Université de Rennes 1
Préparation à l'agrégation de mathématiques
Auteur du document :
M. Coste, A. Paugam, R. Quarez

Codes correcteurs

M. Coste, A. Paugam, R. Quarez

juin 2002

Il faut distinguer les codes correcteurs d'erreurs de la cryptographie. Les codes correcteurs d'erreur servent à protéger l'information d'erreurs de transmission ou de stockage.

On peut trouver dans ce texte, avec des références bibliographiques précises, des idées d'exposés ou d'applications pour plusieurs leçons portant sur l'algèbre linéaire ou les polynômes. (Les titres des leçons sont ceux du rapport du jury 2001, en algèbre et géométrie ou en calcul scientifique.)

- Corps finis. Applications.
- Applications des congruences ou des corps finis aux thèmes du programme.
- Polynômes irréductibles à une indéterminée. Corps de rupture. Exemples et applications. (Extension de $\mathbb{F}_2$ engendrée par les racines primitives n-èmes de l'unité pour les calculs sur les codes BCH)
- Racines des polynômes à une indéterminée. Relations entre les coefficients et les racines d'un polynôme. Exemples et applications. (Formules de Newton, décodage des codes BCH)
- Idéaux d'un anneau commutatif unitaire. Exemples et applications. (Codes cycliques)
- Dimension d'un espace vectoriel. Rang. Exemples et applications. (Borne de Singleton, détermination du nombre d'erreurs pour les codes BCH)
- Opérations élémentaires sur les lignes et les colonnes d'une matrice. Résolution d'un système d'équations linéaires. Applications. (Code sous forme systématique, calcul du polynôme localisateur d'erreurs pour les codes BCH)
- Déterminant. Applications. (Vandermonde pour la distance minimum d'un code cyclique)
- PGCD, PPCM : méthodes de calcul et applications dans les thèmes du programme. (Décodage d'un code BCH par l'algorithme d'Euclide)

Les codes considérés dans ce texte sont tous binaires (sur le corps à deux éléments $\mathbb{F}_2$). Dans les références, on considère souvent plus généralement des codes sur un corps fini $\mathbb{F}_q$. La référence la plus utile est sans doute le livre de Demazure [Dem].

1 Codes en blocs, distance de Hamming

Les messages transmis sont supposés découpés en blocs (ou *mots*) de longueur n écrits avec l'alphabet $\{0,1\}$. Un code est un sous-ensemble C de l'ensemble $\{0,1\}^n$ de tous les mots possibles. On dit que n est la *longueur* de C.

La *distance de Hamming* entre deux mots $x = (x_1, \ldots, x_n)$ et $y = (y_1, \ldots, y_n)$, que l'on notera $d(x, y)$, est le nombre d'indices i tels que $x_i \neq y_i$. C'est bien une distance sur $\{0, 1\}^n$. La *distance minimum* du code C est le minimum des $d(x, y)$ pour x et y des mots différents de C (on suppose que C a au moins 2 mots !). On la notera toujours d.

Le mot $c \in C$ est émis et, après d'éventuelles erreurs de transmission, le mot $r \in \{0, 1\}^n$ est reçu. On décode le mot r selon le principe du maximum de vraisemblance, c.-à-d. qu'on le décode comme un mot de C à distance minimum de r. On dit que C est *t-correcteur* (ou corrige t erreurs) quand toute erreur portant sur au plus t bits est corrigée correctement. On voit donc que le code C est t-correcteur si et seulement si les boules fermées (dans $\{0, 1\}^n$ muni de la distance de Hamming) de centres les éléments de C et de rayon t sont disjointes, ou encore si et seulement si la distance minimum d de C vérifie $d \geq 2t + 1$.

Pour pouvoir travailler avec des codes, il faut mettre plus de structure.

2 Codes linéaires

2.1 Définitions

La première structure que nous ajoutons est la structure *linéaire*. On note $\mathbb{F}_2$ le corps à deux éléments 0 et 1. Les mots de longueurs n sont les éléments de ${\mathbb{F}_2}^n$, que l'on écrira comme des vecteurs lignes. Un *code linéaire de longueur n* est un sous-espace vectoriel $C \subset {\mathbb{F}_2}^n$. La lettre k désignera toujours la *dimension* de C (comme espace vectoriel). Le nombre de mots du code C est 2^k.

Le *poids* d'un mot $x = (x_1, \ldots, x_n) \in {\mathbb{F}_2}^n$, noté $w(x)$, est le nombre d'indices i tels que $x_i \neq 0$. Comme $d(x, y) = w(x - y)$, la distance minimum d d'un code linéaire C est le minimum des poids $w(x)$ pour $x \in C$ non nul. (On suppose que C n'est pas le code nul.)

On regroupe les trois paramètres n, k et d d'un code linéaire C (non nul, c.-à-d. avec $k > 0$) en disant que C est de type (n, k, d). On a toujours $d + k \leq n + 1$ (borne de Singleton).

Exercice 1. Vérifiez cette inégalité en intersectant C avec le sous espace de ${\mathbb{F}_2}^n$ formé des mots dont les $k - 1$ premières coordonnées sont nulles. (On peut voir [Dem] Proposition 9.1.)

La borne de Singleton quantifie le fait qu'on ne peut pas avoir à la fois le beurre (une capacité de correction importante) et l'argent du beurre (un nombre de mots de code important), pour une longueur n fixée.

Exercice 2. Si C est un code linéaire (de type (n, k, d)), on définit le *code étendu* $\overline{C}$ comme le code formé des mots $(x_1, \ldots, x_{n+1}) \in {\mathbb{F}_2}^n$ tels que $(x_1, \ldots, x_n) \in C$ et $\sum_{i=1}^{n+1} x_i = 0$. Quel est le type de $\overline{C}$?

2.2 Matrice génératrice

On peut se donner un sous-espace vectoriel (et donc un code) par une base. Soit C un code linéaire. Une *matrice génératrice* de C est une matrice dont les lignes forment une base de C. Une matrice génératrice G est donc de taille

$k \times n$ et de rang k. Si m est un vecteur ligne de ${\mathbb{F}_2}^k$, le produit mG est un mot du code C et l'application $m \mapsto mG$ est un isomorphisme de ${\mathbb{F}_2}^k$ sur C (que l'on peut voir comme une opération de codage). Si la matrice G est de la forme $(I_k\,,\,P)$, on dit que le codage est *systématique*. Les k premiers bits d'un mot de code portent l'information (on y recopie le vecteur de ${\mathbb{F}_2}^k$), les $n-k$ suivants sont de la redondance.

Exercice 3. On dit que deux codes linéaires de même longueur sont *équivalents* si l'un s'obtient à partir de l'autre par une permutation des coordonnées. Vérifier que deux codes équivalents ont même type. Montrer que tout code est équivalent à un code donné par un codage systématique.

2.3 Matrice de contrôle

On peut ausi se donner un sous-espace vectoriel par un système d'équations indépendantes. Soit C un code linéaire. Une *matrice de contrôle* de C est la matrice d'un système d'équations linéaires homogènes indépendantes dont l'espace des solutions est C. Autrement dit, une matrice de contrôle H est de taille $(n-k) \times n$ et de rang $n-k$, et $C = \{x \in {\mathbb{F}_2}^n \;;\; H{}^t x = 0\}$.

Si C est donné sous forme systématique par la matrice génératrice $G = (I_k\,,\,P)$, alors on peut prendre comme matrice de contrôle $H = (-{}^tP\,,\,I_{n-k})$ (le signe $-$ est superflu en caractéristique 2).

Supposons que $c \in C$ est le mot du code envoyé et $r \in {\mathbb{F}_2}^n$ le mot reçu. La différence $e = r - c$ est le *vecteur d'erreur*. Son poids $w(e)$ est le nombre de bits erronés dans le mot reçu. Soit H une matrice de contrôle de C. Le *syndrome* du mot reçu r est le vecteur $s \in {\mathbb{F}_2}^{n-k}$ défini par ${}^ts = H{}^tr = H{}^te$. Le syndrome est nul si et seulement si $r \in C$. Le syndrome définit un isomorphisme du quotient ${\mathbb{F}_2}^n/C$ sur ${\mathbb{F}_2}^{n-k}$. Si le syndrome est non nul, on corrige le mot reçu r en appliquant le principe du maximum de vraisemblance : on soustrait à r un mot de poids minimum dans sa classe modulo C, c.-à-d. un mot de poids minimum parmi ceux ayant même syndrome que r. Dans le cas où $w(e)$ est strictement inférieur à $d/2$, alors e est l'unique mot de poids minimum dans la classe de r modulo C et on récupère bien le mot de code émis.

Exercice 4. Soit H une matrice de contrôle du code C. Montrer que la distance minimum d de C est caractérisée par les propriétés suivantes :
1) $d-1$ colonnes de H sont toujours linéairement indépendantes.
2) Il y a un système de d colonnes de H qui est lié.

2.4 Codes de Hamming

Le code de Hamming de longueur 7 est donné par la matrice de contrôle

$$H = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} .$$

La matrice H est de rang 3. La dimension du code est donc 4.

Exercice 5. Donner une matrice génératrice de ce code. Peut-on donner ce code par un codage systématique ?

Les colonnes de H sont, en notation binaire, tous les entiers de 1 à 7. Donc (voir l'exercice 4) la distance minimum du code est 3 : il permet de corriger une erreur. On peut décrire facilement un procédé de décodage : si le syndrome est non nul, on l'interprète comme un entier i en binaire et on corrige le i-ème bit du mot reçu. Ceci marche bien car, si le vecteur d'erreur e est de poids 1 avec un 1 en i-ème position, alors $H{}^t e$ est bien la i-ème colonne de H. (On peut voir [Chi], pages 107 et suivantes.)

La code que l'on vient de décrire est de type $(7, 4, 3)$. On peut généraliser cet exemple en prenant comme matrice de contrôle la matrice à r lignes dont les colonnes sont tous les entiers de 1 à $2^r - 1$ en binaire. On obtient alors un code de Hamming de type $(2^r - 1, 2^r - 1 - r, 3)$.

Exercice 6. Montrer que tout code de type $(2^r - 1, 2^r - 1 - r, d)$ avec $d \geq 3$ est équivalent au code de Hamming décrit ci-dessus. (Quelles peuvent être les colonnes d'une matrice de contrôle de parité ?)

Les codes de Hamming ont une propriété remarquable : ils sont *parfaits*. On dit qu'un code de distance minimum $d = 2t + 1$ est parfait quand tout mot est à distance $\leq t$ d'un unique mot du code, autrement dit si et seulement si les boules fermées de rayon t et de centres les mots du code forment une partition de l'ensemble des mots. (On a un empilement « parfait » de ces boules, sans espace entre elles.)

Exercice 7. Montrer que les codes de Hamming sont parfaits.

Il paraît que le code de Hamming étendu de type $(64, 57, 4)$ est utilisé pour les mémoires d'ordinateur.

3 Codes cycliques, codes BCH

On met maintenant encore plus de structure sur l'espace des mots : il s'agit maintenant d'une structure d'algèbre sur $\mathbb{F}_2$.

3.1 Codes cycliques

Un code linéaire $C \subset {\mathbb{F}_2}^n$ est dit *cyclique* quand il est stable par l'automorphisme de décalage cyclique

$$\begin{array}{rcl} T : {\mathbb{F}_2}^n & \longrightarrow & {\mathbb{F}_2}^n \\ (x_1, \ldots, x_n) & \longmapsto & (x_2, \ldots, x_n, x_1) \ . \end{array}$$

On identifie ${\mathbb{F}_2}^n$ à l'algèbre $\mathbb{F}_2[X]/(X^n - 1)$ par

$$(x_1, \ldots, x_n) \longmapsto x_1 X^{n-1} + \cdots + x_{n-1} X + x_n \ .$$

(Note : Demazure [Dem] écrit le polynôme associé à un mot de code dans le sens des puissances croissantes.)

On désigne ici par la même lettre l'indéterminée X et son image dans le quotient $\mathbb{F}_2[X]/(X^n - 1)$. Remarquer que tout polynôme de $\mathbb{F}_2[X]$ est congru modulo $X^n - 1$ à un unique polynôme de degré $< n$ (son reste dans la division euclidienne par $X^n - 1$).

L'endomorphisme T, modulo cette identification, est l'endomorphisme de multiplication par X. Par définition, un code cyclique est un sous-espace vectoriel stable par multiplication par X, et donc par n'importe quel polynôme en X. Donc, un code linéaire C de longueur n est cyclique si et seulement si C est un idéal de $\mathbb{F}_2[X]/(X^n-1)$.

L'homorphisme de passage au quotient $\mathbb{F}_2[X] \to \mathbb{F}_2[X]/(X^n-1)$ induit une bijection entre l'ensemble des idéaux de $\mathbb{F}_2[X]/(X^n-1)$ et l'ensemble des idéaux de $\mathbb{F}_2[X]$ qui contiennent (X^n-1). Puisque $\mathbb{F}_2[X]$ est principal, ce sont exactement les idéaux engendrés par les diviseurs (que l'on prend unitaires pour assurer l'unicité) de X^n-1 dans $\mathbb{F}_2[X]$. Le diviseur unitaire g de X^n-1 ainsi associé à un code cyclique C s'appelle le *polynôme générateur de* C. Si $g \neq X^n-1$ (dans le cas contraire C est nul), le code C est engendré (comme espace vectoriel sur $\mathbb{F}_2$) par $g, Xg, \ldots, X^{n-1-\deg(g)}g$. La dimension de C est dans tous les cas $k = n - \deg(g)$.

Le procédé de codage systématique ${\mathbb{F}_2}^k \to C$ d'un code cyclique de polynôme générateur g est donné par la division euclidienne par g : le vecteur $(x_1, \ldots, x_k) \in {\mathbb{F}_2}^k$ est codé par le polynôme $c = c_I - c_R$, où $c_I = x_1X^{n-1} + \cdots + x_kX^{n-k}$, et c_R (de degré $< n-k$) est le reste de la division euclidienne de c_I par g. (Le polynôme c_I porte l'information, et c_R la redondance.)

3.2 Zéros d'un code cyclique

On a vu qu'un code cyclique est engendré (en tant qu'idéal) par un diviseur unitaire g de X^n-1. On suppose à partir de maintenant que n est premier avec la caractéristique de $\mathbb{F}_2$, c.-à-d. impair. Cette hypothèse entraîne que le polynôme X^n-1 a n racines *distinctes* dans son corps de décomposition sur $\mathbb{F}_2$ (il y a bien n racines n-èmes de l'unité). Notons K ce corps de décomposition, c'est à dire le corps engendré par les racines n-èmes de l'unité sur $\mathbb{F}_2$. On fait le choix d'une racine primitive n-ème de l'unité dans K, que nous noterons α. Le polynôme minimal P de α sur $\mathbb{F}_2$ a pour degré l'ordre r de 2 dans le groupe multiplicatif $(\mathbb{Z}/n\mathbb{Z})^*$, et ses racines sont $\alpha, \alpha^2, \alpha^4, \ldots, \alpha^{2^{r-1}}$. On a $K = \mathbb{F}_2[\alpha] = \mathbb{F}_2[X]/P = \mathbb{F}_{2^r}$. Le polynôme cyclotomique Φ_n factorise sur $\mathbb{F}_2$ en produit de facteurs irréductibles de degré r (par exemple $\Phi_{15} = X^8 - X^7 + X^5 - X^4 + X^3 - X + 1$ se factorise en $(X^4+X+1)(X^4+X^3+1)$ sur $\mathbb{F}_2$), et P est un de ces facteurs. Il est fortement conseillé de revoir ce qui concerne les racines de l'unités sur les corps finis (par exemple [Dem] 8.2, 8.3 ou [Esc] 14.7, 14.8).

Le polynôme générateur g va être déterminé par ses racines dans K, qui forment un sous-ensemble de l'ensemble $\{1, \alpha, \alpha^2, \ldots, \alpha^{n-1}\}$ des racines n-èmes de l'unité (les *zéros du code*). Soit Σ un sous-ensemble de $\mathbb{Z}/n\mathbb{Z}$. Le polynôme $g_\Sigma = \prod_{i\in\Sigma}(X-\alpha^i)$ est un diviseur de X^n-1 à coefficients dans $\mathbb{F}_2$ si et seulement si Σ est stable par multiplication par 2 (se souvenir que $\mathbb{F}_2$ est l'ensemble des éléments de K laissés fixes par l'élévation au carré ou voir [Dem], proposition 9.3). En conclusion, on a une bijection entre les codes cycliques de longueur n et les sous-ensembles de $\mathbb{Z}/n\mathbb{Z}$ stables par multiplication par 2.

La configuration des racines du polynôme générateur nous renseigne sur la distance minimale du code cyclique. Voici l'énoncé de la Proposition 9.4 de [Dem] (avec les notations ci-dessus)

Proposition 1 *Si Σ contient s entiers consécutifs $a+1, a+2, \ldots, a+s$ modulo n, alors le code cyclique de polynôme générateur g_Σ est nul ou a une distance minimum supérieure ou égale à $s+1$*

La démonstration est une application des propriétés du déterminant de Vandermonde.

3.3 Les codes BCH binaires et leur décodage

3.3.1 Définition

Les codes BCH (Bose-Chaudhuri-Hocquenghem) sont des codes cycliques particuliers. La famille des codes BCH contient les codes de Reed-Solomon qui servent pour la lecture des CD (voir [Dem], p. 238).

Nous ne considérerons ici que des codes BCH binaires primitifs. Leur longueur n est de la forme $n = 2^r - 1$. Alors r est l'ordre de 2 dans le groupe multiplicatif $(\mathbb{Z}/n\mathbb{Z})^*$, et le corps K engendré par les racines n-èmes de l'unité sur $\mathbb{F}_2$ est $\mathbb{F}_{2^r}$. Tous les calculs de décodage vont se faire sur ce corps K. Choisissons une racine primitive n-ème de l'unité α dans K. Concrètement, on se donne α par son polynôme minimal P sur $\mathbb{F}_2$ (un facteur irréductible sur $\mathbb{F}_2$ du polynôme cyclotomique Φ_n, de degré r). Tout élément du groupe multiplicatif K^* s'écrit de manière unique sous la forme α^i avec $0 \leq i < n$, et il s'écrit aussi de manière unique comme combinaison linéaire à coefficients dans $\mathbb{F}_2$ de $1, \alpha, \ldots, \alpha^{r-1}$. On peut voir la table de correspondance entre ces deux représentations pour $K = \mathbb{F}_{16}$, avec α vérifiant $\alpha^4 + \alpha + 1 = 0$, dans [Chi], p. 245 ou dans [Dem], p. 213.

On appelle *code BCH de longueur* $n = 2^r - 1$ *et de distance prescrite* δ (δ entier tel que $0 < \delta \leq n$) le code cyclique de polynôme générateur g_Σ, où Σ est le plus petit sous-ensemble de $\mathbb{Z}/n\mathbb{Z}$ contenant $1, 2, \ldots, \delta - 1$ et stable par multiplication par 2. Autrement dit, un polynôme $c = x_1 X^{n-1} + \cdots + x_n \in \mathbb{F}_2[X]$ appartient à ce code si et seulement si

$$c(\alpha) = c(\alpha^2) = \ldots = c(\alpha^{\delta-1}) = 0 \; .$$

On peut trouver des exemples de codes BCH explicités pour $n = 15$ (avec une racine primitive quinzième de l'unité $\alpha \in \mathbb{F}_{16}$ vérifiant $\alpha^4 + \alpha + 1 = 0$) dans [Dem] pp. 240 et suivantes ou [Chi] pp. 245 et suivantes.

Exercice 8. On considère un code BCH de longueur $2^r - 1$ et de distance prescrite 3. Montrer que son polynôme générateur est le polynôme minimal de α sur $\mathbb{F}_2$. En déduire que la dimension du code est $2^r - 1 - r$. En déduire que ce code est équivalent au code de Hamming de type $(2^r - 1, 2^r - 1 - r, 3)$ (voir l'exercice 6).

Le code du minitel (voir [Esc], exercice 14.3 page 201) est un exemple de code BCH du type considéré dans l'exercice ci-dessus. C'est un code de type (127,120,3).

3.3.2 Polynôme localisateur d'erreurs

D'après la Proposition 1, la distance minimum d'un code BCH est au moins égale à sa distance prescrite. On pourra donc corriger t erreurs avec un code BCH de distance prescrite $2t+1$. Expliquons un procédé de décodage qui permet de faire cette correction.

On envoie un mot de code c, qui vérifie donc

$$c(\alpha) = c(\alpha^2) = \ldots = c(\alpha^{2t}) = 0 \ .$$

Le mot reçu est $u = c+e$. *On suppose que le polynôme d'erreur* e *est de poids* $\nu \leq t$ (on ne cherche pas à corriger plus de t erreurs). On a donc $e = X^{\ell_1} + \cdots + X^{\ell_\nu}$, avec $n-1 \geq \ell_1 > \ldots > \ell_\nu \geq 0$. Le but du décodage est de déterminer les entiers $n - \ell_1, \ldots, n - \ell_\nu$ qui sont les numéros des bits erronés du mot reçu. Posons $\beta_j = \alpha^{\ell_j}$ pour $j = 1, \ldots, \nu$. Les β_j sont des éléments distincts et tous non nuls de K. On introduit le *polynôme localisateur d'erreurs*

$$\sigma(Z) = \prod_{j=1}^{\nu} (1 - \beta_j Z) = 1 + \sigma_1 Z + \sigma_2 Z^2 + \cdots + \sigma_\nu Z^\nu \in K[Z] \ ,$$

qui est le polynôme réciproque du polynôme unitaire de racines β_j. Les σ_i sont, au signe près, les polynômes symétriques élémentaires des β_j. Si on connaît le polynôme localisateur d'erreurs σ, on pourra récupérer les $n - \ell_j$:

Exercice 9. Les entiers $n - \ell_j$ sont les entiers i entre 0 et $n-1$ tels que α^i soit racine de σ.

On peut alors retrouver de manière bête les numéros des bits erronés, en essayant successivement tous les α^i pour $i = 0, \ldots, n-1$ pour voir s'ils annulent σ. Dans ce qui suit, nous nous préoccuperons uniquement de la détermination de σ.

À partir du mot reçu u, on peut calculer

$$S_i = u(\alpha^i) = e(\alpha^i) = \sum_{j=1}^{\nu} (\alpha^i)^{\ell_j} = \sum_{j=1}^{\nu} {\beta_j}^i \quad \text{pour } i = 1, \ldots, 2t \ .$$

On connaît donc les $2t$ premières *sommes de Newton* des β_j, et on veut en déduire les σ_i, c.-à-d. au signe près leurs polynômes symétriques élémentaires. Il est naturel de penser aux formules de Newton. Elles s'écrivent ici, pour tout entier $p \geq 1$:

$$(N_p) \qquad S_p + \sum_{\ell=1}^{p-1} \sigma_\ell S_{p-\ell} + p\sigma_p = 0 \ ,$$

où $\sigma_i = 0$ pour $i > \nu$ (habituellement on voit apparaître des $(-1)^\ell$ dans les formules parce qu'elles sont écrites avec les polynômes symétriques élémentaires). Une manière d'obtenir ces formules de Newton est donnée dans l'exercice suivant (voir aussi [Esc] page 28, par exemple).

Exercice 10. On note $\omega(Z) = -\frac{\mathrm{d}}{\mathrm{d}Z}(\sigma(Z))$. Montrer que le développement en série formelle de la fraction rationelle

$$\frac{\omega(Z)}{\sigma(Z)} = \sum_{j=1}^{\nu} \frac{\beta_j}{1 - \beta_j Z}$$

est $\sum_{i=0}^{+\infty} S_{i+1} Z^i$, où $S_i = \sum_{j=1}^{\nu} {\beta_j}^i$. En déduire les formules de Newton (N_p).

3.3.3 Décodage par résolution d'un système linéaire

En caractéristique 0, les formules de Newton permettent d'obtenir $\sigma_1, \ldots, \sigma_p$ par récurrence en fonction de $S_1, \ldots, S_p$. Ceci est utilisé par exemple dans la méthode de Leverrier pour calculer le polynôme caractéristique d'une matrice (voir [Gan] page 88). La caractéristique 0 sert parce qu'on doit diviser par p pour obtenir σ_p à partir de la formule (N_p). Dans le problème qui nous occupe pour les codes BCH, nous sommes en caractéristique 2 et on doit procéder autrement.

Posons $H_\ell = \begin{pmatrix} S_1 & S_2 & \ldots & S_\ell \\ S_2 & S_3 & \ldots & S_{\ell+1} \\ \vdots & \vdots & \vdots & \vdots \\ S_\ell & S_{\ell+1} & \ldots & S_{2\ell-1} \end{pmatrix}$. On vérifie que

$$H_\nu = V \begin{pmatrix} \beta_1 & & 0 \\ & \ddots & \\ 0 & & \beta_\nu \end{pmatrix} {}^tV \ , \quad \text{où } V = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \beta_1 & \beta_2 & \ldots & \beta_\nu \\ \vdots & \vdots & \vdots & \vdots \\ \beta_1^{\nu-1} & \beta_2^{\nu-1} & \ldots & \beta_\nu^{\nu-1} \end{pmatrix} .$$

Puisque les β_j sont distincts et non nuls, H_ν est inversible. Le rang de H_t est donc $\geq \nu$. Par ailleurs, les formules de Newton montrent que chaque colonne de numéro $\geq \nu + 1$ de H_t est combinaison linéaire des ν colonnes précédentes. Le nombre d'erreurs ν est donc égal au rang de H_{2t}. Une fois ν trouvé, on obtient $\sigma_1, \ldots, \sigma_\nu$ en résolvant le système de Cramer donné par les formules de Newton $(N_{\nu+1})$ à $(N_{2\nu})$:

$$H_\nu \begin{pmatrix} \sigma_\nu \\ \vdots \\ \sigma_1 \end{pmatrix} = \begin{pmatrix} -S_{\nu+1} \\ \vdots \\ -S_{2\nu} \end{pmatrix} .$$

On peut voir cette méthode employée sur des exemples dans [Chi], pages 249 et suivantes, ou voir [PW] pages 178 et suivantes.

3.3.4 Décodage par l'algorithme d'Euclide

Cette méthode pour déterminer le polynôme localisateur d'erreur utilise l'algorithme d'Euclide étendu (celui qui calcule, en même temps que le pgcd, les coefficients d'une identité de Bezout). Pour les détails de ce qui suit, se reporter à [Dem], pages 244-245.

En utilisant les notations et les résultats de l'exercice 10, on obtient l'identité $\omega(Z) = \sigma(Z)(\sum_{i=0}^{+\infty} S_{i+1} Z^i)$ dans l'anneau de séries formelles en Z à coefficients dans K. En posant $S(Z) = \sum_{i=0}^{2t-1} S_{i+1} Z^i$, on déduit

$$(\dagger) \qquad \omega(Z) \equiv \sigma(Z) S(Z) \bmod Z^{2t} .$$

Ici le polynôme S est connu, et on cherche σ et ω. On sait que $\deg(\omega) < \deg(\sigma) \leq t$, $\sigma(0) = 1$ et que les polynômes σ et ω sont premiers entre eux. Il existe une seule solution (σ, ω) de la congruence $(\dagger)$ vérifiant ces conditions, et on la trouve en éxécutant l'algorithme d'Euclide étendu à partir de $P_0 = Z^{2t}$ et $P_1 = S$, en s'arrêtant au premier reste P_i tel que $\deg(P_i) < t$. L'algorithme d'Euclide étendu a calculé en même temps des polynômes A_i et B_i tels que $P_i = A_i Z^{2t} + B_i S$. Alors $B_i(0) \neq 0$ et on a $\sigma(Z) = B_i(Z)/B_i(0)$ et $\omega(Z) = P_i(Z)/B_i(0)$. (Note : il

convient de corriger l'énoncé de la Proposition 9.13 de [Dem] comme on vient de l'indiquer ; de même, la phrase précédant cet énoncé doit être remplacée par « Puisque $\sigma(0) = 1$ par construction, on a $C = B_i(0)$ ».)

Références

[Dem] M. Demazure, Cours d'algèbre, Cassini 1997.

[Esc] J-P. Escofier, Théorie de Galois, Dunod 1997.

[Chi] L. Childs, A Concrete Introduction to Higher Algebra, Springer Verlag.

[Gan] F.R. Gantmacher, Théorie des matrices, tome 1, Dunod.

[PW] O. Papini et J. Wolfman, Algèbre discrète et codes correcteurs, Springer Verlag.

Une feuille Maple mettant en oeuvre le décodage des codes BCH par l'algorithme d'Euclide est disponible sur le site de la préparation à l'agrégation :
`http://agreg-maths.univ-rennes1.fr/`