10. Codurile Reed-Solomon

Introducere

Codurile Reed-Solomon (RS) fac parte din categoria codurilor ciclice, însă sunt coduri nebinare. Spre deosebire de celelalte coduri ciclice, alfabetul codului RS nu este câmpul binar {0, 1}, ci un câmp finit de ordin superior, numit câmp Galois şi care va fi descris în Anexa. În acest fel, cuvintele codului RS nu sunt secvenţe (succesiuni) de biţi, ci de caractere. Aceste caractere pot fi reprezentate, la rândul lor, prin secvenţe binare, însă sunt indivizibile din punct de vedere al codării şi decodării Reed-Solomon.

Structural, cuvintele de cod RS au aceeaşi alcătuire ca şi cele de cod ciclic:

$$v = v_{n-1}v_{n-2} \dots v_1v_0 \qquad v_j \in GF(2^q, p(x)) \quad j = 0 \div n-1 \tag{10.1}$$

unde: v – cuvântul de cod, format din n caractere;
$v_{n-1}v_{n-2} \dots v_m$ – caracterele de informaţie, în număr de k;
$v_{m-1}v_{m-2} \dots v_0$ – caracterele de control, în număr de m;
q – ordinul câmpului;
p(x) – polinomul generator al câmpului GF.

Relaţia de codare are aceeaşi formă ca şi la codurile ciclice:

$$v(x) = i(x)\cdot x^m + \text{rest}\,(i(x)\cdot x^m / g(x)) \tag{10.2}$$

unde g(x) este polinomul generator al codului, al cărui construcţie este prezentată mai jos, iar

$$i(x) = v_{n-1}\cdot x^{k-1} + v_{n-2}\cdot x^{k-2} + \dots + v_{m+1}\cdot x + v_m \tag{10.3}$$

este polinomul de informaţie.

Prin relaţia de codare (10.2) se obţine polinomul ataşat cuvântului de cod, polinom ai cărui coeficienţi sunt tocmai caracterele ce alcătuiesc cuvântul de cod dat de (10.1). Relaţia (10.2) indică, deasemenea, că v(x) este un multiplu al lui g(x).

Codul RS, având parametrii n, k şi m, construit după relaţia (10.2), este capabil să corecteze un număr e_c de caractere eronate, unde:

$$2\cdot e_c = m = n-k \tag{10.4}$$

La decodare, spre deosebire de codurile ciclice, într-un cuvânt de cod RS recepţionat, în vederea corecţiei, este necesară atât localizarea erorii, cât şi stabilirea valorii ei.

Polinomul generator, g(x), al codului

Pentru a se corecta t erori dintr-un cuvânt este necesar a se preciza poziţia fiecăreia precum şi valoarea ei. Dacă ne referim la un cuvânt de lungime n, unde:

$$n = 2^q - 1 \tag{10.5}$$

atunci informaţia necesară pentru a preciza un caracter eronat între cele n este:

$$i_{p1} = -\log_2(1/n) \tag{10.6}$$

Pentru a include şi varianta "cuvânt fără eroare", considerăm că în acest caz se "eronează" un al n+1–lea caracter, fictiv, astfel încât informaţia necesară pentru a preciza poziţia unui caracter eronat (dintre n+1 caractere) este:

$$i_{p1} = \log_2(n+1) \qquad (10.7)$$

Această informaţie poate fi conţinută de un caracter din $GF(2^q)$. Deasemenea şi valoarea erorii, ε, poate să fie orice caracter din $GF(2^q)$:

$$w = v + \varepsilon \qquad (10.8)$$

unde: w –caracterul recepţionat $\in GF(2^q)$;
v –caracterul emis $\in GF(2^q)$;
ε -valoarea erorii $\in GF(2^q)\backslash\{0\}$.

Incluzând şi cazul "eronare a caracterului fictiv", rezultă că ε poate lua orice valoare din $GF(2^q)$, adică 2^q valori posibile. Informaţia necesară pentru a preciza valoarea ei este identică cu cea dată de (10.7).

În concluzie, pentru fiecare eroare ce se doreşte a fi corectată este necesară o informaţie egală cu 2q biţi, adică două caractere din $GF(2^q)$. La t erori sunt necesare 2t caractere (cantitate de informaţie).

Obs. *În fapt condiţia anterioară este una suficientă, cea necesară implică mai puţină informaţie deoarece nu se poate erona un caracter de două ori. De exemplu în cazul a două erori:*

$$i_{p2} = -\log_2 \frac{1}{C_{n+1}^2} = \log_2 \frac{(n+1)\cdot n}{2} = \log_2(n+1) + \log_2 n - 1 \qquad (10.9)$$

iar pentru n suficient de mare $i_{p2} \approx 2i_{p1} - 1$.

Cele 2t caractere de informaţie necesare soluţionării problemei corecţiei se află din 2t ecuaţii, care înseamnă tot atâtea legături (proprietăţi) pentru cuvântul receptionat. Aceste 2t proprietăţi pentru cuvintele de cod RS sunt generate prin relaţia de codare (10.2). Prin această relaţie v(x) devine multiplul lui g(x), ceea ce înseamnă că rădăcinile lui g vor fi şi rădăcini pentru v. Rezultă necesitatea ca g să aibă 2t rădăcini. Aceste rădăcini pot fi oricare dintre cele 2^q elemente ale câmpului $GF(2^q)$. Vom alege pentru g rădăcinile α, α^2, α^3, ... α^{2t}, datorită simplităţii şi simetriei:

$$\begin{aligned} g(x) &= (x+\alpha)(x+\alpha^2)\ldots(x+\alpha^{2t}) \\ &= x^m + g_{m-1}x^{m-1} + \ldots + g_1 x + g_0 \end{aligned} \qquad (10.10)$$

Aşadar cuvântul de cod RS, v, rezultat prin codarea cu ajutorul relaţiei (10.2), în care g este dat de (10.10), are proprietatea că elementele α, α^2, α^3, ... α^{2t} sunt rădăcini pentru polinomul ataşat, v(x).

Codarea codului Reed – Solomon

Codarea se poate face utilizând relaţia (10.2).

Spre exemplu în cazul codului RS cu n = 7, k = 5, t = 1, (avem $GF(2^3)$ si fie $p(x) = x^3 + x + 1$), având polinomul generator:

$$g(x) = (x+\alpha)(x+\alpha^2) = x^2 + \alpha^4 x + \alpha^3 = x^2 + 5x + 4 \qquad (10.11)$$

Ecuaţia împărţirii este:

$$i(x)\cdot x^2 = (2x^4 + 5x^3 + x^2 + x + 5)(x^2 + 5x + 4) + x + 1 \qquad (10.12)$$

unde: -$i(x)\cdot x^2$ este deîmpărţitul ($i = [2\ 1\ 7\ 5\ 4] \Rightarrow i(x) = 2x^4 + x^3 + 7x^2 + 5x + 4$);
-$2x^4 + 5x^3 + x^2 + x + 5$ este câtul;
-$g(x) = x^2 + 5x + 4$ este împărţitorul, iar
- $x + 1$ este restul.

Împărţirea polinomului $i(x)\cdot x^m = i(x)\cdot x^2$ la g(x), cerută pentru codare de relaţia (10.2), este prezentată mai jos:

$$\begin{array}{l|l}
2x^6 + x^5 + 7x^4 + 5x^3 + 4x^2 & x^2 + 5x + 4 \\
\underline{2x^6 + 6x^5 + 5x^4} & \overline{2x^4 + 5x^3 + x^2 + x + 5} \\
/\quad 5x^5 + 4x^4 + 5x^3 + 4x^2 & \\
\quad \underline{5x^5 + 2x^4 + x^3} & \\
\quad /\quad x^4 + 6x^3 + 4x^2 & \\
\quad\quad \underline{x^4 + 5x^3 + 4x^2} & \\
\quad\quad /\quad x^3 & \\
\quad\quad\quad \underline{x^3 + 5x^2 + 4x} & \\
\quad\quad\quad /\quad 5x^2 + 4x & \\
\quad\quad\quad\quad \underline{5x^2 + 2x + 1} & \\
\quad\quad\quad\quad /\quad\quad x + 1 &
\end{array}$$

Cuvântul de cod, conform relaţiei (10.2) rezultă:

$$\begin{aligned} v(x) &= i(x)\cdot x^2 + x + 1 \\ &= 2x^6 + x^5 + 7x^4 + 5x^3 + 4x^2 + x + 1 \end{aligned} \qquad (10.13)$$

Decodarea codului Reed-Solomon

Decodarea codului RS poate fi făcută atât în timp cât şi în frecvenţă.

Decodarea codului RS-corector de erori multiple.

Decodarea va fi sistematizată sub forma unor paşi algoritmici. La fiecare pas se va prezenta operaţia necesară a fi executată, scopul urmărit cât şi argumentările necesare. Fie aşadar codul RS corector de t erori.

Pasul I

Conform relaţiei (10.10) g(x) este un polinom de grad $m = 2t > 2$. Prin construcţie, cuvintele de cod RS au proprietatea că polinoamele ataşate lor sunt divizibile cu g(x), adică elementele câmpului $GF(2^2)$ α, α^2, …, α^{2t} sunt rădăcini atât pentru g(x) cât şi pentru orice cuvânt de cod v(x):

$$\begin{cases} g(\alpha^j) = 0 \\ v(\alpha^j) = 0 \qquad j = 1 \div 2t \end{cases} \qquad (10.14)$$

Aceste proprietăţi constituie şi punctul de plecare în decodare. Presupunând că w este un cuvânt recepţionat:

$$w = v + \varepsilon \quad \text{sau} \quad w(x) = v(x) + \varepsilon(x) \qquad (10.15)$$

vom calcula 2t coeficienţi, numiţi coeficienţi sindrom, Sj, în forma:

$$S_j = w(\alpha^j) = v(\alpha^j) + \varepsilon(\alpha^j) = \varepsilon(\alpha^j) \qquad j=1\div 2t \tag{10.16}$$

Evident că dacă nu există erori $S_j = 0$. În acest caz se trece la pasul VI. Evident concluzia poate fi eronată. Un exemplu în argumentarea acestei afirmaţii este situaţia: ε = cuvânt de cod. Dar în acest caz numărul erorilor depăşeşte puterea de corecţie de t erori.

Pasul II

Dacă există erori în limitele corectabile (numărul erorilor este mai mic sau egal cu t) atunci există coeficienţi sindrom diferiţi de zero. Fie cuvântul eroare în forma:

$$\varepsilon(x) = \sum_{i=1}^{t} \alpha^{r_i} \cdot x^{k_i} \tag{10.17}$$

unde:

$$Y_i = \alpha^{r_i} \tag{10.18}$$

reprezintă valoarea erorii $r_i \in \{0, 1, 2, \ldots n-1\}$, iar:

$$X_i = \alpha^{k_i} \tag{10.19}$$

reprezintă locatorul erorii $k_i \in \{0, 1, 2, \ldots, n-1\}$. Cu aceste notaţii coeficienţii sindrom au expresiile:

$$S_j = \sum_{i=1}^{t} Y_i \cdot \left(\alpha^j\right)^{k_i} = \sum_{i=1}^{t} Y_i \cdot X_i^j \qquad , \qquad j = 1\div 2t \tag{10.20}$$

Ecuaţiile (10.20) reprezintă un sistem de 2t ecuaţii cu 2t necunoscute: t locatori ai erorilor X_i şi t valori pentru respectivele erori Y_i.

Rezolvarea acestui sistem de ecuaţii se va face în mai multe etape. La pasul prezent se vor calcula coeficienţii polinomului σ(x) ai cărui rădăcini sunt locatorii erorilor:

$$\sigma(x) = \sum_{i=1}^{t} (x + X_i) = x^t + \sigma_1 \cdot x^{t-1} + \ldots + \sigma_{t-1} \cdot x + \sigma_t \tag{10.21}$$

Pentru că Xi , $1 \le i \le t$ este o rădăcină a lui σ(x) putem scrie:

$$X_i^t + \sigma_1 \cdot X_i^{t-1} + \ldots + \sigma_{t-1} \cdot X_i + \sigma_t = 0\,, \quad i = 1\div t \tag{10.22}$$

Înmulţind ecuaţiile (10.22) pe rând cu $X_i^k Y_i$ şi sumându-le obţinem ecuaţia:

$$\begin{aligned} &\sum_{i=1}^{t} Y_i \cdot X_i^{t+k} + \sigma_1 \cdot \sum_{i=1}^{t} Y_i X_i^{t+k-1} + \ldots + \sigma_{t-1} \cdot \sum_{i=1}^{t} Y_i \cdot X_i^{k+1} + \\ &+ \sigma_t \cdot \sum_{i=1}^{t} Y_i \cdot X_i^k = 0 \end{aligned} \tag{10.23}$$

sau, ţinând cont de (10.20) pentru k luând valorile 1, 2, ..., t:

$$S_{t+k} + \sigma_1 \cdot S_{t+k-1} + \ldots + \sigma_{t-1} \cdot S_{k-1} + \sigma_t \cdot S_k = 0 , \quad k = 1 \div t \tag{10.24}$$

Ecuaţiile (10.24) reprezintă un sistem de t ecuaţii cu t necunoscute (coeficienţii σ_i) a cărui rezolvare constituie obiectivul acestui pas algoritmic. Ecuaţiile (10.24) pot fi puse sub forma compactă:

$$A_s \cdot \sigma = B_s \tag{10.25}$$

unde:

$$A_s = \begin{bmatrix} S_t & S_{t-1} & \ldots & S_1 \\ S_{t+1} & S_t & \ldots & S_2 \\ \ldots & \ldots & \ldots & \ldots \\ S_{2t-1} & S_{2t-2} & \ldots & S_t \end{bmatrix}; \quad \sigma = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \ldots \\ \sigma_t \end{bmatrix}; \quad B_s = \begin{bmatrix} S_{t+1} \\ S_{t+2} \\ \ldots \\ S_{2t} \end{bmatrix} \tag{10.26}$$

Calculând inversa matricii A_s găsim soluţia sistemului (10.24) în forma:

$$\sigma = A_s^{-1} \cdot B_s \tag{10.27}$$

Obs: *Toate calculele trebuiesc făcute în câmpul GF(2²), atât coeficienţii sindrom, S_j, cât şi coeficienţii σ fiind elemente ale respectivului câmp.*

În rezolvarea ecuaţiei (10.25) pot apărea trei situaţii:

1° rangul matricii As este e < t şi este egal cu al matricii [AsBs]. În acest caz numărul de erori este e şi din rezolvarea ec (10.25) rezultă un număr e de coeficienţi σ_I nenuli Rezolvarea ecuaţiei (10.25) presupune restrângerea sistemului (10.24) la un număr e < t de ecuaţii cu e necunoscute, rezolvabil.

2° rangul matricii As este t. În acest caz există As^{-1} iar ecuaţia (10.25) are soluţie dată prin relaţia (10.27).Se vor găsi t erori în acest caz.

3° rangul matricii As este e` <t şi este mai mic decât al matricii [AsBs]. O astfel de situaţie este posibil să apară dacă numărul erorilor depăşeşte t. În acest caz se semnalează prezenţa erorilor în număr necorectabil. Funcţie de aplicaţie se va abandona cuvântul în cauză sau se va cere retransmisia sa.

Pasul III

Ecuaţia (10.22) se poate rescrie în forma:

$$\sum_{j=1}^{t} \sigma_j \cdot X_i^{-j} = 1 \tag{10.28}$$

Ştiind că X_i este de forma $X_i = \alpha^{k_i}$ unde k_i indică rangul pe care îl ocupă eroarea (ex: k_i = n-1 este prima poziţie) se vor putea afla locatorii erorilor printr-o operaţie de căutare:

$$\sum_{j=1}^{t} \sigma_j \cdot \alpha^{k \cdot j} = 1 \quad ? \qquad k = 1, 2, \ldots, n \tag{10.29}$$

Acei k pentru care (10.29) este o identitate, indică prezenţa erorii pe poziţia:

$$r = n - k \tag{10.30}$$

Obs: *Înlocuind pe:*

$$X_r = \alpha^r \tag{10.31}$$

în (10.28) şi utilizând identitatea $\alpha^n = 1$ *obţinem:*

$$\sum_{j=1}^{t} \sigma_j \cdot \alpha^{-j \cdot r} = \sum_{j=1}^{t} \sigma_j \cdot \alpha^{n \cdot j} \alpha^{-j \cdot r} = \sum_{j=1}^{t} \sigma_j \cdot \alpha^{(n-r) \cdot j} = \sum_{j=1}^{t} \sigma_j \cdot \alpha^{k \cdot j} = 1$$

Pasul IV

Dispunând de poziţiile erorilor dispunem implicit de numărul lor. Reţinem că problema are soluţie doar dacă e < t. Cunoscând aşadar e locatori ai erorilor în forma $X_i = \alpha^{k_i}$, i = 1÷e, din sistemul de ecuaţii (10.20) se reţin e ecuaţii în vederea aflării valorilor Y_i pentru cele e erori. Acest sistem este compatibil unic determinat. Rezolvarea sa conduce la aflarea celor e valori necesare Y_i.

Pasul V

Cunoscând atât poziţiile erorilor $X_i = \alpha^i$, i = 1÷e, cât şi valorile lor $Y_i = \alpha^{r_i}$, i=1÷e putem face corecţia caracterelor eronate:

$$v_{k_i} = w_{k_i} + Y_i, \qquad i = 1 \div e \tag{10.32}$$

Pasul VI

Se face selecţia caracterelor de informaţie şi livrarea lor la ieşire.