Algorithmes pour l'arithmétique II
Cours 2

Julien Lavauzelle

Université Paris 8

Master 2 ACC et CSSD – Algorithmes pour l'arithmétique

01/12/2020

Contenu

La semaine dernière, on a vu :

1. algorithme de Gauss-Jordan pour la résolution de systèmes linéaires
2. factorisations LU, QR, Cholesky
3. suites récurrentes linéaires, LFSR
4. algorithme de Berlekamp-Massey pour la recherche du LFSR de dimension minimale

Questions ?

Aujourd'hui :

1. résolution de systèmes d'équations creuses
2. calcul de polynôme minimaux de matrices et de suites sur $\mathbb{F}^n$
3. algorithme de Wiedemann

Plan

Motivation

Soit $\boldsymbol{A} \in \mathbb{F}^{m \times n}$ et $\boldsymbol{b} \in \mathbb{F}^{m}$. On souhaite résoudre efficacement

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$$

Contexte : On suppose que toute ligne de $\boldsymbol{A}$ a, au plus, t éléments non nuls

Méthode de Gauss-Jordan :

- générique
- complexité en temps $O(mn \max\{m, n\})$, en espace $O(mn)$, ne dépend pas de t...
- a tendance à densifier le système d'équations

But : tirer parti du fait que $\boldsymbol{A}$ est creuse !

Motivation

Cas concrets de matrices creuses :

- matrices d'adjacence de graphes (réseaux, big data)
- résolution numérique d'équations aux dérivées partielles
- étape d'algèbre linéaire dans le calcul de logarithmes discrets ou dans la factorisation d'entiers

Exemple « extrême » (mais réel) : dans la factorisation de RSA-768 en 2009

- $n \simeq m \geq 192\,000\,000 \simeq 2^{27}$ lignes/colonnes,
- $27\,000\,000\,000 \simeq 2^{34}$ coefficients non-nuls,
- si stockage creux : 105 Go,
- si stockage dense : $>$ 4000 To (impossible).

Plan

Suite récurrente sur $\mathbb{F}^n$

Définition. Soit $\boldsymbol{v} = (\boldsymbol{v}_k)_{k\in\mathbb{N}}$ une suite de vecteurs $\boldsymbol{v}_k \in \mathbb{F}^n$.
La suite $\boldsymbol{v}$ est *récurrente linéaire sur* $\mathbb{F}^n$ s'il existe des éléments $c_0, \ldots, c_d \in \mathbb{F}$, avec $c_0, c_d \neq 0$, tels que

$$\forall k > L, \quad c_0\boldsymbol{v}_k + c_1\boldsymbol{v}_{k+1} + \ldots c_d\boldsymbol{v}_{k+d} = \mathbf{0}\,.$$

Si $n = 1$, on dit que la suite est *scalaire*.

Remarques par rapport au dernier cours :

- L'ordre des indices des coefficients est inversé (+ commode pour la suite)
- Pour tout $1 \leq j \leq n$, la suite $\boldsymbol{v}^{(j)} \in \mathbb{F}^{\mathbb{N}}$ des j-ème coordonnées de $\boldsymbol{v}$ définit une suite linéaire scalaire.

Définition. On appelle polynôme annulateur (ou polynôme de connexion) tout polynôme $C(X) = \sum_{i=0}^{k} c_i X^i$ tel que $\sum_{i=0}^{k} c_i \boldsymbol{v}_i = \mathbf{0}$.

Exemple

Exemple. La suite $\boldsymbol{v} = (\boldsymbol{v}_k)_{k\in\mathbb{N}} \in \mathbb{F}^2$ définie par

$$\begin{pmatrix} v_k^{(1)} \\ v_k^{(2)} \end{pmatrix} = \begin{pmatrix} v_{k-1}^{(2)} \\ -v_{k-1}^{(1)} \end{pmatrix}$$

pour $k \geq 1$.

On peut l'écrire

$$\boldsymbol{v}_k = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \boldsymbol{v}_{k-1}$$

Elle a pour polynôme annulateur $X^2 + 1$, car $v_k^{(1)} = -v_{k-2}^{(1)}$ et $v_k^{(2)} = -v_{k-2}^{(2)}$.

Polynôme minimal

Lemme. L'ensemble $\mathcal{I}_v$ des polynômes annulateurs de v est un idéal de $\mathbb{F}[X]$.

Preuve. Si $P, Q \in \mathcal{I}_v$, alors on peut vérifier que λP, $P+Q$ et $XP(X)$ sont également dans $\mathcal{I}_v$ (pour tout $\lambda \in \mathbb{F}$) :

$$\sum_{i=0}^{\max\{\deg P, \deg Q\}} (p_i + q_i)\, v_i = \mathbf{0} \quad \text{et} \quad \sum_{i=0}^{d} (\lambda p_i)\, v_i = \mathbf{0} \quad \text{et} \quad \sum_{i=1}^{d+1} p_{i-1}\, v_i = \sum_{i=0}^{d} p_i\, v_{i+1} = \mathbf{0}$$

Définition. Le polynôme minimal de v est le polynôme annulateur de v, unitaire et de plus petit degré. Il est noté $\mu_v(X)$.

Remarques :

- Cette définition a un sens car $\mathbb{F}[X]$ est principal et euclidien, donc $\mathcal{I}_v$ est engendré par $P(X)$ de degré minimal parmi les éléments non-nuls de $\mathcal{I}_v$,
- $\mu_v(0) \neq 0$, car sinon $\frac{\mu_v(X)}{X} \in \mathcal{I}_v$:

$$\sum_{i=1}^{d} p_i\, v_i = \mathbf{0} \quad \Longrightarrow \quad \sum_{i=0}^{d-1} p_{i+1}\, v_{i+1} = \mathbf{0}$$

Lemme. Soit $A \in \mathbb{F}^{n\times n}$ non-nulle. Pour tout $x \in \mathbb{F}^n$ non-nul, la suite $v = (A^k x)_{k\in\mathbb{N}}$ est une suite récurrente linéaire. Par ailleurs, $\mu_v(X)$ divise le polynôme minimal de A.

Preuve. Les vecteurs $\{x, Ax, \ldots, A^n x\}$ forment une famille liée de $\mathbb{F}^n$:

$$\sum_{i=0}^{n} \lambda_i A^i x = 0$$

Cette équation reste vérifiée pour les éléments d'ordre supérieur, en appliquant A à gauche :

$$\forall k \geq n, \quad \sum_{i=0}^{n} \lambda_i \, v_{(k-n)+i} = 0$$

Le polynôme annulateur $\mu_A \in \mathbb{F}[X]$ de la matrice A est bien dans $\mathcal{I}_v$ car

$$\mu_A(A) \cdot x = 0 \cdot x = 0 \,.$$

Donc μ_v divise μ_A.

Exemple

Sur $\mathbb{F}_2$, on définit

$$\boldsymbol{A} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

et

$$\boldsymbol{x} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Le polynôme minimal de $\boldsymbol{A}$ est $X^3 + X^2$.
On a

$$\boldsymbol{A}\boldsymbol{x} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \text{ et } \boldsymbol{A}^2\boldsymbol{x} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

Puis, on note que

$$\boldsymbol{A}^k\boldsymbol{x} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \forall k \geq 2\,.$$

Donc le polynôme minimal de $\boldsymbol{v} = (\boldsymbol{A}^k\boldsymbol{x})$ est $X + 1$, car $\boldsymbol{v}$ est constante à partir d'un certain rang.

Liens entre polynômes minimaux

Lemme. Si $\mu_{\boldsymbol{v}^{(1)}}(X), \ldots, \mu_{\boldsymbol{v}^{(n)}}(X)$ sont les polynômes minimaux des suites scalaires « coordonnées » $\boldsymbol{v}^{(1)}, \ldots, \boldsymbol{v}^{(n)}$, alors on a

$$\mu_{\boldsymbol{v}}(X) = \operatorname{ppcm}(\mu_{\boldsymbol{v}^{(1)}}(X), \ldots, \mu_{\boldsymbol{v}^{(n)}}(X)) .$$

Preuve. Notons $P(X) = \operatorname{ppcm}(\mu_{\boldsymbol{v}^{(1)}}(X), \ldots, \mu_{\boldsymbol{v}^{(n)}}(X))$. On observe que

$$\mu_{\boldsymbol{v}}(X) \in \mathcal{I}_{\boldsymbol{v}^{(1)}} \cap \cdots \cap \mathcal{I}_{\boldsymbol{v}^{(n)}} ,$$

qui est un idéal engendré par $P(X)$. Donc $\mu_{\boldsymbol{v}}(X)$ divise $P(X)$.

Réciproquement, $P(X)$ est un polynôme annulateur de $\boldsymbol{v}$ car il annule toutes ses coordonnées. Donc $P(X)$ divise $\mu_{\boldsymbol{v}}(X)$.

Exercice. Pour tout $\boldsymbol{x} \in \mathbb{F}^n$, la suite scalaire $\boldsymbol{a} = (a_k)_{k\in\mathbb{N}}$ définie par

$$\forall k \in \mathbb{N}, \quad a_k := \langle \boldsymbol{x}, \boldsymbol{v}_k \rangle = \sum_{j=1}^{n} x_j v_k^{(j)} \in \mathbb{F}$$

est récurrente linéaire, et son polynôme minimal $\mu_{\boldsymbol{a}}$ divise $\mu_{\boldsymbol{v}}$.

Plan

Méthode de résolution $Ax = b$

Objectif. On veut trouver **une solution** de

$$Ax = b$$

avec $A \in \mathbb{F}^{n \times n}$ creuse, et $b \in \mathbb{F}^n \setminus \{0\}$.

Supposons que l'on sache calculer le polynôme minimal $\mu_v(X) = \sum_{j=0}^{d} \lambda_j X^j$ de la suite itérée

$$v = (A^k b)_{k \in \mathbb{N}}\,.$$

On a alors

$$0 = \lambda_0 b + \lambda_1 Ab + \cdots + \lambda_d A^d b$$

Donc

$$b = A \cdot \underbrace{\left(\frac{-1}{\lambda_0} \left(\lambda_1 b + \cdots + \lambda_d A^{d-1} b \right) \right)}_{\text{solution}}$$

Méthode de résolution.

1. Calculer le polynôme minimal $\mu_v(X) = \sum_{j=0}^{d} \lambda_j X^j$ de la suite $v = (A^k b)_{k \in \mathbb{N}}$.
2. Calculer $Q(X) = \frac{\lambda_0 - \mu_v(X)}{\lambda_0 X} \in \mathbb{F}[X]$ et retourner $Q(A)b$.

Factorisation de $\mu_v(X)$

But. Déterminer le polynôme minimal $\mu_v(X)$ de $v = (A^k b)_{k\in\mathbb{N}}$.

Idée. On va chercher des facteurs de μ_v en utilisant le résultat de l'Exercice précédent.

Proposition. Soit $P(X)$ un diviseur strict de $\mu_v(X)$ tel que $b' := P(A)b \neq 0$. On pose $v' := (A^k b')_{k\in\mathbb{N}}$. Alors on a :

$$\mu_v(X) = \mu_{v'}(X)\, P(X)\,.$$

Preuve. On a

$$(\mu_{v'} P)(A)b = \mu_{v'}(A)(b') = 0$$

donc μ_v divise $\mu_{v'} P$.
D'autre part, soit $Q(X) = \mu_v(X)/P(X)$. On observe que

$$Q(A)b' = Q(A)(P(A)b) = \mu_v(A)b = 0$$

donc $\mu_{v'}$ divise Q.
Autrement dit, $\mu_{v'} P$ divise $QP = \mu_v$.

Calcul du polynôme minimal d'une suite itérée sur $\mathbb{F}^n$

Algorithme $\mathsf{MinPoly}(\boldsymbol{b}, d)$

Entrée : un vecteur $\boldsymbol{b} \in \mathbb{F}^n$ et $d \in \mathbb{N}$
Sortie : un polynôme $P \in \mathbb{F}[X]$ de degré $\leq d$

1. Si $\boldsymbol{b} = \mathbf{0}$, alors retourner le polynôme 1.
2. Choisir aléatoirement $\boldsymbol{x} \in \mathbb{F}^n$ non nul.
3. Calculer les $2d$ premiers termes de la suite scalaire $\boldsymbol{a}$ définie par $a_k = \langle \boldsymbol{x}, \boldsymbol{A}^k \boldsymbol{b} \rangle$.
4. Si $\boldsymbol{a} = \mathbf{0}$, revenir à 2.
5. Sinon, calculer le polynôme minimal $\mu_{\boldsymbol{a}}(X)$ de la suite $\boldsymbol{a}$ (*via* Berlekamp-Massey ou Euclide étendu).
6. Si $\deg \mu_{\boldsymbol{a}}(X) = d$, alors **retourner** $\mu_{\boldsymbol{a}}(X)$.
7. Sinon, calculer $\boldsymbol{b}' = \mu_{\boldsymbol{a}}(\boldsymbol{A})\boldsymbol{b}$.
8. **Retourner** $\mu_{\boldsymbol{a}}(X) \times \mathsf{MinPoly}(\boldsymbol{b}', d - \deg \mu_{\boldsymbol{a}}(X))$.

Calcul du polynôme minimal d'une suite itérée sur $\mathbb{F}^n$

Algorithme MinPoly$(\boldsymbol{b}, d)$

Entrée : un vecteur $\boldsymbol{b} \in \mathbb{F}^n$ et $d \in \mathbb{N}$
Sortie : un polynôme $P \in \mathbb{F}[X]$ de degré $\leq d$

1. Si $\boldsymbol{b} = \mathbf{0}$, alors retourner le polynôme 1.
2. Choisir aléatoirement $\boldsymbol{x} \in \mathbb{F}^n$ non nul.
3. Calculer les $2d$ premiers termes de la suite scalaire $\boldsymbol{a}$ définie par $a_k = \langle \boldsymbol{x}, \boldsymbol{A}^k \boldsymbol{b} \rangle$.
4. Si $\boldsymbol{a} = \mathbf{0}$, revenir à 2.
5. Sinon, calculer le polynôme minimal $\mu_{\boldsymbol{a}}(X)$ de la suite $\boldsymbol{a}$.
6. Si $\deg \mu_{\boldsymbol{a}}(X) = d$, alors **retourner** $\mu_{\boldsymbol{a}}(X)$.
7. Sinon, calculer $\boldsymbol{b}' = \mu_{\boldsymbol{a}}(\boldsymbol{A})\boldsymbol{b}$.
8. **Retourner** $\mu_{\boldsymbol{a}}(X) \times$ MinPoly$(\boldsymbol{b}', d - \deg \mu_{\boldsymbol{a}}(X))$.

On lance l'algorithme avec $\boldsymbol{b}$ et $d = n$.

Complexité :

– En temps : pour chaque appel, les étapes 3. et 7. coûtent $O(dnt)$, l'étape 5. coûte $O(d^2)$
Avec grande probabilité, nombre constant d'appels récursifs $\implies$ $O(tn^2)$.

– En espace : $O(tn)$

Conclusion partielle

Objectif. On voulait trouver **une solution** de

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$$

avec $\boldsymbol{A} \in \mathbb{F}^{n \times n}$ creuse, et $\boldsymbol{b} \in \mathbb{F}^n$.

Méthode de résolution.

1. Calculer le polynôme minimal $\mu_{\boldsymbol{v}}(X) = \sum_{j=0}^{d} \lambda_j X^j$ de la suite $\boldsymbol{v} = (\boldsymbol{A}^k \boldsymbol{b})_{k \in \mathbb{N}}$.
2. Calculer $Q(X) = \frac{\lambda_0 - \mu_{\boldsymbol{v}}(X)}{\lambda_0 X} \in \mathbb{F}[X]$ et retourner $Q(\boldsymbol{A})\boldsymbol{b}$.

Complexité :

- Calcul du polynôme minimal : temps $O(tn^2)$, espace $O(tn)$.
- Calcul de $Q(X)$: temps $O(n)$, espace $O(n)$.
- Calcul de $Q(\boldsymbol{A})\boldsymbol{b}$: temps $O(tn^2)$, espace $O(n)$.
- **TOTAL :** temps $O(tn^2)$, espace $O(tn)$.

Plan

Méthode de résolution $Ax = 0$

Objectif. On veut trouver **une solution** non-nulle de

$$Ax = 0$$

avec $A \in \mathbb{F}^{n \times n}$ creuse et non-inversible.

Idée de l'**algorithme de Wiedemann :**

1. Trouver le polynôme minimal $\mu_A(X) \in \mathbb{F}[X]$ de A.
2. Observer que $\mu_A(X) = XQ(X)$ et $Q(A) \neq 0$.
3. Calculer $x = Q(A)u$ pour $u \in \mathbb{F}^n$ aléatoire.
4. Si $x = 0$, revenir à l'étape 3.
5. Sinon, **retourner** x.

L'algorithme est **correct** car

$$Ax = A(Q(A)u) = (XQ(X))(A)u = \mu_A(A)u = 0$$

La probabilité que $x = 0$ à l'étape 4. est

$$\mathbb{P}[x = 0] = \mathbb{P}[u \in \ker Q(A)] \simeq \frac{|\mathbb{F}|^{\dim \ker Q(A)}}{|\mathbb{F}|^n} \leq \frac{1}{|\mathbb{F}|}$$

Polynôme minimal d'une matrice creuse

But : calculer le polynôme minimal $\mu_A(X)$ d'une matrice creuse A.

Idée : similaire au calcul du polynôme minimal d'une suite itérée sur $\mathbb{F}^n$.

- On calcule des facteurs de $\mu_A(X)$ en construisant des suites scalaires a définies par

$$a_k := \langle v, A^k w \rangle$$

  où v et w sont tirés uniformément dans $\mathbb{F}^n \setminus \{0\}$.
- Le polynôme minimal recherché $\mu_A(X)$ est le ppcm de tous les μ_a.
- On s'arrête lorsque le polynôme obtenu s'annule en A.

On procède donc de la manière suivante :

1. On initialise $P(X) = 1$.
2. **Tant que** $P(A) \neq 0$:
   - Tirer v et w dans $\mathbb{F}^n \setminus \{0\}$
   - Calculer les termes de $a_k := \langle v, A^k w \rangle$
   - Calculer le polynôme minimal $\mu_a(X)$ de a.
   - Calculer $P = \text{ppcm}(P(X), \mu_a(X))$
3. Retourner P.

Pour aller plus loin : algorithme de Wiedemann par blocs

En pratique, une idée largement développée est de regrouper les opérations par blocs.

Deux objectifs :

- Faire décroître la probabilité d'échec de l'algorithme
- Accélérer les calculs

Idée : au lieu de raisonner sur des scalaires $a_k = \boldsymbol{v}^\top \boldsymbol{A}^k \boldsymbol{w}$, on utilise de (petites) matrices

$$\boldsymbol{M}_k = \boldsymbol{V}^\top \boldsymbol{A}^k \boldsymbol{W} \in \mathbb{F}^{\alpha \times \beta}$$

Typiquement, $\boldsymbol{M}_k$ est de taille 64×64.

Alors :

- Chaque matrice $\boldsymbol{M}_k$ contient plus d'informations que les scalaires a_k.
- On aura alors besoin de moins de termes $\boldsymbol{M}_0, \ldots, \boldsymbol{M}_d$ pour obtenir un facteur de $\mu_{\boldsymbol{A}}$ (même si le calcul du polynôme annulateur devient plus compliqué).
- Par ailleurs, les calculs peuvent être parallélisés, car les opérations sur les colonnes des $\boldsymbol{M}_k$ sont indépendantes.

Remarque. C'est ce type d'algorithme qui est utilisé pour l'étape d'algèbre linéaire lors de factorisations log-discrets records.

Références

Autour de l'algorithme de Wiedemann

 *Solving sparse linear equations over finite fields*. Wiedemann. IEEE TIT. **1986**.

 *Solving linear equations over GF(2) via block Wiedemann algorithm*. Coppersmith. Math. Comp.. **1994**.

 *Subquadratic computation of vector generating polynomials and improvement of the block Wiedemann algorithm*. Thomé. J. Symbolic Comp.. **2002**.

Si vous voulez aller plus loin : adaptation de la méthode de Lanczos

 *Solving large sparse linear systems over finite fields*. LaMacchia, Odlyzko. CRYPTO. **1990**.

 *A Block Lanczos Algorithm for Finding Dependencies over GF(2)*. Montgomery. EUROCRYPT. **1995**.

Plan

Retour sur un exercice du cours

Exercice. Pour tout $\boldsymbol{x} \in \mathbb{F}^n$, la suite scalaire $\boldsymbol{a} = (a_k)_{k\in\mathbb{N}}$ définie par

$$\forall k \in \mathbb{N}, \quad a_k := \langle \boldsymbol{x}, \boldsymbol{v}_k \rangle = \sum_{j=1}^{n} x_j v_k^{(j)} \in \mathbb{F}$$

est récurrente linéaire, et son polynôme minimal $\mu_{\boldsymbol{a}}$ divise $\mu_{\boldsymbol{v}}$.

Solution. On note $\mu_{\boldsymbol{v}}(X) = \sum_{j=0}^{d} m_j X^j$.

Il suffit de démontrer que $\mu_{\boldsymbol{v}}(X)$ est un polynôme annulateur de $\boldsymbol{a}$.

Pour tout $k \geq 0$,

$$\sum_{j=0}^{d} m_j a_{k+j} = \sum_{j=0}^{d} m_j \sum_{i=1}^{n} x^{(i)} v_{k+j}^{(i)} = \sum_{i=1}^{n} x^{(i)} \sum_{j=0}^{d} m_j v_{k+j}^{(i)} = \sum_{i=1}^{n} x^{(i)} \times 0 = 0$$

Exercice : LFSR et polynôme minimal

Soit $\boldsymbol{u} = (u_n)_{n\in\mathbb{N}}$ une suite récurrente non-nulle produite par un LFSR de dimension L. On note $P(x) = \sum_{j=0}^{d} c_j X^j$ son polynôme de connexion. On rappelle qu'on a donc

$$\sum_{j=0}^{d} c_j u_{n-j} = 0, \quad c_0 = 1\,.$$

Soit $\boldsymbol{v} = (v_k)$ définie par

$$v_k := (u_k, u_{k+1}, \ldots, u_{k+L-1})^\top \in \mathbb{F}^L$$

Question 1. Démontrer que $\boldsymbol{v}$ est une suite récurrente sur $\mathbb{F}^L$. En donner une description en fonction de v_0 et d'une matrice $\boldsymbol{A}$ dont les coefficients dépendent de P.

Question 2. Que vaut $\det \boldsymbol{A}$? En déduire une condition suffisante sur P pour que $\boldsymbol{u}$ ne soit pas nulle à partir d'un certain rang.

Question 3. Démontrer que, si $\mathbb{F} = \mathbb{F}_q$, alors $\boldsymbol{u}$ a une période de taille $\leq q^L - 1$.

Soit $\widehat{P}(X) = X^d P(\frac{1}{X})$ le polynôme réciproque de $P(X)$.

Question 4. Comment déduire les coefficients de $\widehat{P}(X)$ en fonction de ceux de $P(X)$?

Question 5. Peut-on relier P et le polynôme minimal de $\boldsymbol{A}$?

Exercice : LFSR et polynôme minimal

Soit $\boldsymbol{u} = (u_n)_{n\in\mathbb{N}}$ une suite récurrente non-nulle produite par un LFSR de dimension L. On note $P(x) = \sum_{j=0}^{d} c_j X^j$ son polynôme de connexion. On rappelle qu'on a $\sum_{j=0}^{d} c_j u_{n-j} = 0$ et on suppose que $c_0 = 1$. Soit $\boldsymbol{v} = (\boldsymbol{v}_k)$ définie par

$$\boldsymbol{v}_k \coloneqq (u_k, u_{k+1}, \ldots, u_{k+L-1})^\top \in \mathbb{F}^L$$

Question 1. Démontrer que $\boldsymbol{v}$ est une suite récurrente sur $\mathbb{F}^L$. En donner une description en fonction de $\boldsymbol{v}_0$ et d'une matrice $\boldsymbol{A}$ dont les coefficients dépendent de P.

On a :

$$\boldsymbol{v}_k = \begin{pmatrix} u_k \\ u_{k+1} \\ \cdots \\ u_{k+L-1} \end{pmatrix} = \begin{pmatrix} -\sum_{j=1}^{d} c_j u_{k-j} \\ -\sum_{j=1}^{d} c_j u_{k+1-j} \\ \cdots \\ -\sum_{j=1}^{d} c_j u_{k+L-1-j} \end{pmatrix} = -\sum_{j=1}^{d} c_j \begin{pmatrix} u_{k-j} \\ u_{k+1-j} \\ \cdots \\ u_{k+L-1-j} \end{pmatrix} = -\sum_{j=1}^{d} c_j \boldsymbol{v}_{k-j}$$

Exercice : LFSR et polynôme minimal

$$v_k := (u_k, u_{k+1}, \ldots, u_{k+L-1})^\top \in \mathbb{F}^L$$

On peut écrire $v_{k+1} = Av_k$ avec :

$$A = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ & 0 & 1 & \ddots & \\ & & \ddots & \ddots & 0 \\ 0 & & & 0 & 1 \\ -c_{L-1} & -c_{L-2} & \cdots & \cdots & -c_0 \end{pmatrix}$$

Puis

$$v_k = A^k v_0$$

A est appelée *matrice compagnon* du polynôme P.

Exercice : LFSR et polynôme minimal

Question 2. Que vaut $\det \boldsymbol{A}$? En déduire une condition suffisante sur P pour que $\boldsymbol{u}$ ne soit pas nulle à partir d'un certain rang.

Question 3. Démontrer que, si $\mathbb{F} = \mathbb{F}_q$, alors $\boldsymbol{u}$ a une période de taille $\leq q^L - 1$.

Réponse 2. On a $\det \boldsymbol{A} = -c_{L-1}$

Donc $\boldsymbol{v}_k = \boldsymbol{A}^k \boldsymbol{v}_0 \neq \boldsymbol{0}$ dès lors que $c_{L-1} \neq 0$, c'est-à-dire si $\deg P$ est maximal.

Réponse 3. Le vecteur $\boldsymbol{v}_k \in \mathbb{F}^L$ peut prendre, au maximum, q^L valeurs possibles.

S'il existe k tel que $\boldsymbol{v}_k = \boldsymbol{0}$, alors pour tout $k' \geq k$, on a $\boldsymbol{v}_{k'} = \boldsymbol{0}$.
Dans ce cas $\boldsymbol{u}$ a période 1.

Sinon, il existe k, k' distants d'au plus $q^L - 1$ tels que $\boldsymbol{v}_k = \boldsymbol{v}'_k$.
On a alors, pour tout $t \geq 0$:

$$\boldsymbol{v}_{k+t} = \boldsymbol{A}^t \boldsymbol{v}_k = \boldsymbol{A}^t \boldsymbol{v}_{k'} = \boldsymbol{v}_{k'+t}$$

Exercice : LFSR et polynôme minimal

Soit $\widehat{P}(X) = X^d P(\frac{1}{X})$ le polynôme réciproque de $P(X)$.

Question 4. Comment déduire les coefficients de $\widehat{P}(X)$ en fonction de ceux de $P(X)$?

Question 5. Peut-on relier P et le polynôme minimal de $\boldsymbol{A}$?

Réponse 4.

$$\widehat{P}(X) = X^d P(\tfrac{1}{X}) = X^d \sum_{j=0}^{d} c_j X^{-j} = \sum_{j=0}^{d} c_j X^{d-j} = \sum_{j=0}^{d} c_{d-j} X^j$$

Pour obtenir $\widehat{P}$, on inverse l'ordre des coefficients de P.

Réponse 5.

$$\mathbf{0} = \mu_{\boldsymbol{A}}(\boldsymbol{A})\boldsymbol{v}_0 = \sum_{j=0}^{\ell} m_j \boldsymbol{A}^j \boldsymbol{v}_0 = \sum_{j=0}^{\ell} m_j \boldsymbol{v}_j$$

En particulier, on a donc

$$\sum_{j=0}^{\ell} m_j u_j = 0$$

c'est-à-dire, $\widehat{\mu_{\boldsymbol{A}}}$ est polynôme annulateur de $\boldsymbol{u}$.

Donc, P divise $\widehat{\mu_{\boldsymbol{A}}}$.

Exercice : algorithme d'Euclide étendu

Exercice.– Soient $\boldsymbol{b} = (b_0, \ldots, b_{2n-1})$ les $2n$ premiers termes d'une suite récurrente d'ordre n. On note $B(X) = \sum_{i=0}^{2n-1} b_i X^i$.
On souhaite retrouver un polynôme de connexion de $\boldsymbol{b}$, c'est-à-dire un polynôme $P(X)$ de degré $\leq n$ tel que $P(X)B(X) \mod X^{2n}$ est de degré $< n$.

Question 1. Rappeler l'algorithme d'Euclide étendu dans $\mathbb{F}[X]$. On pourra en donner une version « matricielle ».

Entrée : deux polynômes A et B

Sortie : des séquences de triplets $(R_i, U_i, V_i) \in \mathbb{F}[X]$ tels que

- $R_i(x) = U_i(x)A(x) + V_i(x)B(x)$
- les degrés des restes $R_i(x)$ décroissent
- le dernier $R_i(x)$ non-nul est le pgcd de $A(x)$ et $B(x)$.

Exercice : algorithme d'Euclide étendu

Les relations sont les suivantes : $R_0 = A$, $R_1 = B$, $U_0 = 0$, $U_1 = 1$, $V_0 = 1$, $V_1 = 0$.
Puis,

$$R_{i-1} = Q_i R_i + R_{i+1}, \quad \deg R_{i+1} < \deg R_i$$

Et on a également

$$U_{i-1} = Q_i U_i + U_{i+1} \quad \text{et} \quad V_{i-1} = Q_i V_i + V_{i+1}$$

Sous forme matricielle :

1. On initialise
$$\begin{pmatrix} R_1 & U_1 & V_1 \\ R_0 & U_0 & V_0 \end{pmatrix} = \begin{pmatrix} B & 0 & 1 \\ A & 1 & 0 \end{pmatrix}$$
2. Tant que $R_i \neq 0$, faire :
   - Calculer le quotient Q_i de R_{i-1} par R_i (division euclidienne)
   - Mettre à jour
$$\begin{pmatrix} R_{i+1} & U_{i+1} & V_{i+1} \\ R_i & U_i & V_i \end{pmatrix} = \begin{pmatrix} -Q_i & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} R_i & U_i & V_i \\ R_{i-1} & U_{i-1} & V_{i-1} \end{pmatrix}$$
3. Retourner $\{(R_0, U_0, V_0), \ldots, (R_i, U_i, V_i)\}$.

Exercice : algorithme d'Euclide étendu

Soit $\boldsymbol{b}$ la suite sur $\mathbb{F}_2$ dont les $2n = 8$ premiers termes sont $(0,1,1,1,0,0,1,0)$.
Question 2. Donner la série formelle $B(X)$ associée à $\boldsymbol{b}$.

Question 3. Déterminer un polynôme de connexion de degré ≤ 4 pour $\boldsymbol{b}$. On pourra utiliser l'algorithme de Berlekamp-Massey si besoin.

Question 4. Appliquer l'algorithme d'Euclide à $B(X)$ et $A(X) = X^{2n} = X^8$.

Réponse 2. On a $B(X) = X + X^2 + X^3 + X^6$.

Réponse 3. Avec Berlekamp-Massey :

k	ℓ_k	P
0	0	1
1	0	1
2	2	1
3	2	$X+1$
4	2	$X+1$
5	3	X^3+X+1
6	3	X^3+X^2+1
7	3	X^3+X^2+1
8	3	X^3+X^2+1

Réponse 4. On a

$$\begin{aligned} A(X) &= X^2 \times B(X) + (X^5 + X^4 + X^3) \\ B(X) &= (X+1) \times (X^5 + X^4 + X^3) + (X^2 + X) \\ (X^5 + X^4 + X^3) &= (X^3 + X + 1) \times (X^2 + X) + X \end{aligned}$$

C'est-à-dire :

k	R_k	U_k	V_k
0	X^8	1	0
1	$X + X^2 + X^3 + X^6$	0	1
2	$X^5 + X^4 + X^3$	1	X^2
3	$X^2 + X$	$X + 1$	$X^3 + X^2 + 1$
4	X	$X^4 + X^3 + X^2$	$X^6 + X^5 + X^4 + X^3 + X + 1$

Pour $k = 3$, on a $\deg R_3 = 2 < 4$, $\deg V_3 = 3 < 4$, et

$$U_3(X)X^8 + V_3(X)B(X) = R_3(X)$$

Autrement dit,

$$V_3(X)B(X) \equiv R_3(X) \mod X^8$$

Exercice : algorithme d'Euclide étendu

On considère l'algorithme d'Euclide étendu, appliqué à $A(x) = x^{2n}$ et $B(x)$. On note $k \geq 1$ le premier indice pour lequel $\deg R_k < n$, c'est-à-dire que $\deg R_{k-1} \geq n$.

Question 5. Quelle est la relation entre $\deg V_k$, $\deg R_{k-1}$ et $\deg A$?

Question 6. Que dire de la suite $(\deg V_i)_i$?

Question 7. Démontrer que $V_k(x)$ est un polynôme de connexion de la suite $\boldsymbol{b}$.

Réponse 5. Pour tout i (avant l'arrêt), on a

$$\deg R_{i-1} = \deg Q_i + \deg R_i \quad \text{et} \quad \deg V_{i+1} = \deg Q_i + \deg V_i \,.$$

Donc :

$$\deg V_{i+1} + \deg R_i = \deg V_i + \deg R_{i-1} = \cdots = \deg V_1 + \deg R_0 = 0 + \deg(A)$$

Puis,

$$\deg V_k + \deg R_{k-1} = 2n$$

Réponse 6. La suite $(\deg V_i)_i$ est strictement croissante (avant l'arrêt).

Réponse 7. On a donc $V_k(X)B(X) \equiv R_k(X) \mod X^{2n}$, avec $\deg R_k < n$ et $\deg V_k \leq 2n - n = n$.