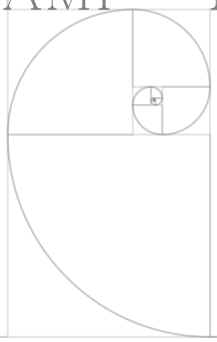
Rendu de projet IN013

Algorithme de Berlekamp – Massey

Andy NGO
21207516

20 juin 2024



SCIENCES SORBONNE UNIVERSITÉ

Table des matières

1

Introduction

Lorsque on transmet des informations, on souhaite que le receveur puisse les obtenir sans qu'elles n'aient été altérés. Ceci est malheureusement impossible, dû à une variété de facteurs, l'un de ces facteur prédominant étant le bruit de mesure. Pour limiter l'impact des erreurs générés, on peut essayer d'ajouter de la redondance dans nos informations avec un algorithme. L'ensemble de ces techniques formant la théorie des codes correcteurs ; le but de notre projet étant d'étudier une partie de l'algorithme de Berkelamp-Massey, qui est capable de décoder les codes BCH qui est une classe de code correcteur.

2

Groupe Z/pZ et calcul d'inverse

2.1 Anneau, groupe et corps

Les codes BCH étant construit par des polynômes sur un corps fini, un corps possédant un nombre fini d'éléments, familiarisons-nous avec la notion d'anneau, de groupe et de corps.

2.1. Un groupe $(G, \times)$ est un ensemble G muni d'une loi interne $\times$ ayant les propriétés suivantes :

- Associativité : $\forall x, y, z \in G : x \times (y \times z) = (x \times y) \times z$.
- Existence de l'élément neutre : $\exists e \in G$, $\forall x \in G : x \times e = e \times x = x$.
- Existence d'un symétrique : $\forall x \in G$, $\exists y \in G : x \times y = y \times x = e$, avec e l'élément neutre.

On dit que un groupe $(G, \times)$ est abélien si : $\forall x, y \in G : x \times y = y \times x$.

$(\mathbb{Z}, +), (\mathbb{R}^*, \times)$ sont des groupes, $(\mathbb{Z}, \times)$ ne l'est pas.

2.2. Un anneau $(A, +, \times)$ est un ensemble A muni de deux lois internes, $+$ et $\times$ ayant les propriétés suivantes :

- $(A, +)$ est un groupe abélien possédant un élément neutre noté 0_A.
- Associativité de la loi $\times$: $\forall x, y, z \in A : x \times (y \times z) = (x \times y) \times z$.
- La loi $\times$ admet un élément neutre noté 1_A.
- La loi $\times$ est distributive avec la loi $+$: $\forall x, y, z \in G : x \times (y + z) = x \times y + x \times z$ et $(x + y) \times z = x \times z + y \times z$.

De même que pour un groupe, on dit qu'un anneau est abélien si la loi $\times$ est commutative.

$(\mathcal{M}[\mathbb{R}], +, \times)$ est un anneau.

2.3. Un corps est un ensemble A muni de deux lois internes, $+$ et $\times$ ayant les propriétés suivantes :

- $(A, +)$ est un groupe abélien possédant un élément neutre noté 0_A.
- $(A \backslash \{0_A\}, \times)$ est un groupe abélien possédant un élément neutre noté 0_A.
- La loi $\times$ est distributive avec la loi $+$: $\forall x, y, z \in G : x \times (y + z) = x \times y + x \times z$ et $(x + y) \times z = x \times z + y \times z$.

R et **Q** munis de leur loi somme et produit usuelles sont des corps non fini.

L'ensemble qui nous intéresse est l'ensemble des relatifs quotienté par la relation d'équivalence suivante avec $n \in \mathbb{N}^*$:

$$x = y \iff x \equiv y \bmod n$$

Cet ensemble est donc constitué des classe d'équivalences de **Z**, et est noté $\mathbb{Z}/n\mathbb{Z}$. On peut rapidement constater que $(\mathbb{Z}/n\mathbb{Z}, +, \times)$ est un anneau fini car $(\mathbb{Z}, +, \times)$ est un anneau, cependant ce n'est pas un corps en général.
En effet dans tout ces anneaux, l'élément neutre de la loi produit est 1, donc si l'ensemble muni des deux lois était un corps, tout élément de $\mathbb{Z}/n\mathbb{Z}$ admettrait un inverse pour la loi produit, cependant en prenant n = 4 :

$$\forall k \in \mathbb{Z} : 2 + k \times 4 \equiv 2 \bmod 4$$

Donc 2 n'admet pas d'inverse, et par conséquent $\mathbb{Z}/4\mathbb{Z}$ n'est pas un corps.
Bien que pour tout entier naturel n l'anneau n'est pas un corps, on peut prendre n dans l'ensemble des nombres premiers, qui sera désormait noté p. On va chercher à montrer que tout élément de $\mathbb{Z}/p\mathbb{Z}$ admet un inverse, et donc que c'est un corps.

2.2 Algorithme d'Euclide étendu et théorème de Bézout

Dans cette partie, on cherche à résoudre l'équation suivante, sachant que l'on connaît a et p :

$$a \times x \equiv 1 \bmod p$$

Pour résoudre cette équation, on introduit le théorème de Bézout :

2.4. *$\forall a, b \in \mathbb{Z}, \exists u, v$ tel que : $a \times u + b \times v = PGCD(a, b)$*

Le lien avec la recherche d'inverse se fait ainsi ; en supposant que $a \not\equiv 0 \bmod p$ (ce qui est équivalent à dire que $a \neq 0$ et que a n'est pas un multiple de p), et en posant $b = p$, le théorème de Bézout nous donne l'existence d'un couple (u, v) tels que :

$$a \times u + p \times v = PGCD(a, p) = 1 \iff a \times u \equiv 1 \bmod p$$

On voit donc que u est l'inverse de a dans l'anneau $\mathbb{Z}/n\mathbb{Z}$, donc tout élément non nul de cet anneau admet un inverse selon la loi produit, et par conséquent cet anneau est aussi un groupe.
La preuve de ce théorème se fait par l'algorithme d'Euclide étendu, qui s'inspire de l'algorithme d'Euclide, qui permet de déterminer le PGCD de deux entiers.

Démonstration. Pour prouver que cet algorithme est valide et se termine, on peut poser les suites suivantes : Avec q_k = quotient de la division euclidienne de r_{k-1} par r_k.

- $(r_n)_{n\in\mathbf{N}} : r_0 = a, r_1 = b, r_{k+1} = r_{k-1} - q_k \times r_k$.
- $(u_n)_{n\in\mathbf{N}} : u_0 = 1, u_1 = 0, u_{k+1} = u_{k-1} - q_k \times u_k$.
- $(v_n)_{n\in\mathbf{N}} : v_0 = 0, v_1 = 1, v_{k+1} = v_{k-1} - q_k \times v_k$.

On peut rapidement voir les égalités suivantes :

- $r_0 = u_0 a + v_0 b$
- $r_1 = u_1 a + v_1 b$
- $r_0 - q_1 \times r_1 = (u_0 - q_1 \times u_1)a + (v_0 - q_1 \times v_1)b \iff r_2 = u_2 a + v_2 b$
- En général : $r_k = u_k a + v_k b$

Algorithme 1 : Algorithme d'Euclide étendu

Entrées : $a, b \in \mathbf{N}^*$
Sorties : u, v tels que : $a \times u + b \times v = PGCD(a, b)$
initialisation : r0 = a, r1 = b, u0 = 1, u1 = 0, v0 = 0, v1 = 1, q = 0, rt, ut, vt ;
tant que $r1 \neq 0$ **faire**
    q = quotient de la division euclidienne de r0 par r1 ;
    rt = r0 ; ut = u0 ; vt = v0 ;
    r0 = r1 ; u0 = u1 ; v0 = v1 ;
    r1 = rt - (r1 * q) ;
    u1 = ut - (u1 * q) ;
    v1 = vt - (vp * q) ;
fin
u = u0 ; v = v0 ; PGCD(a, b) = r0

De plus, on a une propriété du PGCD qui nous donne que : pour $k \in \mathbf{Z}$, $PGCD(a, b) = PGCD(b, a + kb)$.
q_k étant un entier relatif, $PGCD(r_0, r_1) = PGDC(r_1, r_0 - q_1 \times r_1) = PGCD(a, b)$.
Puisque la suite $(r_n)_{n\in\mathbf{N}}$ est une suite strictement décroissante et est constitué de termes positifs ; il existe un rang k tel que $r_k = 0$ et $r_{k-1} \neq 0$, et donc $PGCD(r_k, r_{k-1}) = 0$ et $PGCD(r_{k-1}, r_{k-2}) = PGCD(a, b)$; donc $r_{k-1} = PGCD(a, b) = u_{k-1}a + v_{k-1}b$.
Donc on a deux relatifs $u = u_{k-1}$ et $v = v_{k-1}$ tels que $PGCD(a, b) = au + bv$, ce qui prouve le théorème de Bézout, et par conséquent que si p est premier $\mathbb{Z}/p\mathbb{Z}$ est un corps fini. ■

3 Résolution de système d'équations linéaires et de polynôme du second degrés dans $\mathbf{Z}/p\mathbf{Z}$

Dans cette partie, on va chercher à donner des méthodes et conditions de résolution de systèmes linéaires et de recherche de racine d'un polynôme du second degré dans le groupe $\mathbf{Z}/p\mathbf{Z}$.

3.1 Systèmes linéaires

Soit un système linaire de la forme :

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + ... + a_{1,n}x_n = c_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + ... + a_{2,n}x_n = c_2 \\ \qquad\qquad ... \\ a_{n,1}x_1 + a_{n,2}x_2 + ... + a_{n,n}x_n = c_n \end{cases}$$

Ce système est équivalent au système matriciel suivant :

$$\begin{pmatrix} a_{1,1} & a_{1,2} & ... & a_{1,n} \\ a_{2,1} & a_{2,2} & ... & a_{2,n} \\ & ... & & \\ a_{n,1} & a_{n,2} & ... & a_{n,n} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ ... \\ x_n \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ ... \\ c_n \end{pmatrix} \Longleftrightarrow A \times X = C$$

La condition pour que ce système admette une solution est que A soit inversible, donc que son déterminant ne soit pas nul dans $\mathbf{Z}/p\mathbf{Z}$, c'est à dire que : $det(A) \neq 0 \mod p$. Si ceci est vrai, A est inversible et on peut trouver cet inverse par l'algorithme du pivot de Gauss qui est semblable à celui dans $\mathbf{M}_{n,n}[R]$, les transvections et dilations prenant place dans $\mathbf{Z}/p\mathbf{Z}$.

3.2 Condition de résolution d'un polynôme du second degré

Soit un polynôme du second degré :

$$\alpha x^2 + \beta x + \gamma \text{ avec, } \alpha \neq 0 \text{ et } \alpha, \beta, \gamma \in \mathbf{Z}/p\mathbf{Z}$$

On recherche la conditions d'existence de ces racines. Le déterminant de ce polynôme est $\Delta = \beta^2 - 4\alpha\gamma$. On a donc les situations suivantes :

- $\Delta > 0$: il existe deux racines x_1 et x_2.
- $\Delta = 0$: il existe une racine double.
- $\Delta < 0$: le polynôme n'admet pas de racines dans $\mathbf{R}$.

4 Théorème des restes chinois et résolution

Dans cette section, on cherche à résoudre ce système ci-contre :

$$\begin{cases} a_1 \equiv x \mod m_1 \\ a_2 \equiv x \mod m_2 \\ \quad ... \\ a_n \equiv x \mod m_n \end{cases}$$

Sachant que les m_k sont premiers entre eux (donc que $\forall i, j \leqslant n,\ i \neq j : PGCD(m_i, m_j) = 1$). Pour cela, on peut faire appel au théorème des restes chinois, qui dit qu'il existe une solution unique modulo $M = m_1 \times m_2 \times ... \times m_n$ donné par la formule :

$$x = a_1 M_1 y_1 + a_2 M_2 y_2 + ... + a_n M_n y_n$$

Avec M_i = quotient de la division euclidienne de M par m_i et $y_i = M_i^{-1} \mod m_i$.

Démonstration. Pour l'égalité $a_k \equiv x \mod m_k$; $k \in \{1, ..., n\}$, on peut voir que les M_i avec $i \neq k$ sont les produits de tout les modules sauf m_i, par conséquent : $\exists p$ tel que $M_i = p \times m_k \iff M_i \equiv 0 \mod m_k$. Donc $x \equiv a_k M_k y_k \mod m_k$, et puisque $y_k \equiv M_k^{-1} \mod m_k$; $x \equiv a_k \mod m_k$. ■

5 Recherche d'une relation de récurrence linéaire

On se donne les k premiers termes d'une suite définie par récurrence linéaire, c'est à dire une suite $(u_n)_{n \in \mathbf{N}}$ définie de cette manière :

$$u_{n+p} = a_0 u_n + a_1 u_{n+1} + ... + a_{p-1} u_{n+p-1}, \text{ avec } a_0, ..., a_{p-1} \text{ des relatifs donnés et } a_0 \neq 0$$

Notre but est donc de retrouver la relation de récurrence à partir de ces k premiers termes.

5.1 Une première approche naïve

Une première manière d'approcher ce problème serait de chercher une combinaison de coefficients de manière itérative sur le degré r de la relation de récurrence ; c'est à dire de trouver $a_0, ..., a_r$ tels que : $u_p = a_0u_0 + a_1u_1 + ... + a_{r-1}u_{r-1}$, avec : $a_{r-1} \neq 0$ et de vérifier si cette relation est vraie sur les autres termes de la suite. Si cette relation est vraie sur tout les termes, la relation qu'on a trouvé est la bonne ; si ce n'est pas le cas, on recherche une nouvelle relation de degré supérieure. En général, on peut les chercher en inversant la matrice de gauche donnée par la relation suivante :

$$\begin{pmatrix} u_0 & u_1 & ... & u_{r-1} \\ u_1 & u_2 & ... & u_r \\ & ... & & \\ u_r & u_{r+1} & ... & u_{2r-1} \end{pmatrix} \times \begin{pmatrix} a_0 \\ a_1 \\ ... \\ a_{r-1} \end{pmatrix} = \begin{pmatrix} u_r \\ u_{r+1} \\ ... \\ u_{2r} \end{pmatrix}$$

Bien que cette recherche est facile à mettre en place, elle requiert de faire une inversion de matrice pour chaque degré r, ce qui peut devenir coûteux, surtout avec des matrices de grande taille. On va donc chercher une propriété sur l'échec de la relation qui nous permettrait d'accélérer la recherche.

5.2 Méthode de recherche optimisé

Prenons le cas où notre relation d'ordre 1 $u_k = au_{k-1}$ échoue à un rang r donné, donc :

$$\forall k < r : u_k - au_{k-1} = 0 \text{ et } a_r - au_{r-1} = d \neq 0$$

On pose ce système :

$$\begin{cases} au_0 - u_1 = 0 \\ au_1 - u_2 = 0 \\ \quad ... \quad \text{avec } d \neq 0 \\ au_{r-2} - u_{r-1} = 0 \\ au_{r-1} - u_r = d \end{cases}$$

Que l'on transforme en matrice :

$$M = \begin{pmatrix} u_1 & u_0 \\ u_2 & u_1 \\ ... & ... \\ u_{r-1} & u_{r-2} \\ u_r & u_{r-1} \end{pmatrix}$$

Le calcul du rang de cette matrice se fait par le pivot de Gauss ; après application, on obtient la matrice suivante

$$M = \begin{pmatrix} u_1 & u_0 \\ 0 & -u_0d/u_1 \\ 0 & 0 \\ ... & ... \\ 0 & 0 \end{pmatrix}$$

rang(M) = 2, donc on n'a toujours pas la bonne relation de récurrence. On peut transformer le système de départ afin d'obtenir une matrice de cette forme :

$$M' = \begin{pmatrix} u_2 & u_1 & u_0 \\ u_3 & u_2 & u_1 \\ ... & ... & ... \\ u_{r-1} & u_{r-2} & u_{r-3} \\ u_r & u_{r-1} & u_{r-2} \end{pmatrix}$$

En appliquant le pivot de Gauss, on voit que le rang de M' est toujours de 2. On peut voir que à chaque fois que l'on modifie la matrice en ajoutant une colonne et retirant une ligne, le rang de la matrice est toujours de 2, ce qui se voir avec cette matrice :

$$\begin{pmatrix} u_{r-1} & u_{r-2} & ... & u_0 \\ u_r & u_{r-1} & ... & u_1 \end{pmatrix}$$

Cependant, cette propriété n'est plus vraie quand la matrice n'est que composée d'une ligne :

$$U = \begin{pmatrix} u_r & u_{r-1} & ... & u_0 \end{pmatrix}$$

Dont le rang vaut évidemment 1, par conséquent il existe un seul vecteur $A = \begin{pmatrix} a_r \\ a_{r-1} \\ ... \\ a_0 \end{pmatrix}$ tel que :

$$U \times A = (0) \Longleftrightarrow \sum_{k=0}^{r-1} \frac{a_k}{a_r} u_k - u_r = 0$$

Par conséquent toute relation d'ordre inférieure strictement à r n'est pas correcte, et on peut rechercher une relation d'ordre au moins r. En général, on peut appliquer un processus similaire lorsque une relation de degré d échoue à un rang r ; et de cette idée, on peut en extraire une propriété qui est que :
Lorsque une relation de récurrence échoue à un rang r donné, on peut directement chercher une relation de degré r.

5.3 Introduction à l'algorithme de Berkelamp-Massey

Bien que les méthodes discutés précédemment permettent de trouver une relation de récurrence, il est nécessaire de déterminer à quel rang la relation échoue, ce qui est coûteux. On va donc étudier l'algorithme de Berkelamp-Massey, qui nous permet de trouver le polynôme caractéristique associé aux termes de la suite. Pour commencer, on a les termes $u_0, ..., u_{2r}$ tels que $\exists c_r, ..., c_0$ avec $c_r = 1, \forall i \in \{0, ..., r\} : c_r u_{r+i} + c_{r-1} u_{r+i-1} + ... + c_0 u_r = 0$ .Et on pose les polynômes P et C :

$$P = u_0 x^{2r} + u_1 x^{2r-1} + ... + u_{2r-1} x + u_2 r \text{ et } C = c_r x^r + c_{r-1} x^{r-1} + ... + c_1 x + c_0$$

Dont on va étudier le produit $P \times C = F$. Pour faciliter la lecture, on peut mettre ce calcul sous forme de tableau :

$\times$	u_0x^{2r}	...	u_rx^r	...	u_{2r}
c_0	$c_0u_0x^{2r}$	...	$c_0u_rx^r$	...	c_0u_{2r}
c_1x	$c_1u_0x^{2r+1}$	...	$c_1u_rx^{r+1}$	...	$c_1u_{2r}x$
...	...	...	...	...	...
$c_{r-1}x^{r-1}$	$c_{r-1}u_0x^{3r-1}$	...	$c_{r-1}u_rx^{2r-1}$	...	$c_{r-1}u_{2r}x^{r-1}$
c_rx^r	$c_ru_0x^{3r}$	...	$c_ru_rx^{2r}$	...	$c_ru_{2r}x^r$

Table 1 – Caption

On peut observer que lorsque on factorise notre polynôme produit par les puissances de x, toutes les puissances de x entre r et $2r$ s'annulent, car on obtient la relation de départ qui nous est donné. F est donc de degré $3r$, et contient tout les coefficients (c_i). On pose l'équation suivante

$$P \times C \equiv F \bmod X^{2r-1}$$

Avec $deg(F) < r$. Or on peut observer que cette équation est semblable à celle de la partie 2.2, mais cette fois-ci avec des polynômes. Par conséquent :

$$\exists V \in \mathbf{Z}[X] \text{ tel que : } (P \times C) + (X^{2r-1} \times V) = F \text{ et } deg(C) > deg(F)$$

Pour calculer les coefficients de C, on peut appliquer l'algorithme d'Euclide étendu à X^{2r-1} et P, et puisque les coefficients de C sont les coefficients de la suite linéaire récurrente linéaire, on a donc trouvé la récurrence linéaire de la suite. L'algorithme de Berkelamp-Massey consiste donc à faire ceci