TP RACINES DE L'UNITE DANS UN CORPS FINI ET CODES BCH

N.B. : *Ce TP est adapté du TP N°15, du même nom, publié dans l'ouvrage Guin et Hausberger, Algèbre I : Groupes, Corps et Théorie de Galois, paru à EDP Sciences. Le remaniement consiste en une simplification des procédures (moins systématiques que celles proposées dans l'ouvrage) et l'usage de SageMath plutôt que Maple, afin de tenir compte des conditions de passation actuelles de l'oral de l'option C de l'Agrégation de Mathématiques.*

On se propose dans ce TP de passer en revue la théorie des polynômes cyclotomiques sur un corps fini $\mathbb{F}_q$. Comme application, on génère des codes BCH construits par définition à partir des polynômes minimaux de puissances de racines primitives de l'unité sur $\mathbb{F}_q$. Puis on offre une initiation à la théorie des codes correcteurs d'erreurs : on expose comment coder et décoder un message (ne pas confondre avec la cryptographie dont le propos est d'envoyer un message secret que seul le destinataire puisse décoder), dans le cas des codes BCH, et on teste expérimentalement la capacité de correction du code et la puissance de l'algorithme de décodage (une variante de l'algorithme d'Euclide étendu due à Berlekamp et Massey). Ces méthodes sont fondamentales dans les technologies de transmission de l'information, d'où de multiples applications dans l'industrie.

Racines de l'unité et polynômes cyclotomiques sur un corps fini $\mathbb{F}_q$

Les polynômes cyclotomiques sont par définition les

$$\Phi_n(x) = \prod_{\zeta \in \mathcal{P}_n} (x - \zeta),$$

où $\mathcal{P}_n \subset \mathbb{C}$ désigne l'ensemble des racines primitive n-ièmes de l'unité, constitué des $\zeta_k = e^{\frac{2ik\pi}{n}}$, $\mathrm{pgcd}(k, n) = 1$. Il y en a $\varphi(n)$, où l'on a noté φ la fonction indicatrice d'Euler. La relation

$$x^n - 1 = \prod_{d|n} \Phi_d(x) \tag{1}$$

permet de calculer les Φ_d par récurrence et montre que ces derniers sont à coefficients entiers. On démontre que les Φ_d sont irréductibles dans $\mathbb{Z}[x]$ en réduisant modulo un nombre premier p (voir XV §3 ; le lecteur pourra, à titre d'exercice, démontrer l'irréductibilité sans recourir au groupe de Galois, en adaptant les idées de XV 2.4 (ii)).

Plus généralement, puisque Φ_n est a coefficients entiers, on peut évaluer Φ_n, tout comme $x^n - 1$, sur n'importe quel élément d'un anneau A. On peut aussi regarder Φ_n comme un polynôme de $A[x]$ en considérant que ses coefficients $a_i \in \mathbb{Z}$ sont maintenant $a_i.1_A \in A$, ce qui revient, pour $A = \mathbb{Z}/p\mathbb{Z}$, à réduire les coefficients modulo p. Les racines de $x^n - 1$ dans un corps K sont appelées les *racines de l'unités dans K* ; une telle racine est dite *primitive* si $x^n = 1_K$ mais $x^d \neq 1_K$ pour tout diviseur strict de n.

Proposition 1. *Supposons que la caractéristique de K soit première avec n. Alors les racines de l'unité dans K sont des racines simples du polynôme $x^n - 1 \in K[x]$. Plus généralement, les facteurs irréductibles dans la décomposition de $x^n - 1$ en irréductible dans $K[x]$ sont tous de multiplicité un. Les racines primitives de l'unité dans K sont les racines de Φ_n dans K.*

Démonstration. Le polynôme dérivé de $P = x^n - 1$ est $P' = nx^{n-1}$, qui est non nul car la caractéristique de K ne divise pas n. On en déduit que $\mathrm{pgcd}(P, P') = 1$ (car 0 n'est pas racine

de P), donc les racines de P (dans un corps de décomposition) sont simples et les facteurs irréductibles dans la décomposition sur $K[x]$ sont de multiplicité un.

L'égalité (1) dans $\mathbb{Z}[x]$ se transforme en une égalité dans $K[x]$. Comme les racines de l'unité sont des racines simples, chacune est donc racine d'un unique Φ_d, pour d divisant n. Or les racines qui ne sont pas primitives sont les racines de $x^d - 1$, pour d un diviseur strict de n, donc les racines des Φ_d pour d un diviseur strict de n. Cela démontre que les racines primitives dans K sont les racines de Φ_n dans K. □

Bien que le polynôme Φ_n soit irréductible sur $\mathbb{Q}$, il n'en est pas nécessairement de même lorsqu'on le réduit modulo p. Par exemple, $\Phi_7(x) = x^6 + \ldots + x + 1$ se décompose en $\overline{\Phi}_7(x) = (x^3 + x + \bar{1})(x^3 + x^2 + \bar{1})$ dans $\mathbb{F}_2[x]$.

Nous allons dire ce qu'il advient si l'on regarde Φ_n comme un polynôme $\Phi_{n,q}$ de $\mathbb{F}_q[x]$, où $\mathbb{F}_q$ désigne un corps fini à $q = p^n$ éléments. Comme $K = \mathbb{F}_q$ est de caractéristique p, il contient canoniquement $\mathbb{F}_p = \{m.1_K\}$ et $\Phi_{n,q}$ se déduit de Φ_n en réduisant les coefficients modulo p (et non q !). Parler de la décomposition ou de l'irréductibilité de Φ_n sur $\mathbb{F}_q$ est une commodité de language : il s'agit bien-entendu de $\Phi_{n,q}$.

Proposition 2. *Soit $\mathbb{F}_q$ un corps fini à q éléments et n un entier premier à q. Notons r l'ordre de la classe de q dans $\mathbb{U}(\mathbb{Z}/n\mathbb{Z})$ (i.e. le plus petit entier tel que $q^r \equiv 1 \mod n$). Alors les facteurs irréductibles dans la décompositon du polynôme cyclotomique Φ_n sur $\mathbb{F}_q$ sont tous de degré r et de multiplicité un.*

Démonstration. Les facteurs irréductibles étant tous de multiplicité un en vertu de la proposition précédente, il reste à démontrer qu'ils sont de degré r. Soit donc P un tel facteur et s son degré. On considère le corps $K = \mathbb{F}_q[x]/(P)$, de cardinal q^s. Tout élement non nul $\alpha \in K$ vérifie $\alpha^{q^s-1} = 1$. Soit ζ la classe de x dans $\mathbb{F}_q[x]/(P)$: cet élément annule l'image de P, donc l'image de Φ_n dans K. C'est donc une racine primitive n-ième dans K. Puisque $\zeta^{q^s-1} = 1$ et que ζ est primitive, n divise $q^s - 1$, *i.e.* $q^s \equiv 1 \mod n$. Cela démontre que s est un multiple de l'ordre r de q dans $\mathbb{U}(\mathbb{Z}/n\mathbb{Z})$, et en particulier $s \geqslant r$.

Démontrons maintenant que $s \leqslant r$. Puisque $\zeta^n = 1$ et que n divise $q^r - 1$, on a $\zeta^{q^r-1} = 1$ donc $\zeta^{q^r} = \zeta$. On considère l'ensemble des racines dans K de l'équation $x^{q^r} = x$. C'est un sous-corps de K contenant $\mathbb{F}_q$ et ζ, qui est un élément primitif de l'extension $K/\mathbb{F}_q$. Il s'agit donc de K tout entier. Comme $x^{q^r} - x$ possède au plus q^r racines distinctes, le cardinal q^s de K est plus petit que q^r, d'où $s \leqslant r$. □

Corollaire 1. *Φ_n est irréductible sur $\mathbb{F}_q$ si et seulement si la classe de q est un générateur de $\mathbb{U}(\mathbb{Z}/n\mathbb{Z})$.*

Par exemple, les polynômes Φ_3 et Φ_5 sont irréductibles sur $\mathbb{F}_2$.

Corollaire 2. *Le polynôme Φ_{p^r-1} se décompose sur $\mathbb{F}_p$ en un produit de polynômes irréductibles unitaires de degré r deux à deux distincts. En particulier, il existe des polynômes irréductibles sur $\mathbb{F}_p$ de n'importe quel degré r.*

Les polynômes cyclotomiques peuvent être calculés de façon efficace grâce aux formules ci-dessous :

(i) si p est un nombre premier ne divisant pas n alors $\Phi_{pn}(x)\Phi_n(x) = \Phi_n(x^p)$ (démontrer que $\mathcal{P}_n^{1/p} = \{x \in \mathbb{C},\ x^p \in \mathcal{P}_n\}$ est l'union disjointe $\mathcal{P}_{pn} \bigsqcup \mathcal{P}_n$),

(ii) si chaque diviseur premier de k divise n alors $\Phi_{kn}(x) = \Phi_n(x^k)$.

On en déduit l'algorithme suivant de construction de Φ_n :
— on détermine les diviseurs premiers $p_1, \ldots, p_m$ (distincts) de n,
— on définit par récurrence $f_i(x) = f_{i-1}(x^{p_i})/f_{i-1}(x)$ à partir de $f_0(x) = x - 1$;
— alors $\Phi_n(x) = f_m(x^{\frac{n}{p_1 \cdots p_m}})$.

Puisque l'on dispose d'un algorithme efficace de factorisation sur $\mathbb{F}_p$ (TP.IX.A), le corollaire précédent fournit une méthode de construction des corps finis $\mathbb{F}_{p^r}$ alternative à celle exposée au TP.IX.A où le polynôme irréductible de degré r était obtenu par tirage aléatoire. Cependant, Φ_{p^r-1} est de degré $\varphi(p^r - 1)$, qui est exponentiel en r : c'est impratiquable pour r très grand.

1. Les commandes `F3=GF(3)` et `F9.<a>=GF(3^2)` sous SAGE définissent respectivement des corps finis à 3 et 9 éléments (F3 et F9 sont des noms arbitraires donnés, que nous choisissons de façon à être explicites). Parcourir l'aide en ligne de la commande `GF` (pour Galois Field ; cette commande est identique à `FiniteField`). Lister les éléments de $\mathbb{F}_3$ et de $\mathbb{F}_9$ et effectuer des opérations arithmétiques dans ces deux corps. Quel est le sens mathématique du paramètre a ?

2. La commande `F3X.<x>=PolynomialRing(GF(3))` définit l'anneau de polynômes $\mathbb{F}_3[x]$. On peut ensuite vérifier que $P = x^4 - x^3 + x^2 - x + 1$ est irréductible sur $\mathbb{F}_3$ à l'aide de la commande `P.is_irreducible()` (on notera que la mention de l'indéterminée x dans la déclaration de l'anneau de polynômes est primordiale pour que SAGE sache à quel anneau appartient P). Si b désigne une racine de P dans une clôture algébrique $\overline{\mathbb{F}}_3$, le corps $\mathbb{F}_3(b) \simeq \mathbb{F}_3[x]/(P)$ est donc un corps fini à 3^4 éléments.

Tester ces commandes puis définir `F81.<b>=GF(3^4,modulus=P)`. Calculer b^9 et b^{-1} puis vérifier que l'on obtient respectivement les mêmes résultats que le reste de la division euclidienne de X^9 par P et que le coefficient attendu dans la relation de Bezout entre X et P (la commande `xgcd` fournit une implémentation de l'algorithme d'Euclide étendu).

Enfin, la commande `b.minpoly(x)` renvoie le polynôme minimal de b (sur $\mathbb{F}_3$, puisque x est associé à $\mathbb{F}_3[x]$). Tester et calculer également le "modulus" utilisé par SAGE à la question 1 ; est-ce cohérent avec les calculs arithmétiques menés précédemment ? Ceci vient éclairer les algorithmes employés par SAGE pour calculer dans les corps finis.

3. Il s'agit maintenant, pour construire les corps finis, de produire des polynômes irréductibles sur $\mathbb{F}_p$ du degré voulu. C'est là qu'interviennent les polynômes cyclotomiques. Le texte du TP décrit complètement un algorithme de construction de ces polynômes (par récurrence). Pour gagner du temps, nous utiliserons directement la commande SAGE correspondante, à savoir `cyclotomic_value`. Calculer $\Phi_n(x)$ pour $n = 3^4 - 1$.

4. Nous allons maintenant apprendre à décomposer en irréductibles dans $\mathbb{F}_q[x]$. L'algorithme de Berlekamp exposé au sein du TP.IX.A réalise cette tâche (bien que nous ayons supposé $q = p$ pour simplifier ; le lecteur motivé saura adapter les énoncés). Les résultats des calculs sont appliqués à la détermination de corps de rupture et de décomposition pour des polynômes donnés.
— Vérifier que $P_1 = x^3 + 2x^2 + 2x + 1$ est décomposé sur $\mathbb{F}_3$ à l'aide de la commande `factor`.
— Démontrer qu'un corps de rupture de $P_2 = x^3 + 2x^2 - x - 1$ sur $\mathbb{F}_3$ est corps de décomposition. On définira un corps fini F27 de cardinal 3^3 en prenant $P2$ comme modulus, puis l'anneau de polynômes correspondant `F27X.<x>=PolynomialRing(F27)` ; enfin, on factorisera P_2 sur ce corps à l'aide de la commande `F27X(P2).factor()`. Comparer avec le résultat de `P2.factor()`.

— Soit maintenant $P_3 = x^4 + 2x^3 + 2x^2 + x + 2 \in \mathbb{F}_3[X]$. Comme l'indéterminée x est associée actuellement à l'anneau $\mathbb{F}_{3^3}[x]$ dans la mémoire de SAGE, il s'agit d'utiliser la commande `x=F3X.gen()` avant de définir P_3. Factoriser P_3 sur $\mathbb{F}_9$. Déterminer un corps de rupture et un corps de décomposition (à isomorphisme près).

5. Vérifier que les facteurs irréductibles de Φ_{3^4-1} sur $\mathbb{F}_{3^k}$, $1 \leqslant k \leqslant 4$, sont bien du degré prescrit par la proposition 2, sachant que l'ordre de q dans $\mathbb{U}(\mathbb{Z}/n\mathbb{Z})$ s'obtient avec la commande SAGE suivante : `R1=Integers(n); R1(q).multiplicative_order()`. Il est clair, au vu du corollaire 2, que $\mathbb{F}_{3^4}$ est corps de rupture, donc de décomposition (puisque tous les facteurs sont de même degré).

Les polynômes irréductibles P et P_3 sont-ils des facteurs irréductibles de Φ_{3^4-1} sur $\mathbb{F}_3$?

Polynôme générateur d'un code $BCH(q, n, \delta)$

On suppose que n est premier avec q. Sans expliquer la terminologie (pour le moment), un polynôme générateur $g \in \mathbb{F}_q[x]$ d'un code $BCH(q, n, \delta)$ est le ppcm des polynômes minimaux sur $\mathbb{F}_q$ des $\delta - 1$ puissances consécutives $\beta, \ldots, \beta^{\delta-1}$ d'une racine primitive n-ième β dans $\overline{\mathbb{F}}_q$.

C'est un diviseur de $x^n - 1$. En effet, puisque les β^i annulent $x^n - 1$, leurs polynômes minimaux μ_{β^i} divisent tous $x^n - 1$. On peut donc écrire $g = \prod_{j\in\Sigma}(x - \beta^j)$, où Σ est une partie convenable de $\mathbb{Z}/n\mathbb{Z}$. Or g appartient à $\mathbb{F}_q[x]$ si et seulement si $g(x^q) = g(x)^q$, donc si et seulement si Σ est stable par multiplication par q.

Définition 1. Les classes cyclotomiques sont les orbites Σ_i de la multiplication par q dans $\mathbb{Z}/n\mathbb{Z}$, *i.e.* les classes pour la relation d'équivalence

$$i \sim j \Leftrightarrow \exists k \in \mathbb{Z},\ q^k i = j.$$

La classe Σ_i de i est la plus petite partie stable par q contenant i, ou encore $\Sigma_i = \{i, qi, \ldots, q^{s-1}i\}$, où s est le plus petit entier positif non nul tel que $q^k i \equiv i \mod n$. Les entiers $k \in \mathbb{Z}$ vérifiant cette congruences forment un sous-groupe de $\mathbb{Z}$ contenant l'ordre r de q dans $\mathbb{U}(\mathbb{Z}/n\mathbb{Z})$. On voit donc que s divise r. Enfin, le lecteur justifiera facilement que les différents facteurs irréductibles de $x^n - 1$ sur $\mathbb{F}_q$ corespondent aux $g_{i_k} = \prod_{j\in\Sigma_{i_k}}(x - \beta^j)$ pour les différentes classes cyclotomiques Σ_{i_k}.

Voici comment construire g de façon efficace :
— On calcule Φ_n,
— puis on factorise Φ_n sur $\mathbb{F}_q$ et choisit une racine β d'un facteur irréductible.
— Soit $\Sigma_{i_1}, \ldots, \Sigma_{i_l}$ les classes cyclotomiques distinctes associées à $1, \ldots, \delta - 1$ (on peut même choisir i_k tel que $i_k = \min \Sigma_{i_k}$). On détermine le polynôme minimal g_{i_k} de β^{i_k} à l'aide de la décomposition en facteurs iréductibles sur $\mathbb{F}_q$ de $\Phi_{n/\operatorname{pgcd}(n,i_k)}$ (puisque β^{i_k} est racine primitive sur $\mathbb{F}_q$ d'ordre l'ordre de i_k dans $\mathbb{Z}/n\mathbb{Z}$) : on prend le facteur qui annule β^{i_k}.
— Alors $g = \prod_{k=1}^{l} g_{i_k}$.

6. On reprend l'exemple $q = 3$ et $n = 3^4 - 1$ de la question précédente et se donne une racine primitive n-ième de l'unité β à travers son polynôme minimal μ_β. Quel polynôme peut-on prendre au vu des questions précédentes ?

Calculer `mul(x-beta^(3^i) for i in range(4))`. Le résultat est-il cohérent ?

Vérifier que β^2 est racine de Φ_m pour m l'ordre de 2 dans $\mathbb{Z}/n\mathbb{Z}$. Décomposer Φ_m en facteurs irréductibles sur $\mathbb{F}_3$ puis en déduire le polynôme minimal de β^2. Comme deuxième

méthode, déterminer la classe cyclotomique de 2 et $\prod_{j\in\Sigma_2}(x-\beta^j)$. Enfin, vérifier avec la commande `minpoly`.

7. On se restreint pour simplifier aux codes BCH binaires primitifs : on prend $q=2$ et $n=2^m-1$. La théorie des classes cyclotomiques est alors extrêmement simple : elles sont représentées par les nombres impairs (justifier).

On se donne $m=5$ et $\delta=7$. Quels sont les différentes classes cyclotomiques permettant le calcul de $g=\prod_{k=1}^{l} g_{i_k}$? Choisir μ_β puis déterminer le polynôme générateur du code BCH correspondant en utilisant d'une part sa définition comme un ppcm (commande `lcm`), d'autre part la formule $g=\prod_{k=1}^{l} g_{i_k}$. Les polynômes minimaux seront déterminés à l'aide de la commande `minpoly`.

Codes correcteurs d'erreurs, codage et décodage des codes BCH

Le propos de la théorie des codes correcteurs d'erreurs est la détection et la correction d'erreurs, lors de la transmission d'un message dans un canal qui est en général bruité donc source d'erreurs. En rajoutant une information supplémentaire au message M (opération de codage) avant de le transmettre (on transmet donc le message codé m), on espère pouvoir reconstituer le message d'origine (opération de décodage) à partir du message reçu m'. Si les erreurs ne sont pas trop nombreuses, le message décodé M' est égal à M.

Par exemple, on peut répéter plusieurs fois le message M et décoder en prenant les symboles qui apparaissent majoritairement. Une erreur de transmission se produit avec une probabilité moindre qu'en transmettant simplement le message, cependant le coût de la transmission se trouve accru puisque la longueur du message augmente. Le but est de construire des codes qui réduisent la probabilité d'erreur avec un coût raisonnable, et tels que l'on dispose d'algorithmes de codage et surtout de décodage efficaces. Les bases de la théories des codes ont été établies par Shannon vers 1950. Les applications technologiques sont nombreuses dans les télécommunications (minitel, TV par satellite, *etc...*). C'est également grâce à ces technologies qu'il est possible de lire un CD avec une bonne qualité d'écoute même s'il est raillé.

L'algèbre fournit des codes très utiles. Un *code linéaire sur $\mathbb{F}_q$ de dimension k et longueur n* est un sous-espace C de dimension k de $\mathbb{F}_q^n$. Le choix d'une base définit une application d'encodage $E:\mathbb{F}_q^k\to\mathbb{F}_q^n$ dont l'image est C.

Afin de transmettre un message, on commence par l'identifier à un élément de $\mathbb{F}_q^k$. Si l'on prend, par exemple, $q=2$ et $k=64$, et si l'on désire transmettre un message rédigé en ASCII [1], alors chaque lettre ASCII peut être identifiée à un octet et un bloc de 8 lettres à un "mo" de $\mathbb{F}_2^{64}$.

Pour chaque mot $a=(a_0,\dots,a_{63})\in\mathbb{F}_q^n$, on note $w(a)=\mathrm{Card}(\{i,\ a_i\neq 0\})$ son *poids de Hamming*. La *distance minimale* du code est par définition $d(C)=\min(w(a),\ a\in C\setminus\{0\})$. Comme C est un espace vectoriel, $w(a-b)\geqslant d(C)$ pour deux mots distincts a et b du codes. Le lecteur vérifiera facilement que $d(a,b)=w(a-b)$ définit une véritable distance sur les mots de $\mathbb{F}_q^n$, au sens des espaces métriques. Par exemple, le code de répétition pure $C=\{(a,a,a)\in\mathbb{F}_2^{192},\ a\in\mathbb{F}_2^{64}\}$ possède une distance minimale $d(C)=3$.

Un mot reçu m' est décodé en $c\in C$ tel que $w(c-m')$ soit minimal. Comme les probabilités vont dans ce sens, on parle de décodage selon le principe du maximum de vraissemblance. S'il se produit moins de $d(C)/2$ erreurs, alors le message est décodé correctement. On dit que le

1. La norme American Standard Code for Information Interchange est la norme de codage de caractères la plus connue en informatique

code est t-correcteur, où t désigne la partie entière de $(d(C)-1)/2$: le code peut corriger t erreurs.

Expliquons maintenant le fonctionnement des codes $BCH(q,n,\delta)$, qui constituent une classe populaire de codes introduite par Bose, Ray-Chaudhuri et Hocquenghem.

Définition 2. Soit β une racine primitive n-ième de l'unité dans $\overline{\mathbb{F}}_q$, pour n un entier premier avec q et g le ppcm (unitaire) des polynômes minimaux de $\beta,\dots,\beta^{\delta-1}$. L'espace vectoriel

$$C=\sum_{0\leqslant i<n-\deg g} x^i\bar{g}\cdot\mathbb{F}_q\subset\mathbb{F}_q[x]/(x^n-1)=A\simeq\mathbb{F}_q^n,$$

où $\bar{g}\in A$ désigne la classe de g modulo x^n-1, est appelé *code BCH* et noté $BCH(q,n,\delta)$. Il est de longueur n et dimension $k=n-\deg g$. On dit que g est son *polynôme générateur*, car C est l'idéal de A engendré par g.

Précisons l'isomorphisme $\mathbb{F}_q^n\simeq A$: on identifiera un mot $a=(a_0,\dots,a_{n-1})$ du code avec le polynôme $a(x)=\sum_{i=0}^{n-1}a_ix^i$ (plus exactement sa classe). Faisant de même avec $\mathbb{F}_q^k$, l'application d'encodage n'est donc rien d'autre que la multiplication par g.

Remarque. Les mots du code $BCH(q,n,\delta)$ sont invariants par permutation circulaire : si $(a_0,\dots,a_{n-1})\in C$ alors $(a_{n-1},a_0,\dots,a_{n-2})\in C$. En effet, cette opération correspond à la multiplication par x dans A. On parle de *code linéaire cyclique*. Ces codes correspondent aux idéaux de A (qui sont tous principaux, mais A n'est pas principal puisqu'il n'est pas intègre).

La définition précédente ne reflète pas le fait qu'un code $BCH(q,n,\delta)$ dépend du choix de β. Cependant, les propriétés du code sont essentiellement indépendantes de β, et en particulier la distance minimale.

Théorème 1. *La distance minimal de $C=BCH(q,n,\delta)$ vérifie $d(C)\geqslant\delta$. On pourra donc corriger au moins $E((\delta-1)/2)$ erreurs.*

Démonstration. Un élément $a(x)$ appartient au code si et seulement si $a(\beta^i)=0$ pour $1\leqslant i<\delta$ ou encore si et seulement si

$$\begin{pmatrix}1&\beta&\dots&\beta^{n-1}\\1&\beta^2&\dots&\beta^{2(n-1)}\\\vdots&\vdots&&\vdots\\1&\beta^{\delta-1}&\dots&\beta^{(\delta-1)(n-1)}\end{pmatrix}\begin{pmatrix}a_0\\a_1\\\vdots\\a_{n-1}\end{pmatrix}=0.$$

Parce que tous les déterminants de taille $\delta-1$ extraits de la matrice ci-dessus sont, à une constante non nulle près, des déterminants de Vandermonde dont les coefficients parmi les β^i sont deux à deux distincts, on voit que ce système d'équations n'admet pas de solution $a\neq 0$ tel que $w(a)\leqslant\delta-1$. Tout élément non nul de C vérifie donc $w(a)\geqslant\delta$. □

On a vu que les racines primitives n-ièmes de l'unité sont les racines dans $\overline{\mathbb{F}}_q$ de Φ_n. L'extension cyclotomique engendrée est de degré $r=m$ lorsque $n=q^m-1$ et alors $\mathbb{F}_q(\beta)^*=\{1,\beta,\dots,\beta^{n-1}\}$. On parle de code *BCH primitif*. Dans le cas général, si l'on pose $m=[\mathbb{F}_q(\beta):\mathbb{F}_q]$, on sait juste que l'ordre n de β divise le cardinal q^m-1 de $\mathbb{F}_q(\beta)^*$.

8. Nous avons généré précédemment un code $BCH(2,2^5-1,7)$, de polynôme générateur g. Quelle sont la longueur n et la dimension k du code ? Combien d'erreurs peut-on corriger ?

La commande SAGE suivante permet de générer aléatoirement un message à envoyer (dans le formalisme polynômial) : `M=F2X.random_element(k-1)` où l'on a défini au préalable `F2X.<x>=PolynomialRing(GF(2))`. Générer un tel message puis le mot du code $c = Mg$ qui lui correspond après encodage.

Nous allons maintenant expliquer comment décoder. On suppose que $c \in C$ est transmis et que m' est reçu. Le polynôme d'erreur est $e(x) = \sum_{i=0}^{n-1} e_i x^i$, correspondant au vecteur d'erreur $e = m' - c$. On suppose qu'au plus $t = E((\delta-1)/2)$ erreurs se sont produites, *i.e.* $w(e) \leqslant t$ et définit :

— l'ensemble $I = \{i,\ e_i \neq 0\}$ des positions des erreurs,

— le polynôme $u(x) = \prod_{i\in I}(1-\beta^i x) \in \mathbb{F}_g(\beta)[x]$ appelé *localisateur d'erreur*,

— le polynôme $v = \sum_{i\in I} e_i\beta^i \prod_{j\in I\setminus\{i\}}(1-\beta^j x)$ *évaluateur d'erreur.*

Les polynômes u et v vérifient $\deg u \leqslant t$ et $\deg v < t$. Ils déterminent à eux deux l'emplacement et la valeur des erreurs : il suffit d'évaluer en β^{-i} pour obtenir I ; on calcule e_i à l'aide de $u' = \sum_{i\in I} -\beta^i \prod_{j\in I\setminus\{i\}}(1-\beta^j x)$, d'où

$$v(\beta^{-i}) = e_i\beta^i \prod_{j\in I\setminus\{i\}} (1-\beta^{j-i}) = -e_i u'(\beta^{-i})$$

puis $e_i = -v(\beta^{-i})/u'(\beta^{-i})$.

Il existe différentes façons de calculer u et v. On peut, par exemple, formuler le problème en terme d'équations linéaires à résoudre. On va donner une autre méthode, plus performante en pratique.

On définit

$$w = \frac{v}{u} = \sum_{i\in I} \frac{e_i\beta^i}{1-\beta^i x} = \sum_{i\in I} x^{-1} \sum_{j\geqslant 1} e_i(\beta^i x)^j = \sum_{j\geqslant 1} x^{j-1} \sum_{i\in I} e_i\beta^{ji} = \sum_{j\geqslant 1} e(\beta^j)x^{j-1}.$$

Comme $c(\beta^j) = 0$ pour $1 \leqslant j \leqslant \delta-1$, on a $e(\beta^j) = m'(\beta^j)$ pour $1 \leqslant j \leqslant \delta-1$. On connaît donc w modulo $x^{\delta-1}$: c'est $S(x) = \sum_{j=1}^{\delta-1} m'(\beta^j)x^{j-1}$, appelé parfois *polynôme syndrôme.*

La congruence

$$v(x) \equiv u(x)S(x) \mod x^{2t} \tag{2}$$

(noter que $2t \leqslant \delta-1$) peut se résoudre en utilisant une variante de l'algorithme d'Euclide étendu, appelé *algorithme de Berlekamp-Massey* : on calcule trois suites r_j, u_j et v_j telles que $r_j(x)x^{2t} + u_j(x)S(x) = v_j(x)$ pour tout j, à partir de $(r_0, u_0, v_0) = (1, 0, x^{2t})$ et $(r_1, u_1, v_1) = (0, 1, S(x))$, en effectuant les divisions euclidiennes $v_{i-1} = v_i q_i + v_{i+1}$ puis les soustractions $r_{i+1} = r_{i-1} - r_i q_i$ et $u_{i+1} = u_{i-1} - u_i q_i$ jusqu'à obtenir $\deg v_i < t$ et $\deg v_{i-1} \geqslant t$.

Proposition 3. *L'algorithme de Berlekamp-Massey donne (à facteur constant près) le couple* $(u(x), v(x))$ *recherché, avec* $\deg u \leqslant t$ *et* $\deg v < t$*, vérifiant la congruence (2).*

Démonstration. Comme $\deg v_{i+1} < \deg v_i$, la suite $(\deg v_i)$ est strictement décroissante pour $i \geqslant 1$. Il existe donc j tel que $\deg v_j < t$ et $\deg v_{j-1} \geqslant t$. On a également $\deg q_i = \deg v_{i-1} - \deg v_i$ pour $i \geqslant 1$. Regardons la suite $(\deg u_i)$: on a $u_2 = -u_1 q_1$, d'où $\deg u_2 \geqslant \deg u_1$, puis $u_3 = u_1 - u_2 q_2$, d'où $\deg u_3 = \deg u_2 + \deg q_2 > \deg u_2$. On démontre par récurrence que la suite est strictement croissante à partir de $i = 2$: si $\deg u_i > \deg u_{i-1}$ alors $\deg u_{i+1} =$

$\deg u_i + \deg q_i > \deg u_i$. On obtient également, en sommant les égalité $\deg u_{i+1} - \deg u_i = \deg q_i = \deg v_{i-1} - \deg v_i$ pour $i \geqslant 1$:

$$\deg u_j = \deg u_j - \deg u_1 = \deg v_0 - \deg v_{j-1} = 2t - \deg v_{j-1} \leqslant t.$$

Donc (u_j, v_j) répond au problème. De plus, on a pour tout $i \geqslant 1$:

$$\begin{pmatrix} r_i & u_i \\ r_{i+1} & u_{i+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & -q_i \end{pmatrix} \begin{pmatrix} r_{i-1} & u_{i-1} \\ r_i & u_i \end{pmatrix}$$

d'où $r_i u_{i+1} - r_{i+1} u_i = -(r_{i-1} u_i - r_i u_{i-1})$ puis $r_i u_{i+1} - r_{i+1} u_i = (-1)^i$ par récurrence ($i \geqslant 0$). Cela montre que $\operatorname{pgcd}(r_j, u_j) = 1$.

Soit maintenant (u, v) la solution recherchée correspondant à l'erreur de décodage. Noter que les polynômes localisateur et évaluateur d'erreur sont premiers entre eux par définition, donc $\operatorname{pgcd}(r, u) = 1$. On va prouver que $r_j u = r u_j$, ce qui implique la proportionalité (dans $K[x]$ puis dans K par primalité) des couples (r, u) et (r_j, u_j), donc également de (u, v) et (u_j, v_j). Dans le cas contraire, les formules de Cramer pour le système

$$\begin{pmatrix} r_j & u_j \\ r & u \end{pmatrix} \begin{pmatrix} x^{2t} \\ S \end{pmatrix} = \begin{pmatrix} v_j \\ v \end{pmatrix}$$

nous donneraient $x^{2t} = \frac{v_j u - v u_j}{r_j u - r u_j}$. Comme $\deg(v_j u - v u_j) \leqslant \max(\deg v_j + \deg u, \deg v + \deg u_j) < 2t$, cela est impossible. □

On obtient donc u et v en divisant au besoin les polynômes obtenus par $u(0)$ pour les rendre unitaires.

9. On prend dans un premier temps $e = x^{n-1} + 1 + x^8$ comme polynôme d'erreur. Le message reçu est donc $m' = e + c$. Calculer le polynôme syndrôme $S(x)$ correspondant (on pourra utiliser la commande `add`).

Ecrire ensuite une procédure `Berlekamp(S,t)` renvoyant le polynôme localisateur d'erreur u lorsque S et t sont fournis en entrée.

Tester ces procédures sur le polynôme $S(x)$ précemment calculé.

10. Il s'agit maintenant de déterminer la position des erreurs à l'aide du polynôme localisateur u, ce qui permet de corriger le message (remarquer que la valeur des erreurs est connue ; il est donc inutile, dans le cas des codes BCH binaires, de calculer le polynôme évaluateur d'erreur). Pour cela, il suffit de tester si β^{-i} est racine de u, pour chaque indice i.

Réaliser cette démarche sur notre exemple (ce qui permet de vérifier si la procédure `Berlekamp` est correcte).

11. Tester avec d'autres valeurs du polynôme d'erreur (ainsi que des polynômes possédant plus de 3 coefficients non nuls : qu'observe-t-on ?).