L3 Calcul Formel
Université de Lorraine

TP 4 : Codes correcteurs

Clément Dell'Aiera

1 Codes linéaires

Nos deux protagonistes préférés, Alice (A) et Bob (B) ont réussi les TP 2 et 3. Ils savent donc trouver de très grands entiers premiers (TP 3) et s'en servir pour coder et décoder leurs messages (TP 2). Ils prennent toutefois conscience que leur canal de transmission est bruité ! De façon aléatoire, un voire plusieurs bits de la transmission peuvent être altérés. Pour remédier à ce problème, ils décident d'utiliser la théorie des codes correcteurs.

A et B utilisent toujours un alphabet $\mathcal{A}$ à q éléments. Nous supposerons que $\mathcal{A} = \mathbb{F}_q$ est le corps à q éléments, qui s'évalue en Sage grâce à $A = GF(q)$. On rappelle que q est alors forcément une puissance d'un nombre premier p, i.e. $q = p^f$. Un mot sera un élément x de $\mathcal{A}^n$, qui est un espace vectoriel de dimension n. On muni l'ensemble des mots de la distance de Hamming

$$d(x, y) = Card\{j \in [1, n] : x_j \neq y_j\}$$

et on définit le poids d'un élément comme

$$w(x) = Card\{j \in [1, n] : x_j \neq 0\}.$$

Définition 1.
- Un code $\mathcal{C}$ est un sous ensemble non vide de $\mathcal{A}^n$ qui possède au moins 2 éléments distincts.
- Un code linéaire est un code $\mathcal{C}$ qui est un sous-espace vectoriel de $\mathcal{A}^n$.
- Un code est binaire si $\mathcal{A} = \mathbb{F}_2$, ternaire si $\mathcal{A} = \mathbb{F}_3$.
- La distance d'un code $\mathcal{C}$ est définie comme

$$d(\mathcal{C}) = \min_{x \neq y \in \mathcal{C}} d(x, y).$$

- On définit $t(\mathcal{C})$ comme le nombre maximal d'erreurs qui sont corrigés par le code.

Voici quelques questions d'ordre théorique (i.e. à résoudre avec un crayon).

1. Montrer que $t(\mathcal{C}) = E[\frac{d(\mathcal{C})-1}{2}]$, et $d(\mathcal{C}) = 2t(\mathcal{C}) + 1$ ou $2t(\mathcal{C}) + 2$.

2. Si moins de $d(\mathcal{C}) - 1$ erreurs sont commises, on peut détecter la présence d'erreur(s). Pourquoi ?
3. Montrer que si t erreurs ont été commises lors de la transmission d'un message m, telles que $2t+1 \leq d(\mathcal{C})$, alors il existe un unique $x \in \mathcal{C}$ tel que $d(x, m) \leq d(\mathcal{C})$.

A un code linéaire $\mathcal{C}$, on associe
- Une matrice vérificatrice $H \in \mathfrak{M}_{n-k,n}(\mathcal{A})$ de rang $n-k$ dont les lignes forment une base des formes linéaires s'annulant sur $\mathcal{C}$.
- Une matrice génératrice $G \in \mathfrak{M}_{k,n}(\mathcal{A})$ de rang k dont les lignes forment une base de $\mathcal{C}$.

On a donc $GH^T = 0$ et $HG^T = 0$. De plus $\mathcal{C}$ est le noyau de H.

Si A veut envoyer le message m, il code le message $x = mG$, et l'envoie à B. On suppose qu'au plus une erreur a été commise lors de la transmission, i.e. au plus un bit de x a été changé. Comment B peut-il tester la présence d'erreur et, le cas échéant, retrouver m ? On note $e = y - x$ l'erreur commise.

B calcule Hx.
Si $Hx = 0$, $x \in \mathcal{C}$ et $m = x$.
Sinon, $Hx \neq 0$ et, comme une seule erreur a été commise, il existe un unique j tel que He_j soit proportionnel à Hx, i.e. $\exists! j, \exists a \in \mathcal{A}, Hx = aHe_j = H(ae_j)$.
Alors $m = x - ae_j$.

Code de Hamming $H(2,7)$: Le code $H(2,7)$ est le sous-espace vectoriel de $(\mathbb{F}_2)^7$ engendré par

$$e_0 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, e_1 = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, e_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, e_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

1. Implémenter une fonction **Test** qui prend en entrée un message bruité ainsi qu'une matrice vérificatrice associée à un code $\mathcal{C}$; et retourne $True$ si le message appartient au code défini par H, $False$ sinon.
2. Implémenter une fonction **Code** qui prend en entrée un message à envoyer ainsi qu'une matrice génératrice associée à un code $\mathcal{C}$; et retourne le message codé.
3. Implémenter une fonction **Bruit** qui prend en entrée un message non bruité et le bruite. On prendra pour cela un bit du message au hasard que l'on inverse ($0 \mapsto 1$ et $1 \mapsto 0$).
4. Implémenter une fonction **Decode** qui, étant donné un code et un message reçu, vérifie s'il y a une erreur, et le cas échéant, corrige l'erreur. La fonction doit retourner le message corrigé. En cas d'erreur, en plus du message corrigé, la fonction afficher un message du type "Erreur detectée sur le bit numéro j" avec j la position de l'erreur.

5. Combien le code de Hamming $H(2,7)$ contient-t-il de mots ? Calculer sa distance, sa capacité de détection et de correction.
6. Tester vos fonctions sur le code de Hamming $H(2,7)$: choisir un message à envoyer, le coder, lui ajouter un bruit, puis le décoder.

2 Codes cycliques

On se donne une application appelée **shift**, définie par

$$S : \left\{ \begin{array}{lcl} A^n & \to & A^n \\ (a_0, ..., a_{n-1}) & \mapsto & (a_{n-1}, a_0, a_1, ..., a_{n-2}) \end{array} \right.$$

Définition 2. Un code cyclique de longueur n est un code linéaire $\mathcal{C} \subset A^n$ stable par S.

Comme $\mathcal{A} = \mathbb{F}_q$, on a un isomorphisme de $\mathbb{F}_q$-espace vectoriel

$$\Psi : \left\{ \begin{array}{lcl} \mathcal{A}^n & \to & \mathbb{F}_q[X]/(X^n - 1) \\ (a_0, ..., a_{n-1}) & \mapsto & a_0 + a_1 X + ... + a_{n-1} X^{n-1} \bmod (X^n - 1) \end{array} \right.$$

qui permet d'identifier A^n à l'anneau $R_n = \mathbb{F}_q[X]/(X^n - 1)$ (et donc d'avoir une structure d'anneau sur les mots). L'application Ψ envoie S sur la multiplication par X

$$\Psi \circ S(a) = X\Psi(a) \quad , \forall a \in A^n,$$

et un code cyclique de longueur n $\mathcal{C}$ n'est rien d'autre qu'un sous espace-vectoriel de R_n stable par multiplication par X, i.e. un idéal de R_n. De tels idéaux sont en bijection avec les polynômes unitaires divisant $X^n - 1$ dans $\mathbb{F}_q[X]$. On s'intéresse donc à la décomposition de $X^n - 1$ en polynômes irréductibles, ce qui se fait grâce aux polynômes cyclotomiques.

On note $\mu_n = \{w \in \mathbb{C}, w^n = 1\}$ les racines complexes $n^{\text{ième}}$ de l'unité, et $\mu_n^* = \{e^{2i\pi d/n} : d \text{ premier avec } n\}$ les racines primitives $n^{\text{ième}}$ de l'unité.

Définition 3. Le $n^{\text{ième}}$ polynôme cyclotomique est défini par

$$\Phi_n = \prod_{w \in \mu_n^*} (X - w) \in \mathbb{C}[X]$$

Ces polynômes sont en fait à coefficients entiers, et donnent la décomposition de $X^n - 1$ en facteurs irréductibles dans $\mathbb{Z}[X]$:

Proposition 1. Les polynômes cyclotomiques sont à coefficients entiers $\Phi_n \in \mathbb{Z}[X]$ et sont irréductibles dans $\mathbb{Z}[X]$, $\deg \Phi_n = \varphi(n)$ (indicatrice d'Euler) et

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

La proposition suivante donne une méthode pour calculer effectivement les polynômes cyclotomiques.

Proposition 2. Soit p premier et m non multiple de p. Si $n = p_1^{\alpha_1}...p_k^{\alpha_k}$ est non nul, on note $n' = p_1...p_k$ sa partie sans facteur carré. Alors
- $\Phi_p(X) = 1 + X + ... + X^{p-1}$
- $\Phi_{mp}(X) = \frac{\Phi_m(X^m)}{\Phi_m(X)}$
- $\Phi_n(X) = \Phi_{n'}(X^{\frac{n}{n'}})$

On en déduit l'algorithme suivant.

Polynôme cyclotomique
Entrée : $n \in \mathbb{N}^*$
Sortie : Φ_n

1. Déterminer la décomposition en facteur premier de $n = p_1^{\alpha_1}...p_k^{\alpha_k}$
2. $m = p_1 p_2 ... p_k$
3. $P_0 = X - 1$
4. Pour $j = 1, .., k$ faire $P_j \leftarrow P_{j-1}(X^{p_j})//P_{j-1}(X)$
5. Retourner $P_k(X^{n/m})$

On peut réduire les polynômes cyclotomiques modulo q et les voir comme des polynômes à coefficients modulo q, $\Phi_{n,q} \in \mathbb{F}_q[X]$, où ils ne sont pas forcément irréductibles : la réductibilité de $\Phi_{n,q}$ dépend de la nullité de n modulo q.

Proposition 3.
- Si $n = p^s m$, p ne divisant pas m, alors

$$\Phi_{n,q}(X) = \Phi_{m,q}(X)^{p^s - p^{s-1}}.$$

- Si n et q sont premiers entre eux ($s = 0$), alors $\Phi_{n,q}$ est un produit de $\frac{\varphi(n)}{r}$ polynômes unitaires irréductibles distincts, où r est l'ordre de $q \pmod n$ dans $(\mathbb{Z}/n\mathbb{Z})^*$.

Si on se place dans le cas des codes binaires de longueur impaire $n = 2^s - 1$, on obtient que $\Phi_{n,2}$ se décompose en produit de $\frac{\varphi(n)}{s}$ polynômes irréductibles sur $\mathbb{F}_2$ de degré s. On définit le polynôme suivant :

$$T_d(X) = \sum_{j=0}^{d-1} X^{2^j}.$$

On va chercher un facteur irréductible non trivial de $\Phi_{n,2}$ grâce à l'algorithme de Cantor-Zassenhauss, qui permet de trouver un facteur d'un polynôme dont tous les facteurs irréductibles ont même degré.

Cantor-Zassenhauss
Entrée : $P \in \mathbb{F}_2[X]$ unitaire sans facteur carré ayant tous des facteurs irréductibles de même degré d, $deg\ P = n > 0$.
Sortie : Un facteur non trivial de P, ou $False$.

1. Choisir $Q \in \mathbb{F}_2[X]$ au hasard de degré $< n$.
2. $g = pgcd(P, Q)$
3. Si $g = 1$, alors retourner g
4. $a = T_d(Q) \pmod P$
5. $g \leftarrow pgcd(a - 1, P)$
6. Si $g \neq 1$ et $g \neq P$, retourner g, sinon retouner $False$

1. Implémenter une fonction qui, étant donné un entier n, retourne le $n^{\text{ième}}$ polynôme cyclotomique Φ_n, grâce à l'algorithme **Polynôme Cyclotomique**.
2. Implémenter l'algorithme de Cantor-Zassenhauss.
3. Implémenter une fonction qui, étant donné un entier impair n, retourne la liste des facteurs irrédutibles du $n^{\text{ième}}$ polynôme cyclotomique $\Phi_{n,2}$.

Attention, la première fonction travaille dans $\mathbb{Z}[X]$, la troisième dans $\mathbb{F}_2[X]$. Cette dernière fonction vous donne donc la liste de tous les codes cycliques binaires de longueur impaire n.