ACCQ204

1 Les codes en blocs

Définition des codes en blocs

Principe : $k \underset{m}{\text{bits}} \longrightarrow n \underset{c\in\mathcal{C}}{\text{bits}}$ avec $n > k$. On a $\dim(\mathcal{C}) = k$.

Def. Rendement : $R = \frac{k}{n}$.

Not. $C(k, n, d_{\min})$.

Def. Capacité de détection : $t \leqslant d_{\min} - 1$. Capacité de correction : $t \leqslant \lfloor \frac{d_{\min}-1}{2} \rfloor$.

Les codes linéaires en blocs

$\mathcal{C}$ est un sev de $\mathrm{GF}(2)^n$. Alors $d_{\min} = \min_{c\neq 0} w_H(c)$ (poids de Hamming).

Def. Matrice génératrice : $G = [I_{k\times k} \mid P_{k\times(n-k)}] \in \mathfrak{M}_{k\times n}$ sous forme systématique telle que $c = m \cdot G = [m \mid n-k$ bits de parité$]$.

Def. Matrice de parité : $H \in \mathfrak{M}_{(n-k)\times n}$ la matrice génératrice de $\mathcal{C}^{\perp}$, donc $\forall c \in \mathcal{C}, c \cdot H^{\mathsf{T}} = 0$. Sous forme systématique : $H = [-P^{\mathsf{T}} \mid I_{n-k}]$.

Def. Vecteur syndrôme : $s = rH^{\mathsf{T}}$ avec r le mot reçu. Alors $s = 0$ ssi R est un mot de code.

Th (Borne de singleton). *$d_{\min} \leqslant n - k + 1$, d'où la correction d'erreur $2t \leqslant d_{\min} - 1 \leqslant n - k$.*

On effectue alors un décodage par maximum likelyhood : si les éléments de l'alphabet de départ sont équiprobables, on cherche $\max p(y \mid x)$.

Transformations

- Extension : rajouter des bits de parité.
- Allongement : rajouter des bits d'info.
- Perforation : supprimer des bits de parité.
- Raccourcissement : supprimer des bits d'info.
- Augmentation : ajouter des bits d'info sans modifier la longueur.
- Expurgation : supprimer des bits d'info sans modifier la longueur.

2 Les codes cycliques

Def. **Code cyclique** : code linéaire en bloc $\mathcal{C}$ défini sur $\mathrm{GF}(q)$ tel que $\forall c = (c_0, \dots, c_{n-1}) \in \mathcal{C}, (c_{n-1}, c_0, \dots, c_{n-2}) \in \mathcal{C}$.

Représentation polynomiale

Def. $c(X) = c_0 + c_1X + \ldots + c_{n-1}X_{n-1}$

Décalage cyclique : $Xc(X) \pmod{X^n - 1}$.

Def. **Polynôme générateur** : $g(X)$ l'unique mot de code unitaire de degré minimal.

Tous les mots de code sont multiples de $g(X)$, et $g(X) \mid X^n - 1$. Pour écrire $c(X) = g(X)m(X)$ de manière unique, avec m un mot d'info de degré $< k$, il faut $\deg(g(X)) = n - k$.

Th. *$\mathcal{C}$ est un sous-ensemble cyclique de $\mathrm{GF}(q)[X]/X^n - 1$ ssi $\mathcal{C}$ est un idéal de $\mathrm{GF}(q)[X]/X^n - 1$.*

On a le morphisme $\phi\colon p(X) \in \mathrm{GF}(q)[X] \to p(X) \pmod{X^n - 1}$. $\phi^{-1}(\mathcal{C})$ est un idéal de $\mathrm{GF}(q)[X]$, qui est un corps, donc tous ces idéaux sont principaux et il y a existence et unicité de $g(X)$.

Polynôme de parité

Def. Le **polynôme de parité** est $h(X) = \frac{X^n-1}{g(X)}$.

Pour avoir unicité de l'écriture des mots de code : $\deg(h) < k$.

Forme systématique

$c(X) = m(X)g(X)$: pour rendre $m(X)$ visible dans $g(X)$, on mettra ses coefficients dans les plus hauts degrés.

Forme souhaitée : $c(X) = X^{n-k}m(X) + t(X)$.

Division euclidienne : $X^{n-k}m(X) = \underbrace{q(X)g(X)}_{\in\mathcal{C}} + r(X)$. On écrit donc $c(X) = X^{n-k}m(X) - r(X)$.

Polynôme syndrôme

Le mot reçu est $r(X) = c(X) + e(X)$, où e est la représentation polynomiale de l'erreur. Alors $s(X)$ est égal au reste de $r(X)/g(X)$ ou de $e(X)/g(X)$.

Si $\deg(e(X)) < \frac{d_{\min}}{2}$ alors $s(X)$ est unique.

3 Rappels sur les corps finis

Polynôme sur un corps

Tout polynôme $P(X)$ défini sur un corps F peut être factorisé de façon unique en produit de polynômes premiers (irréductible, unitaire, de degré > 1).

Soit un anneau quotient $F[X]/P(X)$. Si $p(X)$ est unitaire, c'est l'ensemble des polynômes de degré inférieur à $P(X)$. C'est un corps ssi $P(X)$ est premier, et c'est alors une extension de F.

Construction de $\mathrm{GF}(p^m)$

Soit P premier dans $\mathrm{GF}(p)[X]$, de degré m. Alors $\mathrm{GF}(p)[X]/P(X)$ est un corps fini à p^m éléments.

Élément primitif

$\alpha \in \mathrm{GF}(p^m)$ est **primitif** si tout élément de $\mathrm{GF}(p^m) \setminus \{0\}$ est une puissance de α. Tout corps fini en possède au moins un.

$(\mathrm{GF}(p^m) \setminus \{0\}, \cdot)$ est un groupe cyclique généré par α.

Def. $P(X)$ est un polynôme **primitif** ssi il annule un élément primitif.

Factorisation de $X^n - 1$, où $n = p^m - 1$

Soit $\beta \in \mathrm{GF}(p^m) \setminus \{0\}$, d'ordre r. Alors $\beta^{p^m-1} = (\beta^r)^{\frac{p^m-1}{r}} = 1$ donc β est racine de $X^n - 1$ (r divise l'ordre du groupe).

$X^n - 1 = \prod_{\beta \in \mathrm{GF}(p^m)\setminus\{0\}} (X - \beta)$ et on veut factoriser dans $\mathrm{GF}(p)$ maintenant.

Polynôme minimal

Def. Le **polynôme minimal** de $\beta \in \mathrm{GF}(p^m)$ est le polynôme de plus petit degré dans $\mathrm{GF}(p)$ qui annule β.

Prop (de Frobenius). $\forall q \in \mathrm{GF}(p^m)[X], \forall a \in \mathbf{N}, q(X)^{p^a} = \left[\sum_{i=0}^{\deg(q)} q_i X^i\right]^{p^a} = \sum_{i=0}^{\deg(q)} q_i^{p^a} X^{ip^a}$.

Th. *Si $f(X)$ est le polynôme minimal de $\beta \in \mathrm{GF}(p^m)$ alors c'est aussi le polynôme minimal de β^p.*

Deux éléments de $\mathrm{GF}(p^m)$ sont conjugués s'ils ont le même polynôme minimal. Les conjugués de β sont $\{\beta, \beta^p, \beta^{p^2}, \ldots, \beta^{p^{r-1}}\}$ où $r = \min\{i \in \mathbf{N}^* \mid \beta^{p^i} = \beta\}$.

Le polynôme minimal de β s'écrit $f(X) = (X - \beta)(X - \beta^p) \cdots (X - \beta^{p^{r-1}})$.

4 Codes BCH

Construction d'un code cyclique

Code cyclique primitif : de longueur $n = p^m - 1$ avec p premier.

Th. *Soit $\beta_1, \ldots, \beta_r$ les racines dans $\mathrm{GF}(p^m)$ de $g(X)$, polynômes générateur d'un code cyclique primitif. Alors $c(X) \in \mathrm{GF}(p)[X]$ est un mot de code ssi $\forall i, c(\beta_i) = 0$. De plus $g(X) = \mathrm{ppcm}(f_{\beta_1}(X), \ldots, f_{\beta_r}(X))$ où les f_i sont les polynômes minimaux.*

On obtient $g(X)$ en choisissant les racines. On a alors k via $\deg(g(X)) = n - k$.

Les zéros du code sont les i tels que α^i est racine de $g(X)$, avec α un élément primitif.

Les codes BCH

Def. **Code BCH** : code cyclique ayant $2t$ zéros consécutifs (qui corrige t erreurs, $d_{\min} \geqslant 2t + 1$).

Construction d'un code BCH, avec $n = p^m - 1$, p premier :

1) choisir $p(X)$ premier, de degré m sur $\mathrm{GF}(p)[X] \longrightarrow \mathrm{GF}(p^m)$,
2) calculer les polynômes minimaux des α^i pour $1 \leqslant i \leqslant 2t$,
3) calculer $g(X) = \mathrm{ppcm}((f_{\alpha^i})_{i=1,\ldots,2t})$.

Décodage par calcul du syndrôme

On a $R(X) = c(X) + e(X)$. Le polynôme syndrôme $s(X)$ est le reste de $R(X)/g(x)$.

Def. **vecteur syndrôme** : $S = (s_1, \ldots, s_{2t})$ où $s_i = R(\alpha^i)$.

Si $S = O$, $R(x)$ est un mot de code. On a $s_{2i} = R(\alpha^{2i}) = R(\alpha^i)^2 = s_i^2$, donc il y a de la redondance.

Ex (Code de Hamming). $BCH(2^m - 1, 2^m - 1 - m, 3)$. On a $\mathrm{GF}(2^m) = \mathrm{GF}(2)[X]/P(X)$, $g(X) = p(X)$. Le code corrige une erreur. Algorithme de décodage : calculer s_1, si $s_1 = 0$, $e(x) = 0$, sinon $s_1 = \alpha^i \neq 0$ et $e(X) = X^i$.

Ex (Code BCH binaire correcteur de 2 erreurs). $S = (s_1, s_2, s_3, s_4)$, $s_2 = s_1^2$ et $s_4 = s_1^4$. Les composantes non redondantes sont s_1 et s_3.

$e(X) = 0$	pas d'erreur	$s_1 = 0$	$s_3 = 0$
$e(X) = X^i$	erreur en position i	$s_1 = \alpha^i$	$s_3 = \alpha^{3i}$
$e(X) = X^i + X^j$	erreurs en positions i et j	$s_1 = \alpha^i + \alpha^j$	$s_3 = \alpha^{3i} + \alpha^{3j}$

Def. **Polynôme localisateur d'erreurs** : $\Lambda(X)$ dont les racines sont les inverses des positions des erreurs dans $\mathrm{GF}(p^m)[X]$ (corps localisateur d'erreurs).

Dans l'exemple : $\Lambda(X) = (1+\alpha^i X)(1+\alpha^j X) \in \mathrm{GF}(2^m)[X]$ avec 2 erreurs. En développant on obtient $\Lambda(X) = 1 + s_1 X + \left(s_1^2 + \frac{s_3}{s_4}\right) X^2$.

Transformée de Fourier discrète dans les corps finis

- TDF dans $\mathbf{C}^n$: $(h_0, \dots, h_{n-1}) \mapsto (H_0, \dots, H_{n-1})$ avec $H_k = \sum_{l=0}^{n-1} h_l \exp\left(-\frac{2i\pi k}{n} l\right)$.
- TDF dans $\mathrm{GF}(p^m)$: soit $\alpha \in \mathrm{GF}(p^m)$ tel que $\alpha^n = 1$, on a

$$(v_0, \dots, v_{n-1}) \in \mathrm{GF}(p) \longleftrightarrow (V_0, \dots, V_{n-1}) \in \mathrm{GF}(p^m)$$

$$v_i = \frac{1}{n \mod p} \sum_{j=0}^{n-1} V_j \alpha^{-ij} \qquad V_j = \sum_{i=0}^{n-1} v_i \alpha^{ij}$$

Dans le cas $p=2$ et $n = 2^m - 1$, on a donc $v_i = \sum_{j=0}^{n-1} V_j \alpha^{-ij}$ et $V_j = \sum_{i=0}^{n-1} v_i \alpha^{ij}$.
On définit $v(X) = \sum_{i=0}^{n-1} v_i X^i$ et $V(X) = \sum_{j=0}^{n-1} V_j X^j$. On a $v_i = V(\alpha^{-i})$ et $V_j = v(\alpha^j)$.

Produit de convolution cyclique :

$$v_i = h_i u_i \overset{\text{TDF}}{\longrightarrow} V_j = \sum_{i=0}^{n-1} H_i U_{(j-i) \mod n}$$

$$v_i = \sum_{j=0}^{n-1} h_j u_{(j-i) \mod n} \overset{\text{TDF}}{\longrightarrow} V_j = H_j U_j$$

Technique spectrale de décodage des codes BCH

Vecteur reçu : $v(X) = c(X) + e(X)$ ($v_i = c_i + e_i$). Syndromes : $\forall i \in [\![1\,;2t]\!]$, $S_i = v(\alpha^i) = e(\alpha^i) = E_i$.

$$\underset{e_0,\dots,e_{n-1}}{e(X)} \overset{\text{TDF}}{\longrightarrow} \underset{E_0,\dots,E_{n-1}}{E(X)}$$

Or $(E_1, \dots, E_{2t}) = (S_1, \dots, S_{2t})$ est la fenêtre spatiale sur le motif d'erreur.
Supposons ν erreur, avec $\nu \leqslant t$, de positions $i_k, k \in [\![1\,;\nu]\!]$.
On a le polynôme localisateur d'erreurs $\Lambda(X) = \prod_{k=1}^{\nu}(1+\alpha^{i_k}X) = \sum_{i=0}^{n-1} \Lambda_i X^i$. On passe à $\lambda(X) = \sum_{i=0}^{n-1} \lambda_i X^i$, $\lambda_i = \Lambda(\alpha^i) = \sum_{j=0}^{n-1} \Lambda_j \alpha^{-ij} \in \mathrm{GF}(p)$.

Prop. *On a* $\forall i, e_i \lambda_i = 0$.

Démonstration. Si i n'est pas la position d'une erreur, $e_i = 0$, sinon i est la position d'une erreur, donc $e_i \neq 0$ mais $\lambda_i = \Lambda(\alpha^{-1}) = 0$. □

Système fondamental de décodage, t équations avec t inconnues :

$$\begin{cases} \sum_{j=0}^{n-1} \Lambda_j E_{(k-j) \mod n} = 0 \\ \forall i \in [\![t+1\,;2t]\!], \sum_{j=1} t \Lambda_j S_{k-j} = S_k \end{cases}$$

Algorithme de PGZ. Ériture du système sous forme matricielle :

$$\underbrace{\begin{pmatrix} S_1 & S_2 & \dots & S_t \\ S_2 & S_3 & \dots & S_{t+1} \\ \dots & \dots & & \dots \\ S_t & S_{t+1} & \dots & S_{2t} \end{pmatrix}}_{\text{matrice des syndromes}} \cdot \begin{pmatrix} \Lambda_t \\ \Lambda_{t-1} \\ \dots \\ \Lambda_1 \end{pmatrix} = \begin{pmatrix} S_{t+1} \\ S_{t+2} \\ \dots \\ S_{2t} \end{pmatrix}$$

S'il existe t erreurs, la matrice des syndromes est inversible, donc le système est résoluble et il existe une solution unique. Sinon on reprend mais en testant pour une erreur de moins.

5 Codes Reed-Solomon

Cas particulier des codes BCH.

Def. Un code RS correcteur de t erreurs est un code cyclique de longueur $2^m - 1$ ayant uniquement $2t$ zéros consécutifs.

Ex. Avec $\{1, 2, 3, 4\}$, $g(X) = (X - \alpha)(X - \alpha^2)(X - \alpha^3)(X - \alpha^4) \in \mathrm{GF}(2^m)[X]$.

Pour les RS, le corps des symboles et le corps localisateur d'erreurs sont les mêmes. Se sont des codes non binaires.

On a $\deg(g(X)) = 2t$, $g(X) = (X - \alpha) \cdots (X - \alpha^{2t})$ et $n - k = 2t$, d'où $d_{\min} = 2t + 1$ (code à distance maximale).

Ex. On prend $RS(15, 11, 5)$, sur $\mathrm{GF}(16) \simeq \mathrm{GF}(2)[X]/1 + X + X^4$, avec $n = 2^4 - 1$.

Il corrige 2 erreurs : $g(X) = X^4 + \alpha^{13}X^3 + \alpha^6 X^2 + \alpha^3 X + \alpha^{10} \in \mathrm{GF}(16)[X]$. On a $n - k = 4$, donc $k = 11$ et $d_{\min} = 5$.

Ex. DVB : $RS(204, 188)$ (c'est un RS raccourci, $204 \neq 2^m - 1$). La chaîne de transmission (le code source) impose d'utiliser 188 octets, donc sur $\mathrm{GF}(2^8) = \mathrm{GF}(256) \simeq \mathrm{GF}(2)[X]/X^8 + X^6 + X^3 + X^2 + 1$.

Il corrige 8 erreurs : $2t = 16$ et l'on a $g(X) = (X+1)(X+\alpha) \cdots (X+\alpha^{15})$. Alors $\deg(g(X)) = 16 \implies k = 239$ mais on veut $k = 188$. Donc on raccourcit le code : on rajoute des zéros pour le codage, on les enlève pour la transmission et on les rajoute pour le décodage.

Algorithme d'Euclide

Les zéros du code sont $\{0, \ldots, 2t-1\}$. On a $\forall i \in [\![0\,; n-1]\!]$, $\Lambda(\alpha^{-i})E(\alpha^{-i}) = \lambda_i e_i = 0$. Donc $\Lambda(X)E(X) = 0 \mod X^n - 1$, d'où $\Lambda(X)E(X) = \Omega(X)(X^n - 1)$.

Il vient $[\Lambda(X)E(X) = \Omega(X)(X^n - 1)] \mod X^{2t}$, puis $\Lambda(X)[E(X) \mod X^{2t}] = \Omega(X) \mod X^{2t}$ et $\Lambda(X)S(X) = \Omega(X) \mod X^{2t}$. Donc $\Lambda(X)S(X) + q(X)X^{2t} = \Omega(X)$, $\Omega(X)$ est un diviseur commun de $S(X)$ et X^{2t}.

Dans l'algorithe d'Euclide de base on calcule $\mathrm{pgcd}(a(X), b(X))$, en supposant $\deg(b) \leqslant \deg(a)$, par :

$$\begin{aligned} a(X) &= b(X)q_1(X) + r_1(X) \\ b(x) &= r_1(X)q_2(X) + r_2(X) \\ &\cdots\cdots \\ r_i(x) &= r_{i+1}(X)q_{i+2}(X) + r_{i+2}(X) \end{aligned}$$

et on s'arrête dès que le degré d'un reste est nul.

Dans la version généralisée on a

$$f_i(X)a(X) + g_i(X)b(X) = r_i(X)$$

avec

$$\begin{array}{lll} f_i(X) = f_{i-2}(X) + f_{i-1}(X)q_i(X) & & \\ g_i(X) = g_{i-2}(X) + g_{i-1}(X)q_i(X) & & \\ f_{-1}(X) = 1 & f_0(X) = 0 & f_1(X) = 1 \\ g_{-1}(X) = 0 & g_0(X) = 1 & g_1(X) = q_1(X) \end{array}$$

L'algorithme d'Euclide généralisé appliqué à $S(X), X^{2t}$ donne donc $\Lambda(X)$ et $\Omega(X)$. On arrête l'algorithme dès que $r_i(X)$ est de degré $\leqslant t - 1$.

Algorithme de Forney pour connaître les valeurs des erreurs

Soit $\Omega(X)$ le polynôme évaluateur d'erreur. En dérivant $\Lambda(X)E(X) = \Omega(X)(X^n - 1)$ on obtient

$$\Lambda'(X)E(X) + E'(X)\Lambda(X) = \Omega'(X)(X^n - 1) + n\Omega(X)X^{n-1}$$

Pour i une position d'erreur, $\lambda_i = \Lambda(\alpha^{-i}) = 0$ et $e_i = E(\alpha^{-i})$. Donc $\Lambda'(\alpha^{-i})E(\alpha^{-i}) = n\Omega(\alpha^{-i})\alpha^{n-1-i}$.