Coduri corectoare de erori utilizate în sistemele moderne de transmisiuni

- acest capitol al cursului va face o scurtă prezentare a câtorva clase de coduri corectoare de erori utilizate în sistemel moderne de transmisiuni numerice.
- vor fi prezentate codurile LDPC şi turbocodurile, care alături de codurile convoluţionale şi codurile Reed-Solomon sunt utilizate în cea mai mare parte a sistemelor comerciale de transmisiuni.

1. Codurile LDPC – Low Density Parity Check Codes

1.1. Generalităţi

- Codurile LDPC, introduse în 1963 de Robert G. Gallager, reprezintă o familie de coduri bloc liniare care se obţin pornind de la o matrice „rară" de control a parităţii H, care conţine un număr mic de elemente nenule şi un număr mare de elemente de 0 („sparse matrix").
- Codurile LDPC au fost reinventate de MacKay şi Neal la mijlocul anilor 1990, la scurt timp după ce Berrou şi alţii au introdus turbocodurile şi au demonstrat importanţa tehnicilor iterative de decodare pentru apropierea performanţelor sistemelor practice de transmisie de capacitatea ideală a canalului ($C = \log_2(1+\rho)$ [bps/Hz]).
- Codurile LDPC pot fi privite ca şi coduri bloc liniare, descrise de matricea de control a parităţii *H,* ale cărei dimensiuni sunt $M \times N$,
- La aceste coduri ecuaţia: $Hv^t = 0$ este satisfăcută pentru toate cuvintele de cod *v*.
- Fiecare linie a matricii *H* defineşte o ecuaţie de control a parităţii care trebuie să fie satisfăcută de fiecare cuvânt de cod *v*. Rezolvând sistemul format de aceste ecuaţii se determină biţii de control specifici fiecărui cuvânt de cod.
- Spre deosebire de matricea de control *H* a unui cod Hamming, matricile de control a parităţii care determină codurile LDPC au următoarele proprietăţii specifice:

- dimensiunile *M* şi *N* ale matricii *H* sunt mult mai mari faţă de dimensiunile unei matrici de control specifice unui cod Hamming. S-au impus aceste dimensiuni astfel încât, prin codare, să ne apropiem cât mai mult de limita teoretică de codare impusă de teorema lui Shannon, în care se presupune un cod corector cu lungimea infinită a cuvântului de cod
- matricile *H* sunt definite, prin construcţie, într-o formă (cuasi-)nesistematică şi conţin un număr de elemente de 1 (pentru codurile binare) mult mai mic decât numărul de elemente de 0;
- matricea de control a parităţii ce defineşte un cod LDPC H*(j, k)* are exact *j* elemente de 1 pe fiecare coloană şi exact *k* elemente de 1 pe fiecare linie. Acestea înseamnă că fiecare bit al cuvântului de cod intră în *j* ecuaţii de control, iar fiecare ecuaţie de control conţine *k* biţi.

- Clasa matricilor de control *H* pentru codurile LDPC trebuie să satisfacă următoarele constrângeri:

- fiecare coloană trebuie să conţină un număr mic, fix, de elemente de 1, nota cu *j*;
- fiecare linie trebuie să conţină un număr mic, fix, de elemente de 1 , notat cu *k*.

- Aceasta echivalează cu descrierea matricii *H* de către un graf bipartit cu două tipuri de noduri:

- *N* noduri de simbol care corespund fiecărui bit din cuvântul de cod, (coloanele matricii) şi
- *M* noduri de control care corespund ecuaţiilor de control de paritate, noduri reprezentate de liniile matricii- *se va reveni asupra acestui aspect*

- Constrângerile menţionate anterior impun ca fiecare nod de simbol (bit) să fie conectat cu *j* noduri de verificare a parităţii şi fiecare nod de verificare a parităţii să fie conectat cu *k* noduri de simbol (bit).
- Codurile definite prin astfel de grafuri pot fi decodate cu algoritmul Sumă-Produs (Sum Product Algorithm – SPA sau Message-Passing –MP)). Decodarea cu acest algoritm asigură performanţele unui decodor bazat pe principiul ML (*Maximum-Likelihood*) numai dacă grafurile sunt fără cicli cu lungime mică, sau altfel spus, sunt de tipul „*4-cycle free" - se va reveni asupra acestui aspect*.
- Această cerinţă este echivalentă cu condiţia ca matricea *H* **să nu aibă două linii (oricare două linii) în care să existe elemente de 1 suprapuse pe mai mult de o singură poziţie**.
- Pornind de la această condiţie, limitele ce se impun dimensiunilor *M* (linii) şi *N* (coloane) ale matricii *H* sunt intercondiţionate de relaţia:

$$N \le \frac{M(M-1)}{j(j-1)} \tag{1}$$

1.2. Tipuri de coduri LDPC

- După modul de generare a matricii de control H, se pot defini mai multe clase de coduri LDPC, dintre care amintim:

- construite aleator (random)

- construite pe baza unor structuri regulate: de tip „array-based" sau "quasi-cyclic"
- pe baza unor construcţii combinatoriale
- pe baza unor construcţii geometrice
- construite pe baza grafurilor
- există şi alte construcţii mai recente

1.3. Coduri LDPC random

- în literatură sunt prezentate mai multe metode de generare a matricilor H care definesc codurile LDPC de tip random; prezentarea acestor metode depăşeşte cadrul acestui curs
- pentru a asigura existenţa a k biţi de „1" în fiecare linie (adică o ecuaţie de control să depindă de k biţi de cod) şi j biţi de „1" în fiecare coloană (adică un bit de cod să fie inclus în j ecuaţii) ale matricii H, lungimea cuvântului de cod N şi numărul de biţi de control M trebuie să satisfacă, în cazul acestor coduri, condiţiile (1'.a) şi (1'.b).
- rata acestui tip de cod poate fi aproximată prin relaţia (1'.c).

$$N_1 = \frac{k(k+1)}{2}(j-1)) \text{ ; a. ; } j \geq 2 \text{ ; } k > j \text{ ; } N \geq N_1 \text{ şi } M = \left[N \cdot \frac{j}{k} \right] \in N; \quad \text{b. ; } R_c \cong \frac{k-j}{k}; \quad \text{c. ; } \quad (1')$$

1.3. Matrici de control a parităţii bazate pe coduri array pentru coduri LDPC

-Vom defini acest tip de coduri LDPC prin intermediul a trei parametri: un număr prim *p* şi două numere întregi *k* şi *j* astfel încât $j < k \leq p$.
- Se construieşte mai întâi o matrice H de dimensiuni *MxN*, cu $M = jp$, $N = kp$, de forma:

$$H = \begin{bmatrix} I & I & I & \cdots & I \\ I & \alpha & \alpha^2 & \cdots & \alpha^{k-1} \\ I & \alpha^2 & \alpha^4 & \cdots & \alpha^{2(k-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & \alpha^{j-1} & \alpha^{2(j-1)} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix} \quad (2)$$

unde *I* este matricea unitate de dimensiune $p \times p$, iar α este o matrice de dimensiune $p \times p$ obţinută prin deplasare la dreapta sau la stânga, cu câte o poziţie, a coloanelor matricii *I*.
- Pentru $p = 5$ matricea α va avea una din următoarele forme:

$$\alpha_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{sau} \quad \alpha_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (3)$$

- Această formă a matricii H nu are rangul M, pentru a permite generarea unică a M biţi de control prin rezolvarea ecuaţiei (4)

$$H \cdot v^t = 0 \text{ - sistem de N ec. cu M nec. compatibil nedeterminat} \quad (4)$$

- Una din metodele prin care se poate asigura existenţa unei soluţii unice a ecuaţiei (4) constă în aducerea matricii H la o formă triangularizată
- Pentru aceasta se defineşte o matrice H_S' obţinută prin deplasarea ciclică a coloanelor din matricea *H* (fiecare element al matricii *H* este de fapt o matrice de dimensiune $p \times p$). Deplasarea ciclică a fiecărei coloane se face astfel încât pe diagonala principală a submatricii de dimensiune $jp \times jp$ să avem ca elemente doar matrici unitate *I* de dimensiune $p \times p$; prin permutări de coloane (linii), rezultă matricea completă (5):

$$H'_S = \begin{bmatrix} I & I & I & \cdots & I & I & \cdots & I \\ \alpha^{k-1} & I & \alpha & \cdots & \alpha^{(j-2)} & \alpha^{j-1} & & \alpha^{k-2} \\ \alpha^{2(k-2)} & \alpha^{2(k-1)} & I & \cdots & \alpha^{2(j-3)} & \alpha^{2(j-2)} & & \alpha^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & & \vdots \\ \alpha^{(j-1)(k-j+1)} & \alpha^{(j-1)(k-j+2)} & \cdots & \cdots & I & \alpha^{j-1} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix} \quad (5)$$

- Apoi se triangularizează matricea H_s' înlocuind elementele de sub diagonala principală, matrici p x p, cu matrici *O* de dimensiune $p \times p$ care conţin numai elemente de 0. Astfel se va obţine matricea de control triangularizată H_T(6) pentru codurile LDPC care este tot o matrice de tipul "4-cycle free":
- Codul LDPC definit de matricea H_T , este de fapt definit de parametrii p, j, k;

$$H_T = \begin{bmatrix} I & I & I & \cdots & I & I & \cdots & I \\ O & I & \alpha & \cdots & \alpha^{j-2} & \alpha^{j-1} & & \alpha^{k-2} \\ O & O & I & \cdots & \alpha^{2(j-3)} & \alpha^{2(j-2)} & & \alpha^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & & \vdots \\ O & O & \cdots & O & I & \alpha^{j-1} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix} \quad (6)$$

- Aceştia permit construcţia matricii de control H_T, determinarea lungimii $N = k \times p$ a cuvântului de cod, a numărul de ecuaţii de control al parităţii $M = j \times p$ (egal cu numărul biţilor de control) şi a lungimii blocului de informaţie $K = (k - j) \times p$.

- rata codului este dată de relaţia: $$R = \frac{K}{N} = \frac{(k-j)\cdot p}{k\cdot p} = 1 - \frac{j}{k} \quad (7)$$

- matricea H_T poate fi calculată off-line şi memorată, sau se pot determina poziţiile biţilor de „1”, în funcţie de indecşii de linie şi coloană, iar matricea poate fi generată dinamic în timp real, fără a fi necesară ridicarea la putere a matricii α.

1.4. Codarea codurilor LDPC

- Considerând cuvântul de cod $v = [c_0,\ldots,c_{M-1}, i_0,\ldots,i_{(N-M)-1}]$, biţii de control c_m se pot calcula, în funcţie de biţii informaţionali i_l, prin rezolvarea sistemului de M ecuaţii liniare:

$$H\cdot v^t = 0 \quad (8)$$

- Această abordare are două dezavantaje majore:

- pentru valori mari ale parametrilor j şi sau p, M ia valori mari necesitând un mare volum de calcule, ceea ce conduce la creşterea timpului şi/sau resurselor hardware necesare procesării.
- are nevoie de toţi biţii informaţionali i_l, l = 0,..., (N-M)-l, în acelaşi timp; această cerinţă induce o întârziere suplimentară de un cuvânt de cod în sistemul de transmisie.

- Aceste dezavantaje pot fi evitate de metoda de codare, mai simplă şi mai rapidă, descrisă mai jos.
- Matricea H_T, (M x N), este împărţită în două matrici D şi E, reţinând primele M coloane în matricea D şi restul de (N-M) coloane în matricea E. Cele două matrici au dimensiunile:

$$D \rightarrow (M \times M); \quad E \rightarrow (M \times (N-M)); \quad (9)$$

- Matricea D este pătrată şi are determinant nenul, datorită construcţiei matricii H, deci are inversă D^{-1}.
- Rezultă că ecuaţia codării (8) poate fi rescrisă sub forma (10), unde matricea F este matricea de codare:

$$[H]\cdot[v]^t = [0] \Leftrightarrow [D]\cdot\begin{bmatrix} c_0 \\ \vdots \\ c_{M-1} \end{bmatrix} + [E]\cdot\begin{bmatrix} i_0 \\ \vdots \\ i_{(N-M)-1} \end{bmatrix} = [0] \Rightarrow [D]^{-1}\cdot[D]\cdot\begin{bmatrix} c_0 \\ \vdots \\ c_{M-1} \end{bmatrix} + [D]^{-1}\cdot[E]\cdot\begin{bmatrix} i_0 \\ \vdots \\ i_{(N-M)-1} \end{bmatrix} = [0]$$
$$\Rightarrow \begin{bmatrix} c_0 \\ \vdots \\ c_{M-1} \end{bmatrix} = [D]^{-1}\cdot[E]\cdot\begin{bmatrix} i_0 \\ \vdots \\ i_{(N-M)-1} \end{bmatrix} = [F]\cdot\begin{bmatrix} i_0 \\ \vdots \\ i_{(N-M)-1} \end{bmatrix} \quad (10)$$

- Prin dezvoltarea matricii de codare F, (M x (N-M)), dacă notăm cu $[f_i]$ coloanele sale (fiecare din ele un vector de M biţi, relaţia (10) poate fi exprimată sub forma:

$$[f_0]\cdot i_0 + \ldots [f_{(N-M)-1}]\cdot i_{(N-M)-1} = [C] \quad (11)$$

- vezi comentarii pe tablă

- Matricea F, calculată off-line, este stocată coloană cu coloană în memorie; fiecare bit de informaţie se înmulţeşte cu coloana cu acelaşi index, iar vectorii-produs rezultaţi sunt acumulaţi, astfel obţinându-se vectorul biţilor de control [C]; pentru codurile de tip “array” sau cuasi-ciclice matricea F poate fi calculată în timp real
- Matricea F_s a unui cod prescurtat, K’=K-U, se obţine eliminând din matricea F un număr de U coloane care corespund biţilor informaţionali care se elimină din cuvântul de cod complet;
- cele U coloane pot fi – vezi comentarii pe tablă:

- ultimele U coloane din partea dreaptă a matricii F
- coloanele ai căror indecşi corespund biţilor care au “girth”-ul cel mai mic în graful Tanner asociat

- Această metodă de codare este mai rapidă decât cea descrisă de ecuaţia (8) şi calculează biţii de control în mod paralel folosind un bit de informaţie la fiecare tact, ceea ce face ca ea să nu insereze întârzieri suplimentare în sistemul de transmisie.
- De asemenea volumul de calcule necesar determinării biţilor de control este repartizat uniform în timp.

1.5. Decodarea codurilor LDPC – algoritmul Sum-Product (Message Passing)

- Algoritmul de decodare SP (MP) este un algoritm iterativ care nu urmăreşte determinarea cuvântului de cod cel mai apropiat de secvenţa recepţionată, în sensul unei metrici;
- El urmăreşte decodarea fiecărui bit folosind informaţiile primite privitoare la toţi biţii cuvântului de cod şi utilizând corelaţiile dintre aceşti biţi impuse de ecuaţiile de control
- Algoritmul MP utilizează ca date de intrare probabilităţile *aposteriori* ale biţilor cuvântului de cod, furnizate de funcţia de demapare soft (vezi paragraful următor, pag. 9).
- La fiecare iteraţie, algoritmul modifică valorile acestor probabilităţi ale unui bit, în funcţie de probabilităţile *aposteriori* ale celorlalţi biţi care intră in aceleaşi ecuaţii de control cu bitul dat. Aceste actualizări sunt făcute pentru fiecare bit.
- La sfârşitul fiecărei iteraţii, în pasul 3, se ia o decizie asupra valorii logice a biţilor obţinuţi în iteraţia respectivă, folosind criteriul lui Bayes, iar apoi, în pasul 4, cu biţii obţinuţi se calculează sindromul, cu care se verifică dacă toţi biţii cuvântului de cod sunt corecţi.
- Dacă acesta este nul, decodarea se consideră încheiată; dacă nu, atunci se trece la o nouă iteraţie, până la atingerea numărului maxim de iteraţii *I*.
- Dacă s-a atins numărul maxim de iteraţii permise, atunci se livrează biţii decişi după ultima iteraţie, chiar dacă sindromul lor e nenul. Alternativ, în acest caz se poate "şterge" acest cuvânt de cod (erasure).
- Pentru a exemplifica acest concept se consideră codul LDPC (k = 4, j = 3, p = 5, N=20, M=15). Acest cod este descris de matricea H completă dată de (12); sistemul de ecuaţii de control a parităţii, construit pe baza (8) este descris de (13). **Notă: acest cod nu este de tip array.**

$$H = \left[\begin{array}{cccccccccccccccccccc}
1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1\\
\hline
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0\\
0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&1&0&0&0\\
0&0&1&0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&0&0\\
0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0\\
0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1\\
\hline
1&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&1&0&0\\
0&1&0&0&0&0&1&0&0&0&1&0&0&0&0&1&0&0&0&0\\
0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&0&1&0\\
0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&1&0&0&0\\
0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1
\end{array}\right] \qquad (12)$$

$$\left[\begin{array}{cccccccccccccccccccc}
1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1\\
\hline
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0\\
0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&1&0&0&0\\
0&0&1&0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&0&0\\
0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0\\
0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1\\
\hline
1&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&1&0&0\\
0&1&0&0&0&0&1&0&0&0&1&0&0&0&0&1&0&0&0&0\\
0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&0&1&0\\
0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&1&0&0&0\\
0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1
\end{array}\right]
\cdot
\left[\begin{array}{c}
c_1\\c_2\\c_3\\c_4\\c_5\\ \hline \vdots\\c_{12}\\c_{13}\\c_{14}\\c_{15}\\ \hline m_1\\m_2\\m_3\\m_4\\m_5
\end{array}\right]
= [0] \qquad (13)$$

- În (13), c_i, $i=1..M$, sunt cei M= 3·5 biţi de control, iar m_i, $i=1..K$, $K=N-M$ sunt cei K = (4-3)·5 biţi informaţionali din cuvântul de cod. Pentru a obţine biţii c_i asociaţi cuvântului binar m={m_1, m_2,...,m_K}trebuie rezolvat sistemul următor: Notă: fiecare ecuaţie conţine k = 4 biţi, iar fiecare bit intră în j = 3 ecuaţii

$$\begin{cases} c_1 + c_2 + c_3 + c_4 = 0\ (1) \\ c_4 + c_6 + c_7 + c_8 = 0\ (2) \\ c_9 + c_{10} + c_{11} + c_{12} = 0\ (3) \\ c_{13} + c_{14} + c_{15} + m_1 = 0\ (4) \\ m_2 + m_3 + m_4 + m_5 = 0\ (5) \\ c_1 + c_5 + c_9 + c_{13} = 0\ (6) \\ c_2 + c_6 + c_{10} + m_2 = 0\ (7) \\ c_3 + c_7 + c_{14} + m_3 = 0\ (8) \\ c_4 + c_{11} + c_{15} + m_4 = 0\ (9) \\ c_8 + c_{12} + m_1 + m_5 = 0\ (10) \\ c_1 + c_6 + c_{12} + m_3 = 0\ (11) \\ c_2 + c_7 + c_{11} + m_1 = 0\ (12) \\ c_3 + c_8 + c_{13} + m_4 = 0\ (13) \\ c_4 + c_9 + c_{14} + m_2 = 0\ (14) \\ c_5 + c_{10} + c_{15} + m_5 = 0\ (15) \end{cases} \quad (14)$$

- Aceşti biţi (de control) sunt calculaţi la codare folosind metoda echivalentă dată de relaţiile (9)-(11), dar sistemul (14) este folosit pentru construcţia grafului bipartit (Tanner) asociat codului LDPC
- Graful bipartit asociat codului LDPC (k = 4, j = 3, p = 5) este dat în figura 1

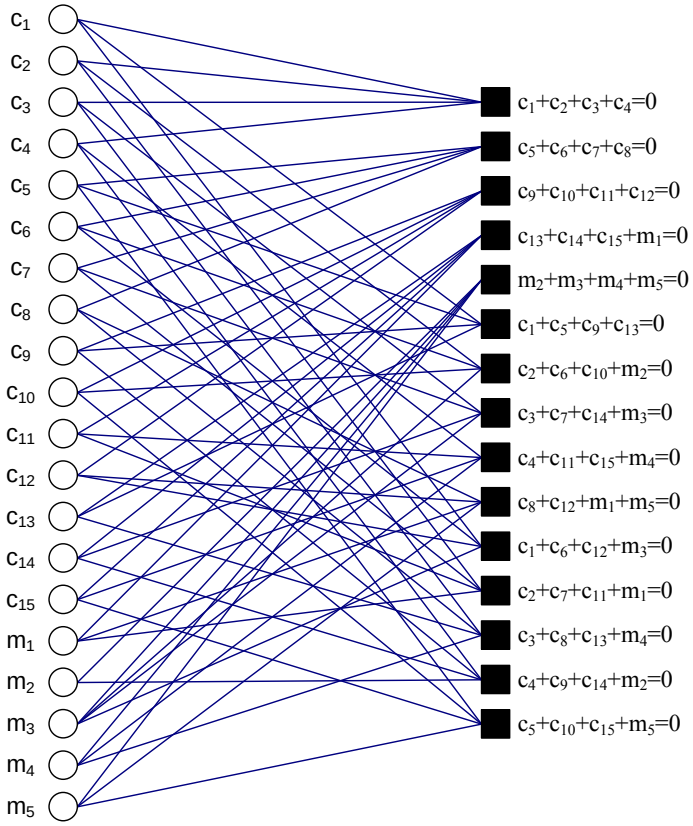


Figura 1 Graful bipartit corespunzător matricii de control a parităţii

Etapele algoritmului de decodare MP

Pasul 0 – iniţializare

- Nodurile de bit sunt iniţializate cu probabilităţile *aposteriori* iniţiale, P_j^0 (probabilitatea ca bitul j din cuvântul recepţionat să fie 0) şi P_j^1(probabilitatea ca bitul j din cuvântul recepţionat să fie 1), determinate de blocul de demapare soft (vezi (26) - par. 1.6) pe baza semnalului recepţionat.

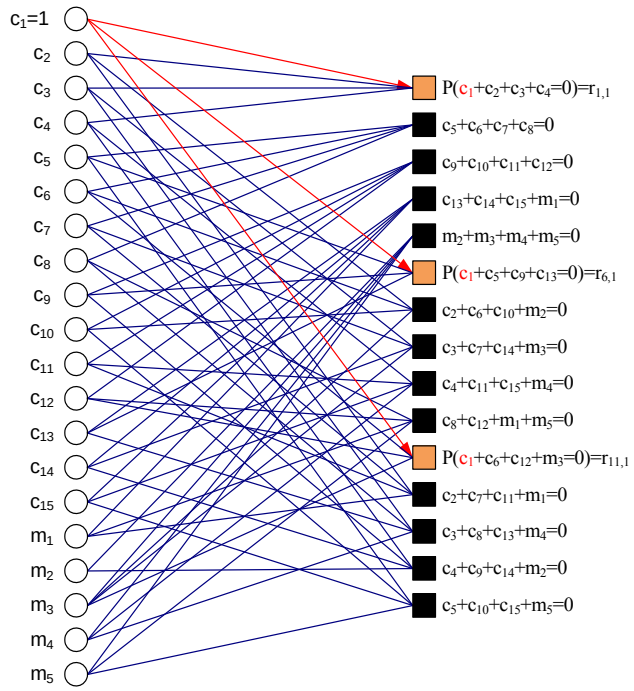


Figura 2 Actualizarea nodurilor de control

Pasul 1- actualizarea nodurilor de control

- Se determină probabilităţiile $r_{m,n}^{x}$, care reprezintă probabilitatea ca ecuaţia de control *m* să fie satisfăcută dacă bitul *n* are valoarea x, (0 sau 1).

- Probabilitatea ca prima ecuaţie de control a sistemului definit de (14) să fie satisfăcută, dacă c_1=1, este:

$$\begin{aligned} &P(c_1 + c_2 + c_3 + c_4 = 0 \,|\, c_1 = 1) \\ &= P(c_2 + c_3 + c_4 = 1) = \\ &= P_2^1 \cdot P_3^0 \cdot P_4^0 + P_2^0 \cdot P_3^1 \cdot P_4^0 + \\ &+ P_2^0 \cdot P_3^0 \cdot P_4^1 + P_2^1 \cdot P_3^1 \cdot P_4^1 \end{aligned} \tag{15}$$

- Din (15) rezultă că, dacă bitul *n* are valoarea 1, atunci ecuaţia *m* de control al parităţii este satisfăcută numai dacă între ceilalţi k-1 biţi care intră în această ecuaţie există un număr impar de biţi care au valoarea 1

- Se poate arăta că probabilitatean $r_{m,n}^{1}$ ca din k_m *-1* biţi independenţi ai ecuaţiei de control *m*, **diferiţi de bitul *n***, un număr impar de biţi să fie nenuli este:

$$r_{m,n}^{1} = \frac{1 - \prod\limits_{\substack{l=1 \\ l \neq n}}^{k_m} \left(1 - 2q_{l,m}^{1}\right)}{2} = \frac{1}{2}\left(1 - \prod_{\substack{l=1 \\ l \neq n}}^{k_m} \left(q_{l,m}^{0} - q_{l,m}^{1}\right)\right) \tag{16}$$

unde $q_{n,m}^{x}$ este probabilitatea ca bitul *n* al cuvântului de cod recepţionat, care intră în ecuaţia *m*, să aibă valoarea *x* =„0” sau „1”, calculată pe baza fazorului *y* recepţionat, vezi (26). Practic $q_{n,m}^{x}$, (17), reprezintă mesajul primit de nodul de control *m* de la nodul de bit *n* pe parcursul unei iteraţii, dacă s-a recepţionat *y*:

$$q_{n,m}^{x} = P_n^{x} = p(y \,|\, b_n = x) \tag{17}$$

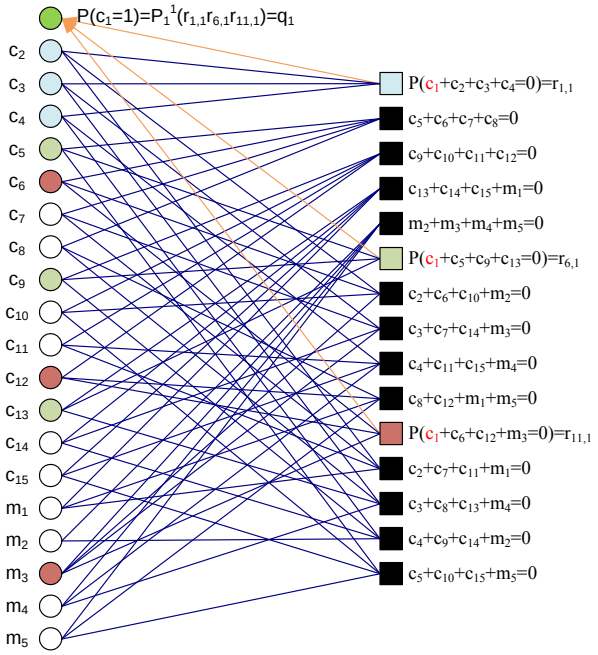
- Pentru fiecare nod de control *m* se determină $r_{m,n}^{1}$ şi $r_{m,n}^{0}$ adică probabilităţile ca ecuaţia de control *m* să fie satisfăcută dacă bitul corespunzător nodului de bit *n* are valoarea „1” respectiv „0”, pentru fiecare nod de bit *n* care este legat cu nodul de control *m*. Notând cu *M(n)* mulţimea ecuaţiilor de control în care intra bitul *n* şi cu *N(m)* mulţimea biţilor care intră în ecuaţia *m*, $r_{m,n}^{1}$ şi $r_{m,n}^{0}$ se calculează cu (18):

$$\begin{aligned} &for\ n = 0, \ldots, N-1 \\ &\quad for\ \ m \in M(n) \\ &\qquad r_{m,n}^{0} = \frac{1}{2}\left(1 + \prod_{\substack{t \in N(m) \\ t \neq n}} \left(q_{t,m}^{0} - q_{t,m}^{1}\right)\right); \\ &\qquad r_{m,n}^{1} = \frac{1}{2}\left(1 - \prod_{\substack{t \in N(m) \\ t \neq n}} \left(q_{t,m}^{0} - q_{t,m}^{1}\right)\right) \\ &\quad end \\ &end \end{aligned} \tag{18}$$

Pasul 2 Actualizarea nodurilor de bit

- La sfârşitul fiecărei iteraţii se determină probabilităţile *a posteriori* pentru fiecare bit, adică probabilităţile $P_n^{1(0)}$ ca bitul transmis pe poziţia *n* să fie „1” (sau „0”) condiţionat de setul de simboluri *y* recepţionate (din care se obţin probabilităţile *aposteriori* ale biţilor) şi de evenimentul $S_1(S_0)$ ca cele *j* ecuaţii de control, în care intră bitul *n*, {*M(n)*}, să fie satisfăcute pentru b_n *=1(0)* , vezi figura 3.

- Fiindcă ecuaţiile de control sunt independente, probabilitatea ca toate cele {M(n)}ecuaţii de control în care intră bitul *n* să fie satisfăcute este produsul probabilităţilor ca fiecare din aceste ecuaţii să fie satisfăcută, adică (19).



Notă: card(M(n)) = j, dar mulţimea {M(n)} diferă pentru fiecare bit.

$$\mathrm{P}\left(x_n = 1 \mid S_1, \{y\}\right) = \alpha_n \cdot P_n^1 \cdot \prod_{i=1}^{M(n)} \left(r_{i,n}^1\right)$$
$$\mathrm{P}\left(x_n = 0 \mid S_0, \{y\}\right) = \alpha_n \cdot P_n^0 \cdot \prod_{i=1}^{M(n)} \left(r_{i,n}^0\right) \qquad (19)$$

- În (19) constanta de scalare α_n se alege astfel încât:

$$\mathrm{P}\left(x_n = 1 \mid S_1, \{y\}\right) + \mathrm{P}\left(x_n = 0 \mid S_0, \{y\}\right) = 1 \qquad (20)$$

-vezi explicaţii pe tablă

- Acest pas reprezintă actualizarea probabilităţilor *aposteriori* ale biţilor

Figura 3 Actualizarea nodurilor de bit

- Noul mesaj transmis de nodul de bit *n* spre nodul de control *m* în iteraţia următoare va fi (21), în care se consideră doar efectul ecuaţiilor în care intră bitul *n*, mulţimea {*M(n*}, diferite de ecuaţia curentă *m*

$$q_{n,m}^x = \alpha_{nm} P_n^x \prod_{\substack{l=1 \\ l \neq m}}^{M(n)} r_{l,n}^x \qquad (21)$$

- În (21) $q_{n,m}{}^0$, $q_{n,m}{}^1$ reprezintă probabilităţile ca bitul n, "0" sau „1", să satisfacă toate grupurile (mulţimile) de M(n) -1 ecuaţii de control. Aceste valori sunt trimise de către nodurile de bit şi vor fi folosite în iteraţia următoare la pasul 1 (actualizarea nodurilor de control) . Notă: mulţimile sunt diferite pentru fiecare bit, dar cardinalul lor e acelaşi fiind egal cu j-1
- În cadrul acestui pas se mai calculează şi probabilităţile $Pl_n{}^0$, $Pl_n{}^1$ca bitul n, „0" sau „1", să satisfacă toate ecuaţiile de control, care vor fi folosite pentru decizia bitului pe baza criteriului lui Bayes.
- Pe baza relaţiilor (19) şi (21) pasul 2 al algoritmului de decodare va fi:

actualizarea nodurilor de bit →
- nu conţin ecuaţia de control *m*

$$\begin{aligned}
&for\ n = 0, \ldots, N-1 \\
&\quad for\ m \in M(n) \\
&\qquad q_{n,m}^0 = \alpha_{nm} P_n^0 \prod_{\substack{t \in M(n) \\ t \neq m}} r_{t,n}^0 \\
&\qquad q_{n,m}^1 = \alpha_{nm} P_n^1 \prod_{\substack{t \in M(n) \\ t \neq m}} r_{t,n}^1 \qquad (22) \\
&\quad end \\
&\quad Pl_n^0 = \beta_n P_n^0 \prod_{t \in M(n)} r_{t,n}^0 \\
&\quad Pl_n^1 = \beta_n P_n^0 \prod_{t \in M(n)} r_{t,n}^0 \\
&end
\end{aligned}$$

calculul probabiltăţilor pt. criteriul luiBayes→
- conţin şi ecuaţia de control *m*

- Factorii de scalare β_n sunt calculaţi cu o relaţie similară cu (20) pentru valorile expresiilor din ultimele două rânduri ale (22).

Pasul 3 Decizia biţilor decodaţi, decizia privind corectitudinea biţilor decodaţi şi decizia de terminare a decodării

- Valoarea logică a fiecărui bit al cuvântului de cod se determină cu o formă a criteriului lui Bayes, $Pl_n{}^1 > 0.5$, atunci bitul decodat este 1, altfel bitul decodat este 0.
- Corectitudinea biţilor decodaţi se determină prin verificarea ecuaţiei $Hv^T = 0$.
- Această înmulţire dintre o matrice şi un vector poate fi realizată prin metoda descrisă de relaţia (11)

- Decizia privind terminarea decodării depinde de doi factori: valoarea sindromului şi numărul de iteraţii efectuate:

- Dacă sindromul este nul, decodarea este încheiată, iar biţii decişi sunt livraţi către blocurile următoare.
- Dacă sindromul nu este nul, $Hv^t \neq 0$, se efectuează o nouă iteraţie, adică se transmit noile mesaje de la nodurile de bit spre nodurile de control în care, pe baza acestor mesaje, se calculează noile mesaje spre nodurile de bit; cu aceste mesaje se determină noile probabilităţi *aposteriori* Pl_n^0, Pl_n^1, numai dacă numărul de iteraţii efectuate nu a atins numărul maxim permis *I*; *I* este un parametru al algoritmului MP.

- Conform consideraţiilor de mai sus, următorul pas al algoritmului este exprimat de (23).

$$\begin{aligned} &for\ n = 0, \ldots, N-1 \\ &\quad daca\ Pl_n^1 \geq 0.5\ atunci\ x_n = 1 \\ &\quad altfel\ x_n = 0 \\ &end \\ &Sind = Hv^t \\ &\quad daca\ Sind = 0\ decodare\ ter\min ata \\ &\quad altfel\ daca\ i = I\ decodare\ ter\min ata \\ &\qquad altfel\ mergi\ la\ pasul\ 1 \end{aligned} \tag{23}$$

- Acest algoritm iterativ continuă până când se găseşte un cuvânt de cod valid (cu sindrom nul) sau până când numărul de iteraţii atinge o valoare *I*, pentru care graful de decodare va deveni un arbore cu *I* nivele, vezi figurile 4 şi 5, caz în care se consideră că decodarea a eşuat -fie se livrează aşa, fie se şterge (erasure)
- Trebuie însă menţionat că obţinerea unui cuvânt de cod valid nu este echivalentă cu obţinerea cuvântului de cod corect, cel transmis.
- În figurile 4 şi 5 sunt prezentate grafurile vecinilor bitului c_1 după una, şi respectiv, două iteraţii, pentru codul definit în relaţia (14)

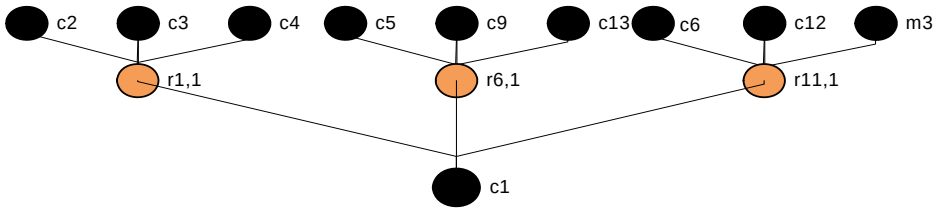


Figura 4 Graful vecinilor nodului de bit c_1 după prima iteraţie (de jos in sus)

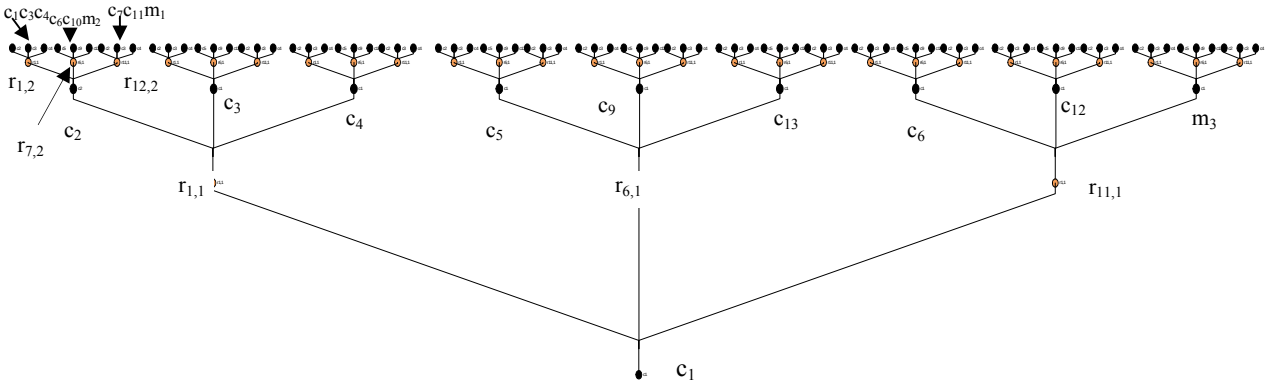


Figura 5 Graful vecinilor nodului de bit c_1 după a doua iteraţie (de jos in sus)

- Generalizând reprezentarea din fig.5, pentru ca toţi vecinii de gradul L ai unui nod de bit să fie independenţi (să apară o singură dată în graful vecinilor) după iteraţia L, lungimea *N* a cuvântului de cod ar trebui să fie :

$$N \geq \left[(k-1) \cdot j \right]^L \tag{24}$$

- Deoarece lungimea cuvintelor de cod în situaţii reale nu respectă (24), subgraful vecinilor nodului de bit *n* rezultat în urma decodării iterative, după un număr de *I* iteraţii, nu va fi un arbore fără bucle.
- Aceasta înseamnă că mesajele de bit care contribuie la determinarea anumitor mesaje de control nu vor mai fi independente din punct de vedere statistic, ecuaţia (16) nu mai este validă (pt. că biţii nu mai sunt independenţi), iar valorile probabilităţilor $q_{n,m}^0$, $q_{n,m}^1$ nu se vor mai modifica (semnificativ).
- Considerând ecuaţiile de control (14) ale codului definit de (12) şi (13), în figura 6 (care reprezintă tot

graful vecinilor bitului c_1) este figurată o buclă de 4 cicli, definită de:
$c_1 \rightarrow r_{11} \rightarrow c_6 \rightarrow r_2 \rightarrow c_8 \rightarrow r_{13} \rightarrow c_{13} \rightarrow r_6 \rightarrow c_1$ – 8 pasi – 4 cicli

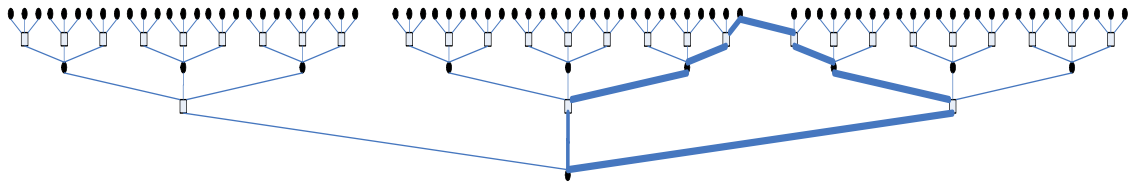


Figura 6 Buclă în subgraful vecinilor; pentru indecşii nodurilor vezi fig. 5 şi (14)

$$\begin{cases} c_1 + c_2 + c_3 + c_4 = 0 \ (1) \\ c_4 + c_6 + c_7 + c_8 = 0 \ (2) \\ c_9 + c_{10} + c_{11} + c_{12} = 0 \ (3) \\ c_{13} + c_{14} + c_{15} + m_1 = 0 \ (4) \\ m_2 + m_3 + m_4 + m_5 = 0 \ (5) \\ c_1 + c_5 + c_9 + c_{13} = 0 \ (6) \\ c_2 + c_6 + c_{10} + m_2 = 0 \ (7) \\ c_3 + c_7 + c_{14} + m_3 = 0 \ (8) \\ c_4 + c_{11} + c_{15} + m_4 = 0 \ (9) \\ c_8 + c_{12} + m_1 + m_5 = 0 \ (10) \\ c_1 + c_6 + c_{12} + m_3 = 0 \ (11) \\ c_2 + c_7 + c_{11} + m_1 = 0 \ (12) \\ c_3 + c_8 + c_{13} + m_4 = 0 \ (13) \\ c_4 + c_9 + c_{14} + m_2 = 0 \ (14) \\ c_5 + c_{10} + c_{15} + m_5 = 0 \ (15) \end{cases}$$

Alte bucle în graful Tanner:

$c_1 \rightarrow r_1 \rightarrow c_2 \rightarrow r_7 \rightarrow c_6 \rightarrow r_{11} \rightarrow c_1$ – 6 pasi – 3 cicli in graful Tanner
$c_1 \rightarrow r_1 \rightarrow c_3 \rightarrow r_8 \rightarrow m_3 \rightarrow r_{11} \rightarrow c_1$ – 6 pasi – 3 cicli in graful Tanner

(14)

- Un parametru care afectează semnificativ capacitatea de corecţie a codului este diametrul minim (numărul de tranziţii) al buclelor din graful Tanner, numit „girth”, şi numărul buclelor cu diametru minim, fiecare buclă corespunzând unui bit al cuvântului de cod.
- Trebuie reţinut că acest algoritm nu încearcă să găsească cel mai apropiat cuvânt de cod, faţă de secvenţa recepţionată, ci încearcă să corecteze fiecare bit al secvenţei recepţionate, folosind informaţiile oferite de ceilalţi biţi cu care bitul dat intră în ecuaţiile de control. Sindromul este folosit pentru verificarea corectitudinii cuvântului decodat; deci acest tip de coduri are CRC inclus în structura sa.
- Datorită acestui fapt, numărul de biţi eronaţi după o decodare nereuşită este mai mic decât numărul biţilor eronaţi înaintea decodării (spre deosebire de codurile bloc ciclice sau de cele convoluţionale), dacă cuvântul de cod este recepţionat în integralitatea sa pe un canal cu parametrii aproximativ constanţi.
- Numărul maxim de iteraţii este un parametru care se stabileşte în funcţie de durata permisă pentru decodare de către aplicaţia în care este utilizat codul.
- Varianta în probabilităţi *a posteriori* a algoritmului MP are dezavantajul că necesită un număr foarte mare de înmulţiri. Ea foloseşte pentru decizie o formă echivalentă a raportului de plauzibilitate, adică raportul probabilităţilor *a priori* ca bitul *n* să fi fost “1” sau „0”, dacă s-a recepţionat fazorul *y* :

$$LR_n = \frac{Pl_n^1}{Pl_n^0} > 1 \text{ bitul n este 1} \qquad (25)$$

- Pentru reducerea resurselor de calcul necesare s-a dezvoltat o variantă a acestui algoritm care utilizează logaritmul raportului de plauzibilitate, Log-Likelihood ratio (LLR), al fiecărui bit ca variabilă în algoritm.
- Ea se bazează pe exprimarea logaritmului raportului de plauzibilitate sub forma (25’):

$$LLR = \ln\frac{P_0}{P_1} \Rightarrow e^{LLR} = \frac{P_0}{P_1} \Rightarrow \tanh(\frac{LLR}{2}) = \frac{e^{LLR}-1}{e^{LLR}+1} = P_0 - P_1 \quad \text{- comparată cu 0} \qquad (25')$$

- Studiul acestei variante depăşeşte cadrul cursului de faţă.
- În general codurile LDPC random cu cuvinte de cod de lungime mai mare (la aceeaşi rată) au bucle cu girth-uri minime mai mari în graful Tanner asociat, ceea ce scade probabilitatea de eroare după decodare.
- Codurile LDPC de tip array au valoarea girth-ului minim independentă de lungimea cuvântului de cod

- Lungimea cuvantului de cod trebuie aleasă şi în funcţie de timpul de coerenţă al canalului şi de volumul resurselor timp-frecvenţă (simboluri QAM) ce pot fi alocate pentru transmiterea sa.

1.6. Demaparea soft

- Deoarece semnalele demodulate cu metoda QAM reprezintă coordonatele (afectate de canal) ale unui fazor pe care au fost mapaţi p biţi, probabilităţile *a posteriori* ca dacă s-a recepţionat fazorul f_r, de coordonate I_r şi Q_r, bitul j să aibă valoarea „1" sau „0", se obţin prin operaţia de demapare soft descrisă de:

$$P\left(b_j = 1\right) = \frac{\sum_{k=1}^{2^p} \frac{1}{\sqrt{\left(2\pi\sigma^2\right)}} \exp\left(-\frac{(Ir - I_k)^2 + (Qr - Q_k)^2}{2\sigma^2}\right) \cdot b_{jk}}{\sum_{k=1}^{2^p} \frac{1}{\sqrt{\left(2\pi\sigma^2\right)}} \exp\left(-\frac{(Ir - I_k)^2 + (Qr - Q_k)^2}{2\sigma^2}\right)} \qquad (26)$$

$$P\left(b_j = 0\right) = \frac{\sum_{k=1}^{2^p} \frac{1}{\sqrt{\left(2\pi\sigma^2\right)}} \exp\left(-\frac{(Ir - I_k)^2 + (Qr - Q_k)^2}{2\sigma^2}\right) \cdot \overline{b_{jk}}}{\sum_{k=1}^{2^p} \frac{1}{\sqrt{\left(2\pi\sigma^2\right)}} \exp\left(-\frac{(Ir - I_k)^2 + (Qr - Q_k)^2}{2\sigma^2}\right)}$$

unde I_k si Q_k reprezintă coordonatele celor $N = 2^p$ fazori ai constelaţiei, b_{jk} valoarea logică a bitului cu index j din grupul de biţi mapat pe fazorul k, k = 0,...,N-1,iar b_j bitul cu index j din grupul de biţi 'demapat' de pe fazorul recepţionat (I_r, Q_r), vezi figura 7.

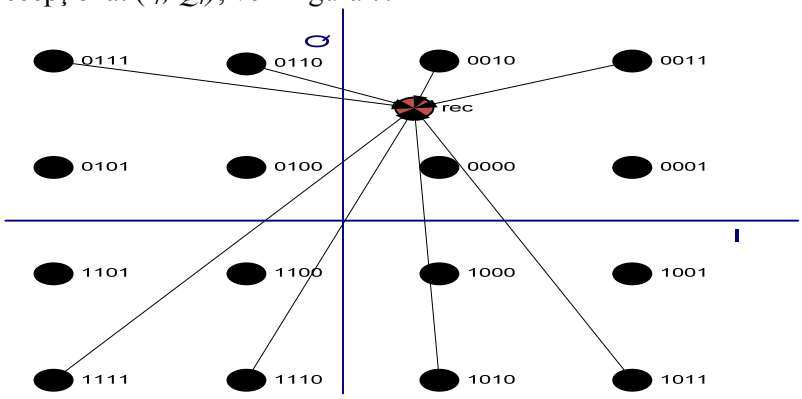


Figura 7 Fazorii din care poate să provină fazorul recepţionat dacă al 3-lea bit este „1"

- Aceste probabilităţi reprezintă P_j^1 şi P_j^0 care sunt introduse ca variabile de intrare pentru fiecare bit în algoritmul MP

- Datorită erorilor de prelucrare, în cazurile practice suma celor două probabilităţi nu va fi egală cu 1, şi aceea valorile obţinute din calcule trebuie normate la suma lor, aşa cum se face şi în ecuaţiile (20) şi (22).

- Valorile probabilităţilor *aposteriori* depind de valoarea SNR (şi implicit a puterii zgomotului σ_2); de aceea, este de preferat ca demaparea tuturor bţilor unui cuvânt de cod să fie făcută la acelaşi SNR, ceea ce impune ca durata de transmisie a cuvântului de cod să fie mai mică decât timpul de coerenţă al canalului si ca banda de frecvenţă a transmisiei sa fie mai mică decât banda de coerenţă a canalului. Neîndeplinirea acestor condiţii conduce la creşterea probabilităţii de eroare la ieşirea decodorului MP care utilizează probabilităţile furnizate de operaţia de soft-demapping.