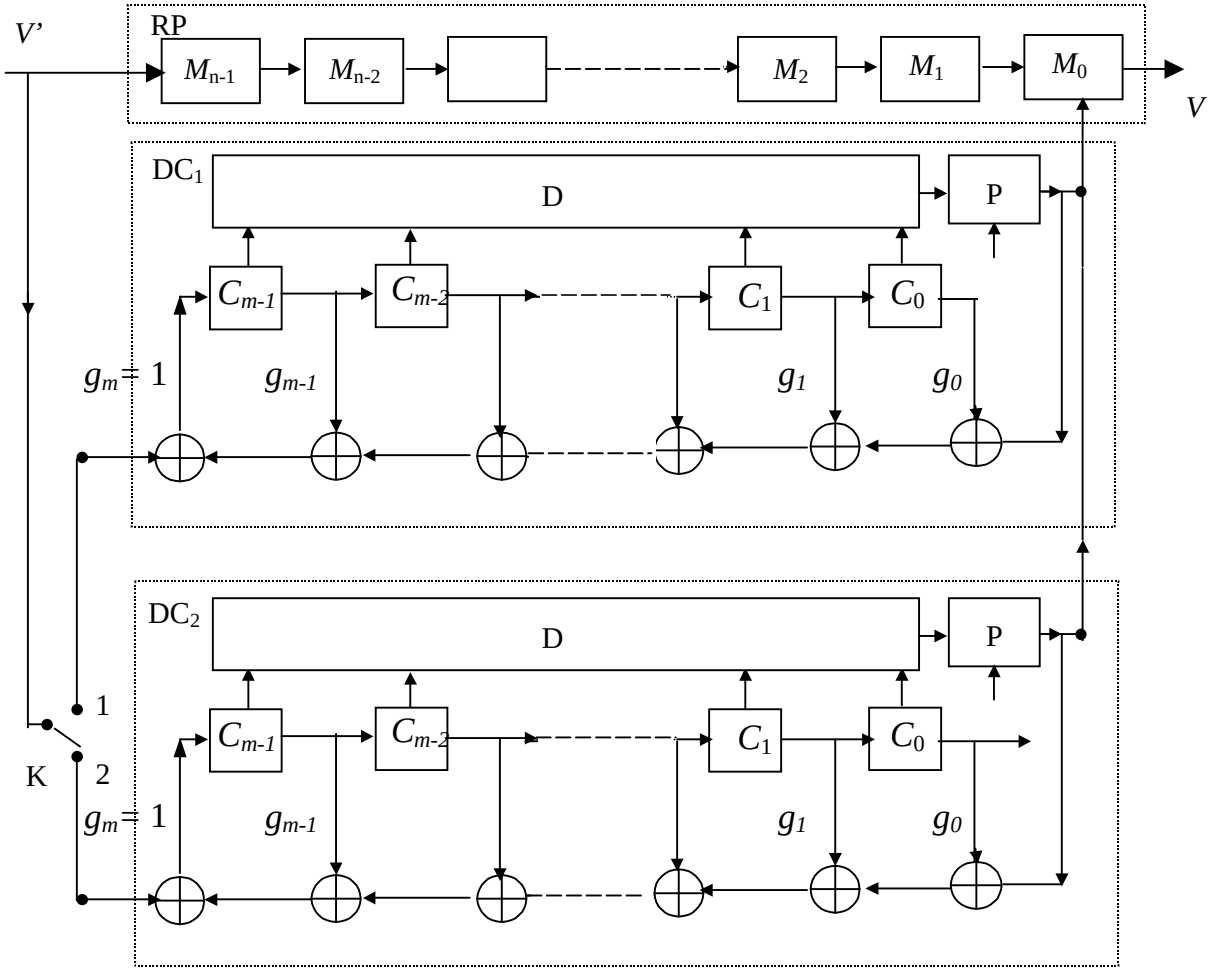
$\mathbf{v} = [a_0\ a_1 \ldots a_6] = [c_0\ c_1\ c_2\ \ i_0\ i_1\ i_2\ i_3] = [011\ 1001]$

Obs.: dupa introducerea secventei informationale (dupa tactul 4) continutul celulelor (0 1 0) nu formeaza simbolurile de control, reprezentand o forma modificata a restului; este necesara prelucrarea in continuare in Σ_1.

- Decodor cu registru de deplasare cu reactie

Sistemul de decodare contine un registru de deplasare principal RP si doua decodoare DC_1 si DC_2. In RP este inmagazinat cuvantul receptionat de lungime *n*. Celulele registrelor din decodoare sunt cuplate la detectoarele de erori D. Aceste detectoare emit un simbol"1" cand simbolul eronat este in M_0, permitand corectia erorii. Acelasi "1" se aplica pentru a reseta registrul.

Simbolurile cuvantului receptionat sunt introduse simultan in RP pentru memorare si in DC_1 (K pe pozitia 1) pentru calculul corectorului, in acest timp D ramane deconectat (P este blocata). Cand ultimul simbol al *V'* a intrat in RP, poarta P se deschide pentru efectuarea corectiei; comutatorul K trece in pozitia 2 si cuvantul urmator se introduce in DC_2 si in RP, DC_1 si DC_2 lucrand in contratimp, adica in timp ce unul determina corectorul, celalalt prelucreaza corectorul pentru realizarea corectiei.



Starea registrului DC_1 in acest moment este chiar corectorul **z.**

Calculul corectorului **z** se face cu relatia:

$\mathbf{z} = a'_0\mathbf{U} + a'_1\mathbf{TU} + \ldots + a'_{n-2}\mathbf{T}^{n-2}\mathbf{U} + a'_{n-1}\mathbf{T}^{n-1}\mathbf{U} = \mathbf{Hv'}^{\boldsymbol{T}}$ unde

$\mathbf{H} = [\mathbf{U}\ \mathbf{TU}\ \mathbf{T}^2\mathbf{U} \ldots \mathbf{T}^{n-1}\mathbf{U}]$

Daca apare o singura eroare in pozitia *n-r*, cuvantul eroare $\boldsymbol{\varepsilon}$ este de forma:

$\boldsymbol{\varepsilon} = [\ldots, \alpha_{n-r}, \ldots]$

iar corectorul va fi dat de relatia:

$\mathbf{z} = \mathbf{H}\boldsymbol{\varepsilon}^{\boldsymbol{T}} = \mathbf{T}^{n-r}\mathbf{U} = \mathbf{T}^{-r}\mathbf{U}$ (deoarece $\mathbf{T}^n = \mathbf{I}$).

Din acest moment DC_1 evolueaza liber, succesiunea de stari fiind:

$\mathbf{z}, \mathbf{Tz}, \mathbf{T}^2\mathbf{z}, \ldots, \mathbf{T}^{r-1}\mathbf{z},$

prin celula M_0 a RP succedandu-se simbolurile:

$a'_{n-1}, a'_{n-2}, \ldots, a'_{n-r}$.

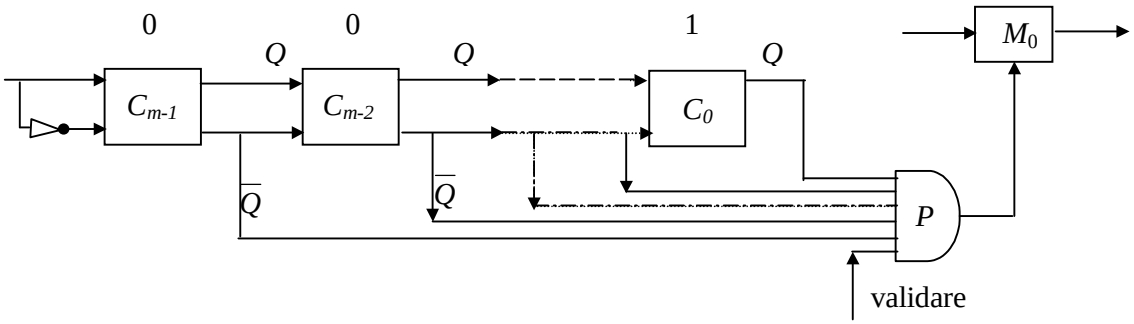
Dupa *r-1* perioade de tact, simbolul eronat a'_{n-r} ajunge in M_0, in timp ce starea DC_1 devine:

$\mathbf{S}_{r-1} = \mathbf{T}^{r-1}\mathbf{z} = \mathbf{T}^{r-1}\mathbf{T}^{-r}\mathbf{U} = \mathbf{T}^{-1}\mathbf{U} = [1\ 0\ \ldots\ldots 0]^T$

Detectorul D recunoaste aceasta stare fixa a DC_1, care apare la un moment ce depinde de pozitia eronata si furnizeaza un impuls de

complementare a starii celulei M_0, asa ca la iesirea RP apare cuvantul corectat, **v**. Impulsul de corectie este folosit si la resetarea DC_1, pregatit astfel pentru un nou cuvant de cod. Ciclul de functionare pentru DC_1 este de $2n$ perioade de tact.

Un exemplu de circuit care detecteaza starea $\mathbf{T}^{-1}\ \mathbf{U}$ se da mai jos: acesta furnizeaza un semnal "1" numai cand starea registrului de decalaj este (0, 0, …, 0, 1); acest "1" complementeaza continutul celulei M_0, corectand eroarea. Poarta P este inchisa cat timp se calculeaza corectorul, altfel la trecerea regisrului prin starea (0 0 … 1) se da o comanda falsa de corectie.



Imediat dupa corectie, la tactul r, starea registrului va fi:

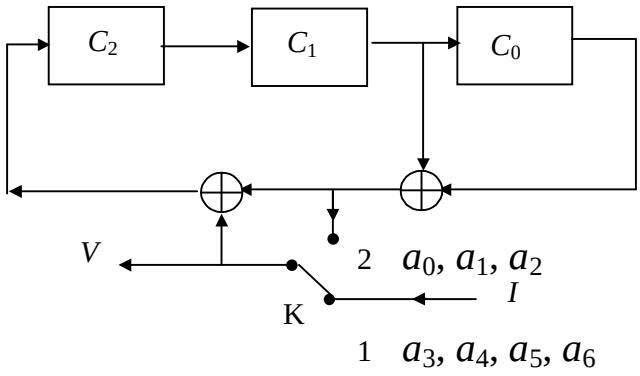
$$\mathbf{T}\,\mathbf{S}_{r-1} + \mathbf{U} = \mathbf{T}\mathbf{T}^{-1}\,\mathbf{U} + \mathbf{U} = \mathbf{U} + \mathbf{U} = 0$$

si, incepand din acest moment, starea nu se va mai modfica pana la sosirea primului simbol al cuvantului urmator care trebuie corectat.

Exemplu: Fie un cod ciclic cu: n = 7, k = 4, m = 3, cuvintele de cod fiind de forma:

$v(x) = a_0 + a_1 x + \ldots + a_6 x^6$ unde a_0, a_1, a_2 sunt simboluri de control iar a_3, a_4, a_5, a_6 sunt simboluri de informatie. Polinomul generator $p(x)$ este: $p(x) = x^7 + 1$ iar polinomul generator $g(x)$ este $g(x) = x^3 + x + 1$

a) Codarea



Pentru determinarea simbolurilor de control se foloseste relatia:

$$c(x) = rest\frac{x^m i(x)}{g(x)} = \frac{x^3(a_3 + a_4 x + a_5 x^2 + a^6 x^3)}{x^3 + x + 1} =$$

$$= (a_3 + a_5 + a_6) + (a_3 + a_4 + a_5)x + (a_4 + a_5 + a_6)x^2$$

Daca se egaleaza coeficientii cu cei ai expresiei:

$c(x) = a_0 + a_1 x + a_2 x^2$

rezulta:

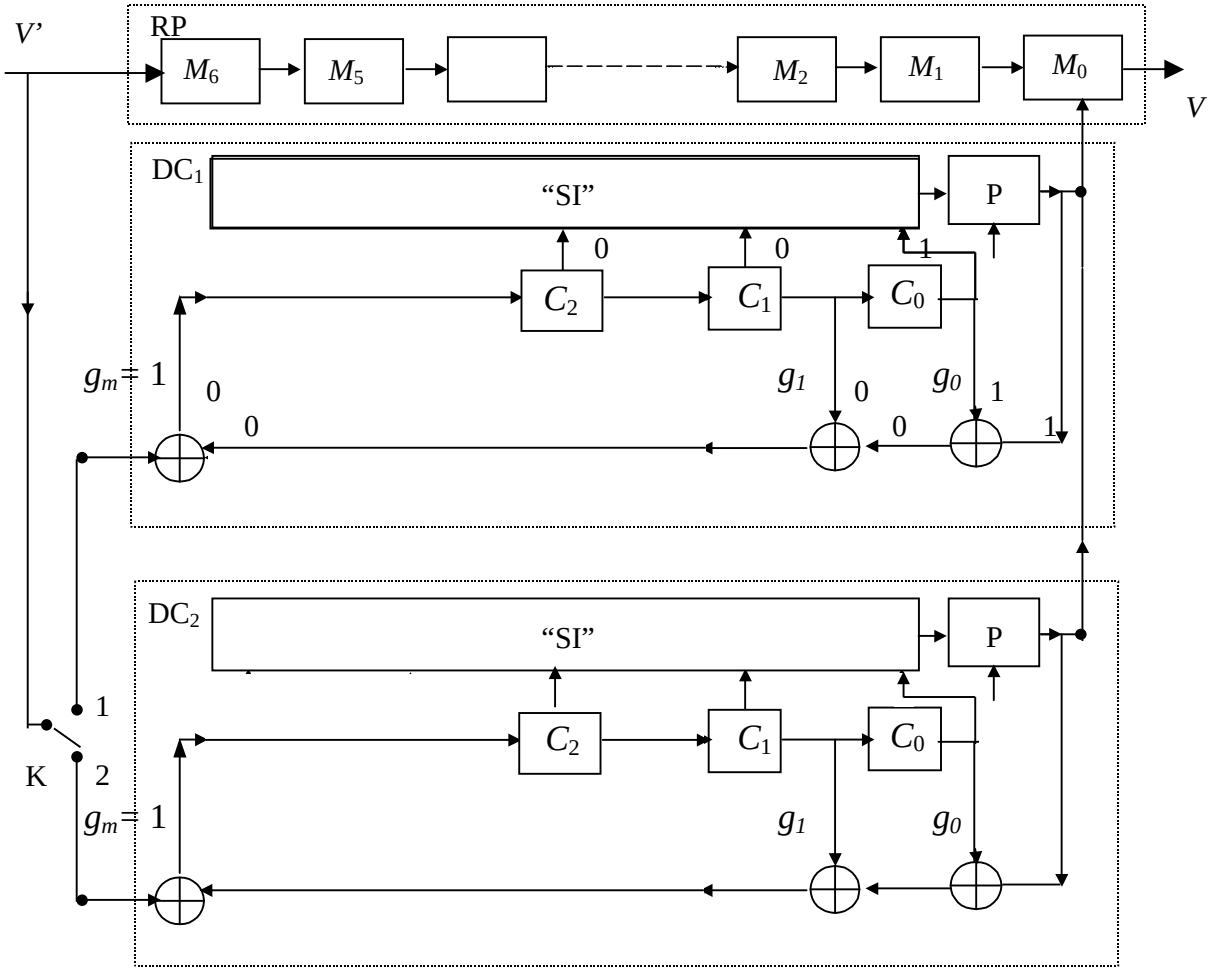
$a_2 = a_4 + a_5 + a_6$

$a_1 = a_3 + a_4 + a_5$

$a_0 = a_3 + a_5 + a_6$

b) Decodarea

Se face cu circuitul din figura de mai jos (particularizat din circuitul general pentru n = 7, k = 4, m = 3):



Daca a_4' = a_4 + 1 este simbolul eronat, $n - r = 4$, si corectorul, corespunzator starii registrului de deplasare dupa introducerea tuturor simbolurilor cuvantului receptionat, va fi:

$$\mathbf{z} = \mathbf{T}^4 \mathbf{U} = \mathbf{T}^{-3} \mathbf{U},$$

cand simbolul a_4' ajunge in celula M_0, dupa doua tacte suplimentare, starea registrului de deplasare cu reactie este:

$$\mathbf{T}^2 \mathbf{T}^{-3} \mathbf{U} = \mathbf{T}^{-1} \mathbf{U}$$

adica (0 0 1); in acest moment in celula M_0 se aplica un simbol "1" dat de decodor astfel incat vom avea:

$$a_4' + 1 = a_4$$

realizandu-se corectia; la tactul urmator, registrul de deplasare cu reactie este adus in starea 0.

XI. Coduri convolutionale

Sunt caracterizate prin faptul ca, spre deosebire de codurile bloc, simbolurile de informatie generate de sursa sunt prelucrate continuu; aici "blocurile" de informatie sau control nu au semnificatia de cuvinte de cod: simbolurile de control dintr-un bloc controleaza simbolurile de informatie si din alte blocuri.

Constrangere (ordinul de memorie): se defineste prin numarul blocurilor precedente in care se gasesc simbolurile de informatie verificate prin simbolurile de control ale blocului considerat (si se noteaza cu M);

Lungime de constrangere: se defineste prin numarul total de simboluri informationale i care concura la calculul unui simbol de control (sau unui simbol din cuvantul de cod) si se noteaza cu $n_C = k\,(M + 1)$; numarul simbolurilor i memorate este kM;

Rata de codare (raportul dintre numarul de simboluri informationale si lungimea unui bloc codat): $R = k\,L\,/\,n(L+M)$, unde L este numarul de blocuri considerate, din secventa de lungime finita codata.

Distanta de cod de ordinul N (minimul distantei Hamming, calculata insa pe N blocuri): $D_N = \min D_H\,(\mathbf{v}_{iN}, \mathbf{v}_{jN})$

Obs.:

a) Daca $L >> M$, $R = k / n$ ca la codurile bloc;
b) Daca nu se lucreaza cu L mari, $R < k / n$ si rata va fi mai mica decat la codurile bloc, reflectand o redundanta relativ ridicata, necesara insa in aplicatiile cu transmiterea datelor prin canale perturbate (canal radio);

– Coduri convolutionale* sistematice: Se considera, in succesiunea continua de simboluri, un bloc de lungime n, constituit din k simboluri de informatie si m simboluri de control;
– Coduri convolutionale nesistematice: Se considera, in succesiunea continua de simboluri, un bloc de lungime n, in care simbolurile de informatie si cele de control nu sunt distincte, ambele se noteaza cu u_i si sunt simboluri ale cuvantului de cod; aici:

Constrangere (M): numarul blocurilor de n simboluri, care sunt legate intre ele prin faptul ca un simbol de informatie oarecare, aparut la

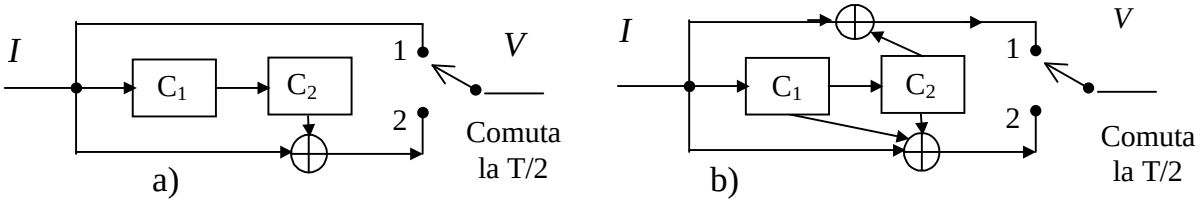
* Denumirea provine de la faptul ca simbolurile de control se obtin prin convolutia numerica dintre secventa informationala $i(x)$ si polinoamele generatoare $g_j(x)$

intrare, este luat in consideratie la calculul a cel putin unul din cele *n* simboluri u_i ale fiecaruia din cele *M* blocuri.

1. Codarea

Exemplu: pentru un codor cu parametrii $M = 2$, $k = 1$, $n = 2$ rezulta $M\,k = 2$, $n_C = 3$, $R = 1 / 2$.

In schemele de mai jos se reprezinta codorul sistematic (a) si cel nesistematic (b), corespunzatoare exemplului de mai sus:



La codurile convolutionale, fiecare din cele *m* simboluri de control (la coduri sistematice) respectiv fiecare din cele *n* simboluri (la coduri nesistematice), se obtin din n_C simboluri informationale prin inmultirea secventei informationale cu polinoamele generatoare corespunzatoare. Numarul polinoamelor generatoare necesare codarii este: $k \times m$, pentru sistematice si $n \times m$ la nesistematice. Din acestea, cel putin unul trebuie sa aiba gradul $n_C - 1$, deoarece la determinarea unui simbol de control *c* sau *u* trebuie sa participe n_C simboluri informationale.

Pentru sistematic:

$$c^{(j)}(x) = i(x)g^{(j)}(x) \qquad sau \quad c^{(j)} = i * g^{(j)}$$

iar pentru nesistematic:

$$u^{(j)}(x) = i(x)g^{(j)}(x) \qquad sau \quad u^{(j)} = i * g^{(j)}$$

unde *j* este pozitia comutatorului din figura de mai sus.

Astfel pentru exemplul de mai sus, $n_C - 1 = 2$ si

a) la sistematice (mai simple),

avem $k \times m = 1$ deci un singur polinom generator de grad 2:

$g(x) = 1 + x^2$ sau $1 + x + x^2$

Daca secventa informationala este $I = (0\ 1\ 1\ 0\ 1\ 0\ 0)$ sau $i(x) = x + x^2 + x^4$

$$c(x) = i(x)g(x) = (x + x^2 + x^4)(1 + x^2) = x + x^2 + x^3 + x^6$$

$$sau \quad C = (0111001)$$

Secventa codata este:

$V = (|\ i_0c_0\ |\ \ i_1c_1|\ \ \ldots) = (00\ \ 11\ \ 11\ \ 01\ \ 10\ \ 00\ \ 01)$

Simbolurile de control se determina cu relatia (termenul de ordinul 1 al convolutiei, avand un singur sumator):

$$c_p = \sum_{i=0}^{m} i_{p-i}\, g_i = i_p g_0 + i_{p-1} g_1 + ... + i_{p-m} g_m \quad unde\ i_{p-i} = 0 \quad \forall p < i$$

$$c_0 = i_0 g_0 = 0$$

$$c_1 = i_1 g_0 + i_0 g_1 = 1$$

$$c_2 = i_2 g_0 + i_1 g_1 + i_0 g_2 = 1$$

$$c_3 = i_3 g_0 + i_2 g_1 + i_1 g_2 = 1$$

$$c_4 = i_4 g_0 + i_3 g_1 + i_2 g_2 = 0$$

$$c_5 = i_5 g_0 + i_4 g_1 + i_3 g_2 = 0$$

$$c_6 = i_6 g_0 + i_5 g_1 + i_4 g_2 = 1$$

iar evolutia starilor se da in tabelul de mai jos:

t_n		t_{n+1}		t_n	
T	i	C_1	C_2	V	
				1 (i)	2 (c)
1	0	0	0	0	0
2	1	1	0	1	1
3	1	1	1	1	1
4	0	0	1	0	1
5	1	1	0	1	0
6	0	0	1	0	0
7	0	0	0	0	1

Obs.: la codurile sistematice cele *k* simboluri informationale se regasesc nemodificate in cele *n* simboluri ale cuvantului de cod: prin comutatorul de iesire se transmit direct la iesire aceste simboluri (vezi poz. 1 la schema din exemplu)

b) la nesistematice (intalnite in majoritatea aplicatiile recente datorita faptului ca la aceeasi rata de codare se pot corecta mai multe erori),

avem $n \times m = 2$ polinoame generatoare, dintre care cel putin unul de grad 2:

$g^{(1)}(x) = 1 + x^2$ si $g^{(2)}(x) = 1 + x + x^2$ sau $1+x$

daca se aleg in mod corespunzator** $g^{(j)}(x)$, pentru nesistematic codarea este:

$$u^{(1)}(x) = i(x)g^{(1)}(x) = (x + x^2 + x^4)(1 + x^2) = x + x^2 + x^3 + x^6$$

$$u^{(2)}(x) = i(x)g^{(2)}(x) = (x + x^2 + x^4)(1 + x + x^2) = x + x^5 + x^6$$

** se alege combinatia de $g^{(1)}(x)$ si $g^{(2)}(x)$ asa incat sa nu aiba factor comun modulo 2, in caz contrar obtinandu-se *erori catastrofice* (un numar finit de erori la transmisie produc un numar infinit de erori la decodare)

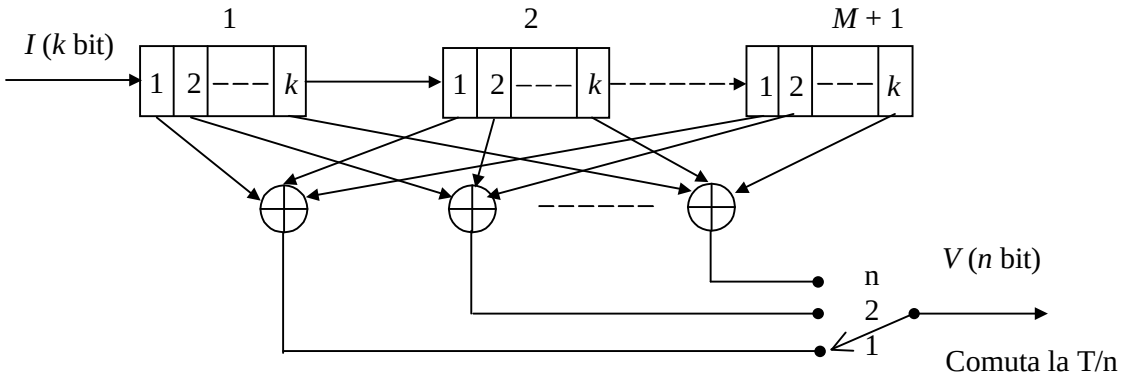
iar secventa de cod va fi:

$V = (|\, u_0^{(1)}\, u_0^{(2)} \,|\;\; u_1^{(1)}\, u_1^{(2)}|\;\; \ldots) = (00\;\; 11\;\; 10\;\; 10\;\; 00\;\; 01\;\; 11)$

Evolutia starilor se da in tabelul care urmeaza:

t_n		t_{n+1}		t_n	
T	i	C_1	C_2	V	
				1 ($u^{(1)}$)	2 ($u^{(2)}$)
1	0	0	0	0	0
2	1	1	0	1	1
3	1	1	1	1	0
4	0	0	1	1	0
5	1	1	0	0	0
6	0	0	1	0	1
7	0	0	0	1	1

In schema de mai jos se da un codor convolutional nesistematic mai general, care se utilizeaza in aplicatiile mai recente, in special la transmisiile de date in banda radio:



Obs.: se observa si aici ca in total sunt luate in consideratie $n_C = k\,(M + 1)$ simboluri informationale (actuale si memorate) pentru calculul a n simboluri ale cuvantului de cod.

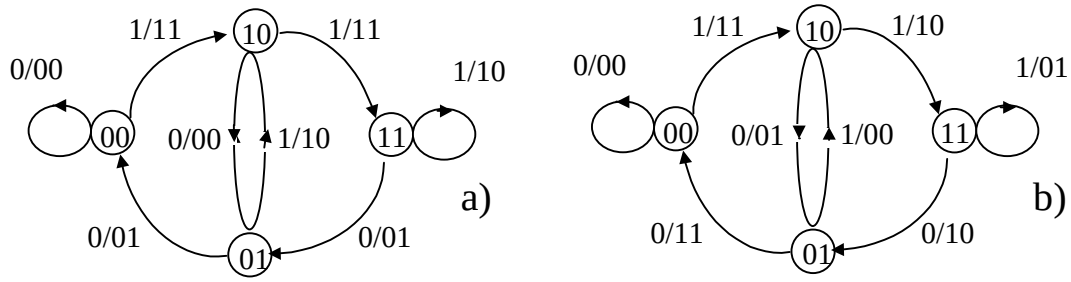
- Reprezentarea grafica a codurilor convolutionale
  - Diagrama de stari

  Codorul se realizeaza cu un RD avand M celule, starea codorului la momentul t_i fiind:

  $S^{(i)} = \left(C_1^{(i)}\; C_2^{(i)} \ldots C_M^{(i)}\right)$ unde $C_j^{(i)}$ este starea celulei C_j la t_i;

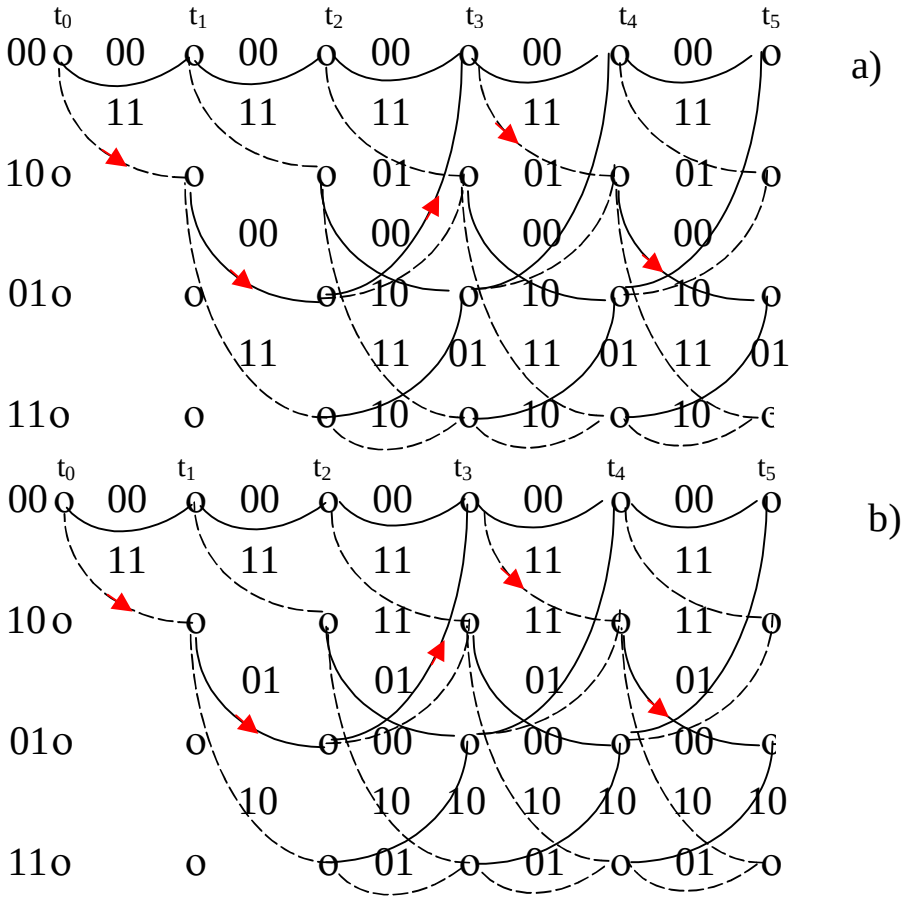
  secventa de cod generata la momentul t_i este complet determinata de starea anterioara a RD, si de simbolul informational i_i; pentru codoarele sistematic (a) si nesistematic (b) date in exemplul

anterior, diagramele de stare, in corespondenta si cu tabelele de evolutia starilor, *pentru starile distincte*, care pentru $M = 2$ sunt doar 4, sunt date mai jos, unde s-a notat pe fiecare arc ce leaga doua stari simbolul informational / secventa codata: *i*/*V*.



- Diagrama trellis (reprezentarea evolutiei starilor in timp)

Este alcatuita din noduri si arce de tranzitie; fiecare nod reprezinta una din cele 2^{kM} stari (determinate de constrangerea sau memoria codorului); din fiecare nod (stare) ies 2^k arce (corespunzator celor 2^k simboluri posibile). Pentru exemplul anterior, la sistematice (a) si nesistematice (b) avem diagramele care urmeaza unde arc plin semnifica tranzitie la primire “0” si arc punctat la primirea “1”:



Traseele marcate cu sageti corespund secventei *i* = (01101)

2. Decodarea (cu algoritmul Viterbi)

Principiul decodarii se prezinta cu ajutorul exemplului pe care s-a facut analiza codarii (cazul nesistematic, respectiv diagrama de stari b); algoritmul Viterbi opereaza pe trellis bloc cu bloc, pe un numar finit de blocuri, pentru a regasi drumul utilizat la codare. In fiecare nod se calculeaza distantele (in alta varianta, gradul de corelatie) intre secventa receptionata si toate secventele de pe trellis, cumuland distanta (gradul de corelatie) de la un nod la altul. In fiecare bloc, in fiecare nod intra doua ramuri, deci vor fi doua distante (grade de corelatie) pentru fiecare nod, dintre care se retine drumul cu distanta minima (corelatie maxima) numit "supravietuitor". Daca intr-un nod exista doua drumuri de distante egale (corelatii egale) se alege unul din ele la intamplare ca "supravietuitor". Cu "supravietuitorii" din blocul j se se analizeaza blocul $j + 1$ etc. Analiza se continua pe atatea blocuri pana cand ramane un singur drum care va fi considerat secventa corecta (numarul de blocuri W pe care se face decodarea se numeste fereastra de decodare). Din practica s-a constatat ca pentru $W = (4 \div 5)\,Mk$ se poate lua o decizie corecta, cu distorsiuni neglijabile fata de cazul $W \to \infty$ (memorie infinita). Secventa receptionata este $V' = (x_{11}x_{12}\ldots x_{i1}x_{i2}\ldots)$ iar secventa aferenta ramurii este $C = (c_{11}c_{12}\ldots c_{i1}c_{i2}\ldots)$. Gradul de corelatie pentru blocul i se defineste:

$$M = x_{i1}\, c_{i1} + x_{i2}\, c_{i2}$$

Daca se atribuie pentru simboluri valori bipolare: $0 \to -1$, $1 \to +1$, M ia valorile din tabel:

$x_{11}x_{12}$	$c_{i1}c_{i2}$	M	Observatie
00	00	(-1)(-1) + (-1)(- 1) = 2	Bitii pt. x si c corespund
11	11	(+1)(+1) + (+1)(+1) = 2	
01	01	(- 1)(- 1) + (+1)(+1) = 2	
00	11	(- 1)(+1) +(-1)(+1) = - 2	Bitii pt. x si c sunt opusi
01	10	(- 1)(+1) +(+1)(-1) = - 2	
10	11	(+1)(+1) +(-1)(+1) = 0	Numai unul din bitii pt. x si c corespunde
01	00	(- 1)(-1) +(+1)(-1) = 0	

Calea de corelatie maxima pentru secventa receptionata V' este data in diagrama trellis de mai jos; ca exemplu de decizie se ia cazul plecarii din nodurile initiale (aferente primului bloc, moment t_0) si se iau decizii pe baza valorilor lui M, calculate pentru fiecare din cele doua cai posibile de iesire din nod. In diagrama, caile posibile sunt reprezentate punctat iar calea

“supravietuitoare” este reprezentata cu linie plina (caile abandonate nu s-au mai reprezentat).

$00 \xrightarrow{00} 00 \quad M = +2$ (calea se valideaza)

$00 \xrightarrow{11} 10 \quad M = -2$ (calea se abandoneaza)

……

$10 \xrightarrow{01} 01 \quad M = 0$ (calea se valideaza la intamplare)

$10 \xrightarrow{10} 11 \quad M = 0$

……

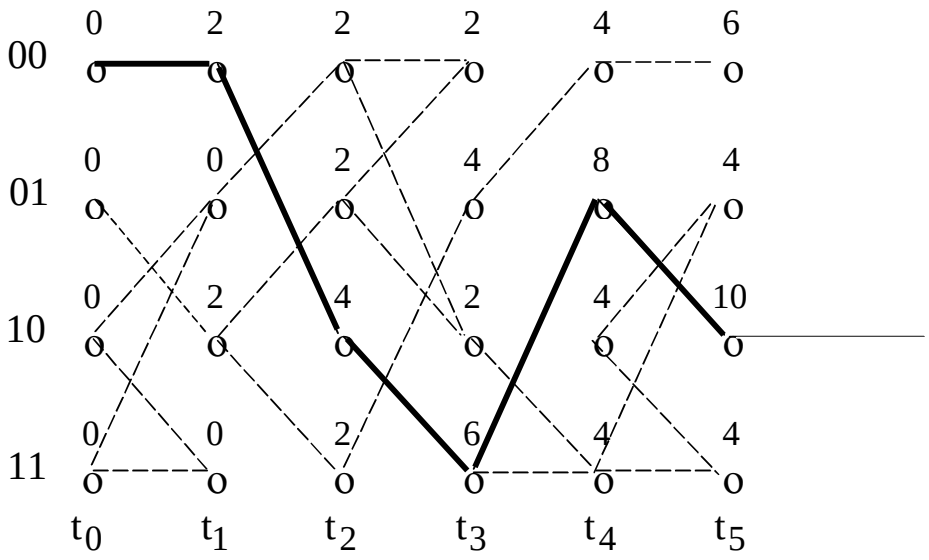
$01 \xrightarrow{11} 00 \quad M = -2$ (calea se abandoneaza)

$01 \xrightarrow{00} 10 \quad M = +2$ (calea se valideaza)

……

$11 \xrightarrow{10} 01 \quad M = 0$

$11 \xrightarrow{01} 11 \quad M = 0$ (calea se valideaza la intamplare)



1) Modul de generare a cuvintelor:

$v(x) = a_0 \oplus a_1 x \oplus a_2 x^2 \oplus \ldots \oplus a_{n-1} x^{n-1}$ $\quad cu\ a_i \in \{0, 1\}$,

de catre un polinom generator $p(x)$ de grad n, este prezentat in tabelul care urmeaza, in care cuvintele sunt reprezentate prin clasele de resturi modulo $p(x)$. De ex., la clasa denumita "0", din prima linie, restul impartirii oricarui element al clasei, prin $p(x)$, este 0.

Continutul clasei	*Elemente ale clasei*	*Denumirea clasei*
0 + multiplii lui $p(x)$	$p(x), xp(x), (1+x)p(x), \ldots$	0
1 + multiplii lui $p(x)$	$1+p(x), 1+xp(x),$ $1+(1+x)p(x), \ldots$	1
$x+$ multiplii lui $p(x)$	$x+p(x), x+xp(x),$ $x+(1+x)p(x), \ldots$	X
$1+x+$ multiplii lui $p(x)$	$1+x+p(x), 1+x+xp(x),$ $1+x+(1+x)p(x), \ldots$	$1+X$
…	…	…
$1+x+\ldots+x^{n-1}+$ multiplii lui $p(x)$	$1+x+\ldots+x^{n-1}+p(x), 1+x+\ldots+x^{n-1}+xp(x), 1+x+\ldots+x^{n-1}+(1+x)p(x), \ldots$	$1+X+\ldots+X^{n-1}$
in general: $a_0+a_1x+\ldots+a_{n-1}x^{n-1}$ + multiplii lui $p(x)$	in general: $a_0+a_1x+\ldots+a_{n-1}x^{n-1}+p(x),$ $a_0+a_1x+\ldots+a_{n-1}x^{n-1}+xp(x),\ldots$	in general: $a_0+a_1X+\ldots+a_{n-1}X^{n-1}$

Exemplu: Elementul $a_0+a_1x+a_2x^2+p(x)= u(x)+p(x)$ *impartit la* $p(x)$ *se noteaza* $a_0+a_1X+a_2X^2+p(X)= a_0+a_1X+a_2X^2 = u(X)$ *deoarece* $p(X) = 0$

2) Se poate da o justificare a ciclicitatii daca se alege $p(x) = 1 + x^n$:

Daca $v(x) = a_0+a_1x+\ldots+a_{n-1}x^{n-1}$ este un cuvant de cod, adica este un polinom divizibil prin $g(x)$ sau $\quad v(x) = g(x)\, i(x)$,

atunci divizibilitatea prin $g(x)$ este valabila si pentru cuvantul

$x\ v(x) = a_0x+a_1x^2+\ldots+a_{n-2}x^{n-1}+ a_{n-1}x^n = a_0x+a_1x^2+\ldots+a_{n-2}x^{n-1}+ a_{n-1}x^n +a^{n-1} +a^{n-1}=(a_{n-1}+ a_0x+a_1x^2+\ldots+a_{n-2}x^{n-1}) +a_{n-1}\,(x^n+1)$

Daca $p(x) = x^n +1$ atunci ultimul termen este un multiplu al lui $p(x)$ si restul partii din dreapta a egalitatii anterioare este un nou cuvant de cod:

$v_1(x) = a_{n-1}+ a_0x+a_1x^2+\ldots+a_{n-2}x^{n-1}$

obtinut din $v(x)$ prin prima permutare ciclica a coeficientilor (simbolurilor) a_j.

3) Daca $p(x) = x^n + 1$, se poate arata ca produsul de forma $h(X)\ v(X)$ intre doua elemente ale claselor de resturi modulo $p(x)$ se poate pune sub o forma similara produsului scalar $\langle h,v \rangle$:

$h(X) = h_0 + h_1X + h_2X^2$

$v(X) = a_0 + a_1X + a_2X^2$

$h(X)\ v(X) = h_0\ a_0 + (\ h_0\ a_1 + h_1\ a_0)X + (\ h_0\ a_2 + h_1\ a_1 + h_2\ a_0)X^2 + (\ h_1\ a_2 + h_2\ a_1)X^3 + h_2\ a_2X^4$

deoarece $p(X) = X^3 + 1 = 0 \rightarrow X^3 = 1,\ X^4 = X, \ldots$

$h(X)\ v(X) = h_0\ a_0 + h_1\ a_2 + h_2\ a_1 + (\ h_0\ a_1 + h_1\ a_0 + h_2\ a_2)X + (\ h_0\ a_2 + h_1\ a_1 + h_2\ a_0)X^2 = 0,$

de unde:

$h_0\ a_0 + h_1\ a_2 + h_2\ a_1 = 0$

$h_0\ a_1 + h_1\ a_0 + h_2\ a_2 = 0$

$h_0\ a_2 + h_1\ a_1 + h_2\ a_0 = 0$

Prima relatie este similara produsului scalar $\langle h,v \rangle = 0$ in care toate componentele unuia din vectori sunt scrise in ordine inversa. In plus, urmatoarele doua relatii pun in evidenta din nou proprietatea de ciclicitate deoarece orice permutare ciclica a coeficientilor conduce la un produs scalar nul. In general, daca gradul lui $h(x)$ este k si gradul lui $v(x)$ este $n-1$ cele m ecuatii corespunzatoare sunt, avand in vedere ca din cele n ecuatii date de permutarile ciclice numai $m = n - k$ sunt independente:

$$\langle a_0 a_1 \ldots a_{n-1} \rangle \langle 00 \ldots 0 h_k \ldots h_0 \rangle = 0$$

$$\langle a_0 a_1 \ldots a_{n-1} \rangle \langle 00 \ldots h_k \ldots h_0 0 \rangle = 0$$

……..

$$\langle a_0 a_1 \ldots a_{n-1} \rangle \langle h_k h_{k-1} \ldots h_0 0 \ldots 0 \rangle = 0$$

care pot fi scrise sub forma matriceala:

$$\begin{bmatrix} 0 & 0 & \ldots & 0 & h_k & h_{k-1} & \ldots & h_1 & h_0 \\ 0 & \ldots & 0 & h_k & h_{k-1} & h_{k-2} & \ldots & h_0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ h_k & \ldots & h_2 & h_1 & h_0 & 0 & \ldots & 0 & 0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \ldots \\ a_{n-1} \end{bmatrix} = 0$$

sau, compact:

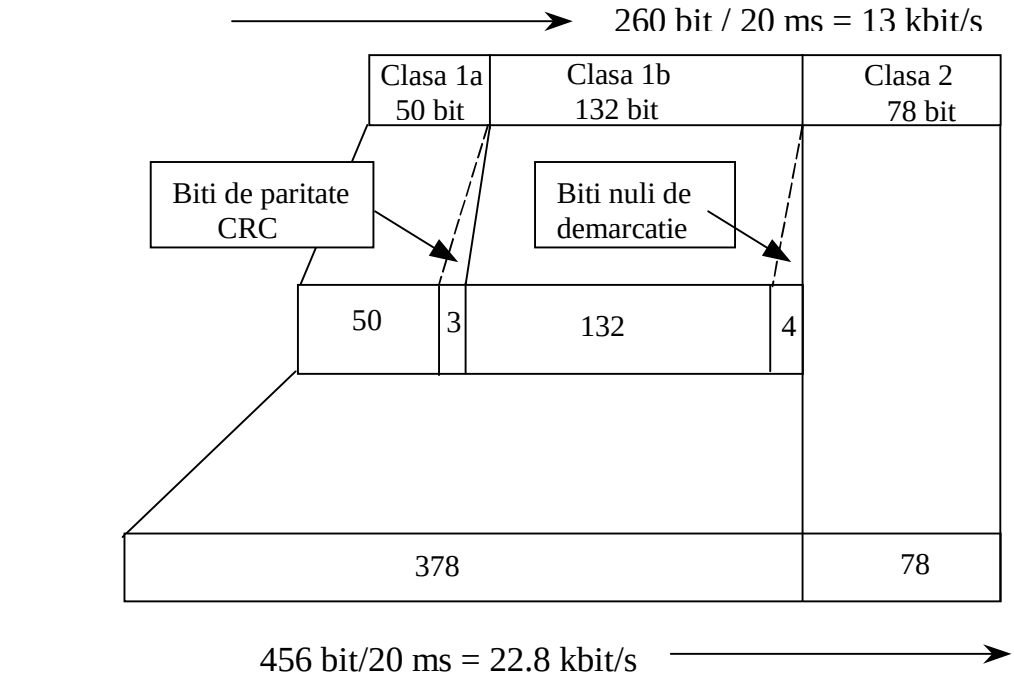
$$\mathbf{H}\mathbf{v}^T = 0$$

Anexa A2

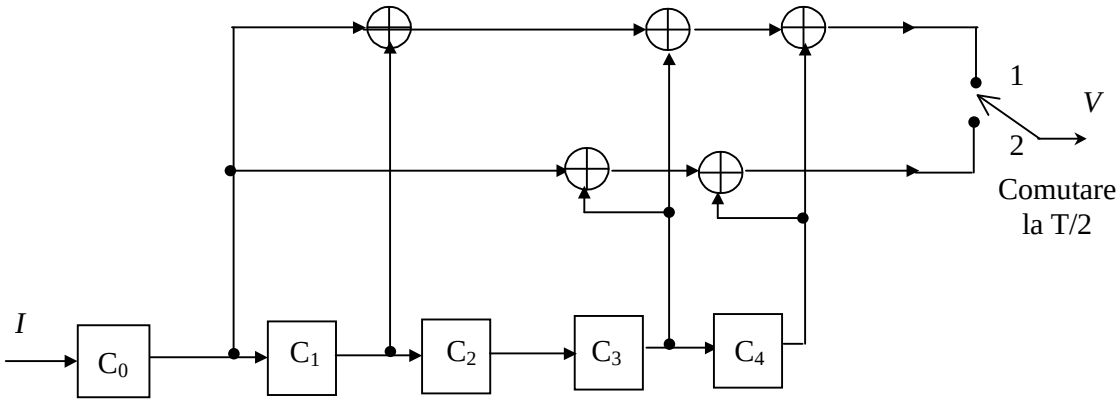
Unele aplicatii ale teoriei codurilor

- Protectia informatiei digitale prin bitul de verificare la paritate.
- Protectia informatiei prin transmiterea ei repetata si decizie majoritara
- Protectia transmisiilor de date prin coduri detectoare de erori care asigura formularea cererii de retransmisie daca blocul transmis este eronat (sisteme ARQ)
- Corectia erorilor in transmisiile de date utilizand coduri iterate
- Protectia memoriilor semiconductoare prin coduri grup corectoare de o eroare si detectoare de erori duble: k = 16, m = 6, n = 22; k = 32, m = 7, n = 39; k = 64, m = 8, n = 72.
- Protectia blocurilor de date sau verificarea accesului in diferite protocoale de comunicatii prin coduri ciclice detectoare de erori bazate pe CRC = (Cyclic Redundancy Check), cu polinoame generatoare de grad m=3, m=16.
- Protectia inregistrarii informatiei pe suport magnetic prin coduri ciclice corectoare de erori simple sau muliple.
- Protectia informatiei in comunicatii satelitare si misiuni spatiale utilizand variante de coduri ciclice nebinare, corectoare de erori multiple si/sau de erori in pachete ca de exemplu coduri Fire, coduri Bose-Chauduri-Hockenguem (BCH), coduri Reed-Solomon.
- Protectia informatiei inregistrate pe CD
- Criptarea informatiei in vederea secretizarii
- Interleaving (intreteserea datelor) pentru a putea proteja prin coduri corectoare de erori independente informatiile afectate de pachete de erori.
- Protectia vorbirii digitale comprimate in comunicatiile mobile (GSM) prin coduri ciclice detectoare de erori combinate cu coduri convolutionale corectoare de erori:

  In sistemul GSM codarea vorbirii se realizeaza utilizand o varianta imbunatatita de sistem de compresie bazat pe metoda predictiei liniare in care fiecare cadru de vorbire avand durata de 20ms este reprezentat prin 260 biti. Dupa cum se vede in figura, bitii sunt impartiti in trei clase: 50 biti foarte importanti pentru calitatea vorbirii, 132 biti importanti si 78 biti nu foarte importanti.



Primilor 50 de biti li se adauga un CRC de 3 biti, calculati cu polinomul generator $g(x) = x^3+x+1$; acesti 53 biti impreuna cu cei 132 biti importanti si patru biti nuli sunt introdusi intr-un codor convolutional cu rata R=1/2 si o lungime de constrangere de 5, reprezentat in figura (n = 2):



Codorul convolutional dubleaza numarul de biti la 378 la care ii adauga si pe cei 78 neprotejati, rezultand 456 biti pe fiecare cadru de vorbire. Rata initiala de 13 kbiti/s devine astfel de 22,8 kbiti/s. Rezulta un factor de compresie de 64/28,8 = 2,16 fata de metoda MIC a telefoniei clasice (in MIC standard rata de bit este de 64 kbit/s).

Anexa A2