



IP PARIS

BERTRAND MEYER

ALGÈBRE COMPUTATIONNELLE

DÉPARTEMENT INFRES



Copyright © 2023 B. Meyer

NOTES DU COURS ACCQ 203

Ce polycopié est (perpétuellement) en cours d'écriture et sera mis à jour en ligne au fur et à mesure de l'avancement du cours. Il est réécrit en fonction des difficultés que vous rencontrez en TP. La version en ligne comprend également quelques corrigés.

Ces notes comportent certainement beaucoup de coquilles. Tout retour sera fortement apprécié.

Version du 29 novembre 2023.

Table des matières

Syllabus 5

I ACCQ 203a : Algorithmes pour l'algèbre 7

TP 1 : Introduction à SageMath 9

TP 2 : Algèbre linéaire sur un anneau principal 19

TP 3 : Réseaux euclidiens 45

TP 4 : Factorisation partielle de polynômes univariés sur un corps fini 61

TP 5 : Factorisation complète de polynômes univariés 77

TP 6 : Bases de Gröbner et systèmes polynomiaux multivariés 89

TP 7 : Applications choisies des bases de Gröbner 113

II ACCQ 203b : Algorithmes pour l'arithmétique 135

TP 8 : Primalité des entiers 137

<i>TP 9 : Factorisation des entiers</i>	153
<i>TP 10 : Invariants de similitude et LFSR</i>	169
<i>TP 11 : Codes correcteurs d'erreurs algébriques</i>	197
<i>TP 12 : Cryptanalyse et cryptographie à base de réseaux</i>	225
<i>TP 13 : Courbes elliptiques</i>	249
<i>TP 14 : Logarithme discret et couplages</i>	269
<i>III Corrigés de certains exercices</i>	295
<i>Index</i>	381
<i>Bibliographie</i>	383

Syllabus

Contenu

Ce cours est formé de deux parties (ACCQ 203a et 203b), chacune comprenant sept séances et demi de cours & T.P. et un examen de 1h30 en fin de période. Le logiciel employé pour les T.P. est le logiciel SageMath.

La partie A traite des techniques de résolution d'équations au sens large : systèmes linéaires à coefficients dans un anneau, factorisation de polynômes, systèmes polynômiaux et leur géométrie.

La partie B aborde plus spécifiquement l'arithmétique et la théorie des nombres (primalité, factorisation des entiers, logarithme discret) et approfondit les liens entretenus avec la cryptographie (notamment en ce qui concerne les LFSR, les courbes elliptiques et les réseaux) ou le codage correcteur d'erreur.

Organisation du cours

Chaque T.P. comprend :

1. des travaux préparatoires à réaliser chez soi avant le T.P. (lecture et exercices théoriques),
2. des exercices à réaliser en classe (programmation d'algorithmes et exercices nécessitant l'aide d'un logiciel de calcul formel).

Vous êtes invités à consulter la littérature pour préparer les exercices théoriques. Les exercices de programmation peuvent être terminés chez soi durant la semaine qui suit. Ils doivent être traités en complétant le fichier réponse fourni pour chaque TP. Chaque TP fait l'objet d'une remise sur E-Campus : seul le fichier réponse complété est attendu. Les travaux demandés ainsi que la compréhension générale du sujet sont évalués lors de la séance de la semaine suivante.

Une séance se déroule selon le plan suivant :

1. présentation du contenu du TP
2. questions / réponses autour des travaux préparatoires ;
3. travail individuel ;

2bis. évaluation du TP de la séance précédente.

Evaluation

Chaque partie du cours donne lieu à une note indépendante. La note finale d'une partie est constituée d'une note de T.P. (sur 8 points) et d'une note d'examen (sur 12 points).

Chaque T.P. est évalué par une discussion en tête-à-tête avec l'enseignant lors de la séance suivante. L'examen final est formé d'une série d'exercices dont la résolution nécessite des raisonnements théoriques et l'usage d'un logiciel de calcul formel.

Fraude et plagiat

Ce cours repose pour une large part sur votre travail personnel. Discuter de vos problèmes avec vos camarades de classe et consulter des ressources extérieures fait naturellement partie de ce travail. Néanmoins, votre production doit être originale et personnelle. Un élève incapable d'expliquer ou de commenter un programme ou une manière de résoudre un exercice avec ses propres mots sera considéré comme fraudeur.

Tout élève surpris à tricher s'expose à un zéro comme note finale, à une convocation devant le jury des études et aux poursuites judiciaires que l'école pourrait décider de conduire.

Première partie

**ACCQ 203a : Algorithmes
pour l'algèbre**

TP 1 : Introduction à SageMath



Buts : Vous initier au logiciel de calcul formel et scientifique SageMath.

Comme prétexte pour découvrir SageMath, nous étudions quelques variations autour du calcul du déterminant et en particulier comme des informations modulaires permettent de reconstituer un calcul trop volumineux pour être mené de front.



Travaux préparatoires : Aucun



Évaluation du TP : Aucune évaluation

POUR UTILISER SageMath, trois options s'offrent à vous :

1. Vous pouvez appeler directement `sage -n jupyter` +  dans un terminal d'une des machines des salles de TP. Il se peut que les machines de l'école possèdent plusieurs versions du logiciel, auquel cas une complétion de commande (touche ) vous permettra de trouver la dernière version installée. En appelant `notebook()` vous aurez accès à une interface graphique¹.
2. Vous pouvez installer et utiliser SageMath directement sur votre machine personnelle en le téléchargeant depuis <http://www.sagemath.org/download.html>. Il est aussi possible de passer par des gestionnaires de paquets : pour Ubuntu, `sudo apt-get install sagemath`; pour MacOS, `brew install sage`.

SageMath fonctionne avec le langage Python : les instructions de programmation sont en Python (boucles `if`, `while`, etc.). Les commentaires sont introduits par le symbole `#`. La transition entre Python 2 et Python 3 est récente. Il se peut que certaines instructions présentées dans ce polycopié soient encore basées sur Python 2 et ne fonctionnent plus telles quelles.

SageMath possède des types plus riches que les objets Python : pour retrouver l'ensemble des méthodes existantes pour un objet donné `obj`, il est recommandé d'utiliser la complétion de commande : taper `obj.` + . On peut accéder à la documentation d'une commande par `obj.methode?` + . Nous recommandons également le manuel donné en référence².

1. Vos calculs sont alors organisés en cellules que l'on exécute avec  + .

2. Alexandre Casamayou, Nathann Cohen, Guillaume Connan, Thierry Dumont, Laurent Fousse, François Maltey, Matthias Meulien, Marc Mezzarobba, Clement Pernet, Nicolas Thiéry, and Paul Zimmermann. *Calcul mathématique avec SAGE*. CreateSpace Independent Publishing Platform, 2013

Mise en garde : vous ne devez pas importer des bibliothèques classiques telles que `numpy` ou `random`.

Pour démarrer

Un exemple de code

RIEN DE TEL que de se lancer pour voir comment SageMath fonctionne. Voici quelques lignes de code.

```
p = random_prime(10000)
q = random_prime(10000)
n = p*q
phi = (p-1)*(q-1)
```

```
e1 = ZZ.random_element(phi)
while (true):
    try:
        e2 = ZZ.random_element(phi)
        e2 = mod(e2, phi)
        d=1/e2
        break
    except ZeroDivisionError:
        pass
```

```
print("Type de e1 : ", type(e1))
print("Parent de e1 : ", parent(e1))
print("Inverse de e1 :", 1/e1)
print("Type de e2 : ", type(e2))
print("Parent de e2 : ", parent(e2))
print("Inverse de e2 :", 1/e2)
```

```
def Chiffrement(a):
    return mod(a,n)^e
Dechiffrement = lambda b : mod(b,n)^d

for a in range(4):
    Dechiffrement(Chiffrement(a))
```

Affectation de variable

Une méthode est appelée sur un objet en l'indiquant après l'objet et un point.

Boucle "si" : noter l'importance du double point et de l'indentation.

ZZ, RR, CC désignent pour SageMath les ensembles \mathbb{Z} , \mathbb{R} et \mathbb{C} .

Boucle "tant que" : noter l'importance du double point et de l'indentation.

Seul le dernier résultat d'une cellule est affiché. On peut forcer un affichage avec `print`

Fonction déclarée selon un paradigme de *programmation impérative*.

Fonction déclarée selon un paradigme de *programmation fonctionnelle*.

Boucle "pour" : noter l'importance du double point et de l'indentation.

ON REMARQUERA que tout peut être codé à la volée ou dans le cadre de fonctions.

Exercice 1. Recopiez les lignes des cartouches ci-dessus et essayez de comprendre ce qu'elles codent. Au fait, quel cryptosystème célèbre a-t-on codé ici? (Voir exemple 515)

Les listes

LES LISTES s'introduisent en Python par des crochets ($L = [1, 2, 5]$) et sont numérotées entre 0 et `len(L)-1`. Les commandes `L[p]` et `L[p:q]` renvoient respectivement l'élément L_p et la liste $[L_p, L_{p+1}, \dots, L_{q-1}]$. Si p ou q sont négatifs, les indices s'entendent comptés depuis la fin de la liste.

On peut insérer l'élément x avec `L.append(x)` ou `L.insert(i, x)`, instructions équivalentes à `L[len(L):] = [x]` et `L[i:i] = [x]`. On peut concaténer des listes avec `L1 + L2` et en répéter une i fois avec `L * i`.

On peut construire les listes *par compréhension* comme dans l'exemple : `[(k, euler_phi(k)+1) for k in range(2, 25) if is_prime(k)]`. On peut appliquer des fonctions à des listes comme suit : `map(cos, [0, pi/6, pi/4])`. On peut filtrer des listes comme suit `filter(is_prime, [1..55])`.

Quelques structures particulières

NOUS DÉTAILLONS quelques objets usuels par des exemples :

vecteurs : `v = vector(ZZ, [3, 4, 5])` qui permet d'affecter à la variable v le vecteur $(3, 4, 5) \in \mathbb{Z}^2$,

matrices : `A = matrix(ZZ, 2, 3, [[2, 3, 6], [-2, 4, 2]])` qui permet d'affecter à la variable A la matrice

$$A = \begin{pmatrix} 2 & 3 & 6 \\ -2 & 4 & 2 \end{pmatrix} \in \mathbb{Z}^{2 \times 3},$$

polynômes : `Pol.<x>=PolynomialRing(QQ)`, ce qui produit deux affectations : la variable Pol est affectée de l'anneau de polynômes $\mathbb{Q}[x]$ et la variable x est affectée de l'indéterminée de cet anneau.

On peut ensuite faire `p=x^4+x` qui affecte à p le polynôme $x^4 + x \in \mathbb{Q}[x]$

et appliquer à p toute sorte de méthodes connues de SageMath : par exemple, `p.roots(multiplicities = false)` pour les racines de p .

corps finis : `F16.<alpha> = FiniteField(16)`, produit deux affectations : à savoir $F16$ comme une version du corps fini \mathbb{F}_{16} et α qui est dans cet exemple la racine d'un polynôme irréductible de degré 4 sur \mathbb{F}_2 . Des arguments optionnels permettent de préciser le polynôme minimal de α .

Exercice 2. 1. On donne $m = 119$ et $n = 435$. Trouver une relation de Bezout entre m et n .

Utiliser la méthode `xgcd`

2. On donne $\alpha = 2$, $\beta = 3$, $m = 45$ et $n = 14$. Résoudre de deux manières possibles le système

Utiliser la méthode `crt`

$$\begin{cases} x \equiv \alpha \pmod{m} \\ x \equiv \beta \pmod{n} \end{cases}$$

d'inconnue x .

Exercice 3 (Endomorphisme de Frobenius). On fixe $p = 157$, $k = 4$ et $q = p^k = 607\,573\,201$. On note σ l'application de Frobenius de \mathbb{F}_q sur \mathbb{F}_p :

$$\sigma = \begin{cases} \mathbb{F}_q & \rightarrow \mathbb{F}_q \\ x & \mapsto x^p \end{cases}$$

On rappelle que σ est une application \mathbb{F}_p -linéaire et que $\sigma^k = \text{Id}$. On représente \mathbb{F}_q par $\mathbb{F}_p[\alpha]$ où α est une racine d'un polynôme irréductible de degré 4 dans $\mathbb{F}_p[x]$.

1. Construire la matrice \mathbf{Q} de l'endomorphisme σ dans la base $\{1, \alpha, \alpha^2, \alpha^3\}$.
Calculer de deux manières $\sigma(z)$ pour

$$z = 130\alpha^3 + 97\alpha^2 + 99\alpha + 18.$$

Vérifier sur une centaine d'exemples tirés au sort que, pour $x_0, x_1, x_2, x_3 \in \mathbb{F}_p$, on a bien

$$\begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 \end{pmatrix} \mathbf{Q} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} = (x_0 + x_1\alpha + x_2\alpha^2 + x_3\alpha^3)^p.$$

2. Définir de deux façons distinctes l'application τ inverse de σ

$$\tau = \begin{cases} \mathbb{F}_q & \rightarrow \mathbb{F}_q \\ x & \mapsto \sqrt[p]{x} \end{cases}$$

Calculer $\tau(z)$. Vérifier que les deux définitions coïncident sur une centaine d'exemples tirés au sort. [Indication : τ peut être vu comme un endomorphisme ou comme une fonction puissance.]

3. On donne le polynôme

$$f(x) = x^4 - 48x^3 + 19x^2 - 2x - 32 \in \mathbb{F}_p[x].$$

Contrôler (avec `is_irreducible`) que f est irréductible. Donner les racines de f dans la plus petite extension de \mathbb{F}_p qui les contient toutes (avec `roots`). Contrôler numériquement que les racines sont conjuguées par le Frobenius (i.e. à partir d'une quelconque des racines, on déduit les trois autres en appliquant σ).

Exercice 4 (Polynômes irréductibles et isomorphismes de corps finis).

Soient p un nombre premier et n un entier.

1. Enumérer tous les polynômes irréductibles unitaires de degré n sur \mathbb{F}_p .
2. Renvoyer tous les polynômes primitifs unitaires parmi les polynômes obtenus précédemment.
3. On note $t = p^n - 1$.
 - (a) Contrôler que la liste des polynômes primitifs unitaires compte $\phi(t)/n$ éléments (où ϕ est l'indicatrice d'Euler).
 - (b) Vérifier que le produit des polynômes primitifs unitaires obtenus est égal au t -ième polynôme cyclotomique Φ_t plongé dans $\mathbb{F}_p[x]$.
On rappelle que Φ_t est le polynôme à coefficient entier

Que font les méthodes `polynomials` et `is_irreducible`?

Voir la fonction `cyclotomic_polynomial`.

$$\Phi_t(x) = \prod_{1 \leq r \leq t \text{ t.q. } r \wedge t = 1} (x - e^{2i\pi r/n}) \in \mathbb{Z}[x].$$

4. Choisir deux polynômes irréductibles p_1 et $p_2 \in \mathbb{F}_p[x]$ de même degré n . On note α_1 et α_2 l'une de leurs n racines respectives. On définit \mathbb{F}_{p^n} de deux manières : en tant que $\mathbb{F}_p[\alpha_1]$ et en tant que $\mathbb{F}_p[\alpha_2]$. Soit $\Psi : \mathbb{F}_p[\alpha_1] \rightarrow \mathbb{F}_p[\alpha_2]$ un isomorphisme de corps. On note $\beta \in \mathbb{F}_p[\alpha_2]$ l'image $\beta = \Psi(\alpha_1)$.
 - (a) Que vaut $p_1(\beta) \in \mathbb{F}_p[\alpha_2]$?
 - (b) Combien d'isomorphismes $\Psi : \mathbb{F}_p[\alpha_1] \rightarrow \mathbb{F}_p[\alpha_2]$ y-a-t-il entre les deux copies de \mathbb{F}_{p^n} ?
 - (c) Construire l'un d'entre eux et calculer $\psi = \Psi(\alpha_1^2)$.

Utiliser la méthode `hom` du corps fini.

Exercice 5. On rappelle le

Théorème 6. Soient E, F et G des \mathbb{K} -espaces vectoriels de dimension finie, $u : E \rightarrow F$ et $v : E \rightarrow G$ des endomorphismes, alors il existe un endomorphisme $w : F \rightarrow G$ tel que $v = w \circ u$ ssi $\ker(u) \subseteq \ker(v)$.

et on donne les matrices

$$\mathbf{A} = \begin{pmatrix} -2 & 1 & 1 \\ 8 & 1 & -5 \\ 4 & 3 & -3 \end{pmatrix} \quad \text{et} \quad \mathbf{C} = \begin{pmatrix} 1 & 2 & -1 \\ 2 & -1 & -1 \\ -5 & 0 & 3 \end{pmatrix} \in \mathbb{Q}^{3 \times 3}$$

et on cherche dans $\mathbb{Q}^{3 \times 3}$ l'ensemble des solutions \mathbf{X} à l'équation

$$\mathbf{A} = \mathbf{XC}.$$

1. Vérifier que $\ker(\mathbf{C}) \subseteq \ker(\mathbf{A})$.
2. Calculer une solution particulière à l'équation $\mathbf{A} = \mathbf{XC}$.
3. Décrire l'ensemble des solutions à l'équation $\mathbf{A} = \mathbf{XC}$.

Utiliser `right_kernel`.

Utiliser `solve_left`.

Autour du calcul de déterminant

Par le pivot de Gauß

TRADITIONNELLEMENT, le déterminant d'une matrice se calcule par la méthode du pivot de Gauß et de la décomposition LU : après triangulation, il suffit de faire le produit des termes diagonaux.

Utiliser la méthode LU.

Exercice 7. Soit A la matrice

$$A = \text{matrix}(\text{ZZ}, 3, 3, [[3, -7, 2], [4, 6, -1], [-4, 3, 2]])$$

$$A = \begin{pmatrix} 3 & -7 & 2 \\ 4 & 6 & -1 \\ -4 & 3 & 2 \end{pmatrix}$$

Calculer la décomposition LU et le déterminant de A . Que peut-on dire de la place en mémoire occupée par les coefficients diagonaux U par rapport au déterminant de A ?

L'algorithme de Gauß-Bareiss

POUR ÉVITER LES DIVISIONS, on peut utiliser l'algorithme suivant, qui découle directement du théorème de Bareiss³.

Théorème 8 (Bareiss). Soit $\mathbf{M}_0 = ((a_{ij}^{(0)}))_{1 \leq i, j \leq n}$ une matrice $n \times n$. On pose $c_0 = 1$ et pour $1 \leq k < n$, on définit par récurrence

$$\forall (i, j) \in \llbracket k+1, n \rrbracket^2, \quad a_{ij}^{(k)} = \frac{1}{c_{k-1}} \begin{vmatrix} a_{k,k}^{(k-1)} & a_{k,j}^{(k-1)} \\ a_{i,k}^{(k-1)} & a_{i,j}^{(k-1)} \end{vmatrix},$$

$$\mathbf{M}_k = (a_{ij}^{(k)})_{k+1 \leq i, j \leq n} \quad \text{et} \quad c_k = a_{k,k}^{(k-1)}$$

et finalement $c_n = a_{n,n}^{(n-1)}$. Alors, toutes les divisions par c_{k-1} sont exactes et $\det(\mathbf{M}_k) = c_k^{n-k-1} \det(\mathbf{M}_0)$. En particulier

$$\det \mathbf{M}_0 = c_n.$$

Exercice 9. Écrire un programme `myGaussBareiss` qui implémenter l'algorithme 1. Faire des tests sur plus de 10000 matrices de tailles différentes.

3. Henri Cohen. *A course in computational algebraic number theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, 1993

L'accès au coefficient (i, j) (avec $0 \leq i, j \leq n-1$) d'une matrice M de taille $n \times n$ se fait avec `M[i, j]`.

Calculs modulaires

Algorithme 1 : Algorithme de Gauß-Bareiss**Entrées** : Matrice $\mathbf{M} = ((m_{i,j}))_{i,j \leq n}$ **Sorties** : Déterminant $\det \mathbf{M}$

```

1  $c \leftarrow 1; s \leftarrow 1$ 
2 pour tous les  $k \in \llbracket 1, n-1 \rrbracket$  faire
3   si  $m_{k,k} = 0$  alors
4     si  $\exists i \in \llbracket k+1, n \rrbracket, m_{i,k} \neq 0$  alors
5       pour tous les  $j \in \llbracket k, n \rrbracket$  faire
6          $m_{i,j} \leftrightarrow m_{k,j}$ 
7        $s \leftarrow -s$ 
8     sinon
9       retourner 0
10  pour tous les  $i \in \llbracket k+1, n \rrbracket$  faire
11    pour tous les  $j \in \llbracket k+1, n \rrbracket$  faire
12       $m_{i,j} \leftarrow (m_{k,k}m_{i,j} - m_{i,k}m_{k,j})/c$ 
13   $c \leftarrow m_{k,k}$ 
14 retourner  $sm_{n,n}$ 

```

ALTERNATIVEMENT, une autre technique permet d'éviter l'explosion des coefficients entiers au court d'un calcul : travailler modulo un grand entier N , voire travailler modulo de nombreux entiers et utiliser le théorème des restes chinois pour reconstituer le résultat.

On rappelle

Proposition 10 (Borne de Hadamard). Soit \mathbf{A} une matrice de $\mathbb{R}^{n \times n}$ et \mathbf{A}_i la i -ème colonne de \mathbf{A} . Alors

$$|\det \mathbf{A}| \leq \|\mathbf{A}_1\|_2 \cdot \|\mathbf{A}_2\|_2 \cdots \|\mathbf{A}_n\|_2.$$

Exercice 11. 1. À quelle différence doit-on s'attendre si l'on calcule le déterminant par la méthode du pivot de Gauß dans \mathbb{Q} et dans \mathbb{F}_{p_k} ?

2. Programmer et expliquer l'algorithme⁴ décrit ci-dessus.

Exemple 12. On souhaite calculer le déterminant de la matrice

$$\mathbf{A} = \begin{pmatrix} 11 & 13 & 1 \\ -1 & 1 & -2 \\ -1 & -1 & 3 \end{pmatrix}.$$

Un calcul du déterminant par la méthode de Gauß donnerait une factorisation LU avec

$$\mathbf{L} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{11} & 1 & 0 \\ -\frac{1}{11} & \frac{1}{12} & 1 \end{pmatrix} \quad \mathbf{U} = \begin{pmatrix} 11 & 13 & 1 \\ 0 & \frac{24}{11} & -\frac{21}{11} \\ 0 & 0 & \frac{13}{4} \end{pmatrix}$$

4. On peut utiliser les méthodes column, norm et crt pour coder l'algorithme modulaire.

Algorithme 2 : Calcul modulaire du déterminant**Entrées** : Matrice \mathbf{M} **Sorties** : Déterminant $\det \mathbf{M}$

```

1  $B \leftarrow$  Borne d'Hadamard sur  $M$ 
2 Générer des nombres premiers distincts  $p_1, \dots, p_k$  tels que
    $N = \prod_{i=1}^k p_i > 2B + 1$ 
3 pour tous les  $i \in [1, k]$  faire
4    $D_k \leftarrow \det \mathbf{M} \pmod{p_k}$ 
5 Résoudre le système suivant dans  $\mathbb{Z} \cap [-B, B]$ 
6   
$$\begin{cases} D \equiv D_1 \pmod{p_1} \\ \vdots \\ D \equiv D_k \pmod{p_k} \end{cases}$$

7 retourner  $D$ 

```

D'où l'on tirerait $\det \mathbf{A} = 11 \cdot \frac{24}{11} \cdot \frac{13}{4} = 78$.

La borne de Hadamard donne

$$|\det \mathbf{A}| \leq 3\sqrt{32718} \leq 543.$$

En travaillant modulo p , on obtient, toujours en appliquant la méthode de Gauß,

$$p = 3, \mathbf{L} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \mathbf{U} = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 0 \end{pmatrix}, \det \mathbf{A} = 0 \pmod{3}$$

$$p = 5, \mathbf{L} = \begin{pmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 4 & 3 & 1 \end{pmatrix}, \mathbf{U} = \begin{pmatrix} 1 & 3 & 1 \\ 0 & 4 & 4 \\ 0 & 0 & 2 \end{pmatrix}, \det \mathbf{A} = 3 \pmod{5}$$

$$p = 7, \mathbf{L} = \begin{pmatrix} 1 & 0 & 0 \\ 5 & 1 & 0 \\ 5 & 3 & 1 \end{pmatrix}, \mathbf{U} = \begin{pmatrix} 4 & 6 & 1 \\ 0 & 6 & 0 \\ 0 & 0 & 5 \end{pmatrix}, \det \mathbf{A} = 1 \pmod{7}$$

$$p = 11, \mathbf{L} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 10 & 1 \end{pmatrix}, \mathbf{U} = \begin{pmatrix} 10 & 1 & 9 \\ 0 & 2 & 1 \\ 0 & 0 & 6 \end{pmatrix}, \det \mathbf{A} = 1 \pmod{11}$$

Le système

$$\begin{cases} x \equiv 0 \pmod{3} \\ x \equiv 3 \pmod{5} \\ x \equiv 1 \pmod{7} \\ x \equiv 1 \pmod{11} \end{cases}$$

a pour solution $x \equiv 78 \pmod{3 \cdot 5 \cdot 7 \cdot 11}$. Par ailleurs, comme $3 \cdot 5 \cdot 7 \cdot 11 \geq 2 \cdot 543 + 1$, on est certain que $\det \mathbf{A} = 78$.

Correction d'erreurs

CALCULER un résultat r modulo différents premiers distincts $(p_i)_{i \leq k}$ et retrouver r par le théorème des restes chinois comme dans l'exemple ci-dessus est utilisé couramment pour contourner des problèmes de taille d'objets à manipuler. Que se passe-t-il lorsque certains calculs modulaires sont erronés? Autrement dit, soient r un entier, $(p_i)_{i \leq k}$ une suite de nombre premiers distincts et r_i le reste de r modulo p_i . On suppose connu une suite $(a_i)_{i \leq k}$ telle que $a_i = r_i$ pour tout $i \leq k$ sauf pour certains indices $i \in I$ pour lesquels une erreur s'est produite. Malheureusement I est inconnu. Est-il encore possible de retrouver r à partir de la suite $(a_i)_{i \leq k}$ seulement (sous réserve que $|I|$ soit petit et k grand)?

Dans ce qui suit, nous supposons que $r \in [0, N-1]$ pour un certain N et nous notons $P = \prod_{i \leq k} p_i$. Par le théorème des restes chinois, il existe $a \in [0, P-1]$ tel que $a \equiv a_i \pmod{p_i}$ pour tout $i \leq k$.

On note E l'entier $E = \prod_{i \in I} p_i$. L'entier E n'est pas calculable a priori mais se prête à l'équation

$$aE \equiv rE \pmod{P}. \quad (1)$$

Soit $t \in \mathbb{Z}$ tel que $aE = rE + tP$. Il s'en suit que

$$0 \leq \frac{a}{P} - \frac{t}{E} < \frac{N}{P}.$$

Sous l'hypothèse $E < 2 \left(\frac{P}{N} \right)^2$, les entiers t et E peuvent être calculés à partir de a et de P en tant que convergent de $\frac{a}{N}$.

Définition 13. On appelle *développement en fractions continues* de $x \in \mathbb{R}_+^*$ les termes de la suite $(y_n)_{n \in \mathbb{N}} \subseteq \mathbb{Z}$ telle que

$$x = y_0 + \frac{1}{y_1 + \frac{1}{y_2 + \frac{1}{y_3 + \frac{1}{y_4 + \dots}}}}$$

et k -ième convergent la même somme tronquée au k -ième terme.

Théorème 14. Soit $x \in \mathbb{R}_+$. S'il existe des entiers p et q tels que

$$\left| x - \frac{p}{q} \right| \leq \frac{1}{2q^2}$$

alors $\frac{p}{q}$ est l'un des convergents de x .

Exemple 15. Nous cherchons toujours à calculer le déterminant de la matrice

$$\mathbf{A} = \begin{pmatrix} 11 & 13 & 1 \\ -1 & 1 & -2 \\ -1 & -1 & 3 \end{pmatrix}.$$

Par la borne de Hadamard, nous savons que $|\det \mathbf{A}| \leq N - 1$ avec $N = 543$. Supposons que nous ayons pu déterminer, par divers calculs modulaires faciles à conduire, que

$$\det \mathbf{A} \equiv \begin{cases} 0 & \text{mod } 2 \\ 0 & \text{mod } 3 \\ 3 & \text{mod } 5 \\ 1 & \text{mod } 7 \\ -1 & \text{mod } 11 \\ 0 & \text{mod } 13 \\ 10 & \text{mod } 17 \\ 2 & \text{mod } 19 \end{cases}$$

mais que, potentiellement, certains d'entre eux soient erronés. Notons $P = 2 \cdot 3 \cdot 5 \cdots 17 \cdot 19 = 9699690$. Le système de congruence admet comme solution $a = 7054398$. Une erreur s'est manifestement introduite dans les calculs puisque ni a , ni $a - P$ ne satisfait la borne d'Hadamard. On étudie le développement de $x = \frac{a}{P} = \frac{90441}{124355}$. Le développement en fraction continue de x est

$$0, 1, 2, 1, 2, 1027, 2, 1, \dots$$

Les convergents successifs sont

$$1, \frac{2}{3}, \frac{3}{4}, \frac{8}{11}, \frac{8219}{11301}, \frac{16446}{22613}, \frac{24665}{33914}, \dots$$

On peut calculer $x - p/q$ où p/q est l'un de ces convergents. Le premier pour lequel $|x - p/q| \leq N/P$ est le convergent $\frac{8}{11}$. On en déduit que $E = 11$ et que la congruence de $(\det \mathbf{A} \bmod 11)$ est incorrecte. On relève le système

$$\det \mathbf{A} \equiv \begin{cases} 0 & \text{mod } 2 \\ 0 & \text{mod } 3 \\ 3 & \text{mod } 5 \\ 1 & \text{mod } 7 \\ 0 & \text{mod } 13 \\ 10 & \text{mod } 17 \\ 2 & \text{mod } 19 \end{cases}$$

qui donne bien 78 comme déterminant.

- Exercice 16.** 1. Programmer un algorithme récursif `myFractionContinue` calculant les k premiers termes du développement en fractions continues (On remarquera que y_0 est la partie entière de x).
2. Donner un algorithme qui, étant donné la suite finie des k premiers termes du développement en fractions continues, calcule le k -ième convergent.
3. Proposer et implémenter un algorithme qui essaye de retrouver r en fonction de la suite $(a_i)_{i \leq k}$.

TP 2 : Algèbre linéaire sur un anneau principal

Buts : S'initier à l'algèbre linéaire sur un anneau principal (et non un corps) via la manipulation de matrices. Savoir résoudre des systèmes d'équations linéaires à coefficient entier. Généraliser le théorème de structure des groupes abéliens

Travaux préparatoires : Cours et exercices 55 (question 1 et 2), 64 (question 1) & 46.

Évaluation du TP : Exercices 55, questions 3 à 6 (mise sous HNF) & 64 questions 3 à 6 (mise sous SNF) puis 59 (système linéaire homogène sur \mathbb{Z}), 60 (image d'une matrice), 67 (système linéaire non-homogène sur \mathbb{Z}), 80 (structure du quotient) & 94 (facteurs invariants).

Des questions aussi simples que « décrire les solutions entières d'un système linéaire entier » ne sont pas aussi facile qu'il n'y paraît. En effet, le traditionnel algorithme du pivot de Gauß utilise des divisions et renvoie un résultat rationnel finalement assez peu exploitable si l'on souhaite seulement des entiers.

Exemple 17. Lorsque qu'on applique la méthode du pivot de Gauß au système

$$\begin{cases} 40x + 70y + 20z = -60 \\ 20x + 50y + 60z = 40 \end{cases}$$

on obtient comme ensemble de solution dans \mathbb{Q} l'ensemble

$$\begin{pmatrix} -29/3 \\ 14/3 \\ 0 \end{pmatrix} + \mathbb{Q} \begin{pmatrix} 1 \\ -5/8 \\ 3/16 \end{pmatrix}.$$

Cette présentation n'est pas opérationnelle pour en déduire l'ensemble des solutions à valeur dans \mathbb{Z} .

Ce T.P. revisite les notions basiques d'algèbre linéaire sur un anneau principal (par exemple l'anneau des entiers \mathbb{Z} ou des polynômes sur un corps $\mathbb{K}[x]$). On rappelle la

Définition 18. Un anneau A est *principal* s'il est intègre et si tout idéal I de A est principal, c-à-d. de la forme $I = aA$ pour un certain $a \in A$. Il est, de plus, *euclidien* s'il possède une division euclidienne.

L'exercice suivant présente un exemple d'anneau principal.

Exercice 19. On note $i = \sqrt{-1}$ et

$$\mathcal{G} = \mathbb{Z}[i] = \{\alpha + i\beta; (\alpha, \beta) \in \mathbb{Z}\}$$

l'anneau des *entiers de Gauß*. Si $a = \alpha + i\beta \in \mathbb{Z}[i]$, on appelle norme de a l'entier $\mathcal{N}(a) = |a|^2 = \alpha^2 + \beta^2$.

1. Montrer que pour tout a et b dans $\mathbb{Z}[i]$, il existe q et $r \in \mathbb{Z}[i]$ tels que

$$a = q \cdot b + r \quad \text{et } \mathcal{N}(r) < \mathcal{N}(b).$$

Que peut-on dire de $\mathbb{Z}[i]$?

2. Trouver $\mathbb{Z}[i]^\times$.
3. Montrer que les nombres de premiers a de $\mathbb{Z}[i]$ sont, à des inversibles près, de la forme :
 - $a = 1 \pm i$ si $\mathcal{N}(a) = 2$
 - $a = \alpha \pm i\beta$ avec $\alpha, \beta \in \mathbb{N}$ si $\mathcal{N}(a) = p$ est premier dans \mathbb{Z} et $p \equiv 1 \pmod{4}$
 - $a = p$ où p est premier dans \mathbb{Z} et $p \equiv 3 \pmod{4}$.
4. Soit $a \in \mathbb{Z}[i]$. Décrire un ensemble de représentants de $\mathbb{Z}[i]/a\mathbb{Z}[i]$. Combien y-a-t-il d'éléments dans le quotient ?

Les modules sur un anneau

Les définitions

AVANT DE COMMENCER, nous devons adapter la notion de \mathbb{K} -espace vectoriel aux anneaux.

Définition 20. Soit $(A, +_A, \times_A)$ un anneau commutatif. Un A -module M est un groupe abélien muni d'une loi de composition interne, notée $+_M$, et d'une loi de composition externe (multiplication scalaire) $A \times M \rightarrow M$, notée \times_M , qui satisfait :

1. $\forall \lambda \in A, \forall \mathbf{v} \in M, \quad 1_A \times_M \mathbf{v} = \mathbf{v}$
2. $\forall \lambda, \mu \in A, \forall \mathbf{v} \in M, \quad (\lambda \times_A \mu) \times_M \mathbf{v} = \lambda \times_M (\mu \times_M \mathbf{v})$
3. $\forall \lambda, \mu \in A, \forall \mathbf{v} \in M, \quad (\lambda +_A \mu) \times_M \mathbf{v} = \lambda \times_M \mathbf{v} +_M \mu \times_M \mathbf{v}$
4. $\forall \lambda \in A, \forall \mathbf{v}, \mathbf{w} \in M, \quad \lambda \times_M (\mathbf{v} +_M \mathbf{w}) = \lambda \times_M \mathbf{v} +_M \lambda \times_M \mathbf{w}.$

Exemple 21. L'ensemble des entiers de Gauß

$$\mathbb{Z}[i] = \{\alpha + \beta i; (\alpha, \beta) \in \mathbb{Z}\}$$

est un \mathbb{Z} -module.

Exemple 22. Soit \mathbb{K} un corps, $A = \mathbb{K}[x]$, E un \mathbb{K} -espace vectoriel et u un endomorphisme de E . Alors E muni de la loi de composition externe

$$\begin{cases} A \times E & \rightarrow & E \\ (p, v) & \mapsto & [p(u)](v) \end{cases}$$

est un A -module.

Remarque 23. Ces axiomes sont exactement les mêmes que les axiomes d'un espace vectoriel. Quelle est la différence avec un espace vectoriel ? Ici, A est seulement un anneau. Lorsque A est un corps, un A -module est un A -espace vectoriel.

Exercice 24. 1. Peut-on munir $\mathbb{Z}/n\mathbb{Z}$ d'une structure de \mathbb{Q} -module ?

2. Soit \mathbb{K} un corps et $\mathbb{K}(x)$ le corps des fractions rationnelles sur \mathbb{K} . Soit $f \in \mathbb{K}[x]$ un polynôme. Peut-on munir $\mathbb{K}[x]/(f(x))$ d'une structure de $\mathbb{K}(x)$ -module ?

Exercice 25. À quelle condition sur le nombre premier p peut-on munir le groupe abélien $(\mathbb{F}_p, +)$ d'une structure de $\mathbb{Z}[\sqrt{-1}]$ -module ?

On peut toujours définir $n \cdot x$ par $x + \dots + x$ pour tout entier n , si bien que :

Proposition 26. La notion de \mathbb{Z} -module et la notion de groupe abélien coïncident.

Exercice 27. Montrer que la notion de $\mathbb{Z}[\sqrt{-1}]$ -module coïncide avec la notion de groupe abélien muni d'un endomorphisme de groupe ϕ tel que $\phi^2 = -\text{Id}$.

Quelques propriétés

Définition 28. On appelle *somme* de deux A -modules M et M' et on note $M + M'$ le A -module

$$M + M' = \{\mathbf{v} + \mathbf{v}'; \mathbf{v} \in M, \mathbf{v}' \in M'\}.$$

On dit que la somme est *directe* et on note $M \oplus M'$ si de plus $M \cap M' = \{0\}$.

Exemple 29. Pour tout entier naturel n , A^n est un A -module appelé *module libre*. \triangle Tout A -module n'est pas de cette forme.

Exemple 30. La somme directe $\mathbb{Z}^r \oplus (\mathbb{Z}/2\mathbb{Z}) \oplus (\mathbb{Z}/3\mathbb{Z})$ est un \mathbb{Z} -module.

Définition 31. Une partie M' d'un A -module M stable par addition et multiplication scalaire s'appelle un *sous- A -module*. (Si $M = A$, on parle d'idéal).

Definition 32. On appelle *homomorphisme de A-modules* toute application $\phi : M \rightarrow M'$ entre deux A-modules M et M' telle que

$$\phi(\mathbf{v} + \mathbf{w}) = \phi(\mathbf{v}) + \phi(\mathbf{w}) \quad \text{et} \quad \phi(\lambda \mathbf{v}) = \lambda \phi(\mathbf{v})$$

pour tous $\mathbf{v}, \mathbf{w} \in M$ et $\lambda \in A$.

Un morphisme bijectif est appelé *isomorphisme de A-module*. Le noyau de ϕ est un sous-module de M noté $\ker \phi$ et l'image de ϕ un sous-module de M' .

Remarque 33. Un morphisme de modules est *injectif* si et seulement si $\ker \phi = \{0\}$.

Definition 34. Étant donné un module M et un sous-module N de M , on définit le *module quotient* M/N comme l'ensemble des classes $[\mathbf{v}] = \mathbf{v} + N$ munies des lois

$$[\mathbf{v}] + [\mathbf{w}] = [\mathbf{v} + \mathbf{w}] \quad \lambda[\mathbf{v}] = [\lambda \mathbf{v}]$$

pour tous $\mathbf{v}, \mathbf{w} \in M$ et $\lambda \in A$.

Proposition 35. L'application canonique $\mathbf{v} \mapsto [\mathbf{v}]$ est un morphisme de A-modules de noyau N .

Le passage au quotient permet de simplifier des morphismes.

Proposition 36. Si $\phi : M \rightarrow N$ est un morphisme surjectif, alors ϕ induit un isomorphisme

$$\bar{\phi} : M / \ker \phi \rightarrow N.$$

Exercice 37. Montrer que si M_1, \dots, M_r sont des A-modules et N_i est un sous-module de M_i pour $1 \leq i \leq r$, alors

$$\left(\bigoplus_{i=1}^r M_i \right) / \left(\bigoplus_{i=1}^r N_i \right) = \bigoplus_{i=1}^r (M_i / N_i)$$

Comme pour les espaces vectoriels, on s'intéresse à des notions de famille génératrice et de base.

Definition 38. Une famille $\mathcal{F} = (\mathbf{v}_i)_{i \in I}$ d'éléments d'un A-module M est dite

1. *génératrice* si M est égal à l'ensemble des combinaisons linéaires

$$\lambda_1 \mathbf{v}_{i_1} + \dots + \lambda_k \mathbf{v}_{i_k} \quad \text{où} \quad k \in \mathbb{N}, (\lambda_{i_j})_{j \leq k} \subseteq A, (i_j)_{j \leq k} \subseteq I.$$

2. *linéairement indépendante* si (avec les mêmes notations et des indices distincts)

$$\lambda_1 \mathbf{v}_{i_1} + \dots + \lambda_k \mathbf{v}_{i_k} = 0$$

implique que tous les coefficients λ_i sont nuls.

3. une *base* de M si elle est génératrice et linéairement indépendante.

Contre-exemple 39. $\mathbb{Z}/n\mathbb{Z}$ ne contient ni famille libre ni base sur \mathbb{Z} .

△

Définition 40. Un module M est dit *de type fini* s'il admet une famille génératrice finie.

Désormais, $\langle \mathbf{v}_1, \dots, \mathbf{v}_k \rangle_A$ désignera le sous- A -module de M engendré par les éléments $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$.

Exemple 41. Le \mathbb{Z} -module \mathbb{Z}^n est de type fini mais le \mathbb{Z} -module \mathbb{Q} n'est pas de type fini.

Exercice 42. Soit M un module possédant une famille de g générateurs, montrer que toute famille de $g + 1$ vecteurs est liée.

Proposition 43. Deux bases d'un même module ont même cardinal.

Démonstration. Pour justifier que la dimension est unique, on peut raisonner avec des matrices. Supposons qu'il existe deux bases $B = (b_i)$ et $B' = (b'_j)$ de cardinal r et s . Soit X la matrice $s \times r$ dont la i -ième colonne exprime les coefficients de b_i dans B' . De même soit X' la matrice $r \times s$ dont la i -ième colonne exprime les coefficients de b'_i dans B . Alors $XX' = I_s$. Si disons $s > r$, on peut ajouter $s - r$ colonnes nulles à X et $s - r$ lignes nulles à X' et on aurait encore $\tilde{X}\tilde{X}' = I_s$. Mais $\det \tilde{X} \det \tilde{X}' = 1_A$ ce qui n'est pas possible. □

Définition 44. Un A -module M possédant une base s'appelle un *module libre*. S'il est de type fini, il est isomorphe à A^n , où n est la taille d'une de ses bases, et n est appelé la *dimension* de M .

Exercice 45. Montrer que le \mathbb{Z} -module (\mathbb{Q}^*, \times) est se décompose sous la forme $\mathbb{Z}/2\mathbb{Z} \oplus M$ où M est libre de base dénombrable. Proposer une base de M .

Exercice 46. Soit \mathbb{K} un corps, $\alpha \in \mathbb{K}$ et $M = \mathbb{K}[x]/(x - \alpha)^r$. Quelle est la dimension de M en temps que \mathbb{K} -espace vectoriel ? M est-il un module de type fini sur $\mathbb{K}[x]$? Si oui, combien de générateurs faut-il au minimum pour engendrer M ? M -est-il libre ?

Exercice 47. Soit $A = \mathbb{C}[\epsilon]$ avec $\epsilon^2 = 0$. Décrire tous les A -modules de type fini.

Operations élémentaires sur une matrice

NOUS RAPPELONS dans cette section les effets d'une multiplication d'une matrice par des matrices élémentaires.

Definition 48. On appelle *matrice élémentaire*⁵ les matrices de $A^{n \times n}$ suivantes :

5. Avec SageMath, `elementary_matrix()`.

1. les matrices de *transposition*

$$\mathbf{P}_{i,j} = \begin{matrix} & i & & & j & \\ & & & & & \\ i & \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & & & 1 & \vdots \\ \vdots & & 1 & & & \vdots \\ \vdots & & & 1 & & \vdots \\ \vdots & 1 & & & 0 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 1 \end{pmatrix} \end{matrix}$$

où $1 \leq i < j \leq n$,

2. les matrices de *dilatation* de rapport inversible $\lambda \in A^\times$

$$\mathbf{D}_{i,\lambda} = \begin{matrix} & i & & & & \\ & & & & & \\ i & \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \lambda & & & & \vdots \\ \vdots & & 1 & & & \vdots \\ \vdots & & & 1 & & \vdots \\ \vdots & & & & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 1 \end{pmatrix} \end{matrix}$$

où $1 \leq i \leq n$ et $\lambda \in A^\times$,

3. les matrices de *transvection* de rapport $\lambda \in A$ quelconque

$$\mathbf{T}_{i,j,\lambda} = \begin{matrix} & & & & j & \\ & & & & & \\ i & \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & & & \lambda & \vdots \\ \vdots & & 1 & & & \vdots \\ \vdots & & & 1 & & \vdots \\ \vdots & & & & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 1 \end{pmatrix} \end{matrix}$$

où $1 \leq i < j \leq n$ et $\lambda \in A$.

Les matrices élémentaires sont inversibles sur A (leur déterminant appartient à A^\times).

Définition 49. On appelle matrice de *Bezout* la matrice inversible

$$\mathbf{B}_{i,j;s,t,u,v} = \begin{matrix} & & & & j \\ i & \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & s & & & t & \vdots \\ \vdots & & 1 & & & \vdots \\ \vdots & & & 1 & & \vdots \\ \vdots & & & & & \vdots \\ j & \vdots & u & & v & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 1 \end{pmatrix} \end{matrix}$$

où $1 \leq i < j \leq n$ et $s, t, u, v \in A$ vérifient $sv - ut \in A^\times$.

Proposition 50. Soit $\mathbf{X} \in A^{m \times n}$ une matrice dont les lignes sont notées L_1, \dots, L_m et les colonnes C_1, \dots, C_n . La multiplication à gauche, respectivement à droite, de \mathbf{X} a pour effet sur les lignes, respectivement sur les colonnes

Matrice	Effet à gauche	Effet à droite
$\mathbf{P}_{i,j}$	$L_i \leftrightarrow L_j$	$C_i \leftrightarrow C_j$
$\mathbf{D}_{i,\lambda}$	$L_i \leftarrow \lambda L_i$	$C_i \leftarrow \lambda C_i$
$\mathbf{T}_{i,j,\lambda}$	$L_i \leftarrow L_i + \lambda L_j$	$C_j \leftarrow C_j + \lambda C_i$
$\mathbf{B}_{i,j;s,t,u,v}$	$\begin{cases} L_i \leftarrow sL_i + tL_j \\ L_j \leftarrow uL_i + vL_j \end{cases}$	$\begin{cases} C_i \leftarrow sC_i + uC_j \\ C_j \leftarrow tC_i + vC_j \end{cases}$

Remarque 51. L'opération de Bezout sert à annuler un coefficient en jouant avec une autre ligne ou une autre colonne comme suit. Supposons qu'une matrice $\mathbf{X} \in A^{m \times n}$ ait pour coefficients particuliers $x_{k,i} = a$ et $x_{k,j} = b$. Notons $d \in A$ le pgcd de a et b . Considérons des scalaires u et $v \in A$ tels que

$$au + bv = d \quad (\text{Relation de Bezout}).$$

Alors en effectuant

$$\begin{cases} C_i \leftarrow uC_i + vC_j \\ C_j \leftarrow -(b/d)C_i + (a/d)C_j \end{cases}$$

on annule le coefficient $x_{k,j}$. Visuellement :

$$\begin{pmatrix} * & * & * & * & * \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ * & * & * & * & * \\ * & \mathbf{a} & * & \mathbf{b} & * \\ * & * & * & * & * \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ * & * & * & * & * \end{pmatrix} \text{ devient } \begin{pmatrix} * & * & * & * & * \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ * & * & * & * & * \\ * & \mathbf{d} & * & \mathbf{0} & * \\ * & * & * & * & * \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ * & * & * & * & * \end{pmatrix}$$

Forme normale de Hermite

L'algorithme

Nous fixons pour tout $a \in A$ un représentant par classe de $A/(a)$ de sorte que $x \bmod a$ ait un sens⁶. Nous notons A^+ un ensemble de représentants des classes d'équivalence de A/A^\times ⁷.

Définition 52 (Forme normale d'Hermite). On dit qu'une matrice $\mathbf{X} \in A^{m \times n}$ est sous forme normale d'Hermite si

- i. toutes les colonnes nulles sont regroupées à gauche dans la matrice \mathbf{X}
- ii. le dernier élément non-nul d'une colonne de \mathbf{X} est réduit multiplicativement modulo A^\times , on l'appelle *directeur*,
- iii. entre deux colonnes successives de \mathbf{X} , le directeur de la colonne de droite se trouve strictement plus bas que le directeur de la colonne de gauche
- iv. dans toute ligne contenant un directeur, les coefficients à droite d'un directeur sont réduits additivement modulo celui-ci.

autrement dit s'il existe un entier $r \leq n$ et une fonction strictement croissante f de $\llbracket r+1, n \rrbracket$ dans $\llbracket 1, m \rrbracket$ tels que

1. les r premières colonnes de \mathbf{X} sont nulles,
2. pour tout $r+1 \leq j \leq n$, la j -ième colonne vérifie
 - (a) $x_{f(k),j}$ est réduit modulo $x_{f(k),k}$, si $k < j$,
 - (b) $x_{f(j),j} \in A^+$, $x_{f(j),j} \neq 0$ et
 - (c) $x_{i,j} = 0$, si $i > f(j)$.

$$\begin{pmatrix} 0 & \times & \times & \times & \times \\ 0 & x_{f(r+1),r+1} & * & * & * \\ \vdots & 0 & \times & \times & \times \\ \vdots & \vdots & \times & \times & \times \\ \vdots & & x_{f(r+2),r+2} & * & * \\ \vdots & & 0 & x_{f(r+3),r+3} & * \\ \vdots & & & \ddots & \times \\ 0 & 0 & \dots & \dots & x_{f(n),n} \end{pmatrix}$$

On rappelle que

Lemme 53. Soient n un entier et $\mathbf{X} \in A^{n \times n}$ une matrice sur un anneau A , alors \mathbf{X} est inversible (i.e. $\mathbf{X} \in \text{GL}_n(A)$) si et seulement si $\det \mathbf{X}$ est une unité de A .

6. On peut choisir $x \in \llbracket 0, a-1 \rrbracket$ quand $a \in \mathbb{Z}$ ou l'unique reste par division euclidienne quand $a \in \mathbb{K}[x]$.
7. On peut choisir $A^+ = \mathbb{N}$ quand $A = \mathbb{Z}$ ou A^+ égal aux polynômes unitaires si $A = \mathbb{K}[x]$.

FIGURE 1: Vue d'une matrice HNF réduite. Les $*$ correspondent à des coefficients contraints et les \times à des coefficients non contraints.

Théorème 54. Soit $\mathbf{X} \in A^{m \times n}$ une matrice. Alors il existe une unique matrice $\mathbf{H} \in A^{m \times n}$ sous forme normale d'Hermite telle que $\mathbf{H} = \mathbf{X}\mathbf{U}$ avec $\mathbf{U} \in GL_n(A)$.

Démonstration. Nous justifions le théorème par l'algorithme 4 que nous présentons par simplicité uniquement dans le cas où $A = \mathbb{Z}$.

L'algorithme consiste à arranger les coefficients de \mathbf{X} en utilisant comme pivot le coefficient $x_{i,k}$ qui part du coin inférieur droit et remonte en direction du coin supérieur gauche. Le pivot sert à annuler le début de la ligne i et à réduire la fin de la ligne i .

Algorithme 3 : Mise sous forme normale d'Hermite (avec les relations de Bezout)

Entrées : Matrice $\mathbf{X} = [X_1 | \dots | X_n] \in A^{m \times n}$

Sorties : Forme normale d'Hermite de \mathbf{X}

```

1  $\ell \leftarrow \max(1, m - n + 1)$ 
2  $i \leftarrow m, k \leftarrow n$ 
3 tant que  $i \geq \ell$  faire
4   si  $\exists j_0 \leq k, x_{i,j_0} \neq 0$  alors
5     Trouver  $j_0$  tel que  $x_{i,j_0} \neq 0$ 
6      $X_{j_0} \leftrightarrow X_k$ 
7     si  $x_{i,k} \notin A^+$  alors
8        $X_k \leftarrow \epsilon X_k$  où  $\epsilon \in A^\times$  tel que  $\epsilon x_{i,k} \in A^+$ 
9     pour tous les  $j < k$  faire
10      Calculer une relation de Bezout  $ux_{i,k} + vx_{i,j} = d$ 
11      Poser  $s \leftarrow -x_{i,j}/d, t \leftarrow x_{i,k}/d$ 
12      Effectuer simultanément  $C_j \leftarrow sC_k + tC_j$  et  $C_k \leftarrow uC_k + vC_j$ 
13    pour tous les  $j > k$  faire
14       $X_j \leftarrow X_j - \lfloor x_{i,j}/x_{i,k} \rfloor X_k$ 
15     $i \leftarrow i - 1, k \leftarrow k - 1$ 
16  sinon
17     $i \leftarrow i - 1$ 
18 retourner  $\mathbf{X}$ 
```

Dans le cas de \mathbb{Z} , on peut se passer de la relation de Bezout par tout un jeu de combinaison des valeurs du début de la ligne i (lignes 5 à 11 du code) pour obtenir la bonne valeur de $x_{i,k}$. Ceci conduit à l'algorithme

Algorithme 4 : Mise sous forme normale d'Hermite (dans \mathbb{Z})

Entrées : Matrice $\mathbf{X} = [X_1 | \dots | X_n] \in \mathbb{Z}^{m \times n}$
Sorties : Forme normale d'Hermite de \mathbf{X}

```

1  $\ell \leftarrow \max(1, m - n + 1)$ 
2  $i \leftarrow m, k \leftarrow n$ 
3 tant que  $i \geq \ell$  faire
4   si  $\exists j_0 \leq k, x_{i,j_0} \neq 0$  alors
5     tant que  $\exists j < k, x_{i,j} \neq 0$  faire
6       Trouver  $j_0$  tel que  $x_{i,j_0} \neq 0$ 
7        $X_{j_0} \leftrightarrow X_k$ 
8       si  $x_{i,k} < 0$  alors
9          $X_k \leftarrow -X_k$ 
10      pour tous les  $j < k$  faire
11         $X_j \leftarrow X_j - \lfloor x_{i,j}/x_{i,k} \rfloor X_k$ 
12      pour tous les  $j > k$  faire
13         $X_j \leftarrow X_j - \lfloor x_{i,j}/x_{i,k} \rfloor X_k$ 
14       $i \leftarrow i - 1, k \leftarrow k - 1$ 
15   sinon
16      $i \leftarrow i - 1$ 
17 retourner  $\mathbf{X}$ 
```

L'algorithme termine car à chaque étape, le coefficient $|b| = |x_{i,k}|$ diminue d'au moins une unité. Lorsque l'algorithme s'achève, il est clair que \mathbf{X} est sous forme normale d'Hermite. Comme seules des opérations inversibles sur les colonnes ont été effectuées, la matrice de transformation est bien inversible.

□

Exercice 55. 1. On donne la matrice

$$\mathbf{A} = \begin{pmatrix} -2 & 3 & 3 & 1 \\ 2 & -1 & 1 & -3 \\ -4 & 0 & -1 & -4 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

Calculer la forme normale d'Hermite de \mathbf{A} à la main.

2. Comment modifier l'algorithme ci-dessus pour obtenir la matrice de passage \mathbf{U} ?
- 3.(a) En vue de la question (3e), écrire une sous-fonction `cherche_pivot_non_nul(i, j)` qui cherche à placer un élément non nul en position (i, j) par une transposition entre la colonne j et une colonne à gauche et renvoie vrai si c'est possible.
- (b) Écrire une sous-fonction `normalise_pivot(i, j)` qui multiplie la colonne j pour que le coefficient (i, j) soit positif.
- (c) Écrire une sous-fonction `annule_a_gauche` qui annule les coefficients à gauche de (i, j) par une opération de Bezout.
- (d) Écrire une sous-fonction `reduit_a_droite` qui réduit les coefficients à droite de (i, j) par une opération de transvection.

- (e) Écrire une fonction `myHNF` qui implémente l'algorithme de mise sous forme normale d'Hermite.
4. Calculer la forme normale d'Hermite \mathbf{H} de \mathbf{A} ainsi que la matrice de changement de base \mathbf{U} avec votre algorithme `myHNF`.
5. La méthode native de SageMath utilise des conventions américaines (vecteurs en ligne et multiplication à gauche). Pour confirmer votre code, essayez les instructions :

```
A = matrix(ZZ, [[-2, 3, 3, 1],
                 [ 2, -1, 1, -3],
                 [-4, 0, -1, -4]])
m = A.nrows()
n = A.ncols()
Mm = identity_matrix(ZZ, m)[::-1, :]
Mn = identity_matrix(ZZ, n)[::-1, :]
AA = (Mm*A).transpose()
HH, UU = AA.hermite_form(transformation=true)
H = (HH*Mm).transpose()*Mn
U = (UU.transpose()*Mn)
A*U-H
```

6. Vérifier que votre programme fonctionne sur une centaine de matrices tirées au sort pour des dimensions et un rang aléatoire

- Exercice 56.** 1. Comment modifier l'algorithme de mise sous forme normale d'Hermite pour qu'il fonctionne avec des polynômes?
2. Programmer votre version `myHNFforPolynomials`.

Les applications

Noyau d'une matrice

Proposition 57. Soit $\mathbf{X} \in \mathbb{Z}^{m \times n}$ une matrice et $\mathbf{H} = \mathbf{X}\mathbf{U}$ sa forme normale d'Hermite, avec $\mathbf{U} \in GL_n(\mathbb{Z})$. Soit r tel que les r premières colonnes de \mathbf{H} sont nulles. Alors les r premières colonnes de \mathbf{U} forment une base du noyau de \mathbf{X} .

Démonstration. $\mathbf{X}\mathbf{U}_i$ représente la i -ième colonne de \mathbf{H} . Les r premières colonnes de \mathbf{U} sont donc dans $\ker \mathbf{X}$.

Par ailleurs, le système d'équation $\mathbf{X}\mathbf{z} = 0$ équivaut à $\mathbf{H}\mathbf{y} = 0$ avec $\mathbf{y} = \mathbf{U}^{-1}\mathbf{z}$. Mais le système $\mathbf{H}\mathbf{y}$ est quasi triangulaire et on voit que les $n - r + 1$ dernière coordonnée de \mathbf{y} sont nulles. Donc \mathbf{z} est la combinaison

$$\mathbf{z} = y_1 \mathbf{U}_1 + \cdots + y_r \mathbf{U}_r$$

des r premières colonnes de \mathbf{U} . □

Exemple 58. Cherchons à résoudre dans \mathbb{Z}^2 le système d'équations

$$\begin{cases} 4x + 5y \equiv 0 \pmod{3} \\ 2x + 7y = 0 \end{cases}.$$

Ceci revient à résoudre dans \mathbb{Z}^3

$$\begin{cases} 4x + 5y = 3k \\ 2x + 7y = 0 \end{cases}.$$

On met la matrice

$$\mathbf{X} = \begin{pmatrix} 4 & 5 & -3 \\ 2 & 7 & 0 \end{pmatrix}$$

sous forme normale d'Hermite, avec

$$\mathbf{H} = \begin{pmatrix} 0 & 3 & 2 \\ 0 & 0 & 1 \end{pmatrix} \quad \mathbf{U} = \begin{pmatrix} 7 & 0 & 4 \\ -2 & 0 & -1 \\ 6 & -1 & 3 \end{pmatrix}$$

On en déduit que les solutions (x, y, k) du système sont les multiples de $(7, -2, 6)$, autrement dit que

$$x = 7\lambda \text{ et } y = -2\lambda \text{ pour un certain } \lambda \in \mathbb{Z}.$$

Exercice 59. Résoudre le système suivant dans \mathbb{Z}^4

$$\begin{cases} -2x + 3y + 3z + t = 0 \\ 2x - y + z - 3t = 0 \\ -4x - z - 4t = 0 \end{cases}.$$

Image et résolution de système d'équations linéaires homogène sur un anneau Soient $\mathbf{v}_1, \dots, \mathbf{v}_n \in A^m$ des vecteurs et M le sous-module de A^m engendré. La mise sous forme normale d'Hermite \mathbf{H} d'une matrice $\mathbf{X} = [\mathbf{v}_1, \dots, \mathbf{v}_n]$ calcule une base de M .

De plus, il est facilement possible de vérifier si un vecteur \mathbf{x} quelconque appartient à M car \mathbf{H} est échelonnée.

Égalité de deux modules libres Deux \mathbb{Z} -modules libres sont égaux si et seulement si les formes normales d'Hermite d'une de leurs familles génératrices sont égales.

Somme de deux modules libres Une base de la somme de deux \mathbb{Z} -modules libres peut être obtenue en calculant la forme normale d'Hermite de la réunion de leurs familles génératrices.

Inclusion On peut tester si deux \mathbb{Z} -modules M et N vérifient $M \subseteq N$ en testant si $M + N = N$.

Exercice 60. Soit

$$\mathbf{A} = \begin{pmatrix} 15 & 8 & -9 & 23 & -9 \\ 22 & 22 & 7 & -8 & 20 \\ 21 & 18 & -1 & -7 & -3 \\ 3 & -1 & 0 & 12 & -16 \end{pmatrix}$$

1. Est-ce que $\text{im}(\mathbf{A}) = \mathbb{Z}^4$?
2. Vérifier en utilisant les instructions

```
Z4 = ZZ^4
M = Z4.submodule(A.transpose())
M==Z4
```

Forme normale de Smith

L'algorithme

Définition 61. On dit qu'une matrice $\mathbf{X} \in A^{m \times n}$ est sous *forme normale de Smith* s'il existe un entier r et des éléments $(a_i)_{i \leq r} \subseteq A \setminus \{0\}$ tels que

1. a_1 divise a_2 , a_2 divise a_3 , ... et a_{r-1} divise a_r ,
2. et \mathbf{X} est la matrice

$$\mathbf{X} = \text{diag}_{m \times n}(a_1, \dots, a_r) = \begin{pmatrix} a_1 & 0 & \cdots & \cdots & 0 \\ 0 & a_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & & a_r & 0 \\ \vdots & & & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 \end{pmatrix} \in A^{m \times n}.$$

Théorème 62. Soit A un anneau principal et $\mathbf{X} \in A^{n \times m}$. Alors il existe deux matrices inversibles $\mathbf{L} \in GL_n(A)$ et $\mathbf{C} \in GL_m(A)$ telles que

$$\Delta = \mathbf{L} \cdot \mathbf{X} \cdot \mathbf{C}$$

est sous forme normale de Smith. De plus, en notant (a_1, \dots, a_r) les termes diagonaux non-nuls de Δ , l'entier r et les idéaux $a_1 A \supseteq a_2 A \supseteq \cdots \supseteq a_r A$ sont uniques.

Remarque 63. On peut voir ce théorème comme un théorème de classification des matrices sous la relation d'équivalence. On dit que deux

matrices \mathbf{X} et \mathbf{X}' sont équivalentes s'il existe deux matrices inversibles \mathbf{L} et \mathbf{R} telles que $\mathbf{X} = \mathbf{L}\mathbf{X}'\mathbf{R}$. Lorsqu'on considère une matrice \mathbf{X} sur un corps, la classe d'équivalence ne dépend que du rang r de \mathbf{X} . On peut toujours se ramener à

$$\mathbf{X}' = \begin{pmatrix} \mathbf{I}_r & 0 \\ 0 & 0 \end{pmatrix}.$$

Nous venons de prouver que sur un anneau principal, les classes d'équivalences ne dépendent que du rang et des idéaux d_1A, d_2A, \dots, d_rA . On dit que les éléments (d_1, \dots, d_r) sont les *facteurs invariants* et que les quotients $(d_2/d_1, \dots, d_r/d_{r-1})$ sont les *diviseurs élémentaires*.

Démonstration. Preuve d'existence : Nous donnons un algorithme de construction de Δ (voir algorithme 5). L'algorithme travaille avec un pivot (le coefficient $x_{k,k}$) qui parcourt la diagonale. Ce pivot doit être non nul (lignes 2 à 7 du code). On fabrique des 0 sur la ligne (lignes 9 à 13 du code) et la colonne (lignes 14 à 18 du code) terminant le pivot. Si le pivot ne divise pas les coefficients restants, une manipulation est nécessaire (ligne 20) et on réapplique la réduction.

L'algorithme s'arrête car pour tout $k \leq \min(m, n)$, la boucle « répéter » fait décroître le coefficient $x_{k,k}$ au sens de la divisibilité. Or nous travaillons avec des anneaux dans lesquels une suite décroissante pour la divisibilité est toujours finie.

Esquisse de preuve d'unicité :

Pour $1 \leq i \leq \min(m, n)$, on note $\delta_i(\mathbf{X})$ le pgcd de l'ensemble des sous-matrices extraites de taille $i \times i$.

1. On remarque d'une part que $\delta_i(\mathbf{X})$ n'est pas modifié lorsqu'on applique une transformation élémentaire à \mathbf{X} . On en déduit que $\delta_i(\mathbf{X}) = \delta_i(\mathbf{LXC})$ pour tout $\mathbf{L} \in GL_m(A)$ et $\mathbf{C} \in GL_n(A)$.
2. On note d'autre part que si $\mathbf{D} = \text{diag}(d_1, \dots, d_r) \in A^{m \times n}$ est sous forme normale d'Hermite, alors $\delta_i(\mathbf{D}) = d_1 d_2 \cdots d_i$.
3. On en déduit que r est unique et les éléments $(d_i)_{i \leq r}$ sont uniques à des éléments inversibles près.

□

Exercice 64. 1. Calculer à la main la forme normale de Smith des matrices entières suivantes :

$$\mathbf{X}_1 = \begin{pmatrix} 40 & 70 & 20 \\ 20 & 50 & 60 \end{pmatrix} \quad \mathbf{X}_2 = \begin{pmatrix} -397 & 423 & 352 \\ 2 & -3 & 1 \\ -146 & 156 & 128 \end{pmatrix}.$$

2. Comment modifier l'algorithme pour que soient également calculées les matrices de passage \mathbf{L} et \mathbf{C} ?
3. Écrire un programme `mySNF` qui implémente l'algorithme 5.
4. Calculer avec `mySNF` la forme normale de Smith des matrices de la première question.

Algorithme 5 : Mise sous forme normale de Smith (avec relations de Bézout)

Entrées : Matrice $X \in A^{m \times n}$ dans un anneau euclidien

Sorties : Forme normale de Smith de X

```

1  pour tous les  $k \leq \min(m, n)$  faire
2      si  $x_{k,k} = 0$  alors
3          si  $\exists (i_0, j_0) \in \llbracket k, m \rrbracket \times \llbracket k, n \rrbracket$  tels que  $x_{i_0, j_0} \neq 0$  alors
4               $L_k \leftrightarrow L_{i_0}$ 
5               $C_k \leftrightarrow C_{j_0}$ 
6          sinon
7              retourner  $X$ 
8      répéter
9          pour  $i \in \llbracket k+1, m \rrbracket$  faire
10              $d \leftarrow x_{k,k} \wedge x_{i,k}$ 
11             Soient  $s$  et  $t$  tels que  $s \cdot x_{k,k} + t \cdot x_{i,k} = d$  (Bézout).
12             si  $x_{k,k} \mid x_{i,k}$  alors
13                  $d \leftarrow x_{k,k}$ 
14                  $s \leftarrow 1$ 
15                  $t \leftarrow 0$ 
16              $u \leftarrow \frac{-x_{i,k}}{d}, v \leftarrow \frac{x_{k,k}}{d}.$ 
17              $\begin{cases} L_k \leftarrow sL_k + tL_i \\ L_i \leftarrow uL_k + vL_i \end{cases}$ 
18         pour  $j \in \llbracket k+1, n \rrbracket$  faire
19              $d \leftarrow x_{k,k} \wedge x_{k,j}$ 
20             Soient  $s$  et  $t$  tels que  $s \cdot x_{k,k} + t \cdot x_{k,j} = d$  (Bézout).
21             si  $x_{k,k} \mid x_{k,j}$  alors
22                  $d \leftarrow x_{k,k}$ 
23                  $s \leftarrow 1$ 
24                  $t \leftarrow 0$ 
25              $u \leftarrow \frac{-x_{k,j}}{d}, v \leftarrow \frac{x_{k,k}}{d}.$ 
26              $\begin{cases} C_k \leftarrow sC_k + tC_j \\ C_j \leftarrow uC_k + vC_j \end{cases}$ 
27         si  $\exists (i_0, j_0) \in \llbracket k+1, m \rrbracket \times \llbracket k+1, n \rrbracket$  tel que  $x_{k,k} \nmid x_{i_0, j_0}$  alors
28              $C_k \leftarrow C_k + C_{j_0}$ 
29     jusqu'à  $\forall i \in \llbracket k+1, m \rrbracket, x_{i,k} = 0$  et  $\forall j \in \llbracket k+1, n \rrbracket, x_{k,j} = 0;$ 
30 retourner  $X$ 

```

5. Calculer la forme normale de Smith de la matrices suivante sur $\mathbb{Q}[x]$, $\mathbb{F}_2[x]$, $\mathbb{F}_3[x]$ et $\mathbb{F}_5[x]$.

Avec
 $\text{F5} = \text{FiniteField}(5)$
 $\text{PolF5}.\langle x \rangle = \text{PolynomialRing}(\text{F5})$
 $\text{matrix}(\text{Pol}, 3, 3, [[x+1, 2, 6], [1, x, -3], [1, 1, x-4]])$

$$\mathbf{A} = \begin{pmatrix} x+1 & 2 & -6 \\ 1 & x & -3 \\ 1 & 1 & x-4 \end{pmatrix}.$$

6. Vérifier vos calculs en comparant avec une méthode native de Sage-Math.

Remarque 65. Dans le cas où l'anneau A est euclidien, on peut utiliser la variante suivante qui a l'avantage d'éviter de calculer des relations de Bézout. Elle est pratique quand on travaille à la main.

Algorithme 6 : Mise sous forme normale de Smith (sans relations de Bézout)

Entrées : Matrice $\mathbf{X} \in A^{m \times n}$ dans un anneau euclidien

Sorties : Forme normale de Smith de \mathbf{X}

```

1  pour tous les  $k \leq \min(m, n)$  faire
2      si  $x_{k,k} = 0$  alors
3          si  $\exists (i_0, j_0) \in \llbracket k, m \rrbracket \times \llbracket k, n \rrbracket$  tels que  $x_{i_0, j_0} \neq 0$  alors
4               $L_k \leftrightarrow L_{i_0}$ 
5               $C_k \leftrightarrow C_{j_0}$ 
6          sinon
7              retourner  $\mathbf{X}$ 
8      répéter
9          tant que  $\exists i \in \llbracket k+1, m \rrbracket, x_{i,k} \neq 0$  faire
10             Poser  $x_{i,k} = q \cdot x_{k,k} + r$  (division euclidienne)
11              $L_i \leftarrow L_i - qL_k$ 
12             si  $r \neq 0$  alors
13                  $L_k \leftrightarrow L_i$ 
14             tant que  $\exists j \in \llbracket k+1, n \rrbracket, x_{k,j} \neq 0$  faire
15                 Poser  $x_{k,j} = q \cdot x_{k,k} + r$  (division euclidienne)
16                  $C_j \leftarrow C_j - qC_k$ 
17                 si  $r \neq 0$  alors
18                      $C_k \leftrightarrow C_j$ 
19             si  $\exists (i_0, j_0) \in \llbracket k+1, m \rrbracket \times \llbracket k+1, n \rrbracket$  tel que  $x_{k,k} \nmid x_{i_0, j_0}$  alors
20                  $C_k \leftarrow C_k + C_{j_0}$ 
21             jusqu'à  $\forall i \in \llbracket k+1, m \rrbracket, x_{i,k} = 0$  et  $\forall j \in \llbracket k+1, n \rrbracket, x_{k,j} = 0$ ;
22 retourner  $\mathbf{X}$ 

```

Les applications

Résolution de système d'équations linéaires non homogène sur un anneau
Soit $\mathbf{X} \in A^{m \times n}$ une matrice et $\mathbf{b} \in A^m$ un vecteur. Soient $\mathbf{L} \in GL_m(A)$ et $\mathbf{C} \in GL_n(A)$ des matrices inversibles telles que $\Delta = \mathbf{LXC}$ est sous forme normale de Smith. On note a_1, \dots, a_r les coefficients diagonaux non-nuls de la matrice Δ . On note encore $\mathbf{b}' = \mathbf{Lb} \in A^m$. Alors le système d'équation linéaire non-homogène

$$\mathbf{Xz} = \mathbf{b}$$

d'inconnue $\mathbf{z} \in A^n$ admet une solution si et seulement si pour tout a_i divise b'_i quand $i \leq r$ et $b'_i = 0$ quand $i \geq r+1$. De plus, dans ce cas, les solutions du système sont de la forme

$$\mathbf{z} = \frac{b'_1}{a_1} \mathbf{C}_1 + \dots + \frac{b'_r}{a_r} \mathbf{C}_r + \sum_{i=r+1}^n z'_i \mathbf{C}_i$$

où $(z'_i)_{r+1 \leq i \leq n} \in A^{n-r}$.

Démonstration. Comme \mathbf{C} est une matrice inversible, les colonnes de \mathbf{C} forment une base de \mathbb{Z}^n . Il nous est loisible de chercher les solutions \mathbf{z} comme combinaison des colonnes de \mathbf{C} , autrement dit sous la forme

$$\mathbf{z} = \mathbf{Cz}' = \sum_{i=1}^n z'_i \mathbf{C}_i$$

Mais le système $\mathbf{Xz} = \mathbf{b}$ équivaut à $\mathbf{XCz}' = \mathbf{b}$ ou encore à $\mathbf{LXCz}' = \mathbf{Lb}$. Ce nouveau système est simplement

$$\begin{cases} a_1 z'_1 = b'_1 \\ a_2 z'_2 = b'_2 \\ \vdots \\ a_r z'_r = b'_r \\ 0 = b'_{r+1} \\ \vdots \\ 0 = b'_n \end{cases}.$$

□

Exemple 66. On cherche à résoudre dans \mathbb{Z}^3 l'équation $\mathbf{Xz} = \mathbf{b}$ où

$$\mathbf{X} = \begin{pmatrix} 5 & 28 & -6 \\ 3 & 18 & -4 \\ 1 & -4 & 2 \end{pmatrix} \text{ et } \mathbf{b} = \begin{pmatrix} 7 \\ 5 \\ -5 \end{pmatrix};$$

Une mise sous forme normale de Smith de \mathbf{X} conduit aux matrices

$$\Delta = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{L} = \begin{pmatrix} 5 & -8 & 0 \\ -3 & 5 & 0 \\ -5 & 8 & 1 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & 2 & -2 \\ 0 & 0 & 1 \\ 0 & -1 & 3 \end{pmatrix}$$

telles que $\mathbf{LXC} = \Delta$. On peut poser

$$\mathbf{b}' = \mathbf{Lb} = \begin{pmatrix} -5 \\ 4 \\ 0 \end{pmatrix}.$$

Les solutions sont alors

$$\begin{aligned} \mathbf{z} &\in \frac{-5}{1}\mathbf{C}_1 + \frac{4}{2}\mathbf{C}_2 + \mathbb{Z}\mathbf{C}_3 \\ &= \begin{pmatrix} -1 \\ 0 \\ -2 \end{pmatrix} + \mathbb{Z} \begin{pmatrix} -2 \\ 1 \\ 3 \end{pmatrix} \end{aligned}$$

Exercice 67. 1. Résoudre dans \mathbb{Z} l'équation $\mathbf{X}_1\mathbf{z} = \mathbf{b}_1$ où

$$\mathbf{X}_1 = \begin{pmatrix} -6 & 12 & 6 \\ 12 & -16 & -2 \\ 12 & -16 & -2 \end{pmatrix} \text{ et } \mathbf{b}_1 = \begin{pmatrix} -6 \\ 4 \\ 4 \end{pmatrix};$$

2. Résoudre dans $\mathbb{F}_5[x]$ l'équation $\mathbf{X}_2\mathbf{z} = \mathbf{b}_2$ où

$$\mathbf{X}_2 = \begin{pmatrix} x+1 & 2 & 4 \\ 1 & x & 2 \\ 1 & 1 & x+1 \end{pmatrix} \text{ et } \mathbf{b}_2 = \begin{pmatrix} 3x+2 \\ 0 \\ -1 \end{pmatrix}.$$

3. Résoudre le système suivant dans \mathbb{Z}^3

$$\begin{cases} 2x - 3y + 4z &= & -4 \\ 4x + 4y + 53z &\equiv & 2 \pmod{5} \end{cases}.$$

Sans passer par

```
X = matrix(ZZ, [[-6,12,6],[12,-16,-2],[12,-16,-2]]
b = vector(ZZ, [-6,4,4])
X.solve_right(b)
```

Structure d'un A -module

Le théorème de la base adaptée

Théorème 68. Soit L un A -module libre de dimension n et M un sous-module. Alors M est libre et sa dimension est inférieure à celle de L .

Démonstration. Si la dimension n vaut 1, L est isomorphe à A et M est isomorphe à un idéal de A . Comme A est principal, M est de la forme $A\mathbf{v}$ avec $\mathbf{v} \in L$, cqfd.

Nous traitons le cas général par récurrence sur n . Soient $(\mathbf{b}_i)_{i \leq n}$ une base de L , L_r le sous-module engendré par les r premiers vecteurs $L_r = \langle (\mathbf{b}_i)_{i \leq r} \rangle$ et $M_r = M \cap L_r$. Par hypothèse de récurrence, M_{n-1} est un sous-module libre de L_{n-1} de dimension inférieure à $n-1$.

Soit \mathfrak{A} l'image de la projection π sur la dernière coordonnée

$$\pi : \begin{cases} L & \rightarrow A \\ x = \sum_{i=1}^n \beta_i \mathbf{b}_i & \mapsto \beta_n \end{cases}$$

Comme \mathfrak{A} est un idéal de A , \mathfrak{A} est de la forme αA pour un certain $\alpha \in A$.

Si $\alpha = 0$, alors $M = M_{n-1}$ (fin de preuve). Sinon, il existe \mathbf{w} tel que $\pi(\mathbf{w}) = \alpha$. Les sous-modules M_{n-1} et $A\mathbf{w}$ sont clairement en somme directe et $M \subseteq M_{n-1} + A\mathbf{w}$. Comme M_{n-1} est libre de dimension $\leq n-1$, le module $M = M_{n-1} \oplus A\mathbf{w}$ est libre de dimensions $\leq n$. \square

Contre-exemple 69. Soit $A = \mathbb{K}[x, y]$ l'anneau des polynômes à deux variables sur un corps \mathbb{K} ; cet anneau n'est pas principal. Les modules libres sont les $(\mathbb{K}[x, y])^n$. Pour $n = 1$ (ce qui revient à revenir au niveau de la théorie des anneaux et leurs idéaux), on peut considérer le module libre $M = A$ et

$$\mathfrak{I} = \langle x, y \rangle = \{p(x, y) \in \mathbb{K}[x, y]; \exists p_1, p_2 \in \mathbb{K}[x, y], p = p_1x + p_2y\}.$$

Alors \mathfrak{I} un A -module. C'est un sous-module de M . Mais il n'existe pas de polynôme $p_0(x, y)$ tel que

$$\mathfrak{I} = A \cdot p_0(x, y).$$

Cela veut dire que \mathfrak{I} n'est pas libre.

Corollaire 70. Soit M un A -module de type fini et N un sous-module de M , alors N est aussi de type fini.

Démonstration. Soient $(\mathbf{v}_i)_{i \leq n}$ des générateurs de M , $(\mathbf{x}_i)_{i \leq n}$ une base du module $M' = A^n$ et p l'homomorphisme $M' \rightarrow M$ qui envoie \mathbf{x}_i sur \mathbf{v}_i . L'image inverse $N' = p^{-1}(N)$ est un sous-module de M' , qui d'après le théorème 68, est libre. Si $(\mathbf{y}_i)_{i \leq r}$ désigne une base de N' , alors $(p(\mathbf{y}_i))_{i \leq r}$ engendre finiment N . \square

Exemple 71. $\mathbb{Z}/6\mathbb{Z}$ est de type fini (on peut prendre $\bar{1}$ comme générateur), alors ses sous-modules qui sont $\mathbb{Z}/3\mathbb{Z} \simeq \{\bar{0}, \bar{2}, \bar{4}\}$ et $\mathbb{Z}/2\mathbb{Z} \simeq \{\bar{0}, \bar{3}\}$ sont aussi de type fini (engendrés respectivement par $\bar{2}$ et par $\bar{3}$). Si nous reprenons la preuve de ce corollaire, nous aurions choisi $M' = \mathbb{Z}$ et $N' = 2\mathbb{Z}$ ou $N' = 3\mathbb{Z}$ respectivement.

Remarque 72. Quand le corollaire 70 est vérifié, on dit que M est *noethérien*. Il suffit qu'un anneau soit noethérien, c'est-à-dire que ses idéaux soient de type fini, pour que le corollaire soit juste.

Exercice 73. Montrer dans les cas suivants que \mathfrak{I} est un idéal non finiment engendré de A :

1. A est l'anneau des polynômes à une infinité de variables $\mathbb{K}[(x_i)_{i \in \mathbb{N}}]$ (les éléments de A sont des combinaisons linéaires finies de monômes) et

$$\mathfrak{I} = \langle x_i, i \in \mathbb{N} \rangle.$$

2. $A = \mathcal{C}^0(\mathbb{R})$ est l'anneau des fonctions continues $\mathbb{R} \rightarrow \mathbb{R}$ et

$$\mathfrak{I} = \{f \in \mathcal{C}^0(\mathbb{R}); \exists x_0 \in \mathbb{R}, \forall x > x_0, f(x) = 0\}$$

Théorème 74 (Base adaptée). Soit M un module libre de dimension n sur un anneau principal A et N un sous-module de M . Alors il existe une base $(\mathbf{e}_i)_{i \leq n}$ de M et des éléments (d_1, \dots, d_r) de A (avec $r \leq n$) tels que

1. $(d_1 \mathbf{e}_1, d_2 \mathbf{e}_2, \dots, d_r \mathbf{e}_r)$ soit une base de N ,
2. on ait les relations de divisibilité $d_1 | d_2 | \dots | d_r$.

Démonstration. D'après le théorème 68, le module N est libre. Soient \mathcal{B} et \mathcal{K} des bases respectives quelconques des modules M et N . Notons $\mathbf{X} \in A^{n \times m}$ la matrice telle que $\mathcal{K} = \mathcal{B}\mathbf{X}$. Selon la décomposition de Smith (théorème 62), il existe des matrices inversibles $\mathbf{L} \in GL_n(A)$ et $\mathbf{C} \in GL_m(A)$ et des coefficients d_1, \dots, d_m tels que

$$\mathbf{LXC} = \text{diag}(d_1, d_2, \dots, d_m)$$

avec $d_1 | d_2 | \dots | d_m$. Comme \mathbf{L} et \mathbf{C} sont inversibles, $\mathcal{B}' = \mathcal{B}\mathbf{L}^{-1}$ et $\mathcal{K}' = \mathcal{K}\mathbf{C}$ sont aussi des bases respectives de M et de N . De plus, si $\mathcal{B}' = \{\mathbf{e}_1, \dots, \mathbf{e}_n\}$, alors $\mathcal{K}' = (d_1 \mathbf{e}_1, \dots, d_m \mathbf{e}_m)$. \square

Exercice 75. Soit $M = \mathbb{Z}^2$ et N le sous- \mathbb{Z} -module engendré par $(1, 0)$ et $(2, 2)$. Trouver une base adaptée.

Exercice 76. Soit $A = \mathbb{Z}$. Calculer une base adaptée à M et à N dans les cas suivants (on commencera par faire des expérimentations avec SageMath et on confirmera par un raisonnement) :

1. $M = \mathbb{Z}^{n+1}$ et

$$N = \left\{ \mathbf{x} \in M; \sum_{i=0}^n x_i = 0 \right\}$$

2. $M = \mathbb{Z}^n$ et

$$N = \left\{ \mathbf{x} \in M; \sum_{i=1}^n x_i \equiv 0 \pmod{2} \right\}$$

[Indication : utiliser la proposition 57 pour calculer une base quelconque de N .]

Application 77. Détermination du quotient. Nous nous plaçons dans le cas où $M = A^n$ et N est engendré par les vecteurs $\mathbf{x}_1, \dots, \mathbf{x}_m \in A^n$. Soit $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m]$ la matrice de $A^{n \times m}$ correspondante. Soit $\Delta = \mathbf{LXC}$ la forme normale de Smith de \mathbf{X} . Alors $N = \text{im}(\mathbf{X}) = \text{im}(\mathbf{XC})$ puisque les vecteurs colonnes de \mathbf{XC} forment une autre famille génératrice de N . Mais alors $N = \text{im}(\mathbf{L}^{-1}\Delta)$. Notons ℓ_i le i -ème vecteur colonne de \mathbf{L}^{-1} . Les vecteurs $(\ell_i)_{i \leq n}$ forment une base de A^n et les vecteurs $(a_i \ell_i)_{i \leq r}$ forment la base adaptée de N . De plus

$$M/N = (A/a_1 A)\ell_1 \oplus \dots \oplus (A/a_r A)\ell_r \oplus A\ell_{r+1} \oplus \dots \oplus A\ell_n.$$

Exemple 78. Sur l'anneau $A = \mathbb{Z}$, on considère le module N engendré par les vecteurs colonnes de la matrice

$$\mathbf{X} = \begin{pmatrix} -6 & 9 & 39 & 237 \\ -1 & 2 & 9 & 54 \\ 11 & -16 & -69 & -420 \end{pmatrix}.$$

Une décomposition de Smith de \mathbf{X} est

$$\underbrace{\begin{pmatrix} 2 & 0 & 1 \\ 1 & 0 & 0 \\ -2 & 1 & -1 \end{pmatrix}}_{\mathbf{L}} \cdot \mathbf{X} \cdot \underbrace{\begin{pmatrix} 3 & -2 & -4 & -1 \\ 2 & -1 & -29 & -5 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\mathbf{C}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Comme $\text{im}(\mathbf{X}) = \text{im}(\mathbf{AC}) = \text{im}(\mathbf{L}^{-1}\Delta)$, on peut remarquer que les colonnes de \mathbf{L}^{-1} forment une base adaptée à \mathbb{Z}^3 et à N . D'abord précisons que

$$\mathbf{L}^{-1} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & -2 & 0 \end{pmatrix}$$

Notons

$$\ell_1 = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \quad \ell_2 = \begin{pmatrix} 1 \\ 0 \\ -2 \end{pmatrix} \quad \text{et} \quad \ell_3 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}.$$

Comme $\mathbf{L} \in GL_3(\mathbb{Z})$, la famille (ℓ_1, ℓ_2, ℓ_3) est une base de \mathbb{Z}^3 , et N est en particulier engendré par

$$N = \langle \ell_1, 3\ell_2 \rangle = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} + \mathbb{Z} \begin{pmatrix} 3 \\ 0 \\ -6 \end{pmatrix}$$

Supposons maintenant que l'on s'intéresse au module $M = \mathbb{Z}^3/N = \text{coker } \mathbf{X}$ (le conoyeau de \mathbf{X}). Nous pouvons utiliser l'exercice 37. Nous avons, en réutilisant la base adaptée de N et \mathbb{Z}^3 que nous avons calculée que

$$\begin{aligned} M = \mathbb{Z}^3/N &= (\mathbb{Z}/\mathbb{Z}) \ell_1 \oplus (\mathbb{Z}/3\mathbb{Z}) \ell_2 + (\mathbb{Z}/0\mathbb{Z}) \ell_3 \\ &\simeq \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}. \end{aligned}$$

De plus, le premier terme $\mathbb{Z}/3\mathbb{Z}$ correspond à la classe du deuxième vecteur colonne de \mathbf{L}^{-1} modulo $\text{im}(\mathbf{X})$ et le second terme \mathbb{Z} correspond à la classe du troisième vecteur de \mathbf{L}^{-1} modulo $\text{im}(\mathbf{X})$

Application 79. Présentation d'un A -module fini. Supposons que M soit A -module de type fini et que par certains moyens théorique, on

ait déterminé un ensemble x_1, \dots, x_n de générateurs. Alors

$$f : \begin{cases} A^n & \rightarrow M \\ (\alpha_i)_{1 \leq i \leq n} & \mapsto \sum_{i=1}^n \alpha_i x_i \end{cases}$$

et une bijection et $M \simeq A^n / \ker f$. Soit (f_1, \dots, f_m) une base de $N = \ker f$ et X sa matrice. Alors la réduction de Smith permet de retrouver la structure de M .

Exercice 80. Soit N le \mathbb{Z} -module engendré dans \mathbb{Z}^3 par les colonnes de la matrice

$$A = \begin{pmatrix} -630 & 735 & 0 & 735 & -630 \\ 1275 & -1485 & -15 & -1470 & 1275 \\ 630 & -630 & 0 & -630 & 630 \end{pmatrix}$$

Donner la structure du quotient $M = \mathbb{Z}^3 / N$.

Exercice 81. On dit qu'un sous-groupe H d'un groupe G est d'*indice fini* si le quotient G/H est fini. Le cardinal $|G/H|$ s'appelle alors l'*indice* de H dans G .

Soit $\phi : \mathbb{Z}^n \rightarrow \mathbb{Z}^n$ une application \mathbb{Z} -linéaire de matrice A . Montrer que l'image de ϕ est d'indice fini si et seulement si $\det A \neq 0$ et que, dans ce cas, l'indice égale $|\det A|$.

Exercice 82. Le but de cet exercice est de vérifier numériquement un théorème de combinatoire algébrique.

On appelle *complexité* d'un graphe $G = (V, E)$ le nombre de d'arbres couvrant du graphe et *Laplacien* de G la matrice $((\Delta_{u,v}))_{u,v \in V^2}$ telle que $\Delta_{u,u} = \deg u$ et $\Delta_{u,v} = -1$ si $(u, v) \in E$. On note $\Phi(G)$ le sous-groupe de torsion de $\mathbb{Z}^V / \text{Im}(\Delta)$.

1. Calculer l'ordre de $\Phi(G)$ pour un arbre G aléatoirement choisi.
2. Calculer l'ordre de $\Phi(G)$ pour un cycle G aléatoirement choisi.
3. Calculer l'ordre de $\Phi(G)$ pour un arbre complété d'une arête G aléatoirement choisi.
4. Conjecturer une relation entre la complexité de G et l'ordre de $\Phi(G)$ et la vérifier expérimentalement.

Exercice 83. Sur l'anneau euclidien des entiers de Gauß $\mathcal{G} = \mathbb{Z}[i]$ (cf. exercice 19), on considère le module N engendré par les vecteurs colonnes de la matrice

$$A = \begin{pmatrix} 3 - 2i & 2 + 3i & -1 + 2i \\ 2 + 2i & 6i & -8 + 6i \\ 1 & i & -1 \end{pmatrix}$$

Quelle est la structure et le cardinal de \mathcal{G}^3 / N ?

On pourra créer $\mathbb{Z}[i]$ avec
`Qi.<i>= QuadraticField(-1)`
`ZZi=Qi.ring_of_integers()`

Structure d'un A -module

Définition 84. On dit qu'un module est *cyclique* s'il est de la forme A/aA avec $a \in A$.

Exemple 85. $\mathbb{Z}/5\mathbb{Z}$ ou $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z}$ sont cycliques (lemme chinois) mais $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$ n'est pas cyclique (pas d'élément d'ordre 4).

Soient $\omega_1, \dots, \omega_n$ des éléments d'un corps \mathbb{K} . Alors $\mathbb{K}[x]/(x - \omega_1) \oplus \dots \oplus \mathbb{K}[x]/(x - \omega_n)$ est un module cyclique si et seulement si les ω_i sont distincts (lemme chinois).

Exercice 86. Montrer qu'un sous-module d'un module cyclique est encore cyclique.

Corollaire 87 (Facteurs invariants). Soit M un module de type fini sur un anneau A principal, alors il existe⁸ $d_1, \dots, d_s \in A$ non nuls et non inversibles, tels que M soit isomorphe à

$$A^r \oplus \bigoplus_{i=1}^s A/d_i A$$

avec $r \in \mathbb{N}$ et $d_1 | d_2 | \dots | d_s$.

8. Avec SageMath, utiliser la fonction `invariants()`

Démonstration. Soient $(\mathbf{g}_i)_{i \leq p}$ une famille génératrice de M et $\phi : A^p \rightarrow M$ la surjection $(\alpha_i)_{i \leq p} \mapsto \sum_{i=1}^p \alpha_i \mathbf{g}_i$. Le module M est isomorphe à $A^p / \ker \phi$. Le théorème de la base adaptée fournit une base $(\mathbf{e}_i)_{i \leq p}$ de A^p telle que

$$A^p = A \mathbf{e}_1 \oplus \dots \oplus A \mathbf{e}_p$$

$$\ker \phi = A d_1 \mathbf{e}_1 \oplus \dots \oplus A d_{s'} \mathbf{e}_{s'}$$

Mais alors directement

$$A^p / N = (A/d_1 A) \mathbf{e}_1 \oplus \dots \oplus (A/d_{s'} A) \mathbf{e}_{s'} \oplus A \mathbf{e}_{s'+1} \oplus \dots \oplus A \mathbf{e}_p.$$

Il reste à nettoyer l'écriture pour obtenir le résultat voulu. \square

Remarque 88. La suite (d_1, d_2, \dots, d_s) décrite ci-dessus s'appelle la *suite des facteurs invariants*. La classe d'isomorphisme d'un A -module est caractérisée par son rang et sa suite de facteurs invariants.

Exercice 89. Déterminer la structure des \mathbb{Z} -modules suivants

$$\text{Hom}_{\mathbb{Z}}(\mathbb{Z}^2, \mathbb{Z}), \quad \text{Hom}_{\mathbb{Z}}(\mathbb{Z}^2, \mathbb{Z}^2),$$

$$\text{Hom}_{\mathbb{Z}}(\mathbb{Z}, \mathbb{Z}/n\mathbb{Z}), \quad \text{Hom}_{\mathbb{Z}}(\mathbb{Z}/n\mathbb{Z}, \mathbb{Z}/m\mathbb{Z}).$$

Exercice 90. Soit $M = \mathbb{Z}[i] = \{\alpha + i\beta; (\alpha, \beta) \in \mathbb{Z}^2\}$. Donner les facteurs invariants de $M/\langle 2i \rangle$ en tant que A -module lorsque

1. $A = \mathbb{Z}$,
2. $A = \mathbb{Z}[i]$.

Définition 91. Dans la situation du corollaire 87, on appelle r le *rang* de M .

Définition 92. Soit M un A -module, un élément $\mathbf{v} \in M$ est un *élément de torsion* s'il existe $\lambda \in A$ non nul tel que $\lambda \mathbf{v} = 0$.

Un *module de torsion* est un module dont tous les éléments sont de torsion.

Un scalaire $c \in A$ est un *exposant* de M si $cM = 0$.

Exemple 93. Le \mathbb{Z} -module $\mathbb{Z}/n\mathbb{Z}$ est un module de torsion d'exposant n , le \mathbb{Z} -module \mathbb{Q}/\mathbb{Z} est un module de torsion sans exposant (il n'est pas de type fini non plus).

Exercice 94. Soit N le \mathbb{Z} -module engendré dans \mathbb{Z}^3 par les colonnes de la matrice

$$A = \begin{pmatrix} -630 & 735 & 0 & 735 & -630 \\ 1275 & -1485 & -15 & -1470 & 1275 \\ 630 & -630 & 0 & -630 & 630 \end{pmatrix}$$

Quel est le rang et quels sont les facteurs invariants du quotient $M = \mathbb{Z}^3/N$? Le module M est-il un module de torsion? Si oui, décrire l'ensemble des exposants de M .

Exercice 95. Soient A un anneau principal et $\mathcal{F} = (\mathbf{f}_i)_{i \leq n}$ une famille de vecteurs linéairement indépendants de A^n et N le sous-module de A^n engendré par \mathcal{F} . Montrer que le déterminant $\det \mathcal{F}$ est un exposant du module quotient A/N .

Exercice 96. Soit U le \mathbb{Z} -module (infini) des racines de l'unité dans \mathbb{C} . Est-il de torsion?

Exercice 97. On appelle *idéal annulateur* $\text{Ann}(M)$ l'ensemble des exposants de M .

1. Vérifier que $\text{Ann}(M)$ est bien un idéal de A .
2. Quel est l'idéal annulateur dans les situations suivantes?
 - (a) $A = \mathbb{Z}$, $M = \mathbb{Z}/4\mathbb{Z} \oplus \mathbb{Z}/7\mathbb{Z} \oplus \mathbb{Z}/10\mathbb{Z} \oplus \mathbb{Z}/16\mathbb{Z}$
 - (b) $A = \mathbb{K}[x]$, M est le module $E = \mathbb{K}^n$ de l'exemple 22, u est un endomorphisme diagonalisable.
 - (c) $A = \mathbb{F}_{81}[x]$, $M = A/(x+1)A \oplus A/(x^2+2)A \oplus A/(x^4+2x^3+2x^2+2x+2)A$.

Définition 98. Étant donné un A -module M , on appelle *sous-module de torsion* le module M_{tors} formé des éléments de torsion de M .

Exercice 99. Vérifier que M_{tors} forme réellement un sous-module.

Nous pouvons reformuler le corollaire 87.

Théorème 100. Soit M un module de type fini. Alors il existe un sous-module N de M tel que M soit la somme directe :

$$M = N \oplus M_{tors}.$$

Remarque 101. Un module de type fini est donc libre si et seulement s'il est sans torsion. L'hypothèse « de type fini » est bien nécessaire comme le montre l'exemple de \mathbb{Q} vu comme \mathbb{Z} -module (voir exercice ??).

Exercice 102. Donner un exemple de module sur $A = \mathbb{K}[x, y]$ de type fini et sans torsion mais cependant non libre. De même pour $A = \mathbb{Z}[x]$.

Exercice 103. Soient M et M' deux A -modules de type fini sur un anneau principal A . Décrire le rang et la partie de torsion de $M \oplus M'$.

Exercice 104. On note $S = \mathbb{Z}^{\mathbb{N}}$ l'ensemble des suites d'entiers. C'est un \mathbb{Z} -module. On appelle ϵ_n la suite $\epsilon_n = (0, \dots, 0, 1, 0, \dots)$ avec un seul 1 en position n .

1. Montrer que S est sans torsion.
2. On suppose dans ce qui suit que S possède une base notée B et que la décomposition de ϵ_n dans B est

$$\epsilon_n = \sum_{b \in B} \lambda_{n,b} b.$$

Soit $A = \{b \in B; \exists n \in \mathbb{N}, \lambda_{n,b} \neq 0\}$. Montrer que A est dénombrable.

3. Soient M et M' les sous-modules de S engendrés par A et $B \setminus A$ respectivement. Montrer que $S = M \oplus M'$.
On note p la projection $p : S \rightarrow S/M$.
4. Pour toute suite a d'entiers positifs $a = (a_1, a_2, \dots)$, on pose

$$x_a = (2, 0, \dots, 0, 2^2, 0, \dots, 0, 2^3, 0, \dots, 0, 2^4, 0, \dots)$$

avec a_i zéros entre 2^i et 2^{i+1} . Quel est le cardinal de l'ensemble de telles suites ?

5. Montrer qu'il existe une suite a_0 telle que $x_{a_0} \notin M$. On note $x = x_{a_0}$.
6. Montrer que pour tout $n \in \mathbb{N}$, il existe $y_n \in S$ tel que $p(x) = 2^n p(y_n)$.
7. Montrer que dans un \mathbb{Z} -module libre L , seul l'élément nul est divisible par 2^n pour tout entier n .
8. Conclure.

TP 3 : Réseaux euclidiens

Buts : Découvrir la notion de réseau, comprendre l'algorithme LLL , en voir quelques exemple d'utilisation.

Travaux préparatoires : Exercice 113, 137 & 136 (à la main).

Évaluation du TP : Exercices 114 (base d'un réseau), 139 (applications numériques de LLL), 135 (algorithme LLL), 118 (réseau \mathbb{E}_8), 145 (densité de réseaux).

Alors que la notion de réseau est conceptuellement très simple, elle n'a pas vraiment été étudiée en mathématiques avant Minkowski. À cette époque, les réseaux n'ont pas vraiment été étudiés en tant que tels mais comme une astuce pour rendre géométrique l'étude des formes quadratiques sur les entiers.

Quant à l'algorithme LLL , il a été publié dans le contexte de la factorisation de polynômes sans rapport avec les réseaux. Il était déjà connu de l'un de ses pères qui ne voyait pas d'intérêt à cet algorithme ; aujourd'hui, LLL est considéré comme l'un des dix algorithmes les plus importants découverts au XXIème siècle.

Définitions

Qu'est ce qu'un réseau ?

On rappelle qu'un *espace euclidien* est un espace vectoriel muni d'un produit scalaire.

Définition 105 (Réseau euclidien). Un *réseau euclidien* d'un espace euclidien $(E, \langle \cdot, \cdot \rangle)$ est un sous-groupe de $(E, +)$ de la forme

$$\mathcal{L} = \left\{ \sum_{i=1}^n x_i \mathbf{e}_i; (x_i)_{i \leq n} \in \mathbb{Z}^n \right\}$$

où $(\mathbf{e}_i)_{i \leq n} \subseteq E$ est une famille libre de E .

L'entier n s'appelle le *rang* et $m = \dim E$ la *dimension* du réseau \mathcal{L} . On dit que $\mathcal{B} = (\mathbf{e}_i)_{i \leq n}$ est une *base* de \mathcal{L} et on utilise la matrice $\mathbf{B} = [\mathbf{e}_1, \dots, \mathbf{e}_n] \in \mathbb{R}^{n \times m}$ pour représenter \mathcal{L} .

Le parallélotope $\{\sum_{i=1}^n x_i \mathbf{e}_i; 0 \leq x_i \leq 1\}$ s'appelle le *domaine fondamental* du réseau.

Lemme 106. Si \mathcal{B}_1 et \mathcal{B}_2 sont deux bases d'un même réseau, alors il existe $\mathbf{U} \in GL_n(\mathbb{Z})$ (il faut et il suffit pour cela que $\det \mathbf{U} = \pm 1$) tel que $\mathcal{B}_1 = \mathbf{U}\mathcal{B}_2$.

Démonstration. En effet, les vecteurs de \mathcal{B}_1 possèdent une décomposition dans la base \mathcal{B}_2 et vice versa, ce qui fournit des matrices $\mathbf{U}, \mathbf{V} \in \mathbb{Z}^{n \times n}$ telles que $\mathcal{B}_1 = \mathbf{U}\mathcal{B}_2$ et $\mathcal{B}_2 = \mathbf{V}\mathcal{B}_1$. Mais alors $\mathbf{UV} = \mathbf{I}$, donc $\mathbf{U} \in GL_n(\mathbb{Z})$. De plus $1 = \det(\mathbf{UV}) = \det \mathbf{U} \det \mathbf{V} \in \mathbb{Z}$, donc $\det \mathbf{U} = \pm 1$. Réciproquement, si $\det \mathbf{U} = \pm 1$, les formules de Cramer permettent d'inverser \mathbf{U} dans \mathbb{Z} . \square

Du point de vue théorique, on pourrait se contenter de fixer $m = n$, mais il est algorithmiquement intéressant de les distinguer (e.g. réseau \mathbb{A}_n).

Un réseau est bien sûr un \mathbb{Z} -module libre de rang n . On remarque que la mention du produit scalaire est primordiale, sans quoi tous les réseaux ne seraient rien de plus que le \mathbb{Z} -module libre de dimension n (isomorphe à \mathbb{Z}^n) et la théorie serait très pauvre.

Exemple 107. — Les ensembles $\mathbb{Z}^n, \mathbb{A}_n = \{(x_i)_{0 \leq i \leq n} \in \mathbb{Z}^{n+1}; \sum_{i=0}^n x_i = 0\}$ et $\mathbb{D}_n = \{(x_i)_{1 \leq i \leq n} \in \mathbb{Z}^n; \sum_{i=1}^n x_i \equiv 0 \pmod{2}\}$ sont des réseaux de rang n l'espace euclidien canonique.

- Les entiers de Gauß $\mathbb{Z}[i] = \{\alpha + i\beta; (\alpha, \beta) \in \mathbb{Z}^2\} \subseteq \mathbb{C}$ forment un réseau de rang 2 et plus généralement $\mathbb{Z}[\sqrt{-d}]$ ($d > 0$).
- L'ensemble $\mathbb{Z}[\sqrt{2}] = \{\alpha + \sqrt{2}\beta; (\alpha, \beta) \in \mathbb{Z}^2\} \subseteq \mathbb{R}$ n'est pas un réseau euclidien de \mathbb{R} car la famille $(1, \sqrt{2})$ n'est pas linéairement indépendante (et $\mathbb{Z}[\sqrt{2}]$ est dense dans \mathbb{R} au lieu d'être discret).

Définition 108. Étant donné un réseau \mathcal{L} , on appelle *réseau dual* le réseau $\mathcal{L}^* = \{z \in \text{Vect}(\mathcal{L}); \forall x \in \mathcal{L}, \langle x, z \rangle \in \mathbb{Z}\}$. La base duale de B en est une base.

Remarque 109. Le réseau \mathbb{A}_2 est le réseau hexagonal et $\mathbb{A}_3 \simeq \mathbb{D}_3$ est le réseau cubique face centrée des chimistes. Par ailleurs, \mathbb{A}_3^* est le réseau cubique (à corps) centré des chimistes. La structure du diamant n'est pas un \mathbb{Z} -module donc, n'est pas un réseau mais peut être obtenue comme la réunion de deux copies de \mathbb{A}_3 .

Les attributs d'un réseau

Définition 110. On appelle la matrice des produits scalaires $\mathbf{G} = (\langle \mathbf{b}_i, \mathbf{b}_j \rangle)_{i,j \leq n}$ la *matrice de Gram* de \mathcal{L}

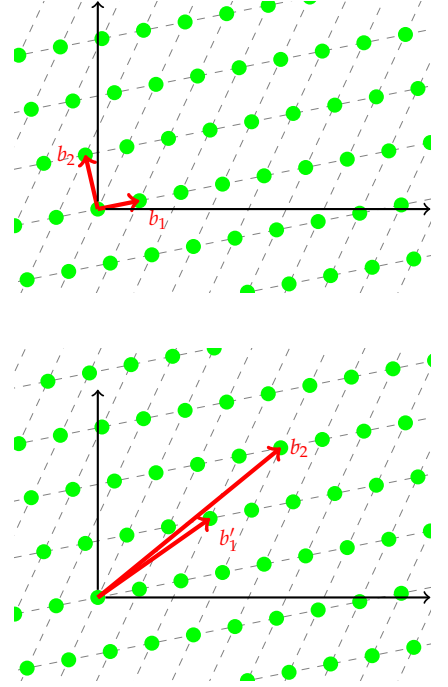


FIGURE 2: Un premier réseau quelconque avec deux bases distinctes

Toutes les notions de réseau discutées ici pourraient en réalité être traitées de façon équivalente sur la forme quadratique définie positive $x \in \mathbb{Z}^n \mapsto \mathbf{G}[x] = x' \mathbf{G} x \in \mathbb{R}$. Le point de vue historique est celui des formes quadratiques, Minkowski étant le premier à faire le lien avec les réseaux et à utiliser ses résultats en théorie algébrique des nombres.

Définition 111 (Déterminant). On appelle *déterminant* d'un réseau \mathcal{L} la quantité $\det(\mathcal{L}) = \det(\mathbf{G})$ et *discriminant* $\text{disc}(\mathcal{L}) = \sqrt{\det \mathcal{L}}$.

Le déterminant d'un réseau ne dépend pas de la base choisie, puisque toute autre base se déduit par transformation unimodulaire (changement de base de déterminant 1).

Quand le produit scalaire est le produit usuel de \mathbb{R}^n , on a simplement $\mathbf{G} = \mathbf{B}'\mathbf{B}$ et $\det(\mathcal{L}) = \det(\mathbf{G}) = \det(\mathbf{B})^2$.

Remarque 112. Le discriminant $\text{disc}(\mathcal{L})$ représente le covolume de \mathcal{L} , c'est-à-dire le volume du domaine fondamental $\{\sum_{i=1}^n x_i \mathbf{e}_i; 0 \leq x_i \leq 1\}$ ou du quotient $\mathbb{R}^n / \mathcal{L}$ (ou encore d'une cellule de Voronoï, cf. infra). Ainsi le discriminant d'un réseau donne une idée de la taille de ce réseau et l'on appréciera la longueur d'un vecteur (est-il court ou long?) relativement à $(\text{disc}(\mathcal{L}))^{1/n}$ (la puissance $1/n$ permettant l'homogénéité des grandeurs).

Exercice 113. Pour chacun des ensembles suivants, décider s'il s'agit d'un réseau. Si oui, en donner une base et calculer le déterminant.

1. $\mathcal{L}_1 = \{(x, y, z) \in \mathbb{Z}^3; 10x + 6y + 3z = 0\}$
2. $\mathcal{L}_2 = \{(x, y, z) \in \mathbb{Z}^3; 10x + 6y + 3z = 0 \pmod{11}\}$
3. $\mathcal{L}_3 = \{(x, y, z) \in \mathbb{Z}^3; 3x \equiv y \pmod{4}, y \equiv x + z \pmod{7}\}$
4. $\mathcal{L}_4 = \{(x, y, z) \in \mathbb{Z}^3; x^3 = 2x\}$

Exercice 114. On donne l'ensemble

$$\mathcal{L} = \left\{ (x, y, z) \in \mathbb{Z}^3; \begin{cases} 2x - 3y + 4z = 0 \\ 4x + 4y + 53z \equiv 0 \pmod{5} \end{cases} \right\}$$

plongé dans l'espace euclidien \mathbb{R}^3 muni du produit scalaire canonique. Montrer que \mathcal{L} est un réseau et en calculer une base.

Exercice 115. Soient \mathcal{L} et \mathcal{L}' deux réseaux de rang plein tels que $\mathcal{L}' \subseteq \mathcal{L} \subseteq \mathbb{Z}^n$. Montrer que $\det \mathcal{L}$ divise $\det \mathcal{L}'$.

Définition 116. On appelle *k-ième minimum successif* et on note $\lambda_k(\mathcal{L})$ le plus petit réels r tel que \mathcal{L} admette k vecteurs \mathbb{R} -linéairement indépendants de longueur inférieure à r .

Les minima successifs sont atteints par des vecteurs mais en dimension ≥ 5 ceux-ci ne forment par forcément une base de \mathcal{L} .

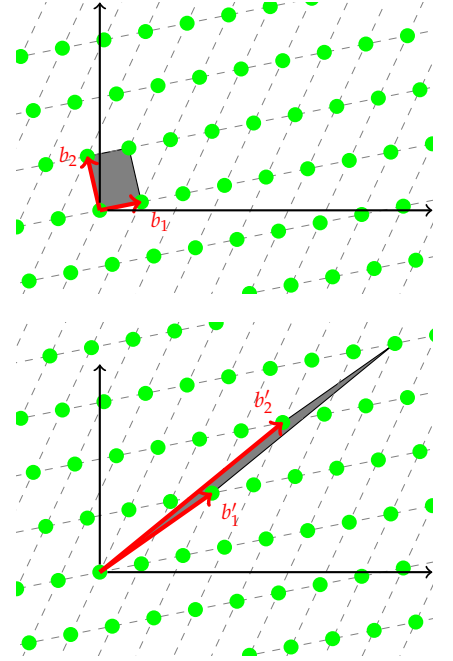


FIGURE 3: Discriminant du réseau

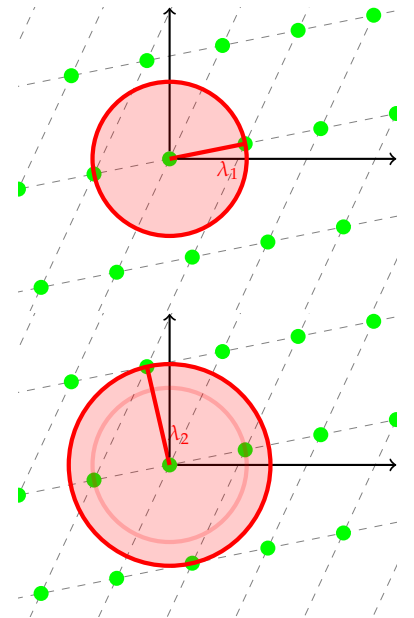


FIGURE 4: Minima successifs

Definition 117. On appelle *vecteur minimal* d'un réseau \mathcal{L} tout vecteur x de \mathcal{L} de longueur $\lambda_1(\mathcal{L})$ et on note $\min \mathcal{L} = \lambda_1(\mathcal{L})^2$.

Exercice 118. Nous reprenons les réseaux \mathbb{A}_n et \mathbb{D}_n définis à l'exemple 107. Soit e le vecteur

$$e = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \right)$$

On appelle \mathbb{E}_8 le réseau $\mathbb{E}_8 = \mathbb{D}_8 + \mathbb{Z}e$.

1. Trouver une base et le déterminant de \mathbb{A}_n en fonction de n .
2. Trouver une base et le déterminant de \mathbb{D}_n en fonction de n .
3. Trouver une base de \mathbb{E}_8 .
4. Montrer que \mathbb{E}_8 est de discriminant 1 (on dit que le réseau est *uni-modulaire*).
5. Vérifier que pour tout $x, y \in \mathbb{A}_n, \mathbb{D}_n$ ou \mathbb{E}_8 , $\langle x, x \rangle \in 2\mathbb{Z}$ et $\langle x, y \rangle \in \mathbb{Z}$ (on dit que le réseau est *pair*).
6. Montrer que les coordonnées d'un vecteur de \mathbb{E}_8 sont soit toutes entières, soit toutes demi-entières.
7. Énumérer les vecteurs x de $\mathbb{A}_n, \mathbb{D}_n$ ou \mathbb{E}_8 tels que $\langle x, x \rangle = 2$.
8. Combien y a-t-il de vecteurs minimaux ?

Exercice 119. Soit $B \in \mathbb{Q}^{n \times n}$ la matrice d'une base d'un réseau \mathcal{L} dont les vecteurs sont deux à deux orthogonaux. Prouver que les vecteurs minimaux de \mathcal{L} se trouvent parmi les vecteurs colonnes de B ou leur opposé.

Remarque 120. On peut remarquer que les ensembles des vecteurs minimaux de $\mathbb{A}_n, \mathbb{D}_n$ et \mathbb{E}_8 sont stables par réflexion d'hyperplan perpendiculaire à un vecteur minimal. On dit que ces ensembles des vecteurs minimaux sont des *systèmes de racines*. Les systèmes de racines sont un outil essentiel dans la classification des groupes de Lie (les groupes possédant une structure de variété différentielle tels que SL_n , O_n etc.) Comme ils sont engendrés par un système de racine, on qualifie de *réseaux de racine* les réseaux $\mathbb{A}_n, \mathbb{D}_n$ et \mathbb{E}_8 (il existe encore les réseaux \mathbb{E}_6 et \mathbb{E}_7).

Théorie de la réduction et application

La réduction

Un réseau possède une infinité de bases. Mais comme le montre la figure 2, seul un petit nombre de bases, formées de vecteurs peu longs, semblent sympathiques.

Réduire un réseau, c'est ramener une de ses bases à une forme prescrite et possédant certaines propriétés fixées à l'avance. (On peut penser par analogie aux bases orthonormées et à la procédure de Gram-Schmidt pour les espaces euclidiens). Une bonne théorie de la réduction conduit à une unique base réduite par réseau. Idéalement, on rêverait que cette base soit orthogonale et formée des minima successifs. Malheureusement, pour les réseaux, ces contraintes sont trop fortes pour que de telles exigences fonctionnent.

Il existe différentes théories de la réduction, selon l'objectif en vue, et aucune n'est algorithmiquement viable. On appelle par exemple *réduction de Minkowski* le choix d'une base $(\mathbf{u}_i)_i$ construite par récurrence où \mathbf{u}_i est le plus petit vecteur tel que $\mathbf{u}_1, \dots, \mathbf{u}_i$ se complète en une base de \mathcal{L} (attention, les longueurs des vecteurs peuvent être distinctes des minima successifs). Mais le calcul d'une telle base est un problème NP-difficile.

Pour les calculs pratiques, on se contente la réduction LLL, qui strictement parlant n'est qu'une pseudo-réduction, car la base réduite n'est pas unique, mais qui a l'avantage d'être calculable en temps polynomial.

L'algorithme LLL

Réseaux de rang 2 On commence par étudier la réduction en dimension 2. Idéalement, on souhaiterait une base de Gram-Schmidt. Ce n'est pas possible mais on peut se contenter de la notion suivante.

Définition 121. Une base réduite d'un réseau de rang 2 est une base formée par deux vecteurs $\mathbf{b}_1, \mathbf{b}_2$ tel que $\|\mathbf{b}_1\| \leq \|\mathbf{b}_2\|$ et $-\frac{1}{2} \leq \frac{\langle \mathbf{b}_1, \mathbf{b}_2 \rangle}{\|\mathbf{b}_1\|^2} \leq \frac{1}{2}$.

Géométriquement, cela revient à choisir le vecteur \mathbf{v}_2 dans le domaine décrit par la figure 5. Le vecteur \mathbf{v}_2 se trouve dans le demi-plan supérieur, car on peut toujours le remplacer par son opposé; \mathbf{v}_2 se trouve hors du demi-cercle car il est plus long que \mathbf{v}_1 ; et \mathbf{v}_2 se trouve dans la bande bleue d'épaisseur $\|\mathbf{v}_1\|$ car on peut toujours retrancher à \mathbf{v}_2 un multiple de \mathbf{v}_1 pour l'y amener.

Définition 122. On note $\lfloor u \rfloor$ l'entier le plus proche d'un réel $u \in \mathbb{R}$. Avec SageMath, on peut utiliser `round`.

Proposition 123. Pour que la base $[\mathbf{b}_1, \mathbf{b}_2]$ d'un réseau de rang 2 soit réduite il faut et il suffit que $\|\mathbf{b}_1\| \leq \|\mathbf{b}_2\| \leq \|\mathbf{b}_1 + \mathbf{b}_2\|, \|\mathbf{b}_1 - \mathbf{b}_2\|$.

Démonstration. Il suffit de voir que

$$\|\mathbf{b}_1 \pm \mathbf{b}_2\|^2 - \|\mathbf{b}_2\|^2 = \|\mathbf{b}_1\|^2 \pm 2\langle \mathbf{b}_1, \mathbf{b}_2 \rangle.$$

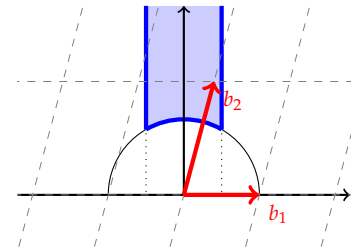


FIGURE 5: Domaine fondamental pour les réseaux de rang 2

Donc $\|\mathbf{b}_2\| \leq \|\mathbf{b}_1 + \mathbf{b}_2\|, \|\mathbf{b}_1 - \mathbf{b}_2\|$ si et seulement si $\|\mathbf{b}_1\|^2 \pm 2\langle \mathbf{b}_1, \mathbf{b}_2 \rangle \geq 0$, soit encore si et seulement si $-\frac{1}{2} < \frac{\langle \mathbf{b}_1, \mathbf{b}_2 \rangle}{\|\mathbf{b}_1\|^2} \leq \frac{1}{2}$. \square

Théorème 124. Une base d'un réseau \mathcal{L} est réduite si et seulement si les longueurs de ses vecteurs sont les minima successifs $\lambda_1(\mathcal{L})$ et $\lambda_2(\mathcal{L})$.

Démonstration. Si les vecteurs \mathbf{b}_1 et \mathbf{b}_2 sont des minima successifs, alors il est clair que $\|\mathbf{b}_1\| \leq \|\mathbf{b}_2\| \leq \|\mathbf{b}_1 + \mathbf{b}_2\|, \|\mathbf{b}_1 - \mathbf{b}_2\|$. Ainsi la proposition 123 montre que la base est réduite.

Réciproquement, supposons que la base soit réduite. Comme $\|\mathbf{b}_1\| \leq \|\mathbf{b}_2\|$, il nous suffit de montrer que pour tout vecteur \mathbf{v} non colinéaire ni à \mathbf{b}_1 ni à \mathbf{b}_2 , $\|\mathbf{v}\| \geq \|\mathbf{b}_2\|$. Soit $\mathbf{v} = \lambda\mathbf{b}_1 + \mu\mathbf{b}_2$ un vecteur non-nul tel que $\lambda\mu \neq 0$. On a alors

$$\|\mathbf{v}\|^2 - \|\mathbf{b}_2\|^2 = \lambda^2\|\mathbf{b}_1\|^2 + (\mu^2 - 1)\|\mathbf{b}_2\|^2 + 2\lambda\mu\langle \mathbf{b}_1, \mathbf{b}_2 \rangle$$

Comme $\mu^2 - 1 \geq 0$, en utilisant le fait que la base est réduite,

$$\|\mathbf{v}\|^2 - \|\mathbf{b}_2\|^2 \geq \lambda^2\|\mathbf{b}_1\|^2 + (\mu^2 - 1)\|\mathbf{b}_1\|^2 - |\lambda||\mu|\|\mathbf{b}_1\|^2$$

Finalement

$$\|\mathbf{v}\|^2 - \|\mathbf{b}_2\|^2 \geq \left(\left(|\lambda| - \frac{1}{2}|\mu| \right)^2 + \frac{3}{4}|\mu|^2 - 1 \right) \|\mathbf{b}_1\|^2.$$

Étudions au cas par cas la valeur entre parenthèse pour montrer qu'elle est toujours positive. Si $|\mu| \geq 2$, $\frac{3}{4}|\mu|^2 - 1 \geq 2$. Si $\mu = 0$, il ne reste que $|\lambda|^2 - 1$ qui est bien positif. Enfin, si $|\mu| = 1$, seul les choix $\lambda \in \{-1, 0, 1\}$ minimisent l'expression ciblée qui vaut alors 0. Donc \mathbf{b}_2 est bien un vecteur de norme inférieure à celle de \mathbf{v} . \square

L'algorithme suivant permet d'obtenir une base réduite. Son idée consiste à raccourcir le plus long des deux vecteurs en le faisant glisser le long du plus court et de répéter l'opération autant de fois que possible.

Algorithme 7 : Réduction de réseau de rang 2

Entrées : Base quelconque b_1, b_2 de \mathcal{L}

Sorties : Base réduite b_1, b_2 de \mathcal{L}

```

1 si  $\|b_1\| > \|b_2\|$  alors
2   Echanger( $b_1, b_2$ )
3 tant que  $\|b_1\| < \|b_2\|$  faire
4    $u \leftarrow \langle b_1, b_2 \rangle / \langle b_1, b_1 \rangle$ 
5    $b_2 \leftarrow b_2 - \text{Arrondi}(u)b_1$ 
6   Echanger( $b_1, b_2$ )
7 Echanger( $b_1, b_2$ )
8 retourner  $b_1, b_2$ 
```

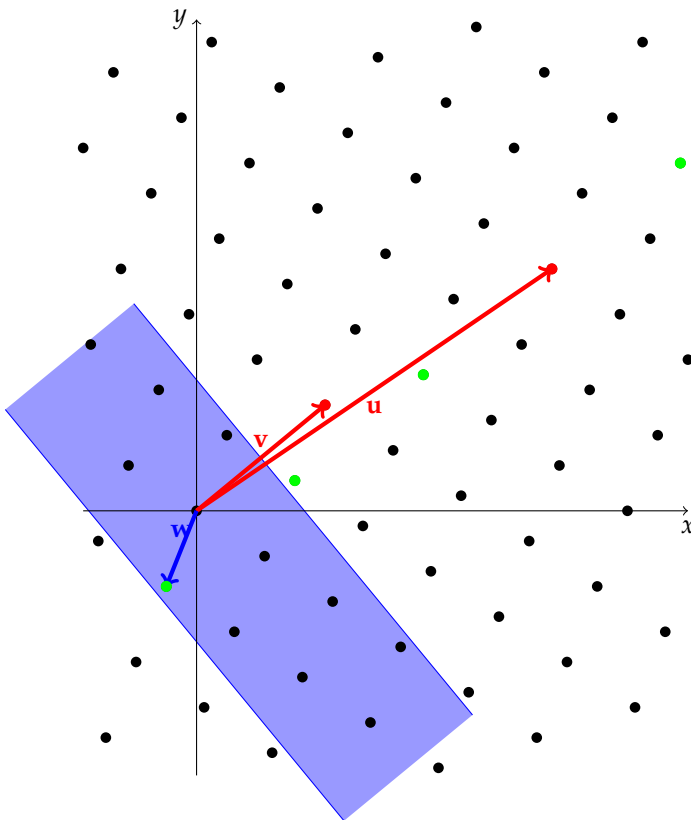
On remarquera que cet algorithme fonctionne comme l'algorithme d'Euclide de calcul du pgcd.

Théorème 125. *L'algorithme 7 se termine et calcule correctement une base LLL réduite d'un réseau.*

Démonstration. Après chaque étape de la boucle tant que, le nouveau vecteur \mathbf{b}_2 est le plus court parmi $\mathbf{b}_2 + \mathbb{Z}\mathbf{b}_1$, en particulier $\|\mathbf{b}_2\| \leq \|\mathbf{b}_2 \pm \mathbf{b}_1\|$. Si la boucle s'arrête, les conditions de la proposition (123) sont donc vérifiées. D'autre part, si la boucle se poursuit, la valeur $\lfloor u \rfloor$ doit être non nulle et dans ce cas la somme $\|\mathbf{b}_1\| + \|\mathbf{b}_2\|$ décroît. Mais cette somme ne prend que des valeurs discrètes positives, ce qui interdit une boucle infinie. \square

Exemple 126. On considère le réseau du plan euclidien usuel engendré par les vecteurs

$$\mathbf{u} = \begin{pmatrix} 47 \\ 32 \end{pmatrix} \quad \text{et} \quad \mathbf{v} = \begin{pmatrix} 17 \\ 14 \end{pmatrix}$$

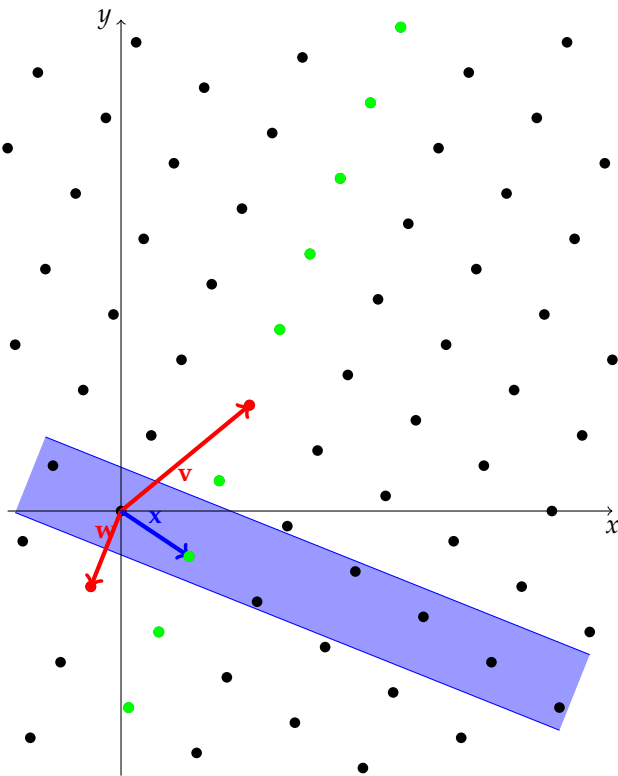


Le plus long des vecteurs est \mathbf{u} qui est de norme $\|\mathbf{u}\|^2 = 3233$ alors

que $\|\mathbf{v}\|^2 = 485$. On peut le raccourcir en posant

$$\begin{aligned}\mathbf{w} &= \mathbf{u} - \left\lfloor \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{v}\|^2} \right\rfloor \mathbf{v} \\ &= \mathbf{u} - \left\lfloor \frac{1247}{485} \right\rfloor \mathbf{v} \\ &= \mathbf{u} - 3\mathbf{v} \\ &= \begin{pmatrix} -4 \\ -10 \end{pmatrix}\end{aligned}$$

On obtient une meilleure base à savoir (\mathbf{w}, \mathbf{v}) avec $\|\mathbf{w}\|^2 = 116$. On peut continuer à essayer de réduire \mathbf{v} à l'aide de \mathbf{w} .



Le plus long des vecteurs est \mathbf{v} . On peut le raccourcir en posant

$$\begin{aligned}\mathbf{x} &= \mathbf{v} - \left\lfloor \frac{\langle \mathbf{v}, \mathbf{w} \rangle}{\|\mathbf{w}\|^2} \right\rfloor \mathbf{w} \\ &= \mathbf{v} - \left\lfloor \frac{-52}{29} \right\rfloor \mathbf{w} \\ &= \mathbf{v} + 2\mathbf{w} \\ &= \begin{pmatrix} 9 \\ -6 \end{pmatrix}\end{aligned}$$

On obtient une meilleure base à savoir (\mathbf{w}, \mathbf{x}) . Comme $\|\mathbf{x}\|^2 = 117$, il n'est plus possible de diminuer la longueur d'un des vecteurs. L'algorithme s'arrête. On a obtenu une base LLL réduite.

Remarque 127. Notons $\mathbf{b}_1^* = \mathbf{b}_1$, $\mathbf{b}_2^* = \mathbf{b}_2 - \mu_{2,1}\mathbf{b}_1^*$, la base de Gram-Schmidt associée. La condition de quasi-orthogonalité $-\frac{1}{2} < \frac{\langle \mathbf{b}_1, \mathbf{b}_2 \rangle}{\|\mathbf{b}_1\|} \leq \frac{1}{2}$ revient à écrire que $|\mu_{2,1}| \leq \frac{1}{2}$.

La condition d'ordre entre les longueurs des vecteurs $\|\mathbf{b}_1\| \leq \|\mathbf{b}_2\|$ revient à écrire que $\|\mathbf{b}_1^*\| \leq \|\mu_{2,1}\mathbf{b}_1^* + \mathbf{b}_2^*\|$. Or $\|\mathbf{b}_1^*\|^2 \leq \|\mu_{2,1}\mathbf{b}_1^* + \mathbf{b}_2^*\|^2 = |\mu_{2,1}|^2\|\mathbf{b}_1^*\|^2 + \|\mathbf{b}_2^*\|^2$ implique, sous l'hypothèse $|\mu_{2,1}| \leq \frac{1}{2}$, $\frac{3}{4}\|\mathbf{b}_1^*\|^2 \leq \|\mathbf{b}_2^*\|^2$.

Ce sont ces conditions que nous généralisons en les relachant.

Réseau de rang quelconque On rappelle que la base orthogonale de Gram-Schmidt se calcule par récurrence avec les formules :

$$\begin{cases} \mathbf{b}_1^* &= \mathbf{b}_1 \\ \forall i \geq 2 & \mathbf{b}_i^* = \mathbf{b}_i - \sum_{j < i} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^* \end{cases}$$

L'idée de la réduction LLL (ou réduction de Lovász) est de requérir les mêmes conditions qu'en dimension 2 aux vecteurs d'une base de notre réseau pris deux à deux. Afin d'obtenir un algorithme fonctionnant en temps polynomial, on est obligé toutefois de relâcher d'un facteur certain la condition d'ordre sur les longueurs.

Définition 128. Soit $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]$ une base d'un réseau et $\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^i \mu_{j,i} \mathbf{b}_j^*$ la base orthogonale de Gram-Schmidt associée. La base \mathbf{B} est dite LLL réduite si

1. $\forall 1 \leq j < i \leq n, \quad |\mu_{j,i}| \leq \frac{1}{2}$
2. $\forall 1 \leq i < n, \quad \|\mathbf{b}_i^*\|^2 \leq 2\|\mathbf{b}_{i+1}^*\|^2 \quad (\text{condition de Lovász})$

Attention : nous parlons de la base orthogonale et non pas de la base orthonormale de Gram-Schmidt associée. Utiliser la base orthonormale serait très maladroit car introduirait des racines carrées.

Exercice 129. Implémenter votre propre algorithme myGS de calcul de la base orthogonale de Gram-Schmidt (inutile de renormer les vecteurs).

Remarque 130. En vérité, le coefficient 2 dans la condition de Lovász est arbitraire et l'on pourrait faire la théorie plus généralement en fonction d'un paramètre δ en exigeant $\delta \cdot \|\mathbf{b}_i^*\|^2 \leq \|\mu_{i,i+1}\mathbf{b}_i^* + \mathbf{b}_{i+1}^*\|^2$ pour un certain $\delta \in]\frac{1}{4}, 1[$, δ proche de 1 donnant le meilleur résultat mais étant le plus coûteux à obtenir.

Comment comprendre cette définition ? La première condition insiste sur le fait que les vecteurs sont aussi orthogonaux que possibles et que l'on ne peut raccourcir la longueur de l'un en lui retranchant un précédent vecteur de la base. La seconde condition assure que $\|\text{proj}_{\langle \mathbf{b}_1, \dots, \mathbf{b}_{i-2} \rangle^\perp}(\mathbf{b}_{i-1})\|^2 \leq$

$2\|\text{proj}_{\langle \mathbf{b}_1, \dots, \mathbf{b}_{i-1} \rangle^\perp}(\mathbf{b}_{i-1})\|^2$, autrement dit que dans l'espace qui reste à conquérir une fois les $(i-2)$ premiers vecteurs choisis, à savoir dans $\langle \mathbf{b}_1, \dots, \mathbf{b}_{i-2} \rangle^\perp$, les « restes » des vecteurs qui suivent arrivent dans un ordre plutôt croissant.

Globalement, on espère que les vecteurs de la base réduite arrivent aussi orthogonaux que possibles et la suite des longueurs soit plutôt proche du minimum par rapport à l'ordre lexicographique. Plus précisément, on a les garanties suivantes conservant le minimum.

Lemme 131. *Si \mathbf{B} est une base LLL - réduite de \mathcal{L} , alors*

$$\lambda_1(\mathcal{L}) \geq \min_{1 \leq i \leq n} \|\mathbf{b}_i^*\|.$$

Démonstration. Soit $\mathbf{v} \neq 0$ un vecteur quelconque de \mathcal{L} , qui se décompose en

$$\mathbf{v} = \sum_{i \leq n_0} \lambda_i \mathbf{b}_i$$

avec $(\lambda_i)_{i \leq n_0} \subseteq \mathbb{Z}$ et $\lambda_{n_0} \neq 0$. Alors en réécrivant \mathbf{v} dans la base orthogonale, on a pour des réels $v_i \subseteq \mathbb{R}$

$$\mathbf{v} = \sum_{i \leq n_0} v_i \mathbf{b}_i^*$$

avec en particulier $v_{n_0} = \lambda_{n_0} \in \mathbb{Z}^*$. Mais alors, comme la base est orthogonale,

$$\|\mathbf{v}\|^2 = \sum_{i \leq n_0} |v_i|^2 \|\mathbf{b}_i^*\|^2 \geq |\lambda_{n_0}|^2 \|\mathbf{b}_{n_0}^*\|^2 \geq \min_{1 \leq i \leq n} \|\mathbf{b}_i^*\|^2.$$

comme voulu. \square

Théorème 132. *Soit $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]$ une base LLL -réduite. Alors*

$$\|\mathbf{b}_1\| \leq 2^{(n-1)/2} \lambda_1(\mathcal{L}). \quad (2)$$

Démonstration. D'une part $\|\mathbf{b}_1^*\| \leq 2^{1/2} \|\mathbf{b}_2^*\| \leq \dots \leq 2^{(n-1)/2} \|\mathbf{b}_n^*\|$, ce qui prouve que pour tout $i \leq n$,

$$\|\mathbf{b}_1^*\| \leq 2^{(n-1)/2} \|\mathbf{b}_i^*\|.$$

Mais $\mathbf{b}_1 = \mathbf{b}_1^*$ d'une part. D'autre part, $\min_{i \leq n} \|\mathbf{b}_i^*\| \leq \lambda_1(\mathcal{L})$ à cause du lemme précédent. Donc

$$\|\mathbf{b}_1\| \leq 2^{(n-1)/2} \lambda_1(\mathcal{L}),$$

ce qu'il fallait démontrer. \square

Passons à présent à l'algorithme qui justifie une telle notion de réduction. L'algorithme suivant, au fonctionnement très simple, calcule une base LLL -réduite. En pratique, le premier vecteur d'une base LLL

réduite obtenue par l'algorithme suivant est bien plus proche du minimum que prédit par le théorème.

Algorithme 8 : Réduction LLL d'un réseau

Entrées : Base quelconque $\mathcal{B} = \mathbf{b}_1, \dots, \mathbf{b}_n$ de \mathcal{L}

Sorties : Base LLL -réduite $\mathbf{b}_1, \dots, \mathbf{b}_n$ de \mathcal{L}

```

1 Repeter  $\leftarrow$  vrai
2 tant que Repeter faire
3   Calculer la base de Gram-Schmidt  $(\mathbf{b}_i^*)_{1 \leq i \leq n}$  et la matrice de
   passage  $\mu = ((\mu_{j,i}))_{1 \leq j \leq i \leq n}$  triang. sup. telles que
    $[\mathbf{b}_i] = [\mathbf{b}_i^*] \cdot \mu$ 
4   pour tous les  $i$  de 2 à  $n$  faire           // Etape de réduction
5     pour tous les  $j$  de  $i-1$  à 1 faire
6        $\mathbf{b}_i \leftarrow \mathbf{b}_i - \lfloor \mu_{j,i} \rfloor \mathbf{b}_j$ 
7        $\mu_i \leftarrow \mu_i - \lfloor \mu_{j,i} \rfloor \mu_j$ 
8   si  $\exists i_0$  tq  $\|\mathbf{b}_{i_0}^*\|^2 > 2\|\mathbf{b}_{i_0+1}^*\|^2$  alors           // Etape d'échange
9     Echanger( $\mathbf{b}_{i_0}, \mathbf{b}_{i_0+1}$ ) (Un seul échange)
10  sinon
11    Repeter  $\leftarrow$  faux
12 retourner  $\mathcal{B}$ 

```

Noter l'importance dans l'algorithme 8 de parcourir les valeurs de j dans l'ordre décroissant afin de ne plus modifier un coefficient $\mu_{j,i}$ une fois qu'il a été ajusté.

Théorème 133. *L'algorithme 8 se termine et calcule correctement une base LLL réduite d'un réseau.*

Démonstration. Les opérations effectuées sur la base sont toutes unimodulaires (échanges et transvections), les vecteurs obtenus continuent de former une base du même réseau. De plus, si l'algorithme s'arrête, la condition 2 d'une base LLL -réduite est clairement vérifiée à cause de l'étape d'échange éventuel. La condition 1 est prise en charge par le bloc de réduction. En effet, la base de Gram-Schmidt n'a pas besoin d'être mise à jour pendant cette boucle. De plus, après avoir retranché le bon

multiple de \mathbf{b}_j , on a $\left| \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \right| = |\lfloor \mu_{j,i} \rfloor| \leq \frac{1}{2}$ car $\langle \mathbf{b}_j, \mathbf{b}_j^* \rangle = \langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle$.

Reste à s'assurer que l'algorithme s'arrête. Pour simplifier la preuve, nous démontrons la terminaison de l'algorithme sous l'hypothèse que \mathcal{L} est un réseau défini sur \mathbb{Z} ; les idées restent les mêmes dans le cas général mais ajoutent des complications techniques.

On construit un invariant dont on va prouver qu'il décroît strictement à chaque étape de la boucle répéter. Notons le produit des déterminants des sous-réseaux

$$\Delta(\mathbf{B}) = \prod_{i=1}^n \det(\langle \mathbf{b}_1, \dots, \mathbf{b}_i \rangle_{\mathbb{Z}})$$

Nous voyons que $\Delta(\mathbf{B})$ ne prend que des valeurs entières. Les formules classiques de calcul de volume d'un pavé montrent que

$$\Delta(\mathbf{B}) = \left(\prod_{i=1}^n \|\mathbf{b}_i^*\|^{n-i+1} \right)^2$$

Cet invariant Δ ne change pas durant l'étape de réduction, puisque la base de Gram ne change pas. Au cours d'un échange entre \mathbf{b}_i et \mathbf{b}_{i+1} , seul le réseau engendré par $\mathbf{b}_1, \dots, \mathbf{b}_i$ change et en particulier $\|\mathbf{b}_i^*\|^2$ devient $\|\mu_{i,i+1}\mathbf{b}_i^* + \mathbf{b}_{i+1}^*\|^2$, les $\|\mathbf{b}_j^*\|$ pour $j < i$ restant inchangés. Mais

$$\frac{\|\mu_{i,i+1}\mathbf{b}_i^* + \mathbf{b}_{i+1}^*\|^2}{\|\mathbf{b}_i^*\|^2} = \frac{\mu_{i,i+1}^2 \|\mathbf{b}_i^*\|^2 + \|\mathbf{b}_{i+1}^*\|^2}{\|\mathbf{b}_i^*\|^2} < \frac{1}{4} + \frac{1}{2} = \frac{3}{4},$$

ce qui assure que

$$\Delta(\mathbf{B}_{\text{nouvelle}}) \leq \frac{3}{4} \cdot \Delta(\mathbf{B}_{\text{ancienne}}).$$

L'invariant $\Delta(\mathbf{B})$ tend vers 0 selon une progression au moins géométrique. Mais comme $\Delta(\mathbf{B})$ est toujours entier, il ne peut décroître indéfiniment. L'algorithme doit s'arrêter. Avec un peu de soin, on notera que le temps d'exécution est polynomial en $\max(n, \ln(\max_i \|\mathbf{b}_i\|))$ \square

Remarque 134. L'algorithme LLL est considéré comme un accomplissement majeur du XXIème siècle. Son importance ne se cantonne pas à la recherche approchée du minimum d'un réseau. Suprénement, il a été initialement introduit pour factoriser des polynômes de $\mathbb{Z}[x]$. Il a trouvé en 30 ans des applications considérables en mathématiques au delà de seules questions de réseaux, notamment en théorie des nombres, en programmation entière (optimisation) et en cryptographie.

- Exercice 135.** 1. Que suffit-il de contrôler dans l'algorithme 8 pour être sûr que la base soit réduite?
2. Écrire un programme `myLLL` qui implémente l'algorithme LLL (algorithme 8).
3. Comment améliorer l'algorithme 8 pour que la réduction de Gram-Schmidt ne soit pas recalculée entièrement à chaque étape?

Vous pouvez utiliser les méthodes `swap_columns` et `add_multiple_of_column` et le corrigé de l'exercice 129 ou une méthode native.

Exercice 136. On se place dans l'espace euclidien \mathbb{R}^3 muni du produit scalaire usuel. Dans chacun des cas suivants, trouver une base LLL réduite du réseau

$$\mathcal{L}_1 = \mathbb{Z} \begin{pmatrix} 3 \\ -5 \\ 0 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -3 \\ 6 \\ -2 \end{pmatrix},$$

$$\mathcal{L}_2 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ -4 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 11 \\ -22 \\ 11 \end{pmatrix},$$

$$\mathcal{L}_3 = \mathbb{Z} \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ 4 \\ 4 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ 0 \\ 7 \end{pmatrix}.$$

Exercice 137. Soit \mathcal{L} le réseau de l'espace euclidien \mathbb{R}^3 canonique engendré par les deux vecteurs

$$\mathbf{b}_1 = \begin{pmatrix} 4 \\ 4 \\ 1 \end{pmatrix} \quad \text{et} \quad \mathbf{b}_2 = \begin{pmatrix} 3 \\ 2 \\ 0 \end{pmatrix}$$

Donner une base LLL-réduite de \mathcal{L} ainsi que son déterminant et ses minima successifs.

Application 138 (Relations entières entre réels). On suppose que r réels n_1, n_2, \dots, n_r satisfont une relation à coefficients entiers de petite taille $\alpha_1 n_1 + \dots + \alpha_r n_r = 0$ inconnue. Afin de retrouver $(\alpha_i)_{i \leq r}$, on choisit un grand nombre M et on considère le réseau \mathcal{L} de rang r , plongé dans \mathbb{R}^{r+1} et de base \mathbf{B} suivante. On définit aussi un vecteur $\mathbf{v} \in \mathbb{R}^{r+1}$ qui se trouve appartenir au réseau.

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \\ \lfloor Mn_1 \rfloor & \lfloor Mn_2 \rfloor & \cdots & \lfloor Mn_r \rfloor \end{pmatrix} \quad \mathbf{v} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_r \\ \simeq 0 \end{pmatrix}$$

Les vecteurs de la base \mathbf{B} ont une longueur en $O(M)$ tandis que \mathbf{v} a une longueur en $O(1)$. Aussi \mathbf{v} est typiquement un vecteur court dans le réseau. Pour M assez grand, on peut espérer que $\pm \mathbf{v}$ ressorte comme premier vecteur de la base LLL réduite, ce qui dévoile les $(\alpha_i)_{i \leq r}$.

Exercice 139. Vous pouvez utiliser la fonction LLL de SageMath pour répondre aux questions.

1. Trouver la formule de Machin entre $\arctan(1)$, $\arctan(1/5)$ et $\arctan(1/239)$.
2. On donne l'entier

$$r = -2.5468182768840820791359975088097915$$

et la promesse que r est un nombre algébrique de degré ≤ 3 (i.e. il existe un polynôme entier π tels que $\alpha_0 + \alpha_1 r + \dots + \alpha_3 r^3 = 0$). Retrouver un polynôme π plausible. Vérifier sa réponse en calculant les racines de π avec la méthode `roots`.

Remarque 140. L'utilisation de LLL pour trouver des relations entre réels est aujourd'hui dépassée par des algorithmes tels que l'algorithme PSLQ. Ce type d'approche a permis de deviner et découvrir des formules telles que celle de Bailey-Borwein-Plouffe qui permettent désormais de calculer le i -ième bit de nombres tels que π sans calculer les précédents.

Problèmes classiques en lien avec les réseaux

Dans cette section, nous passons en revue plusieurs questions qui ne sont pas à strictement parler des problèmes de réseaux mais dont les solutions connues utilisent les réseaux pour être décrites.

Nombre de contact

Problème 141 (Kissing number). Le problème du *nombre de contact* consiste à trouver κ_n , le nombre de sphères-unité qui peuvent être placées contre d'une sphère-unité centrale sans qu'elles s'interpénètrent.

La solution à ce problème est connue en petite dimensions (≤ 4) : la configuration optimale provient des vecteurs minimaux de \mathbb{A}_2 , \mathbb{A}_3 et \mathbb{D}_4 . Elle est aussi connue en dimension 8 et 24 (vecteurs minimaux de \mathbb{E}_8 et du réseau de Leech). Ainsi $\kappa_2 = 6$, $\kappa_3 = 12$, $\kappa_4 = 24$, $\kappa_8 = 240$ et $\kappa_{24} = 196560$. Pour les autres valeurs de n , on connaît seulement des encadrements de κ_n . On sait par exemple que κ_n croît exponentiellement et qu'il existe, en dimension n , des réseaux dont le nombre de contact est supérieur à $(1,015)^{n-o(n)}$ (résultat de 2018 qui utilise des codes correcteurs d'erreurs algébriques).

Le problème du kissing number est un cas particulier de codes (correcteurs d'erreurs) sphériques. Les mots de code sont des éléments de la sphère unité, la distance entre deux mots est l'angle qu'ils forment. Le kissing number revient à requérir une distance minimale de $\frac{\pi}{3}$.

Empilement de sphères

Définition 142. La *densité d'un réseau* désigne la densité de l'empilement de boules dont les centres sont placés selon les points du réseau et dont le rayon est le plus grand tel que les boules ne se superposent pas. Elle vaut par conséquent

$$\frac{\omega_n(\lambda_1(\mathcal{L})/2)^n}{\text{disc } \mathcal{L}} = \frac{\pi^{n/2}}{(n/2)! 2^n} \frac{(\min \mathcal{L})^{n/2}}{(\det \mathcal{L})^{1/2}}$$

(où ω_n est le volume de la boule unité et $x!$ s'obtient via la fonction Γ). On dit encore que $\lambda_1(\mathcal{L})/2$ est le rayon d'empilement.

Proposition 143. Le rayon d'empilement correspond au rayon inscrit de la cellule de Voronoï.

Remarque 144. Les réseaux les plus denses sont connus en dimension ≤ 8 et en dimension 24. Il s'agit notamment de \mathbb{A}_2 , \mathbb{A}_3 , \mathbb{D}_4 , \mathbb{D}_5 , \mathbb{E}_8 . Le problème est ouvert dans les autres dimensions, bien qu'il existe un algorithme (l'algorithme de Voronoï) permettant en théorie d'en calculer la solution.

En dimension 8, \mathbb{E}_8 se construit à partir d'une première version de \mathbb{D}_8 et d'une copie de \mathbb{D}_8 placée dans les trous profonds de la première. En dimension 24, on retrouve le fabuleux *réseau de Leech* (Un réseau si extraordinairement symétrique que son groupe d'automorphisme possède 8 315 553 613 086 720 000 éléments. On retrouve ce réseau notamment dans les *monstrous moonshine conjectures*).

Exercice 145. Calculer la densité des réseaux \mathbb{A}_2 , \mathbb{A}_3 , \mathbb{D}_4 , \mathbb{D}_5 (voir exemple 107) et \mathbb{E}_8 (voir exercice 118).

Problème 146 (Empilement de sphère). On appelle *problème de l'empilement de sphère* (sphere packing) la question de trouver la densité maximale d'un empilement de sphère quelconque (sans structure). C'est un problème difficile. Seul le cas de la dimension 3 est résolu (anciennement conjecture de Kepler) : c'est la densité d'un empilement selon un réseau \mathbb{A}_3 . À partir de la dimension 9, on pense que les empilements de sphères en réseau ne sont pas toujours optimaux : à l'heure actuelle, on connaît pour certaines dimensions des empilements périodiques de sphères plus denses que les meilleurs empilements en réseau connus.

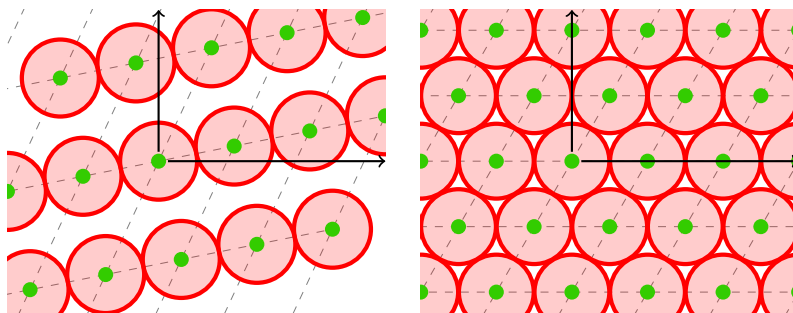


FIGURE 6: Empilements de sphère

Application 147 (Codes réseau pour le canal gaussien). Sous certaines conditions (grand rapport signal à bruit), le problème du codage d'un canal gaussien s'approche du problème d'empilement de sphère.

Recouvrement

Définition 148. On appelle *rayon de recouvrement* $\rho(\mathcal{L})$ d'un réseau \mathcal{L} le plus petit rayon r tel que la collection des boules de rayon r centrées en les points de \mathcal{L} recouvrent l'espace engendré par le réseau. On note $\Theta(\mathcal{L}) = \rho(\mathcal{L})^n \omega_n / (\text{disc } \mathcal{L})$ la densité de recouvrement.

Proposition 149. Le rayon de recouvrement correspond au rayon circonscrit de la cellule de Voronoï.

Théorème 150. On a $\lambda_n(\mathcal{L}) \leq 2\rho(\mathcal{L}) \leq \sqrt{n}\lambda_n(\mathcal{L})$.

De manière analogue au problème de l'empilement de sphère en réseau, on s'intéresse au minimum de $\Theta(\mathcal{L})$. On connaît le minimum en dimension ≤ 5 , il est atteint par les réseaux \mathbb{A}_5^* .

Problème 151 (Recouvrement). Le *problème du recouvrement de sphères* (*sphere covering problem*) consiste à minimiser la densité d'un recouvrement quelconque de l'espace par des sphères. Ce problème est largement ouvert ; on ne sait pas si ce minimum est atteint par un empilement en réseau, même en dimension 3.

Application 152. On peut voir le problème de recouvrement comme un problème de disposition de capteurs isotropes tels que leur zone de fonctionnement recouvre tout l'espace.

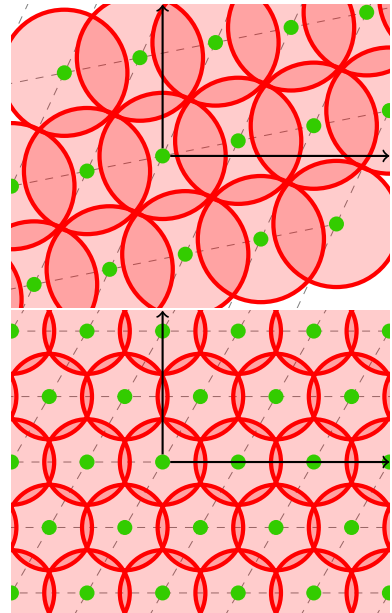


FIGURE 7: Recouvrement de sphère

TP 4 : Factorisation partielle de polynômes univariés sur un corps fini

Buts : Comprendre comment factoriser algorithmiquement un polynôme sur un corps fini.

Travaux préparatoires : Exercices 157, 163 (question 1), 170, 193 (question 1).

Évaluation du TP : Exercices 163 (sauf question 1, factorisation des puissances), 172 (factorisation SFC), 176 (factorisation EDD), 190 (Cantor-Zassenhauss), 192 (factorisation complète), 193 (question 2, racines), 191 (Etude de Cantor-Zassenhauss).

Les algorithmes de factorisation de polynômes sur des corps exacts (par exemple : les corps finis, les extensions finies de \mathbb{Q} , etc) s'appuient tous sur une première factorisation dans $\mathbb{F}_q[x]$. Nous expliquons dans le détail les algorithmes de factorisation sur un corps fini et mentionnons comment les utiliser en général sur $\mathbb{Z}[x]$ et $\mathbb{Q}[x]$.

Factoriser un polynôme sur d'autres corps comme \mathbb{R} ou \mathbb{C} relèvent plutôt du domaine de l'analyse numérique et utilisent des méthodes fondamentalement différentes, typiquement dérivées de la méthode de Newton pour la recherche de racine d'une fonction .

À toutes fins utiles, le code suivant pourra vous rappeler quelques fonctions de SageMath.

```
p=3; q=9; Fq.<alpha>=FiniteField(q)
Pol.<x> = PolynomialRing(Fq)
f=Pol.random_element(degree=7)
g=Pol([1,2,0,4])
```

Vous risquez de rencontrer des confusions entre déclarations locales et globales. Il pourra être utile de redéfinir les paramètres à l'intérieur de chacune de vos fonctions en suivant le modèle suivant.

```

def MyFunction(f):
    Pol=f.parent()
    x=Pol.gen()
    p=Pol.base_ring().characteristic()
    q=Pol.base_ring().cardinality()
    return True

```

Factorisation sur les corps finis

Nous nous plaçons sur l'anneau euclidien $\mathbb{F}_q[x]$ où q est une puissance d'un nombre premier p .

Definition 153. Un polynôme est dit *irréductible* s'il n'est pas divisible par un autre polynôme non-constant.

Definition 154. Un polynôme est dit *sans facteurs carrés* s'il n'est pas divisible par le carré d'un polynôme non-constant.

Comme l'anneau $\mathbb{F}_q[x]$ est euclidien, $\mathbb{F}_q[x]$ est automatiquement factoriel, ce qui signifie ici que :

Proposition 155. *Tout polynôme $f(x) \in \mathbb{F}_q[x]$ se décompose de manière unique en un produit*

$$f(x) = \omega \cdot \prod_{i=1}^k f_i(x)^{e_i}$$

où $\omega \in \mathbb{F}_q$ est le coefficient dominant, $(f_i)_{i \leq k}$ une famille de polynômes irréductibles unitaires appelés facteurs irréductibles de f et $(e_i)_{i \leq k}$ sont des entiers naturels non-nuls appelés multiplicités.

Definition 156. *Factoriser f signifie trouver la décomposition de la proposition ci-dessus.*

La plupart des algorithmes de factorisation de polynômes sur un corps fini repose sur les étapes suivantes :

1. La *factorisation sans facteurs carrés*, c'est-à-dire obtenir à partir de $f(x)$ la suite des polynômes (g_1, g_2, \dots) tels que

$$f(x) = \prod_{i=1}^s g_i(x)^{i},$$

$g_i(x)$ n'est pas divisible par un carré et les g_i sont premiers entre eux. Autrement dit,

$$g_i(x) = \prod_{\substack{1 \leq j \leq k \\ e_j = i}} f_j(x)$$

2. La *factorisation étagée en degrés distincts*, c'est-à-dire obtenir, pour $1 \leq i \leq s$, à partir de $g_i(x)$ la liste des polynômes $g_{i,d}(x)$ tels que

$$g_i(x) = \prod_{1 \leq d \leq t_i} g_{i,d}$$

où $g_{i,d}(x)$ contient exactement les facteurs irréductibles de degré d de $g_i(x)$. Autrement dit,

$$g_{i,d} = \prod_{\substack{1 \leq j \leq k \\ \deg(f_j)=d, e_j=i}} f_j(x).$$

3. la *factorisation de degrés égaux*, c'est-à-dire obtenir la liste des $\deg(g_{i,d})/d$ polynômes f_i à partir du polynôme $g_{i,d}$ sans facteurs carrés et produits d'irréductibles de degré d .

Pour la dernière étape, c'est-à-dire finir de rompre un polynôme en facteurs irréductibles, il existe différentes façons de faire. Nous présenterons les deux algorithmes historiques : l'un déterministe (Berlekamp, 1967) et l'autre probabiliste (Cantor-Zassenhaus, 1981). Les algorithmes plus récents continuent d'utiliser les idées de base de Berlekamp et Cantor-Zassenhaus.

L'outil principal sur lequel repose ces algorithmes est le calcul du pgcd autrement dit l'*algorithme d'Euclide*. En effet, si f est le polynôme à factoriser et h un polynôme quelconque, le pgcd $t = (f \wedge h)$ est un facteur de f . Lorsque h est bien choisi, t est un facteur non trivial de f et il ne reste plus qu'à factoriser t et f/t au lieu de f .

Exercice 157. (Sans SageMath) Quelle est la factorisation du polynôme $x^4 + 1 \in \mathbb{F}_p[x]$ en fonction du nombre premier p ? [Indication : utiliser l'identité $x^4 + 1 = x^4 \pm 2x^2 + 1 \mp 2x^2$]

Factorisation sans facteurs carrés

Définition 158. Étant donné un anneau A et un polynôme $f(x) = \sum_{i=0}^d f_i x^i \in A[x]$, on appelle *polynôme dérivé* et on note $f'(x)$ le polynôme

$$f'(x) = \sum_{i=1}^d f_i \cdot i \cdot x^{i-1}.$$

Exemple 159. Le polynôme dérivé de $x^4 + x^3 + x^2 + 1 \in \mathbb{F}_2[x]$ est x^2 .

Attention toutefois à la nuance entre caractéristique 0 (les surcorps de \mathbb{Q}) et caractéristique p (les surcorps de \mathbb{F}_p).

Lemme 160. Soit p premier, q une puissance de p et $f = \sum_i f_i x^i \in \mathbb{F}_q[x]$. Si $f'(x) = 0$, alors f est de la forme

$$f(x) = u(x^p) = (\tilde{u}(x))^p$$

où

$$u(x) = \sum_{i=0}^{\frac{\deg f}{p}} f_{p \cdot i} x^i,$$

$$\tilde{u}(x) = \sum_{i=0}^{\frac{\deg f}{p}} \sqrt[p]{f_{p \cdot i}} x^i$$

et $z \mapsto \sqrt[p]{z}$ est en fait simplement donné par $z \mapsto z^r$ avec $r = \frac{q}{p} = p^{-1} \pmod{q-1}$.

Démonstration. D'abord, $f' = 0$ implique directement que $if_i = 0$, soit $f_i = 0$ pour tout i non multiple de p . Donc f est de la forme $f = u(x^p)$. Rappelons que $(a+b)^p = a^p + b^p$ dans \mathbb{F}_q . Donc

$$\begin{aligned} \tilde{u}(x)^p &= \left(\tilde{u}_d x^d + \cdots + \tilde{u}_1 x + \tilde{u}_0 \right)^p \\ &= \tilde{u}_d^p x^{pd} + \cdots + \tilde{u}_1^p x^p + \tilde{u}_0^p \end{aligned}$$

il suffit d'identifier les coefficients. \square

Remarque 161. Pour calculer le polynôme \tilde{u} , il est nécessaire d'inverser l'application $\mathbb{F}_q \rightarrow \mathbb{F}_q, x \mapsto x^p$. Cette application est en réalité l'endomorphisme \mathbb{F}_p -linéaire de Frobenius (voir aussi exercice 3). On peut l'inverser soit en écrivant la matrice puis en inversant cette matrice (par un pivot de Gauss par exemple), soit en notant que $\text{Frob}_{\mathbb{F}_q/\mathbb{F}_p} \circ \cdots \circ \text{Frob}_{\mathbb{F}_q/\mathbb{F}_p} = \text{Id}$ (k fois) et donc que $\text{Frob}_{\mathbb{F}_q/\mathbb{F}_p}^{-1} = \text{Frob}_{\mathbb{F}_q/\mathbb{F}_p}^{k-1}$ (où k est le degré de l'extension $[\mathbb{F}_q : \mathbb{F}_p]$, i.e. $q = p^k$), auquel cas on voit que $\sqrt[p]{x} = x^{q/p}$.

Exemple 162. Le polynôme $f(x) = x^{125} + 2x^{30} + x^5 + 2 \in \mathbb{F}_5[x]$ est de dérivée nulle. En posant $u(x) = x^{25} + 2x^6 + x + 2$, on a

$$\begin{aligned} f(x) &= u(x^5) = u(x)^5 \\ &= (x^5)^{25} + 2(x^5)^6 + (x^5) + 2 \\ &= (x^{25} + 2x^6 + x + 2)^5 \end{aligned}$$

Exercice 163. 1. On définit \mathbb{F}_{1849} comme $\mathbb{F}_{43}[\omega]$ où $\omega^2 = -1$.

- (a) Pourquoi $x^2 + 1$ est-il irréductible dans $\mathbb{F}_{43}[x]$?
- (b) Soit $x = a + \omega b \in \mathbb{F}_{1849}$ avec $a, b \in \mathbb{F}_{43}$. Vérifier manuellement que $y = a - \omega b$ est une racine 43-ième de x .
- (c) Soit

$$\begin{aligned} f(x) &= x^{172} + (3 - 2\omega) x^{129} - 5\omega x^{86} \\ &\quad + (2 + 4\omega) x^{43} - 1 - \omega \in \mathbb{F}_{1849}[x] \end{aligned}$$

Trouver à la main un polynôme \tilde{u} tel que $\tilde{u}(x)^{43} = f(x)$.

2. Écrire un algorithme racine-p-polynome qui, à partir d'un polynôme f de dérivée nulle, calcule un polynôme \tilde{u} tel que $\tilde{u}(x)^p = f(x)$.
3. Vérifier racine-p-polynome pour le polynôme

$$f = x^{30} - x^{15} + \alpha x^3 + 1 \in \mathbb{F}_9[x]$$

où α est un générateur de \mathbb{F}_9 sur \mathbb{F}_3 .

4. Vérifier racine-p-polynome sur f^p pour 100 polynômes f tirés au hasard.

Les règles de dérivation formelle d'un produit sont les mêmes que pour la dérivée classique.

Proposition 164. Soit f le polynôme $f(x) = \prod_{i=1}^k f_i(x)^{e_i}$, où les $(f_i)_{i \leq k}$ sont ses facteurs irréductibles alors

$$f'(x) = \sum_{i=1}^k \left(\prod_{\substack{j=1 \\ j \neq i}}^k f_j(x)^{e_j} \right) \cdot e_i \cdot f_i'(x) \cdot f_i(x)^{e_i-1}.$$

Proposition 165. Soit f le polynôme $f(x) = \prod_{i=1}^k f_i(x)^{e_i} \in \mathbb{F}_q[x]$ et p la caractéristique de \mathbb{F}_q , alors

$$t(x) = f \wedge f'(x) = \left(\prod_{p \nmid e_i} f_i(x)^{e_i-1} \right) \left(\prod_{p \mid e_i} f_i(x)^{e_i} \right).$$

Démonstration. Les facteurs irréductibles de $t(x)$ sont forcément parmi les polynômes f_j car $t(x)$ divise $f(x)$. Nous notons v_{f_ℓ} la valuation d'un polynôme par rapport à f_ℓ (cf. définition 382). Alors $v_{f_\ell}(f) = e_\ell$. Par ailleurs, pour tout i compris entre 1 et k ,

$$\begin{aligned} v_{f_\ell} \left(\left(\prod_{\substack{j=1 \\ j \neq i}}^k f_j(x)^{e_j} \right) \cdot e_i \cdot f_i'(x) \cdot f_i(x)^{e_i-1} \right) \\ = \begin{cases} \infty & \text{si } p \mid e_i \\ e_\ell - 1 & \text{si } \ell = i \text{ et } p \nmid e_i, \\ e_\ell & \text{si } \ell \neq i \text{ et } p \nmid e_i \end{cases} \end{aligned}$$

donc

$$v_{f_\ell}(f') = \begin{cases} e_\ell - 1 & \text{si } p \nmid e_\ell \\ e_\ell \text{ ou } \infty & \text{si } p \mid e_\ell \end{cases},$$

La valuation de $f \wedge f'$ s'obtient en prenant le minimum de celle de f et de f' . D'où le résultat. \square

Remarque 166. En caractéristique 0 (i.e. sur un surcorps de \mathbb{Q}), on aurait tout simplement $t(x) = f(x) \wedge f'(x) = (\prod_i f_i(x)^{e_i-1})$.

Avec SageMath, `list` permet d'obtenir les coefficients d'un polynôme. `L[a:b:r]` permet d'extraire d'une liste L des termes en progression arithmétique.

Nous pouvons donc obtenir la partie sans facteurs carrés et le premier terme de la décomposition sans facteurs carrés comme suit, toujours en notant $t = f \wedge f'$

$$u(x) = \frac{f(x)}{t(x)} = \frac{\prod_{i=1}^k f_i(x)^{e_i}}{\left(\prod_{p \nmid e_i} f_i(x)^{e_i-1}\right) \left(\prod_{p \mid e_i} f_i(x)^{e_i}\right)} = \prod_{p \nmid e_i} f_i$$

Comme

$$t(x) \wedge u(x) = \prod_{\substack{p \nmid e_i \\ e_i \neq 1}} f_i,$$

nous en déduisons que

$$g_1(x) = \prod_{e_i=1} f_i = \frac{u(x)}{t(x) \wedge u(x)}.$$

Proposition 167. *Un polynôme $f \in \mathbb{F}_q[x]$ est sans facteurs carrés si et seulement si son polynôme dérivé f' est non nul et f est premier avec f' . En particulier, être sans facteur carré est invariant par extension de corps.*

On continue la factorisation sans facteurs carrés en prenant $t(x)$ à la place de $f(x)$. Lorsque $f' = 0$, on est arrivé à la situation du lemme 160 : f est de la forme $u(x^p) = \tilde{u}(x)^p$. On remplace f par \tilde{u} (abusivement noté ci-dessous $\sqrt[p]{f}$) et on reprend l'algorithme tant que f n'est pas constant.

Nous pouvons donc obtenir la décomposition sans facteurs carrés par un algorithme récursif, où la notation $\sqrt[p]{f} f^{1/p}$ désigne par abus le polynôme \tilde{u} du lemme 160. et où la notation $p \cdot \text{FsFC}(u)$ signifie qu'il faut multiplier toutes les multiplicités renvoyées par p .

Algorithme 9 : Factorisation sans facteurs carrés (FsFC)**Entrées :** Polynôme $f \in \mathbb{F}_q[x]$ unitaire**Sorties :** Liste de couples $(g_i, i) \in \mathbb{F}_q[x] \times \mathbb{N}$ tels que

$$f = \prod_{i=1}^s (g_i)^i \text{ et } g_i \text{ sans facteur carré}$$

```

1   $p \leftarrow$  caractéristique de  $\mathbb{F}_q$ 
2  si  $\deg f \leq 0$  alors
3    retourner  $\emptyset$ 
4  sinon si  $f' \neq 0$  alors
5     $i \leftarrow 1$ 
6     $L \leftarrow \emptyset$ 
7     $t \leftarrow (f \wedge f')$ 
8     $u \leftarrow f/t$ 
9    tant que  $u \neq 1$  faire
10    $y \leftarrow (t \wedge u)$ 
11   si  $p \nmid i$  et  $u/y \neq 1$  alors
12      $L \leftarrow L \cup \{(u/y, i)\}$ 
13    $i \leftarrow i + 1$ 
14    $u \leftarrow y$ 
15    $t \leftarrow t/y$ 
16   si  $t \neq 1$  alors
17      $L \leftarrow L \cup \{(s, pi); (s, i) \in \text{FsFC}(\sqrt[p]{t})\}$ 
18 sinon
19    $L \leftarrow \{(s, pi); (s, i) \in \text{FsFC}(\sqrt[p]{f})\}$ 
20 retourner  $L$ 

```

Exemple 168. Soit f le polynôme suivant sur $\mathbb{F}_3[x]$. Nous donnons les factorisations pour mieux comprendre l'algorithme, mais bien sûr elles sont inconnues de celui qui exécute l'algorithme.

$$f(x) = x^{11} - x^{10} + x^9 + x^8 + x^7 + x^6 - x^4 = (x^2 + x - 1)(x - 1)^2(x + 1)^3x^4$$

1. Premier appel à FsFC :

On a

$$f'(x) = -x^{10} - x^9 - x^7 + x^6 - x^3 = -(x - 1)(x + 1)^3x^3(x^3 - x^2 - x - 1)$$

et

$$t = f \wedge f' = x^7 - x^6 + x^4 - x^3 = (x - 1)x^3(x + 1)^3$$

On en tire

$$u = f/t = x^4 + x^2 + x = (x^2 + x - 1)(x - 1)x$$

au moment d'entrer dans la boucle "tant que".

- (a) Au premier passage dans la boucle, on a $y = (t \wedge u) = x^2 - x = (x - 1)x$, on sort $(1, u/y = x^2 + x - 1)$ comme voulu.
On continue avec $i = 2$, $u = x^2 - x = (x - 1)x$ et $t = x^5 + x^2 = x^2(x + 1)^3$.
- (b) Au deuxième passage dans la boucle, on a $y = (t \wedge u) = x$, on sort $(2, u/y = x - 1)$ comme voulu.
On continue avec $i = 3$, $u = x$ et $t = x^4 + x = x(x + 1)^3$.
- (c) Au troisième passage dans la boucle, on a $y = (t \wedge u) = x$, on sortirait $(3, u/y = 1)$.
On continue avec $i = 4$, $u = x$ et $t = x^3 + 1 = (x + 1)^3$.
- (d) Au quatrième passage dans la boucle, on a $y = (t \wedge u) = 1$, on sort $(4, u/y = x)$ comme voulu.
On continue avec $i = 5$, $u = 1$ et $t = (x + 1)^3$.

Finalement, $u = 1$ ce qui arrête la boucle “tant que”. Mais $t \neq 1$, on calcule la « racine cubique » de $t = x^3 + 1$, il s’agit de $x + 1$.

2. Deuxième appel à FsFC. L’appel $3 \cdot \text{FsFC}(x + 1)$ sort $(3, x + 1)$ comme voulu.

Remarque 169. Sur un surcorps de \mathbb{Q} , l’algorithme présenté fonctionne sans les complications de la caractéristique p , i.e. sans avoir besoin de l’appliquer récursivement.

Exercice 170. Vrai ou faux ?

1. Le polynôme $f(x) = x^{1000} + 2 \in \mathbb{F}_3[x]$ est sans facteur carré. Même question dans $\mathbb{F}_5[x]$.
2. Soit \mathbb{K} un corps quelconque et $f, g \in \mathbb{K}[x]$ deux polynômes, alors la partie sans facteurs carrés de fg est le produit des parties sans facteurs carrés de f et partie sans facteurs carrés de g .

Exercice 171. Nous avons vu que pour calculer la partie sans facteurs carrés, il suffit de connaître un algorithme pour le pgcd. Réciproquement, on suppose que l’on dispose d’un algorithme de calcul de la décomposition sans facteurs carrés et on souhaite calculer le pgcd de deux polynômes.

1. En supposant que f et g sont deux polynômes sans facteurs carrés, résoudre le problème.
2. Supposons que la décomposition sans facteurs carrés de f et de g soit $f = \prod_{i=1}^s f_i^i$ et $g = \prod_{i=1}^t g_i^i$. Montrer que

$$(f \wedge g) = \prod_{i=1}^{\min(s,t)} (f_i \cdots f_s \wedge g_i \cdots g_t)$$

3. Conclure

Exercice 172. 1. Écrire un programme myFsFC qui implémente l'algorithme de factorisation sans facteurs carrés (attention lors des divisions à ne pas sortir de l'anneau des polynômes : forcer le type ou utiliser des divisions entières //).

2. Trouver la factorisation sans facteurs carrés de

$$f(x) = x^{10} + 6x^9 + 3x^7 + 3x^3 + 4x^2 + 2 \in \mathbb{F}_7$$

et vérifier avec la méthode factor de SageMath.

3. Tester myFsFC sur 1000 polynômes tirés au hasard.

Factorisation étagée en degrés distincts

Théorème 173. Pour tout $d \geq 1$, le polynôme $x^{q^n} - x$ est le produit de l'ensemble des polynômes irréductibles unitaires dans $\mathbb{F}_q[x]$ dont le degré divise n .

Démonstration. Si f est un polynôme irréductible de degré k divisant $x^{q^n} - x$, alors ses racines α engendrent une extension \mathbb{F}_{q^k} de degré k . Mais comme f divise $x^{q^n} - x$, $\alpha^{q^n} = \alpha$, donc $\alpha \in \mathbb{F}_{q^n}$, donc $\mathbb{F}_q[\alpha] = \mathbb{F}_{q^k} \subseteq \mathbb{F}_{q^n}$ donc $k|n$.

Réciproquement, si f est irréductible de degré k divisant n , alors les racines de f engendrent une extension de degré k . Mais comme $k|n$, $\mathbb{F}_{q^k} \subseteq \mathbb{F}_{q^n}$, donc ses racines α appartiennent à \mathbb{F}_{q^n} et vérifient $\alpha^{q^n} = \alpha$, donc f divise $x^{q^n} - x$. \square

Corollaire 174 (Critère de Rabin). Un polynôme $f \in \mathbb{F}_q[x]$ de degré n est irréductible si et seulement si f divise $x^{q^n} - x$ et f est premier avec $(x^{q^{n/d}} - x)$ pour tout diviseur premier d de n .

Démonstration. Le premier point permet de s'assurer que f est un produit de facteurs irréductibles de degré divisant n de multiplicités 1. Le second point permet de s'assurer que f ne possède pas de facteurs irréductibles de degré divisant n/d pour d diviseur premier de n . Mais tout diviseur propre de n divise l'un des n/d , donc ceci élimine tout facteur de degré $< n$. Comme f est de degré n et n'a que des facteurs de degrés n , il est irréductible. \square

Si f est sans facteurs carrés, il est dès lors facile d'obtenir la factorisation en degrés distincts de f comme le pgcd de f avec $x^{q^d} - x$ en faisant varier d . Pour éviter une explosion de la taille de $x^{q^d} - x$, nous calculerons successivement $x^{q^d} \bmod f$ plutôt que de travailler avec x^{q^d} tout court.

Algorithme 10 : Factorisation étagée en degrés distincts**Entrées** : Polynôme $f \in \mathbb{F}_q[x]$ sans facteurs carrés**Sorties** : Suite de polynômes $(g_d)_d$ donnant la factorisation en degrés distincts

```

1  $h_0 \leftarrow x,$ 
2  $f_0 \leftarrow f,$ 
3  $i \leftarrow 0$ 
4 répéter
5    $i \leftarrow i + 1$ 
6    $h_i \leftarrow h_{i-1}^q \bmod f$ 
7    $g_i \leftarrow \text{pgcd}(h_i - x, f_{i-1})$ 
8    $f_i \leftarrow \frac{f_{i-1}}{g_i}$ 
9 jusqu'à  $f_i = 1;$ 
10 retourner  $(g_1, g_2, \dots)$ 

```

Exemple 175. Soit le polynôme

$$\begin{aligned}
f(x) &= x^{10} - 2x^9 + x^8 + x^7 - x^6 - 2x^5 + 2x^4 + 2x^3 - x \in \mathbb{F}_5[x] \\
&= \underbrace{[(x)(x+1)]}_{g_1} \cdot \underbrace{[(x^2+x+2)]}_{g_2} \cdot \underbrace{[(x^3+x+1)(x^3+x^2+2)]}_{g_3}
\end{aligned}$$

où encore une fois, l'exécutant de l'algorithme ne connaît pas la factorisation.

La trace de l'algorithme est

i	h_i	g_i
1	x^5	$x^2 + x$
2	$x^8 + 3x^7 + x^6 - x^5 + 3x^4 + x^3 + x^2 - x$	$x^2 + x + 2$
3	$x^9 + 3x^8 - x^7 + 2x^6 + 3x^4 + 2x^3 + x^2 + 3x$	$x^6 + x^5 + x^4 - x^3 + x^2 + 2x + 2$

i	f_i
1	$x^8 + 2x^7 - x^6 + 2x^5 + 2x^4 + x^3 + x^2 + x - 1$
2	$x^6 + x^5 + x^4 - x^3 + x^2 + 2x + 2$
3	1

Dans l'application de l'algorithme, on utilisera les techniques d'exponentiation rapide (cf. algorithme 38) pour calculer $h_{i-1}^q \bmod f$.

Exercice 176. Écrire un programme myFEDD qui implémente l'algorithme de factorisation étagée en degrés distincts. Vérifier son fonctionnement sur une série d'exemples significatifs.

Exercice 177. Soit $p \geq 5$ un premier, $f \in \mathbb{F}_p[x]$ un polynôme de degré 4. On suppose que f est premier avec $x^p - 1$ et $x^{p^2} - 1$. Que peut-on dire de f ?

Exercice 178. Soit $q \in \mathbb{N}$ une puissance d'un nombre premier et $r \in \mathbb{N}$ une puissance d'un nombre premier telle que $q \wedge r = 1$.

1. Montrer que $(q^r - q)/r$ est entier.
2. Montrer que, si r est premier, $\mathbb{F}_q[x]$ possède exactement $(q^r - q)/r$ polynômes irréductibles unitaires de degré r .
3. Trouver une formule pour le nombre de polynômes irréductibles unitaires de degré r dans $\mathbb{F}_q[x]$.

Exercice 179. Soit f un polynôme de $\mathbb{F}_{19}[x]$ de degré 30. Comment vérifier rapidement que f est le produit de trois facteurs irréductibles distincts de degré 10 sans calculer sa factorisation complète.

Exercice 180. Soit n un nombre premier. Montrer qu'il existe un binôme $f(x)$ irréductible de degré n sur $\mathbb{F}_q[x]$ si et seulement si $n|q - 1$.

Remarque 181. Les polynômes irréductibles introduits dans l'exercice 180 permettent de construire des cas particuliers de classes d'extensions de corps appelées *optimal extension fields* (OEFs) et utilisés en cryptographie. En général, on choisit p inférieur et proche d'une puissance de 2 (ce qui permet de représenter sans trop de perte des éléments de \mathbb{F}_p en base 2). Le fait d'utiliser un binôme $f(x) = x^n - \alpha$ permet de calculer facilement des réductions modulo f et donc d'implémenter une arithmétique rapide. Enfin, le fait d'utiliser un nombre premier n permet à des cryptosystèmes basés sur des courbes elliptiques de résister à des attaques par descente de Weil.

Exercice 182. Soit γ un élément primitif de \mathbb{F}_q . Montrer que le polynôme $x^{q-1} - \gamma$ est irréductible.

Exercice 183. Montrer qu'une somme d'un nombre pair de monômes de $\mathbb{F}_2[x]$ n'est jamais irréductible. En déduire qu'il n'existe pas de binôme irréductible sur $\mathbb{F}_2[x]$.

L'algorithme de Cantor Zassenhaus

Proposition 184. On suppose que la caractéristique p de \mathbb{F}_q est ≥ 3 . Soit $f(x) \in \mathbb{F}_q[x]$ le produit d'un ou de plusieurs polynômes irréductibles distincts de degré d . Alors, pour tout polynôme $u(x) \in \mathbb{F}_q[x]$, on a

$$f(x) = [f \wedge u] \cdot \left[f \wedge \left(u^{\frac{q^d-1}{2}} - 1 \right) \right] \cdot \left[f \wedge \left(u^{\frac{q^d-1}{2}} + 1 \right) \right].$$

Démonstration. Comme p est impair, on a la factorisation suivante

$$u^{q^d}(x) - u(x) = u(x) \left(u^{\frac{q^d-1}{2}}(x) - 1 \right) \left(u^{\frac{q^d-1}{2}}(x) + 1 \right).$$

Le polynôme f se scinde dans \mathbb{F}_{q^d} et toutes ses racines sont de multiplicité 1. Il nous suffit donc de voir que pour tout $\alpha \in \mathbb{F}_{q^d}$, α annule un et un seul des trois facteurs ci-dessus. Or on sait (par appartenance à \mathbb{F}_{q^d}) que $u^{q^d}(\alpha) = u(\alpha)$ ce qui montre qu'au moins un des facteurs s'annule. De plus, il est facile de voir que les trois facteurs sont premiers entre eux. Donc un seul des facteurs s'annule. \square

Remarque 185. Compte tenu des hypothèses sur f ,

$$f(x) = \prod_{\substack{\alpha \in \mathbb{F}_{q^d} \\ f(\alpha)=0}} (x - \alpha).$$

Par suite, la factorisation de la proposition correspond à

$$\begin{aligned} f \wedge u &= \prod_{\substack{\alpha \in \mathbb{F}_{q^d} \\ f(\alpha)=0, u(\alpha)=0}} (x - \alpha) \\ f \wedge (u^{(q^d-1)/2} - 1) &= \prod_{\substack{\alpha \in \mathbb{F}_{q^d} \\ f(\alpha)=0 \\ u(\alpha) \text{ est un carré}}} (x - \alpha) \\ f \wedge (u^{(q^d-1)/2} + 1) &= \prod_{\substack{\alpha \in \mathbb{F}_{q^d} \\ f(\alpha)=0 \\ u(\alpha) \text{ est un non-carré}}} (x - \alpha) \end{aligned} \tag{3}$$

On en déduit l'algorithme suivant, qui date de 1981.

Algorithme 11 : Algorithme de Cantor Zassenhaus ($p \geq 3$)

Entrées : Polynôme $f \in \mathbb{F}_q[x]$ produit de polynômes distincts irréductibles de degré d , entier d

Sorties : Facteurs de f

```

1 si  $\deg f = d$  alors
2   retourner  $[f]$ 
3 répéter
4    $u \leftarrow \text{PolynomeAleatoire}()$ 
5    $b \leftarrow f \wedge (u^{(q^d-1)/2} - 1)$ 
6 jusqu'à  $0 < \deg b < \deg f$ ;
7 retourner  $\text{CantorZassenhaus}(b, d) \cup \text{CantorZassenhaus}(f/b, d)$ 
```

La probabilité qu'un facteur irréductible f_i de f soit attrapé dans b est proche de $1/2$. Compte tenu de la remarque 185, il s'agit de la probabilité que $u(x_0)$ soit un carré de \mathbb{F}_{q^d} où x_0 est l'une des racines de f_i . Mais en représentant \mathbb{F}_{q^d} comme $\mathbb{F}_q[x]/\langle f_i \rangle$, on s'aperçoit qu'il s'agit aussi du fait que $u(x)$ est un carré modulo f_i . On peut donc

reformuler l'équation 3 :

$$\begin{aligned} f \wedge u &= \prod_{i \text{ tq } u \equiv 0 \pmod{f_i}} f_i \\ f \wedge (u^{(q^d-1)/2} - 1) &= \prod_{i \text{ tq } u \equiv \square \pmod{f_i}} f_i \\ f \wedge (u^{(q^d-1)/2} + 1) &= \prod_{i \text{ tq } u \equiv \square \pmod{f_i}} f_i \end{aligned}$$

Aussi, la probabilité de trouver un facteur non trivial b dans la boucle de l'algorithme est proche de $1 - \frac{1}{2^{k-1}}$ (où k est le nombre de facteurs de f), ce qui explique que l'algorithme termine avec probabilité 1. (La valeur exacte étant $1 - \left(\frac{q^d-1}{2q^d}\right)^k - \left(\frac{q^d+1}{2q^d}\right)^k$).

Exemple 186. On cherche à factoriser

$$f = x^4 + 1 = \underbrace{(x^2 + x - 1)}_{f_1} \underbrace{(x^2 - x - 1)}_{f_2} \in \mathbb{F}_3[x].$$

Les carrés modulo f_1 sont $\{1, -1, -x+1, x-1\}$ et les non-carrés $\{x, -x, x+1, -x-1\}$. Les carrés modulo f_2 sont $\{1, -1, x+1, -x-1\}$ et les non-carrés $\{x, -x, x-1, -x+1\}$. La figure 8 illustre les issues possibles d'une étape de l'algorithme de Cantor-Zassenhaus en fonction de la valeur de $(u \pmod{f_1}, u \pmod{f_2})$. La zone bleue correspond au cas où une étape renvoie $b = f_1$ et la zone verte au cas où $b = f_2$. La probabilité de factoriser f est exactement 40/81.

Par exemple, avec $u_0(x) = x^3 - x^2 - x - 1$, on obtient $f \wedge (u_0^4 - 1) = f_2$ (et $f \wedge (u_0^4 + 1) = f_1$). Par ailleurs, $(u_0 \pmod{f_1}, u_0 \pmod{f_2}) = (-x, -1)$.

Remarque 187. Dans la ligne 4 de l'algorithme, il n'y a pas d'intérêt à choisir un polynôme de degré $\geq \deg f$. On peut en fait se contenter de choisir u de degré $\leq 2d - 1$.

Remarque 188. Pour accélérer les choses, le calcul de $u^{(p^d-1)/2} - 1$ se fait modulo f en utilisant l'algorithme 38 d'exponentiation rapide.

Remarque 189. En caractéristique 2, on opère ainsi. Rappelons que la trace de l'extension \mathbb{F}_{2^m} sur \mathbb{F}_2 est la somme des itérés des endomorphismes de Frobenius, soit

$$\text{Tr}_m = \begin{cases} \mathbb{F}_{2^m} & \rightarrow \mathbb{F}_2 \\ x & \mapsto \sum_{i=0}^{m-1} x^{2^i} \end{cases}.$$

On peut vérifier l'identité polynomiale $x^{2^m} - x = (\text{Tr}_m(x) - 1) \cdot \text{Tr}_m(x)$. Aussi, lorsque $f \in \mathbb{F}_q[x]$ avec $q = 2^k$ et f est un produit d'irréductibles distincts de degré d , on a pour tout polynôme u

$$f = (f \wedge T_{kd} \circ u) \cdot (f \wedge (T_{kd} \circ u + 1)).$$

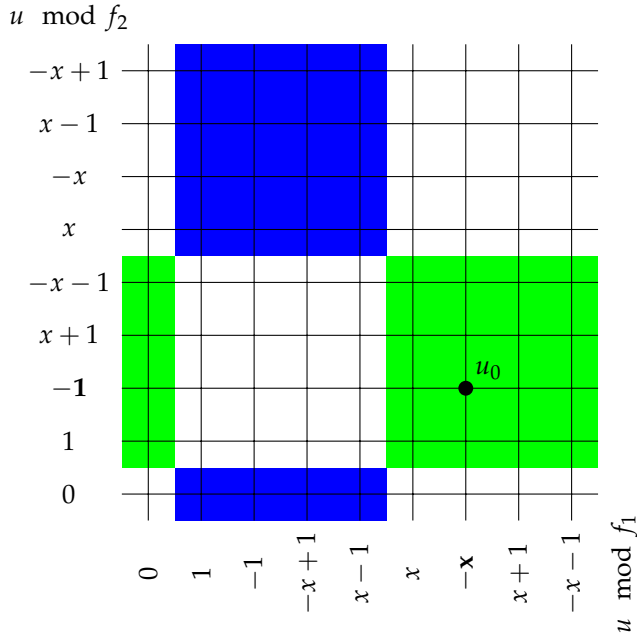


FIGURE 8: Fonctionnement de l'algorithme de C.-Z. avec $f = (x^2 + x - 1)(x^2 - x - 1) \in \mathbb{F}_3[x]$.

avec comme facteurs

$$f \wedge \text{Tr}_{kd} \circ u = \prod_{\substack{\alpha \in \mathbb{F}_{q^d} \\ f(\alpha)=0, \text{Tr}_{kd}(u(\alpha))=0}} (x - \alpha)$$

et

$$f \wedge (\text{Tr}_{kd} \circ u - 1) = \prod_{\substack{\alpha \in \mathbb{F}_{q^d} \\ f(\alpha)=0, \text{Tr}_{kd}(u(\alpha))=1}} (x - \alpha).$$

- Exercice 190.** 1. Écrire un programme `myCZ` qui implémente l'algorithme de factorisation de Cantor-Zassenhaus pour la caractéristique impaire. (On choisira u de degré $< 2d$.)
2. Écrire un programme `myCZ2` qui implémente l'algorithme de factorisation de Cantor-Zassenhaus pour la caractéristique 2.
3. Tester votre algorithme en choisissant un produit aléatoire de facteurs tirés de la liste

```
[f for f in Pol.polynomials(of_degree=d)
if f.is_irreducible()
and f.leading_coefficient()==1]
```

Exercice 191. Choisir un polynôme $f \in \mathbb{F}_q[x]$ comme produit de deux polynômes irréductibles f_1 et f_2 de même degré d . Construire la liste L_i des carrés modulo f_i et L'_i des non carrés modulo f_i . Que renvoie l'algorithme de Cantor-Zassenhaus appliqué avec u lorsque u

$\text{mod } f_1 \in L_1$ et $u \text{ mod } f_2 \in L_2$ (utiliser crt pour construire u) ? Même question pour les trois autres cas. Généraliser.

Exercice 192. Écrire un programme qui effectue la factorisation dans $\mathbb{F}_q[x]$.

Exercice 193. Soit $f \in \mathbb{F}_q[x]$ un polynôme.

1. Proposer un algorithme complet calculant les racines du polynôme f dans \mathbb{F}_q . On ne s'intéresse pas à leur multiplicité.
2. Implémenter votre solution.

TP 5 : Factorisation complète de polynômes univariés

Buts : Terminer l'étude des méthodes de factorisation sur un corps fini. Apprendre à factoriser un polynôme en général en utilisant les méthodes propres aux corps finis

Travaux préparatoires : Exercices 201 (question 1), & 208.

Évaluation du TP : Exercices 201 (Berlekamp, sauf question 1), 211 (Hensel) & 214 (Factorisation finale avec LLL).

Algorithme de factorisation de Berlekamp

Nous supposons que $f(x) = \prod_{i=1}^k f_i(x) \in \mathbb{F}_q[x]$ est un polynôme sans facteurs carrés et que les polynômes f_i sont ses facteurs irréductibles. L'algorithme de Berlekamp, qui date de 1967, est historiquement le premier algorithme de factorisation sur $\mathbb{F}_q[x]$. Il est basé sur la décomposition en composantes primaires (voir aussi définition 386) suivante :

$$\mathbb{F}_q[x]/\langle f \rangle = \bigoplus_{i=1}^k \mathbb{F}_q[x]/\langle f_i \rangle, \quad (4)$$

que l'on obtient par application du lemme chinois (les f_i sont premiers entre eux). Bien entendu, si l'on ne connaît que f , on ne peut a priori que travailler dans $\mathbb{F}_q[x]/\langle f \rangle$ en représentant les éléments comme des restes de polynômes dans la division par f . Comme les $(f_i)_{i \leq k}$ sont irréductibles, on a même l'identification

$$\mathbb{F}_q[x]/\langle f \rangle \simeq \bigoplus_{i=1}^k \mathbb{F}_{q^{\deg f_i}}. \quad (5)$$

Dans cette écriture, un élément u de degré $< \deg f$ dont la i -ième composante dans $\bigoplus_{i=1}^k \mathbb{F}_q[x]/\langle f_i \rangle$ est nul est divisible par f_i . Donc le pgcd $(u \wedge f)$ est un facteur non trivial de f .

A priori nous ne pouvons pas trouver un tel élément. Par contre nous allons voir qu'il est facile de calculer le sous- \mathbb{F}_q -espace vectoriel \mathcal{B} de $\mathbb{F}_q[x]/\langle f \rangle$ qui s'identifie à la partie $\bigoplus_{i=1}^k \mathbb{F}_q \subseteq \bigoplus_{i=1}^k \mathbb{F}_{q^{\deg f_i}}$. En effet, $\mathbb{F}_q \subseteq \mathbb{F}_{q^{\deg f_i}}$ est exactement l'ensemble des zéros de $\sigma(x) - x$ où σ est l'endomorphisme de Frobenius $\sigma : x \mapsto x^q$ et par suite \mathcal{B} est aussi égal

à l'ensemble des polynômes $h(x) \in \mathbb{F}_q[x]/\langle f \rangle$ tels que $h(x)^q \equiv h(x) \pmod{f}$. Mais l'application $h(x) \mapsto h(x)^q$ est une application linéaire sur $\mathbb{F}_q[x]/\langle f \rangle$ et trouver le noyau de $\sigma - \text{Id}$ est un simple problème d'algèbre linéaire sur \mathbb{F}_q (voir aussi exercice 3).

L'algorithme de Berlekamp consiste à calculer la matrice \mathbf{Q} de l'endomorphisme $\sigma : z \mapsto z^q$ sur $\mathbb{F}_q[x]/\langle f \rangle$, calculer une base $\mathbf{b}_1, \dots, \mathbf{b}_k$ du noyau de $\mathbf{Q} - \mathbf{I}_n$ puis casser récursivement les facteurs de f en de plus de petits facteurs en calculant $f \wedge a$ où $a(x) = b_i(x) - \alpha$ pour $1 \leq i \leq k$ et $\alpha \in \mathbb{F}_q$ ($b_i(x)$ étant le polynôme s'identifiant au vecteur \mathbf{b}_i).

Algorithme 12 : Algorithme de Berlekamp

Entrées : Polynôme $f \in \mathbb{F}_q[x]$ produit de polynômes sans facteurs carrés de degré n

Sorties : Facteurs de f

```

1  pour  $0 \leq j < n$  faire
2    Calculer les coefficients de la matrice
       $\mathbf{Q} = ((q_{i,j}))_{0 \leq i,j < n} \in \mathbb{F}_q^{n \times n}$  tels que  $\sum_{0 \leq i < n} q_{i,j} x^i \equiv (x^j)^q \pmod{f}$ 
3  Calculer (par le pivot de Gauß) une base  $[\mathbf{b}_1, \dots, \mathbf{b}_k]$  du noyau
      de  $\mathbf{Q} - \mathbf{I}_n \in \mathbb{F}_q^{n \times n}$ ,  $\mathbf{b}_1$  étant le vecteur  $(1, 0, \dots, 0)$ .
4   $F = \{f\}$  ;                               /* Liste des facteurs */
5   $j \leftarrow 1$  ;                               /* Indice d'un vecteur de base */
6  tant que Longueur( $F$ )  $< k$  faire
7     $j \leftarrow j + 1$ 
8     $C = \{\tilde{f} \in F; \deg \tilde{f} > 1\}$ 
9    pour  $\tilde{f} \in C$  faire
10      $B \leftarrow \emptyset$ 
11     pour  $\alpha \in \mathbb{F}_q$  faire
12        $a \leftarrow \tilde{f} \wedge (b_j - \alpha)$ 
13       si  $\deg a \geq 1$  alors
14          $B \leftarrow B \cup \{a\}$ 
15      $F \leftarrow (F \setminus \{\tilde{f}\}) \cup B$ 

```

Définition 194. La matrice \mathbf{Q} s'appelle la *matrice de Petr-Berlekamp*; le sous-module \mathcal{B} s'appelle la *sous-algèbre de Berlekamp*.

La dimension de $\ker(\mathbf{Q} - \mathbf{I}_n)$ égale le nombre de facteurs irréductibles de f .

Théorème 195. L'algorithme de Berlekamp est correct.

La preuve repose sur les deux observations :

Lemme 196. Soit g un diviseur de f (polynôme sans facteurs carrés), alors

pour tout b dans la sous-algèbre de Berlekamp \mathcal{B} ,

$$g = \prod_{\alpha \in \mathbb{F}_q} g(x) \wedge (b(x) - \alpha). \quad (6)$$

Démonstration. Réduit modulo f_i , b est une constante de \mathbb{F}_q . Aussi $g \wedge (b - \alpha)$ attrape l'ensemble des facteurs f_i de g tels que $b(x) \equiv \alpha \pmod{f_i}$. En faisant varier α , on est certain de capturer tous les facteurs. \square

Lemme 197. Pour tout $i \neq i'$, il existe au moins un élément b_j de la base de \mathcal{B} tel que l'équation 6 sépare les facteurs f_i et $f_{i'}$.

Démonstration. Si tel n'était pas le cas, tous les b_j auraient même réduction modulo f_i et $f_{i'}$ et ne pourraient engendrer que le sous-espace des $b \in \mathcal{B}$ tels que $b \pmod{f_i} = b \pmod{f_{i'}}$ ce qui est impossible. \square

Remarque 198. Lorsque q est grand, le temps de trouver le bon $\alpha \in \mathbb{F}_q$ peut être long. On peut préférer choisir a comme une combinaison aléatoire des (b_i) et calculer le pgcd de f avec $a^{(p-1)/2} - 1$ en utilisant la même astuce que pour l'algorithme de Cantor-Zassenhaus.

Remarque 199. Avec l'algorithme de Berlekamp, l'étape de factorisation en degrés distincts peut être omise, seule l'hypothèse « f est sans facteurs carrés » est importante. Il est toutefois utile de la conserver car elle factorise partiellement f à peu de frais et accorde un gain de temps.

Exemple 200. Soit $f(x) = x^4 + x^3 + x^2 + 1$ le polynôme de $\mathbb{F}_4[x]$ à factoriser. Pour représenter $\mathbb{F}_4[x]/\langle f \rangle$, nous utilisons la base des monômes :

$$\mathbb{F}_4[x]/\langle f \rangle = \mathbb{F}_4 \oplus \mathbb{F}_4 x \oplus \mathbb{F}_4 x^2 \oplus \mathbb{F}_4 x^3.$$

Les images de la base de $\mathbb{F}_4[x]/\langle f \rangle$ par l'application $\sigma : h(x) \mapsto (h(x))^4 \pmod{f}$ sont

$$\sigma : \begin{cases} 1 & \mapsto 1 \pmod{f} \\ x & \mapsto x^4 = x^3 + x^2 + 1 \pmod{f} \\ x^2 & \mapsto x^8 = x \pmod{f} \\ x^3 & \mapsto x^{12} = x^2 + x + 1 \pmod{f} \end{cases}$$

On en déduit la matrice \mathbf{Q} de σ :

$$\mathbf{Q} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Nous recherchons maintenant le noyau de

$$\mathbf{Q} - \mathbf{I}_4 = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

La matrice est de rang 2. Donc le noyau est de dimension $r = 2$ (et par suite, il y a deux facteurs irréductibles). Sa base est

$$\mathbf{b}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ et } \mathbf{b}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

Le vecteur \mathbf{b}_1 s'identifie au polynôme constant $b_1(x) = 1$; le vecteur \mathbf{b}_2 s'identifie au polynôme $b_2(x) = x + x^3$. Nous essayons de calculer les pgcd $f \wedge x^3 + x + \alpha$ pour $\alpha \in \mathbb{F}_4$. (Remarquer qu'il est inutile de chercher un facteur avec b_1 et que ceci est toujours le cas). On a déjà pour $\alpha = 0$: $f \wedge x^3 + x = x + 1$, de plus pour $\alpha = 1$, on a $f \wedge (x^3 + x + 1) = x^3 + x + 1$. Nous avons trouvé deux facteurs et il n'y en a pas d'autres, donc f est décomposé :

$$f = (x + 1)(x^3 + x + 1).$$

À postériori, nous pouvons analyser que $\mathbb{F}_4[x]/\langle x^4 + x^3 + x^2 + 1 \rangle \simeq \mathbb{F}_4 \oplus \mathbb{F}_{64}$ avec

$$\begin{array}{ccc} \mathbb{F}_4[x]/\langle x^4 + x^3 + x^2 + 1 \rangle & \rightarrow & \underbrace{\mathbb{F}_4[x]/\langle x + 1 \rangle}_{\mathbb{F}_4} \oplus \underbrace{\mathbb{F}_4[x]/\langle x^3 + x + 1 \rangle}_{\mathbb{F}_{64}} \\ x^3 + x + 0 & \mapsto & (0, 1) \\ x^3 + x + 1 & \mapsto & (1, 0) \end{array}$$

Exercice 201. 1. On donne $f = x^3 - x^2 - 1 \in \mathbb{F}_3[x]$.

(a) Que valent x^3 et x^6 modulo f ? En déduire la matrice Petr-Berlekamp Q .

(b) Montrer que le noyau de $\mathbf{Q} - \mathbf{I}_3$ est engendré par

$$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}.$$

(c) En déduire la factorisation de f .

2. Écrire un programme `myB` qui implémente l'algorithme de Berlekamp.

Conseils : ne pas réécrire l'algorithme du pivot de Gauß mais utiliser `right_kernel` ou `eigenvectors_right`. En SageMath, `p[i]` désigne

le coefficient du monôme d'exposant i du polynôme p (et vaut 0 au besoin). Enfin, $Q.column(j)$ renvoie le j -ième vecteur colonne de la matrice Q .

3. Vérifier votre algorithme sur des exemples significatifs.
4. Assembler les algorithmes du TP précédent avec l'algorithme de Berlekamp pour former un algorithme `myFactor` factorisant n'importe quel polynôme.

Exercice 202. Comment construire la base canonique de la sous-algèbre de Berlekamp \mathcal{B} à l'aide de polynômes interpolateurs de Lagrange.

Factorisation sur d'autres anneaux

Nous ne mentionnons que les cas de $\mathbb{Z}[x]$ et $\mathbb{Q}[x]$. La situation est bien moins évidente à traiter : nous ne faisons qu'illustrer les techniques employées à ce jour. Un algorithme complet de factorisation n'est pas attendu au terme de ce TP.

Principes généraux

Factoriser un polynôme dans $\mathbb{Q}[x]$ peut se ramener à la factorisation dans $\mathbb{Z}[x]$ pour peu que l'on multiplie le polynôme par un certain entier qui chasse tous les dénominateurs. Nous ne discuterons que la factorisation dans $\mathbb{Z}[x]$.

Définition 203. Si $f = f_n x^n + \dots + f_1 x + f_0 \in \mathbb{Z}[x]$, on note sa norme p

$$\|f\|_p = \left(\sum_{i=0}^n |f_i|^p \right)^{1/p} \quad \text{et} \quad \|f\|_\infty = \max_{0 \leq i \leq n} |f_i|.$$

Lemme 204 (Borne de Mignotte). *Si le polynôme $g \in \mathbb{Z}[x]$ divise le polynôme $f = f_n x^n + \dots + f_1 x + f_0 \in \mathbb{Z}[x]$, alors*

$$\|g\|_\infty \leq (n+1)^{1/2} 2^n |f_n| \cdot \|f\|_\infty.$$

Démonstration. [Admis]

□

Les algorithmes de factorisation dans $\mathbb{Z}[x]$ fonctionnent tous de la même manière. Supposons que f se factorise en

$$f(x) = f_1(x) f_2(x) \cdots f_k(x) \in \mathbb{Z}[x]$$

Alors, lorsque l'on réduit l'égalité modulo m , on a encore :

$$\overline{f}(x) = \overline{f_1}(x) \overline{f_2}(x) \cdots \overline{f_k}(x) \in \mathbb{Z}/m\mathbb{Z}[x].$$

Mais pour m assez grand ($m \geq 2B + 1$ où B est la borne de Mignotte), lorsque l'on écrit \bar{f}_i en centrant les coefficients entre $\lfloor -m/2 \rfloor$ et $\lceil m/2 \rceil$, l'écriture de \bar{f}_i se confond avec celle de f_i . On peut donc retrouver la factorisation de f_i par relèvement des \bar{f}_i .

Est-ce suffisant pour factoriser ? Malheureusement, il arrive que la factorisation de \bar{f} soit plus fine que celle de f : i.e. la factorisation de \bar{f} comporte plus de facteurs que celle de f .

Exemple 205. Soit

$$f(x) = x^6 + x^5 + x^4 + x^2 + x + 1 = \underbrace{(x^2 + x + 1)}_{f_1(x)} \underbrace{(x^4 + 1)}_{f_2(x)} \in \mathbb{Z}[x]$$

Nous tirons au hasard quelques grands nombres premiers m et factorisons \bar{f} dans $\mathbb{F}_m[x]$ à l'aide des méthodes des sections précédentes, on obtient comme facteurs irréductibles :

$$\begin{aligned} \bar{f}(x) &= \underbrace{(x + 3105)(x - 3104)}_{\bar{f}_1} \underbrace{(x^2 + 825)(x^2 - 825)}_{\bar{f}_2} \in \mathbb{F}_{6421}[x] \\ \bar{f}(x) &= \underbrace{(x^2 + x + 1)}_{\bar{f}_1} \underbrace{(x + 225)(x + 2975)(x - 225)(x - 2975)}_{\bar{f}_2} \in \mathbb{F}_{7121}[x] \\ \bar{f}(x) &= \underbrace{(x + 480)(x - 479)}_{\bar{f}_1} \underbrace{(x^2 + 1758x - 1)(x^2 - 1758x - 1)}_{\bar{f}_2} \in \mathbb{F}_{5347}[x]. \end{aligned}$$

Telles quelles, ces factorisations dans $\mathbb{F}_m[x]$ ne donnent pas la factorisation dans $\mathbb{Z}[x]$. Il est nécessaire de trouver comment recombinaer les facteurs obtenus pour retomber sur f_1 et f_2 .

En général, il faut toujours recomposer les facteurs de \bar{f} obtenus et tester s'ils sont vraiment des facteurs de f en tant que polynômes de $\mathbb{Z}[x]$.

Algorithme 13 : Algorithme de factorisation sur \mathbb{Z}

Entrées : Polynôme $f \in \mathbb{Z}[x]$ sans facteurs carrés

Sorties : Facteurs de f

- 1 $B \leftarrow (n + 1)^{1/2} 2^n \|f_n\| \|f\|_\infty$; /* borne de Mignotte */
 - 2 $m \leftarrow p^e$ puissance de premier telle que $p^e \geq 2B + 1$ et $p \nmid f_n$
 - 3 Factoriser \bar{f} dans $\mathbb{Z}/m\mathbb{Z}[x]$: $\bar{f} = r_1 r_2 \dots r_s \in \mathbb{Z}/m\mathbb{Z}[x]$
 - 4 **pour** $S \subseteq [1, s]$ **faire**
 - 5 $\tilde{g} \leftarrow \prod_{i \in S} r_i$
 - 6 Relever g dans $\mathbb{Z}[x]$ en centrant les coefficients entre $-m/2$ et $m/2$
 - 7 Tester si g divise f dans $\mathbb{Z}[x]$.
-

Deux types de choix de $m = p^e$ sont possibles :

1. soit on choisit un grand nombre premier p (et $e = 1$)

2. soit, pour un nombre premier p fixé, on choisit un grand exposant e .

Attention : on parle de factorisation dans $\mathbb{Z}/p^e\mathbb{Z}$ et non pas dans \mathbb{F}_{p^e} . Les deux structures n'ont rien à voir entre elle : ne les confondez pas !

Dans le second cas, factoriser \bar{f} se fait grâce au lemme et relèvement de Hensel. C'est la méthode la plus efficace et celle utilisée en pratique.

Pour accomplir le dernier point (regroupement des facteurs), on peut soit énumérer toutes les possibilités (il peut y avoir un nombre exponentiel en s de cas à traiter), soit utiliser judicieusement l'algorithme LLL pour trouver le regroupement de facteurs S .

Remarque 206. Si l'on essaye de regrouper les facteurs par force brute, il est utile de commencer par contrôler si le coefficient constant de g divise celui de f . On économise ainsi ses forces.

Exercice 207. Soit $f \in \mathbb{Z}[x]$ le polynôme $f(x) = f_n x^n + \cdots + f_1 x + f_0$, on note son polynôme réciproque

$$f^*(x) = f_0 x^n + \cdots + f_{n-1} x + f_n.$$

1. Montrer que si $f = gh \in \mathbb{Z}[x]$, alors $f^* = g^* h^*$.
2. Comment peut-on améliorer l'algorithme de factorisation dans $\mathbb{Z}[x]$ (penser à la borne de Mignotte) ?

Exercice 208. En reprenant l'exercice 157, montrer qu'il existe des polynômes tels que quel que soit le choix de p , l'étape de regroupement des facteurs est incontournable dans l'algorithme de factorisation sur $\mathbb{Z}[x]$.

Relèvement de Hensel

Soit m un entier. Le relèvement de Hensel est une technique qui permet, étant donné un début de factorisation $f \equiv gh \pmod{m}$ dans $\mathbb{Z}/m\mathbb{Z}[x]$, de prolonger cette factorisation en $f \equiv \hat{g}\hat{h}$ dans $\mathbb{Z}/m^2\mathbb{Z}[x]$. Pour y arriver, nous supposons connue une relation de Bezout $sg + th \equiv 1 \pmod{m}$. Brièvement, nous voyons que si

$$e = f - gh, \quad \hat{g} = g + te, \quad \hat{h} = h + se,$$

alors

$$f - \hat{g}\hat{h} = [\cdots] = (1 - sg - th)e - ste^2 \equiv 0 \pmod{m^2}.$$

Le seul inconvénient de cette approche est que les degrés de \hat{g} et \hat{h} peuvent être plus grand que ceux de g et h . Pour dépasser ce problème, nous sommes obligés de recourir à certaines divisions euclidiennes.

Pour simplifier la présentation, nous supposons que f , g et h sont des polynômes unitaires.

Nous présentons une version qui permet de passer de

$$\left\{ \begin{array}{l} g, h, s, t \in \mathbb{Z}[x] \\ f \equiv gh \pmod{m} \\ sg + th \equiv 1 \pmod{m} \\ \deg s < \deg h, \quad \deg t < \deg g, \end{array} \right. \quad (7)$$

à

$$\left\{ \begin{array}{l} g^*, h^*, s^*, t^* \in \mathbb{Z}[x] \\ f \equiv g^* h^* \pmod{m^2} \quad \text{et} \quad s^* g^* + t^* h^* \equiv 1 \pmod{m^2} \\ g \equiv g^* \pmod{m}, \quad h \equiv h^* \pmod{m}, \\ s \equiv s^* \pmod{m}, \quad t \equiv t^* \pmod{m}, \\ \deg s^* < \deg h^*, \quad \deg t^* < \deg g^*, \end{array} \right. \quad (8)$$

Algorithme 14 : Relèvement de Hensel

Entrées : Entier m . Éléments $f, g, h, s, t \in \mathbb{Z}[x]$ tels que (7)

Sorties : Éléments $g^*, h^*, s^*, t^* \in \mathbb{Z}[x]$ tels que (8)

- 1 $e \leftarrow f - gh \pmod{m^2}$
 - 2 $(q, r) \leftarrow$ division euclidienne de se par h modulo m^2
 - 3 $g^* \leftarrow g + te + qg \pmod{m^2}$
 - 4 $h^* \leftarrow h + r \pmod{m^2}$
 - 5 $b \leftarrow sg^* + th^* - 1 \pmod{m^2}$
 - 6 $(c, d) \leftarrow$ division euclidienne de sb par h^* modulo m^2
 - 7 $s^* \leftarrow s - d \pmod{m^2}$
 - 8 $t^* \leftarrow t - tb - cg^*$.
-

Exercice 209. Montrer que (8) est vérifié à l'issue de l'algorithme.

Exemple 210. Soit

$$f(x) = x^4 + 78x^3 - 2556x^2 + 4389x - 722 \in \mathbb{Z}[x]$$

un polynôme à factoriser. Un agent a déjà déterminé que

$$f(x) = \underbrace{(x^2 + x - 1)}_{g_0} \underbrace{(x^2 - x - 1)}_{h_0} \pmod{3}$$

et souhaite trouver une factorisation modulo 3^6 . Par l'algorithme d'Euclide dans \mathbb{F}_3 , on peut calculer

$$s_0(x) = -x + 1 \quad t_0(x) = x + 1 \quad s_0 g_0 + t_0 h_0 = -2 \equiv 1 \pmod{3}.$$

Après une étape de relèvement, on arrive à

$$g_1(x) = x^2 + 4x - 1, \quad h_1(x) = x^2 + 2x + 2,$$

$$s_1(x) = -x + 1, \quad t_1(x) = x + 1$$

et

$$\begin{cases} f - g_1 h_1 &= 72x^3 - 2565x^2 + 4383x - 720 \equiv 0 \pmod{9} \\ s_1 g_1 + t_1 h_1 &= 9x + 1 \equiv 1 \pmod{9} \end{cases}.$$

Après une seconde étape de relèvement, on arrive à

$$g_2 = x^2 + 22x - 19, \quad h_2 = x^2 - 25x + 38, \\ s_2 = -x + 10, \quad t_2 = x + 37$$

et

$$\begin{cases} f - g_2 h_2 &= 81x^3 - 2025x^2 + 3078x \equiv 0 \pmod{81} \\ s_2 g_2 + t_2 h_2 &= -648x + 1216 \equiv 1 \pmod{81} \end{cases}.$$

Après une troisième étape de relèvement, on arrive à

$$g_3 = x^2 + 103x - 19, \quad h_3 = x^2 - 25x + 38, \\ s_3 = 2591x + 1630, \quad t_3 = -2591x + 1333$$

et même

$$f - g_3 h_3 = 0.$$

On a obtenu non seulement une factorisation modulo 3^8 (qui reste vraie modulo 3^ℓ pour tout $\ell \leq 8$) mais finalement même la vraie factorisation dans $\mathbb{Z}[x]$.

Exercice 211. 1. Écrire un programme `polynomeCentre` qui renvoie, à partir de $f \in \mathbb{Z}[x]$ et de $m \in \mathbb{N}$, un polynôme $\tilde{f} \in \mathbb{Z}[x]$ tel que $\tilde{f} \equiv f \pmod{m}$ et dont les coefficients sont compris entre $-m/2$ et $m/2$.

Par exemple en appliquant `lift_centered()` à `f.list()`.

2. Écrire un programme `myHensel` qui code l'algorithme 30 de relèvement d'Hensel.
3. Écrire un algorithme qui reçoit en entrée des polynômes f, g et h (g et h premiers entre eux) et des entiers p premier et ℓ tels que

$$f \equiv gh \pmod{p}$$

et renvoie des polynômes \hat{g} et \hat{h} tels que

$$f \equiv \hat{g}\hat{h} \pmod{p^\ell}.$$

(Remarquer qu'il suffit d'aller jusqu'à p^{2^t} avec $2^t \geq \ell$.)

4. Tester les algorithmes à partir de

$$x^4 - 1 \equiv \underbrace{(x^3 + 2x^2 - x - 2)}_{g(x)} \underbrace{(x - 2)}_{h(x)} \pmod{5}$$

pour obtenir une factorisation modulo 25 et 625. Indication : On pourra utiliser après contrôle

$$s(x) = -2 \quad t(x) = 2x^2 - 2x - 1.$$

ou bien (pour les puristes)

```

PolZZ.<x> = PolynomialRing(ZZ)
m = 5; g = x^3+2*x^2-x-2; h = x-2
d,ss,tt = xgcd(g,h)
s=PolZZ(ss/mod(d,m)); t=PolZZ(tt/mod(d,m))

```

5. Comment réutiliser l'algorithme ci-dessus lorsque f se factorise en un produit de plus de 2 facteurs ?

Regroupement des facteurs avec LLL

On admet le résultat suivant

Lemme 212. Soient f et g deux polynômes de $\mathbb{Z}[x]$ de degrés n et k (non-nuls). Soit m un entier tel que $(\|f\|_2)^k (\|g\|_2)^n < m$. On suppose que $u \in \mathbb{Z}[x]$ est un polynôme non constant, unitaire divisant f et g dans $\mathbb{Z}/m\mathbb{Z}[x]$. Alors $f \wedge g \in \mathbb{Z}[x]$ est non-constant.

Ce résultat s'utilise comme suit : f est le polynôme que l'on souhaite factoriser, u est un des facteurs modulaires de f et g est un polynôme construit exprès pour satisfaire aux conditions du lemme. Pour ce faire, on choisit un polynôme de faible norme de la forme

$$g = qu + rm.$$

Ce problème peut être résolu en regardant les polynômes comme des vecteurs et chercher un court vecteur, via l'algorithme LLL, dans le réseau engendré par

$$\{u(x)x^i; 0 \leq i < j-d\} \cup \{mx^i; 0 \leq i < j\} \quad (9)$$

où d est le degré de u et j un paramètre à choisir.

Exemple 213. On cherche à factoriser $x^3 - 1$ dont la factorisation (encore inconnue) est

$$f = x^3 - 1 = (x-1)(x^2 + x + 1) \in \mathbb{Z}[x].$$

On a fixé $p = 7$, $e = 8$ et $m = p^e = 5764801$ (qui excède la borne de Mignotte). On commence par calculer une factorisation dans \mathbb{F}_7 qui est

$$f = (x-2)(x-1)(x+3) \in \mathbb{F}_7[x].$$

On applique le lemme de Hensel (2 fois) pour remonter cette factorisation en

$$f = (x+19)(x-1)(x-18) \in \mathbb{Z}/7^2\mathbb{Z}[x]$$

où les termes se correspondent. On réapplique le lemme de Hensel pour obtenir

$$f = (x+1048)(x-1)(x-1047) \in \mathbb{Z}/7^4\mathbb{Z}[x]$$

et enfin

$$f = (x + 3376854)(x - 1)(x - 3376853) \in \mathbb{Z}/7^8\mathbb{Z}[x].$$

À ce stade, on cherche un facteur de f par application du lemme. On choisit $u(x) = x - 3376853$ (arbitrairement), g doit donc être de la forme

$$g(x) = (x - 3376853)v(x) + 5764801w(x), \quad v, w \in \mathbb{Z}[x]$$

On cherche g comme vecteur du réseau engendré par les vecteurs colonnes de

$$\begin{pmatrix} -3376853 & 0 & 5764801 & 0 & 0 \\ 1 & -3376853 & 0 & 5764801 & 0 \\ 0 & 1 & 0 & 0 & 5764801 \end{pmatrix}$$

dont une base LLL -réduite est

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1424 \\ 1345 \\ 80 \end{pmatrix}, \begin{pmatrix} 80 \\ -1424 \\ 1345 \end{pmatrix}.$$

Le premier vecteur est un vecteur relativement court. Il s'identifie au polynôme $g(x) = x^2 + x + 1$. On peut vérifier que $f \wedge g = x^2 + x + 1$ qui est bien un facteur de f dans $\mathbb{Z}[x]$.

Exercice 214. Soit

$$f = x^4 - x^3 - 5x^2 + 12x - 6 \in \mathbb{Z}[x].$$

un polynôme à factoriser.

1.(a) Montrer que f possède 4 racines $\alpha, \beta, \gamma, \delta$ dans \mathbb{F}_{13} .

(b) Donner 4 entiers $\hat{\alpha}, \hat{\beta}, \hat{\gamma}$ et $\hat{\delta} \in \mathbb{Z} \cap [-\frac{13^4}{2}, \frac{13^4}{2}]$ tels que

$$\hat{\alpha} \equiv \alpha \pmod{13}, \quad \hat{\beta} \equiv \beta \pmod{13},$$

$$\hat{\gamma} \equiv \gamma \pmod{13}, \quad \hat{\delta} \equiv \delta \pmod{13},$$

$$\text{et } f(x) = (x - \hat{\alpha})(x - \hat{\beta})(x - \hat{\gamma})(x - \hat{\delta}) \in \mathbb{Z}/13^4\mathbb{Z}[x].$$

[Si votre algorithme de relèvement d'Hensel ne fonctionne pas, « tricher » et employer la force brute pour obtenir ces nombres.]

(c) Montrer que $\hat{\alpha}, \hat{\beta}, \hat{\gamma}$ et $\hat{\delta}$ ne sont pas des racines de f .

2. On pose $u(x) = x + 7626$ et $j = 3$.

(a) Vérifier que $u(x)$ est un diviseur de f modulo 13^4 . Donner une base LLL -réduite du réseau décrit par l'équation 9.

(b) En déduire un facteur de f .

(c) Calculer la factorisation complète de f sur \mathbb{Z} .

TP 6 : Bases de Gröbner et systèmes polynomiaux multivariés

Buts : Apprendre à manipuler des systèmes d'équations polynomiales, notamment trouver leurs zéros ou éliminer une variable d'un système.

Travaux préparatoires : Cours et exercice 231

Évaluation du TP : Exercices 218 (fonctions SageMath), 221 (division multivariée), 252 (base de Groebner), 253 (appartenance à un idéal), 256 (résolution d'un système), 257 (optimisation), 263 (manipulations algébriques), 262 (ovales de Descartes).

Nous abordons dans ce T.P. la manipulation de systèmes d'équations polynomiales. Le mécanisme de recherche d'une base de Gröbner que nous allons étudier « généralise trois techniques classiques » : la méthode du pivot de Gauß pour la triangularisation de systèmes linéaires, l'algorithme d'Euclide pour la recherche du pgcd et la méthode du simplexe pour l'optimisation de programmes linéaires.

Dans ce qui suit, \mathbb{K} désigne un corps et $\mathbb{K}[\mathbf{x}]$ l'ensemble des polynômes de la variable $\mathbf{x} = (x_1, \dots, x_n)$. On note \mathbf{x}^α le monôme $x_1^{\alpha_1} \cdots x_n^{\alpha_n}$.

Existence de bases de Gröbner

Ordres, division et reste

Définition 215. Un *ordre monomial* est une relation binaire \preceq sur \mathbb{N}^n telle que

1. \preceq est un ordre total,
2. pour tous α, β et $\gamma \in \mathbb{N}^n$, si $\alpha \preceq \beta$, alors $\alpha + \gamma \preceq \beta + \gamma$ et
3. \preceq est un bon-ordre (toute partie non-vide de \mathbb{N}^n possède un plus petit élément).

Exemple 216. L'ordre lexicographique est un ordre monomial.

Soit $f(\mathbf{x}) = \sum_{\alpha \in \mathbb{N}^n} c_\alpha \mathbf{x}^\alpha \in \mathbb{K}[x_1, \dots, x_n]$ un polynôme multivarié non nul (les coefficients $c_\alpha \in \mathbb{K}$ sont tous nuls sauf un nombre fini

d'entre eux). On appelle *terme* chaque produit $c_\alpha \mathbf{x}^\alpha$ (où $c_\alpha \neq 0$). La notion d'ordre monomial permet de définir un *multidegré* :

$$\text{mdeg}(f) = \max\{\alpha \in \mathbb{N}^n; c_\alpha \neq 0\} \in \mathbb{N}^n$$

et des notions de *coefficient dominant* $c_{\text{mdeg } f}$, *monôme dominant* $\mathbf{x}^{\text{mdeg } f}$ et *terme dominant* $\text{td}(f) = c_{\text{mdeg } f} \mathbf{x}^{\text{mdeg } f}$. Par extension, si G est une famille de polynômes, $\text{td}(G)$ désigne $\{\text{td}(g); g \in G\}$.

Exemple 217. Pour l'ordre lexicographique, le terme dominant du polynôme $f = 9x^2y^7 - 2x^3y^2 + 4x^3$ est $\text{td}(f) = -2x^3y^2$.

Exercice 218. Expliquer les lignes de code suivante :

```
MPol.<x,y,z> = PolynomialRing(QQ,3, order='lex')
x<y^2
f = 2*x^2*y+7*z^3
f.lt()
f.lc()
f.lm()
```

Quels sont les autres ordres monomiaux implémentés dans SageMath ?

Rappelons nous que l'algorithme de division euclidienne revient à chercher autant que possible à écrire un élément comme un multiple d'un autre. L'algorithme suivant permet d'adapter l'algorithme de division euclidienne au cas multivarié.

Algorithme 15 : Division multivariée avec reste

Entrées : Polynômes $f, f_1, \dots, f_s \in \mathbb{K}[\mathbf{x}]$ muni d'un ordre monomial \preceq

Sorties : Polynômes $q_1, \dots, q_s, r \in \mathbb{K}[\mathbf{x}]$ tels que $f = q_1f_1 + \dots + q_sf_s + r$ et nul monôme de r n'est divisible par $\text{td}(f_i)$ pour $i \in \llbracket 1, s \rrbracket$

```
1  $r \leftarrow 0, p \leftarrow f$ 
2 pour  $i \in \llbracket 1, s \rrbracket$  faire
3    $q_i \leftarrow 0$ 
4 tant que  $p \neq 0$  faire
5   si  $\exists i \in \llbracket 1, s \rrbracket$  tq  $\text{td}(f_i) \mid \text{td}(p)$  alors
6      $q_i \leftarrow q_i + \text{td}(p) / \text{td}(f_i)$ 
7      $p \leftarrow p - (\text{td}(p) / \text{td}(f_i)) \cdot f_i$ 
8   sinon
9      $r \leftarrow r + \text{td}(p)$ 
10     $p \leftarrow p - \text{td}(p)$ 
11 retourner  $(q_1, \dots, q_s, r)$ 
```

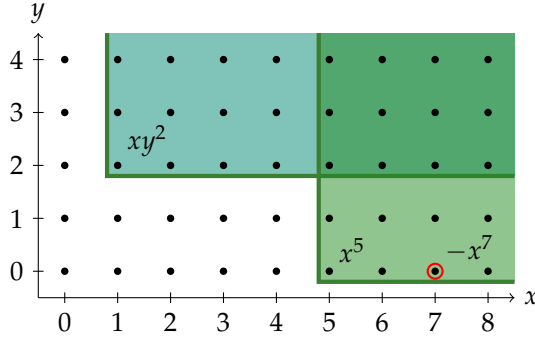
Définition 219. On appelle *quotients* et *reste* les polynômes (q_1, \dots, q_s) et r issus de l'algorithme 15 lorsque les indices i choisis sont toujours les plus petits possibles. On note $r := f \operatorname{rem}(f_i)_{i \leq s}$.

Exemple 220. On se place dans l'anneau $\mathbb{Q}[x, y]$ muni de l'ordre lexicographique avec les polynômes

$$f = -x^7 + x^6y + 2x^5 - 2x^4y - 5x^2 + 3xy^3 + 5xy + 11y^3 + 10$$

$$f_1 = xy^2 + 2y^2 \quad \text{et} \quad f_2 = x^5 + 5.$$

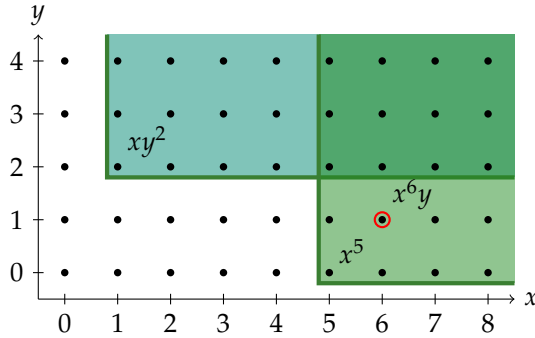
Étape 1 : Le terme dominant de f est $-x^7$. Il n'est pas divisible par le terme dominant de f_1 mais il est divisible par celui de f_2 .



On peut donc écrire

$$f = (-x^2) \cdot f_2 + \underbrace{x^6y + 2x^5 - 2x^4y + 3xy^3 + 5xy + 11y^3 + 10}_{p_1}.$$

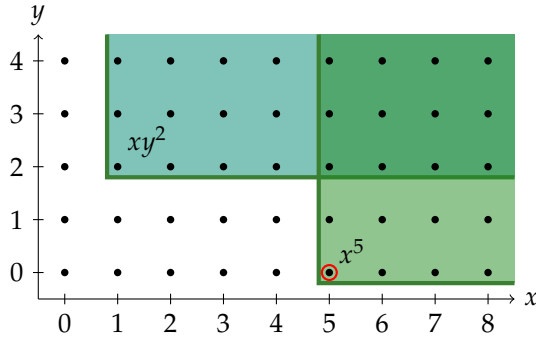
Étape 2 : Le terme dominant de p_1 est x^6y . Il n'est pas divisible par le terme dominant de f_1 mais il est divisible par celui de f_2 .



On peut donc écrire

$$f = (-x^2 + xy) \cdot f_2 + \underbrace{2x^5 - 2x^4y + 3xy^3 + 11y^3 + 10}_{p_2}.$$

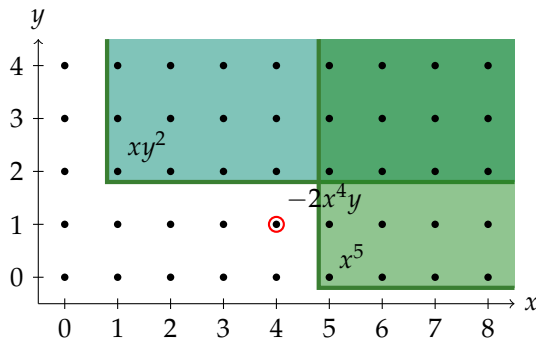
Étape 3 : Le terme dominant de p_2 est $2x^5$. Il n'est pas divisible par le terme dominant de f_1 mais il est divisible par celui de f_2 .



On peut donc écrire

$$f = (-x^2 + xy + 2) \cdot f_2 \quad \underbrace{-2x^4y + 3xy^3 + 11y^3}_{p_3}.$$

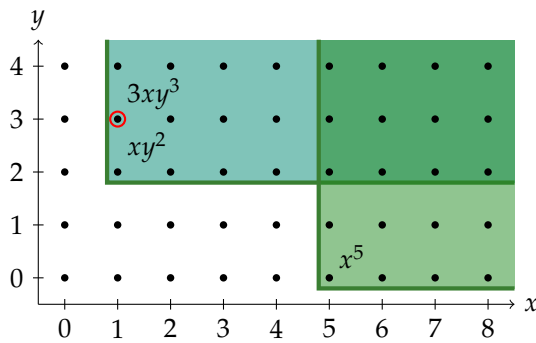
Étape 4 : Le terme dominant de p_3 est $-2x^4y$. Il n'est ni divisible par le terme dominant de f_1 ni par le terme dominant de f_2 .



On continue avec

$$f = (-x^2 + xy + 2) \cdot f_2 \quad + \underbrace{3xy^3 + 11y^3}_{p_4} \quad \underbrace{-2x^4y}_{r_4}.$$

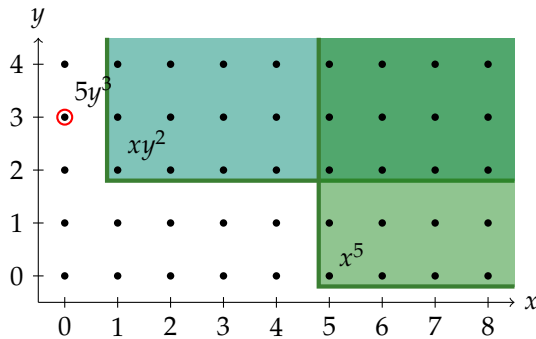
Étape 5 : Le terme dominant de p_4 est $3xy^3$. Il est divisible par le terme dominant de f_1 .



On continue avec

$$f = (-3y) \cdot f_1 + (-x^2 + xy + 2) \cdot f_2 + \underbrace{5y^3}_{p_5} - \underbrace{2x^4y}_{r_5}.$$

Étape 6 : Le terme dominant de p_3 est $5y^3$. Il n'est ni divisible par le terme dominant de f_1 ni par le terme dominant de f_2 .



L'algorithme de division s'arrête.

En conclusion, on a obtenu la décomposition

$$f = q_1 f_1 + q_2 f_2 + r$$

avec comme quotients et comme reste

$$q_1 = -3y, \quad q_2 = -x^2 + xy + 2 \quad \text{et} \quad r = -2x^4y + 5y^3.$$

Plus généralement, on note dans cet exemple que la division d'un polynôme par f_1 et par f_2 fournit un reste dont les monômes sont $x^\alpha y^\beta$ avec soit $\alpha = 0$ et β quelconque, soit $\alpha \leq 4$ et $\beta \leq 1$.

Exercice 221. Écrire une fonction `myDivision` qui implémente l'algorithme 15.

Remarque 222. La division euclidienne dans $\mathbb{K}[x]$ permet de montrer qu'un polynôme (le dividende) appartient à un idéal (celui engendré par le diviseur) en vérifiant que le reste est nul. Cette propriété ne demeure pas vraie en général. Nous allons développer un type de famille génératrice d'un idéal (les bases de Gröbner) pour lequel cette propriété est vraie.

Exemple 223. On se place dans l'anneau $\mathbb{Q}[x, y]$ muni de l'ordre lexicographique avec les polynômes

$$f = 5y^3$$

$$f_1 = xy^2 + 2y^2 \quad \text{et} \quad f_2 = x^5 + 5.$$

Le reste « $f \bmod (f_1, f_2)$ » de f dans sa division par f_1 et f_2 donne encore f . Cependant, on peut remarquer qu'on pourrait avoir un reste nul, autrement dit

$$5y^3 = q_1 f_1 + q_2 f_2$$

en choisissant

$$q_1 = \frac{5x^4y - 10x^3y + 20x^2y - 40xy + 80y}{27}$$

et

$$q_2 = -\frac{5y^3}{27}.$$

Idéaux monomiaux et engendrement fini par des monômes

Définition 224. Un idéal \mathfrak{J} de $\mathbb{K}[\mathbf{x}]$ est dit *monomial* s'il est généré par une famille de monôme, i.e. il existe un ensemble d'exposants $A \subseteq \mathbb{N}^n$ tel que

$$\mathfrak{J} = \langle \mathbf{x}^A \rangle = \langle \{\mathbf{x}^\alpha; \alpha \in A\} \rangle.$$

Lemme 225. Soit $\mathfrak{J} = \langle \mathbf{x}^A \rangle \subseteq \mathbb{K}[\mathbf{x}]$ un idéal monomial et $\beta \in \mathbb{N}^n$ un exposant. Alors, le monôme \mathbf{x}^β appartient à l'idéal \mathfrak{J} si et seulement s'il existe un exposant $\alpha \in A$ tel que \mathbf{x}^α divise \mathbf{x}^β .

Démonstration. Si le monôme \mathbf{x}^β appartient à l'idéal \mathfrak{J} , alors il existe une relation $\mathbf{x}^\beta = \sum_i q_i \mathbf{x}^{\alpha_i}$ avec des polynômes $q_i \in \mathbb{K}[\mathbf{x}]$ et des exposants $\alpha_i \in A$. L'un des termes $q_{i_0} \mathbf{x}^{\alpha_{i_0}}$ contient le terme \mathbf{x}^β et alors $\mathbf{x}^{\alpha_{i_0}}$ divise \mathbf{x}^β . \square

Lemme 226. Soit $\mathfrak{J} \subseteq \mathbb{K}[\mathbf{x}]$ un idéal monomial et $f \in \mathbb{K}[\mathbf{x}]$ un polynôme. Les affirmations suivantes sont équivalentes :

1. le polynôme f appartient à l'idéal \mathfrak{J} ;
2. chaque terme du polynôme f appartient à l'idéal \mathfrak{J} ;
3. le polynôme f est une combinaison linéaire sur \mathbb{K} de monômes de l'idéal \mathfrak{J} .

Démonstration. Seul 1. \Rightarrow 2. nécessite une preuve. Mais si $f = \sum_i q_i \mathbf{x}^{\alpha_i}$, alors chaque terme de f apparait dans l'un des $q_i \mathbf{x}^{\alpha_i}$ et est donc divisible par \mathbf{x}^{α_i} . \square

Remarque 227. On peut visualiser un idéal monomial \mathfrak{J} par un diagramme en escalier dans \mathbb{N}^n : on se contente de noter quels exposants de monômes sont possibles. Si $\mathbf{x}^\alpha \in \mathfrak{J}$, alors tout le quadrant supérieur $\alpha + \mathbb{N}^n$ appartient aussi au diagramme.

Exemple 228. Soit $A = \{(\alpha, \beta) \in \mathbb{N}^2; -7\alpha^2 + 19\alpha\beta - 7\beta^2 + 13 = 0\}$ et $\mathcal{I} = \langle \mathbf{x}^A \rangle$. Quelques uns des éléments de A sont représentés sur la figure 9. On a aussi porté pour chacun d'entre eux le quart de plan qu'ils engendrent.

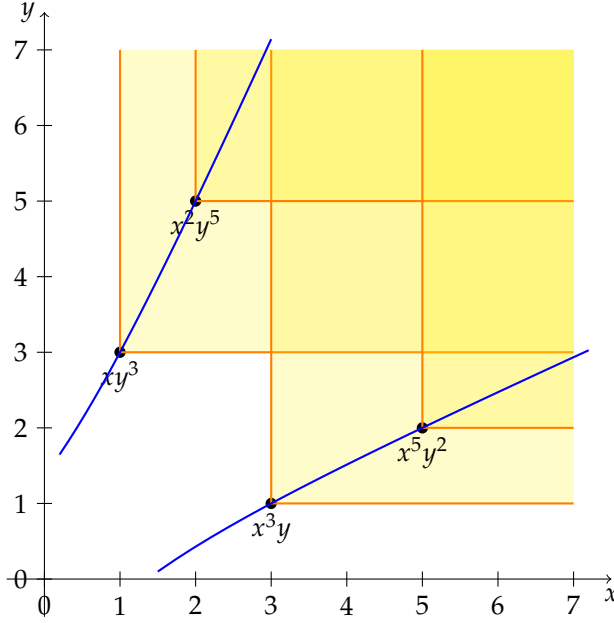


FIGURE 9: Représentation de l'idéal $\langle \mathbf{x}^A \rangle$ où $A = \{(\alpha, \beta) \in \mathbb{N}^2; -7\alpha^2 + 19\alpha\beta - 7\beta^2 + 13 = 0\}$.

Corollaire 229. Deux idéaux monômiaux sont égaux si et seulement s'ils possèdent les mêmes monômes.

Théorème 230 (Lemme de Dickson). Tout idéal monomial est engendré par une famille finie de monôme.

Intuitivement, le lemme de Dickson revient à dire que le diagramme en escalier d'un idéal monomial ne possède qu'un nombre fini de coins.

Démonstration. Soit $\mathcal{I} = \langle \mathbf{x}^A \rangle$ un idéal monomial. On introduit \leq l'ordre produit sur \mathbb{N}^n , c'est-à-dire :

$$\alpha \leq \beta \text{ ssi } \mathbf{x}^\alpha \mid \mathbf{x}^\beta \text{ ou encore ssi } \forall i \in \llbracket 1, n \rrbracket, \alpha_i \leq \beta_i.$$

et B l'ensemble des éléments minimaux de A pour \leq . Comme \leq est un ordre bien fondé (pas de suite strictement décroissante infinie), pour tout exposant α , il existe un exposant $\beta \in B$ tel que $\beta \leq \alpha$. Aussi $\langle \mathbf{x}^B \rangle = \langle \mathbf{x}^A \rangle$.

Reste à montrer que B est fini, ce que nous montrons par récurrence. Si $n = 1$, l'ordre est total et B ne contient qu'un élément. Sinon, soit $A' = \{\alpha' \in \mathbb{N}^{n-1}; \exists \alpha_n \in \mathbb{N}, (\alpha', \alpha_n) \in A\}$. Par hypothèse de récurrence, l'ensemble des éléments minimaux B' de A' est fini. Pour tout $\beta' \in B'$,

on fixe $b_{\beta'}$ tel que $(\beta', b_{\beta'}) \in B$ et notons $b = \max_{\beta' \in B'} b_{\beta'}$. Prouvons que pour tout $\beta = (\beta', \beta_n) \in B$ on a $\beta_n \leq b$. Soit $\alpha = (\alpha', \alpha_n) \in A$. Alors, il existe un certain $\beta' \in B'$ tel que $\beta' \leq \alpha'$. Si $\alpha_n > b$, on aurait

$$(\beta', b_{\beta'}) \leq (\beta', b) \leq \alpha$$

et α n'est pas minimal. Par le même raisonnement, nous montrons que chaque coordonnée des éléments de B est bornée, donc que B est fini. \square

Exercice 231. Soit $A = \{(\alpha, \beta) \in \mathbb{N}^2; -7\alpha^2 + 19\alpha\beta - 7\beta^2 + 13 = 0\}$ et $\mathcal{I} = \langle \mathbf{x}^A \rangle$, l'idéal de la figure 9.

1. Les polynômes $x^6y^3 + x^2y$ et $x^2y^4 + x^7y^3$ font-ils partie de \mathcal{I} ?
2. Donner un ensemble fini de générateurs de l'idéal de la figure 9.

Base de Gröbner

Dans ce qui suit, $\langle \text{td}(E) \rangle$ désigne l'idéal engendré par les termes dominants de polynômes de E .

Définition 232. Soit \preceq un ordre monomial et $\mathcal{I} \subseteq \mathbb{K}[\mathbf{x}]$ un idéal. Un ensemble fini de polynômes $G \subseteq \mathcal{I}$ est une *base de Gröbner* de l'idéal \mathcal{I} par rapport à l'ordre \preceq si les termes dominants de G et de \mathcal{I} engendrent le même idéal monomial, autrement dit si

$$\langle \text{td}(G) \rangle = \langle \text{td}(\mathcal{I}) \rangle.$$

Remarque 233. Le mot « base » est assez mal choisi car une base de Gröbner en reste une si on lui adjoint des éléments. Le lemme suivant montre qu'une base de Gröbner est automatiquement une famille génératrice de l'idéal.

Lemme 234. Soit \mathcal{I} un idéal de $\mathbb{K}[\mathbf{x}]$. Si $G \subseteq \mathcal{I}$ est base de Gröbner de \mathcal{I} , alors G engendre \mathcal{I} .

Démonstration. Soit $G = \{g_1, \dots, g_s\}$ une famille de polynômes qui soit une base de Gröbner d'un idéal \mathcal{I} . Considérons un polynôme quelconque f de \mathcal{I} . Notons (q_1, \dots, q_s) et r les quotients et reste de f dans sa « division » par G . Alors $r = f - \sum_{i=1}^s q_i g_i \in \mathcal{I}$. Donc $\text{td}(r) \in \text{td}(\mathcal{I}) \subseteq \langle \text{td}(g_1), \dots, \text{td}(g_s) \rangle$. Le lemme 225 montre que $\text{td}(r)$ est divisible par l'un des $\text{td}(r) \text{td}(g_i)$ tandis que l'algorithme 15 nous assure du contraire. Donc $r = 0$ et $f \in \langle G \rangle$. \square

Exemple 235. On considère, dans l'anneau $\mathbb{Q}[x, y]$ muni de l'ordre lexicographique, l'idéal \mathcal{I} engendré par les polynômes

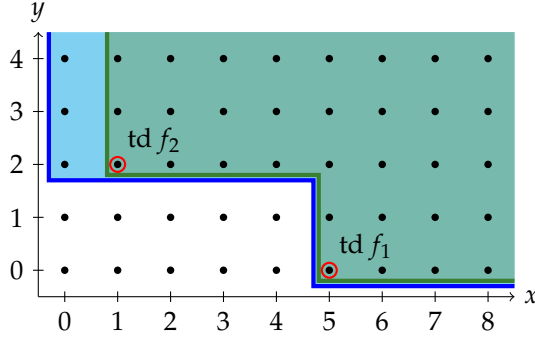
$$f_1 = xy^2 + 2y^2 \quad \text{et} \quad f_2 = x^5 + 5.$$

L'ensemble des monômes dominants des polynômes de \mathcal{I} est (d'après calcul ultérieur de l'exemple 250)

$$\left\{ x^\alpha y^\beta; \alpha \geq 5 \text{ ou } \beta \geq 2 \right\}$$

tandis que $\text{td } f_1$ et $\text{td } f_2$ n'engendrent que les monômes

$$\left\{ x^\alpha y^\beta; \alpha \geq 5 \text{ ou } (\beta \geq 2 \text{ et } \alpha \geq 1) \right\}.$$



Par conséquent la famille $\{f_1, f_2\}$ n'est pas une base de Gröbner de l'idéal \mathfrak{I} qu'elle engendre.

Théorème 236. *Tout idéal \mathfrak{I} de $\mathbb{K}[\mathbf{x}]$ possède une base de Gröbner.*

Démonstration. On applique le lemme de Dickson à l'idéal monomial $\langle \text{td}(\mathfrak{I}) \rangle$. On obtient une famille finie de $\text{td}(\mathfrak{I})$ que l'on relève en une famille de \mathfrak{I} pour obtenir une base de Gröbner. \square

Corollaire 237 (Hilbert). *Tout idéal \mathfrak{I} de $\mathbb{K}[\mathbf{x}]$ est noethérien (i.e. est engendré par un ensemble fini d'éléments).*

Test d'appartenance

Lemme 238. *Lorsqu'une famille de polynôme G est une base de Gröbner de l'idéal \mathfrak{I} , le reste d'un polynôme f dans sa division par G est l'unique polynôme r tel que $f - r \in \mathfrak{I}$ et aucun terme du polynôme r n'est divisible par un monôme de $\text{td}(G)$.*

Démonstration. Supposons qu'il existe deux polynômes r_1 et r_2 satisfaisant aux hypothèses. Alors $r_1 - r_2 \in \mathfrak{I}$. Par le lemme 225, $\text{td}(r_1 - r_2)$ serait divisible par l'un des $\text{td}(g)$ pour $g \in G$, ce qui est exclu par nos hypothèses. \square

Théorème 239. *Soit G une base de Gröbner d'un idéal $\mathfrak{I} \subseteq \mathbb{K}[\mathbf{x}]$ et $f \in \mathbb{K}[\mathbf{x}]$ un polynôme. Alors f appartient à \mathfrak{I} si et seulement si $f \text{ rem } G = 0$.*

Exemple 240. On se place dans l'anneau $\mathbb{Q}[x, y]$ muni de l'ordre lexicographique avec les polynômes

$$f = 5y^3$$

$$f_1 = xy^2 + 2y^2 \quad \text{et} \quad f_2 = x^5 + 5.$$

On note \mathfrak{J} l'idéal $\mathfrak{J} = \langle f_1, f_2 \rangle$. On a déjà vu que (voir exemple 223) que la division de f par f_1 et f_2 ne donne pas 0 alors que f est bien dans \mathfrak{J} . Ceci indique que $\{f_1, f_2\}$ n'est pas une base de Gröbner de \mathfrak{J} . Une base de Gröbner de \mathfrak{J} est $(x^5 + 5, y^2)$ (voir exemple 250). Il est facile de voir que la division de f par cette base donne bien le polynôme nul.

Exercice 241. On considère les polynômes

$$f = x^2, \quad f_1 = x^3 - 2xy \quad \text{et} \quad f_2 = x^2y - 2y^2 + x$$

1. Quel est le reste de f par (f_1, f_2) ?
2. Quel est le ppcm entre $\text{td}(f_1)$ et $\text{td}(f_2)$? Par combien faut-il multiplier f_1 et f_2 respectivement pour qu'ils aient le même terme dominant ?
3. Montrer qu'il existe des polynômes q_1 et q_2 tels que

$$f = q_1 f_1 + q_2 f_2 + r$$

avec $r = 0$

4. Conclure.

Construction de bases de Gröbner

Caractérisation des bases de Gröbner

Definition 242. Soient $g, h \in \mathbb{K}[x]$ deux polynômes non-nuls. On appelle *polynôme de syzygie* de g et h le polynôme

$$S(g, h) = \frac{\text{ppcm}(\text{td}(g), \text{td}(h))}{\text{td}(g)} g - \frac{\text{ppcm}(\text{td}(g), \text{td}(h))}{\text{td}(h)} h. \quad (10)$$

Exemple 243. On se place dans l'anneau $\mathbb{Q}[x, y]$ muni de l'ordre lexicographique avec les polynômes

$$g_1 = xy^2 + 2y^2 \quad \text{et} \quad g_2 = x^5 + 5.$$

Le terme dominant de f_1 est xy^2 , celui de f_2 est x^5 . Leur ppcm vaut x^5y^2 . Pour que ces deux termes s'annulent l'un contre l'autre, on multiplie f_1 par x^4 et f_2 par y^2 . Le polynôme de syzygie est

$$\begin{aligned} S(g_1, g_2) &= x^4 \cdot (xy^2 + 2y^2) - y^2 (x^5 + 5) \\ &= 2x^4y^2 - 5y^2. \end{aligned}$$

Le lemme suivant montre que lorsqu'une combinaison de polynômes annule le terme dominant, cela provient forcément de syzygies

Lemme 244. Soit $G = \{g_1, \dots, g_s\} \subseteq \mathbb{K}[\mathbf{x}]$ une famille de polynôme. On suppose qu'il existe un exposant $\delta \in \mathbb{N}^n$, des exposants $(\alpha_1, \dots, \alpha_s) \subseteq \mathbb{N}^n$ et des scalaires $(c_1, \dots, c_s) \subseteq \mathbb{K}^n$ tels que d'une part, pour tout i compris entre 1 et s , $\alpha_i + \text{mdeg } g_i = \delta$ et d'autre part le polynôme

$$f = \sum_{i=1}^s c_i \mathbf{x}^{\alpha_i} g_i$$

vérifie $\text{mdeg } f \prec \delta$ (annulation des termes dominants). On pose $\mathbf{x}^{\gamma_{ij}} = \text{ppcm}(\text{td}(g_i), \text{td}(g_j))$ pour $1 \leq i < j \leq s$. Alors $\mathbf{x}^{\gamma_{ij}}$ divise \mathbf{x}^δ et il existe des scalaires $c_{i,j} \in \mathbb{K}$ telles que

$$f = \sum_{1 \leq i < j \leq s} c_{i,j} \mathbf{x}^{\delta - \gamma_{ij}} S(g_i, g_j)$$

et $\text{mdeg}(\mathbf{x}^{\delta - \gamma_{ij}} S(g_i, g_j)) \prec \delta$ pour tout $1 \leq i < j \leq s$.

Démonstration. Quitte à changer la valeur du scalaire c_i , on peut supposer que le coefficient dominant du polynôme g_i est toujours 1.

Les exposants $\gamma_{i,j}$ ont été choisis de sorte que

$$S(g_i, g_j) = \frac{\mathbf{x}^{\gamma_{i,j}}}{\text{td}(g_i)} g_i - \frac{\mathbf{x}^{\gamma_{i,j}}}{\text{td}(g_j)} g_j.$$

et

$$\mathbf{x}^{\delta - \gamma_{i,j}} S(g_i, g_j) = \frac{\mathbf{x}^\delta}{\text{td}(g_i)} g_i - \frac{\mathbf{x}^\delta}{\text{td}(g_j)} g_j.$$

(sans problème de division). Mais dans cette somme, les termes de plus haut degrés s'annulent. Donc

$$\text{mdeg}(\mathbf{x}^{\delta - \gamma_{i,j}} S(g_i, g_j)) \prec \delta. \quad (11)$$

Nous prouvons le lemme par récurrence sur s . Soit $s \geq 2$. Nous posons

$$g = f - c_1 \mathbf{x}^{\delta - \gamma_{1,2}} S(g_1, g_2).$$

Nous voyons que

$$\begin{aligned} g &= c_1 \mathbf{x}^{\alpha_1} g_1 + c_2 \mathbf{x}^{\alpha_2} g_2 + \sum_{3 \leq i \leq s} c_i \mathbf{x}^{\alpha_i} g_i - c_1 \mathbf{x}^{\delta - \gamma_{1,2}} \left(\frac{\mathbf{x}^{\gamma_{1,2}}}{\text{td}(g_1)} g_1 - \frac{\mathbf{x}^{\gamma_{1,2}}}{\text{td}(g_2)} g_2 \right) \\ &= c_1 (\mathbf{x}^{\alpha_1} - \mathbf{x}^{\delta - \text{mdeg } g_1}) g_1 + (c_2 \mathbf{x}^{\alpha_2} + c_1 \mathbf{x}^{\delta - \text{mdeg } g_2}) g_2 + \sum_{3 \leq i \leq s} c_i \mathbf{x}^{\alpha_i} g_i \end{aligned}$$

Comme $\alpha_1 + \text{mdeg } g_1 = \delta = \alpha_2 + \text{mdeg } g_2$, l'expression du polynôme g se simplifie en

$$g = (c_1 + c_2) \mathbf{x}^{\alpha_2} g_2 + \sum_{3 \leq i \leq s} c_i \mathbf{x}^{\alpha_i} g_i.$$

Dans le cas où $s = 2$, on doit avoir $c_2 = -c_1$ pour que se produise l'annulation du terme de plus haut degré. Ceci termine la preuve.

Dans le cas où $s \geq 3$, on raisonne par récurrence. Compte tenu de l'équation (11) et du degré de f , on doit avoir $\text{mdeg } g \prec \delta$. On peut appliquer notre hypothèse de récurrence au polynôme g pour conclure. \square

Exemple 245. Pour illustrer ce lemme, nous nous plaçons dans $\mathbb{Q}[x, y, z]$ muni de l'ordre lexicographique. Nous nous donnons

$$\begin{cases} g_1 &= x^2 + 5x \\ g_2 &= xy + y + 1 \\ g_3 &= 2xyz + z \end{cases}$$

Nous choisissons $\mathbf{x}^{\alpha_1} = yz$, $\mathbf{x}^{\alpha_2} = xz$, $\mathbf{x}^{\alpha_3} = x$ et $c_1 = c_2 = 1$, $c_3 = -2$. Nous obtenons

$$f = 6xyz = yz \cdot g_1 + xz \cdot g_2 - x \cdot g_3.$$

Les monômes de plus haut degré, à savoir de degré $\delta = (2, 1, 1)$, s'annulent entre eux dans la somme. Il ne subsiste qu'un terme de degré $(1, 1, 1)$. Les polynômes de syzygies sont les suivants :

$$\begin{cases} S_{1,2} = S(g_1, g_2) &= yg_1 - xg_2 &= 4xy - x \\ S_{1,3} = S(g_1, g_3) &= 2yzg_1 - xg_3 &= 10xyz - xz \\ S_{2,3} = S(g_2, g_3) &= 2g_2z - g_3 &= 2yz + z \end{cases}$$

Le ppcm entre $\text{td } g_1$ et $\text{td } g_2$ vaut x^2y (soit un degré $\gamma_{1,2} = (2, 1, 0)$ dans les notations de la preuve). La preuve du lemme commence par introduire

$$\begin{aligned} g &= f - \mathbf{x}^{\delta - \gamma_{1,2}} S_{1,2} \\ &= f - zS_{1,2} \\ &= (yz \cdot g_1 + xz \cdot g_2 - x \cdot g_3) - z(y \cdot g_1 - x \cdot g_2) \\ &= 2xz \cdot g_2 - x \cdot g_3 \end{aligned}$$

Comme prévu par la preuve du lemme, il ne reste qu'une combinaison linéaire des deux derniers polynômes. Le ppcm entre $\text{td } g_2$ et $\text{td } g_3$ vaut xyz (soit un degré $\gamma_{2,3} = (1, 1, 1)$ dans les notations de la preuve). On calcule

$$\begin{aligned} g' &= g - \mathbf{x}^{\delta - \gamma_{2,3}} S_{2,3} \\ &= g - xS_{2,3} \\ &= (2xz \cdot g_2 - x \cdot g_3) - x(2zg_2 - g_3) \\ &= 0 \end{aligned}$$

En remontant les équations, on a trouvé la décomposition de f en fonction des polynômes de syzygie

$$f = zS_{1,2} + xS_{2,3}.$$

Théorème 246. Soit \mathcal{I} un idéal engendré par une famille finie de polynômes $G = \{g_1, \dots, g_s\}$, alors G est une base de Gröbner de \mathcal{I} si et seulement si

$$\forall 1 \leq i < j \leq s, \quad S(g_i, g_j) \text{ rem } G = 0.$$

Démonstration. Condition nécessaire. Soit G une base de Gröbner d'un idéal \mathfrak{J} . Alors le polynôme de syzygie $S(g_i, g_j)$ appartient encore à l'idéal \mathfrak{J} . D'après le théorème 239, son reste est nul quand on divise par G .

Condition suffisante. Soit G une famille finie de polynômes, \mathfrak{J} l'idéal qu'elle engendre et f un polynôme dans l'idéal \mathfrak{J} . Nous devons montrer que le terme dominant de f appartient à l'idéal engendré par les termes dominants de G . Introduisons des polynômes $(q_i)_{i \leq s} \in \mathbb{K}[\mathbf{x}]$ tels que

$$f = \sum_{1 \leq i \leq s} q_i g_i. \quad (12)$$

Nous appelons δ l'exposant

$$\delta = \max_i \text{mdeg}(q_i g_i).$$

Il vient $\text{mdeg } f \preceq \delta$.

Si $\text{mdeg } f = \delta$, il n'y a rien à faire car alors

$$\text{td}(f) = \sum_{i \text{ tq } \text{mdeg } q_i g_i = \delta} \text{td}(q_i) \text{td}(g_i).$$

Donc le terme dominant $\text{td}(f)$ appartient bien à l'idéal monomial $\langle \text{td}(G) \rangle$ et la famille de polynôme G est une base de Gröbner de l'idéal \mathfrak{J} .

Si $\text{mdeg } f \prec \delta$ nous allons montrer que nous pouvons trouver une nouvelle écriture identique à l'équation (12) dans laquelle se produit l'égalité $\text{mdeg } f = \delta$. En vérité, il nous suffit de trouver une nouvelle écriture identique à celle de (12) avec une valeur de δ strictement plus petite. Comme l'ordre monomial est un bon-ordre, une application un nombre fini de fois de ce processus conduit au cas $\text{mdeg } f = \delta$. Nous posons

$$f^* = \sum_{1 \leq i \leq s \text{ tq } \text{mdeg } q_i g_i = \delta} \text{td}(q_i) g_i.$$

Nous voyons que le polynôme f^* est comme dans le lemme précédent. Par conséquent, f^* est une combinaison des polynômes de syzygie :

$$f^* = \sum_{1 \leq i < j \leq s} c_{i,j} \mathbf{x}^{\delta - \gamma_{i,j}} S(g_i, g_j)$$

et $\text{mdeg}(\mathbf{x}^{\delta - \gamma_{i,j}} S(g_i, g_j)) \prec \delta$ pour tout $1 \leq i < j \leq s$. Or, d'après notre hypothèse, le reste des polynômes de syzygies dans la division par G est nul. En combinant les produits de $\mathbf{x}^{\delta - \gamma_{i,j}}$ par les quotients des $S(g_i, g_j)$ par G , on obtient une décomposition de f^* sous la forme

$$f^* = \sum_{1 \leq i \leq s} q_i^* g_i.$$

dans laquelle $\text{mdeg}(q_i^*) + \text{mdeg}(g_i) \prec \delta$. Par ailleurs, le polynôme $f - f^*$ a une représentation de la forme 12 avec une valeur strictement plus petite de δ . Donc $f = f^* + (f - f^*)$ a une représentation de la forme de l'équation (12) avec une valeur strictement plus petite de δ , comme nous voulions le démontrer. \square

L'algorithme suivant est une version grossière de l'algorithme de Buchberger (1965).

Algorithme 16 : Calcul d'une base de Gröbner

Entrées : Polynômes $f_1, \dots, f_s \in \mathbb{K}[x]$ muni d'un ordre monomial \preceq

Sorties : Base de Gröbner G de l'idéal $\mathfrak{J} = \langle f_1, \dots, f_s \rangle$.

```

1  $G \leftarrow \{f_1, \dots, f_s\}$ 
2 répéter
3    $S \leftarrow \emptyset$ 
4   pour  $g, h \in G$  faire
5      $r \leftarrow S(g, h) \bmod G$ 
6     si  $r \neq 0$  alors
7        $S \leftarrow S \cup \{r\}$ .
8    $G \leftarrow G \cup S$ .
9 jusqu'à  $S = \emptyset$ ;
10 retourner  $G$ 

```

Lemme 247. Soit G est une base de Gröbner d'un idéal \mathfrak{J} . Si pour un certain élément $g \in G$, le terme dominant $\text{td}(g)$ appartient à l'idéal monomial $\langle \text{td}(G \setminus \{g\}) \rangle$, alors la famille $G \setminus \{g\}$ est aussi une base de Gröbner de \mathfrak{J} .

Définition 248. On dit qu'une base de Gröbner G d'un idéal $\mathfrak{J} = \langle G \rangle$ est *minimale* si pour tout élément $g \in G$, le coefficient dominant de g est 1 et le terme dominant $\text{td}(g)$ n'appartient pas à l'idéal $\langle \text{td}(G \setminus \{g\}) \rangle$.

Une base minimale est dite *réduite* si pour tout élément $g \in G$, aucun monôme de g n'est dans l'idéal monomial $\langle \text{td}(G \setminus \{g\}) \rangle$.

On obtient une base minimale réduite à partir d'une base de Gröbner quelconque G en appliquant le lemme 247 itérativement : on remplace les éléments g de G par leur reste modulo $G \setminus \{g\}$ et on normalise les coefficients dominants à 1.

Théorème 249. Tout idéal de $\mathbb{K}[x]$ possède une unique base de Gröbner minimale réduite.

Exemple 250. On se place dans l'anneau $\mathbb{Q}[x, y]$ muni de l'ordre lexicographique avec les polynômes

$$f_1 = xy^2 + 2y^2 \quad \text{et} \quad f_2 = x^5 + 5.$$

On note \mathfrak{J} l'idéal $\mathfrak{J} = \langle f_1, f_2 \rangle$. Le polynôme de syzygie

$$S_{1,2} = S(f_1, f_2) = 2x^4y^2 - 5y^2$$

On remarque, de plus, que par division

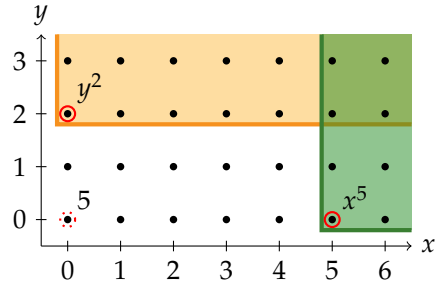
$$S_{1,2} = (2x^3 - 4x^2 + 8x - 16) \cdot f_1 + 0 \cdot f_2 + 27y^2$$

La famille $\{f_1, f_2, y^2\}$ est ainsi encore une famille génératrice de \mathfrak{J} . Nous pouvons de plus simplifier par division $f_1 = (x + 2) \cdot y^2$. Donc la famille $\{f_2, y^2\}$ est aussi une famille génératrice de \mathfrak{J} . Le polynôme de syzygie suivant vaut

$$S(f_2, y^2) = 5y^2 = 0 \text{ rem}(f_2, y^2).$$

Il n'ajoute pas de nouveau terme. La base de Gröbner (minimale réduite) de \mathfrak{J} est finalement $\{f_2, y^2\}$.

Il s'agit bien d'une base minimale réduite parce que d'une part les deux polynômes sont de coefficient dominant égal à 1 et d'autre part, x^5 et 5 ne sont pas divisibles par y^2 et y^2 n'est pas divisible par x^5 .



Exemple 251. Soit \mathfrak{J} l'idéal engendré par $x^2 - y$, $xy - z$ et $z^4 + xy$ dans $\mathbb{Q}[x, y, z]$. La base de Gröbner minimale réduite pour l'ordre lexicographique est $\mathcal{B} = \{f_1, f_2, f_3, f_4, f_5\}$ où

$$f_1 = x^2 - y \text{ avec } \mathbf{d}_1 = \text{mdeg}(f_1) = (2, 0, 0),$$

$$f_2 = xy - z \text{ avec } \mathbf{d}_2 = \text{mdeg}(f_2) = (1, 1, 0),$$

$$f_3 = xz - y^2 \text{ avec } \mathbf{d}_3 = \text{mdeg}(f_3) = (1, 0, 1),$$

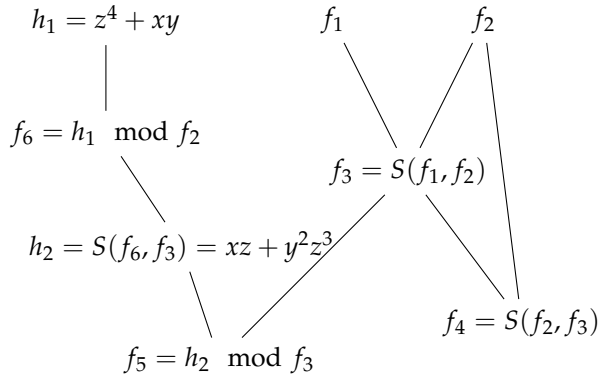
$$f_4 = y^3 - z^2 \text{ avec } \mathbf{d}_4 = \text{mdeg}(f_4) = (0, 3, 0),$$

$$f_5 = y^2z^3 + y^2 \text{ avec } \mathbf{d}_5 = \text{mdeg}(f_5) = (0, 2, 3),$$

$$f_6 = z^4 + z \text{ avec } \mathbf{d}_6 = \text{mdeg}(f_6) = (0, 0, 4).$$

(Voir fig. 10 pour une représentation : noter que l'ensemble des degrés correspond aux « pointes » du domaine)

Dans le détail, voici comment ont été dérivés les termes



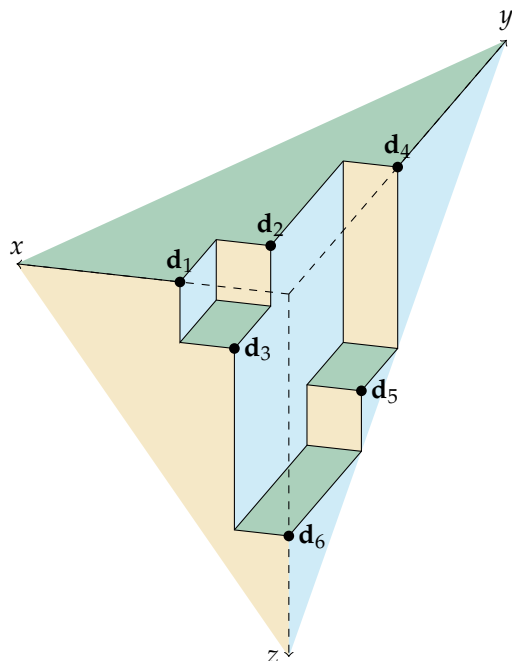


FIGURE 10: Représentation l'idéal des termes dominants de l'exemple ci-contre.

Les autres syzygies possibles sont nulles après division et n'ont pas été représentées.

- Exercice 252.** 1. Implémenter un algorithme naïf `myGroebner` de calcul de base de Gröbner.
2. Implémenter une fonction `myRedGroebner` qui transforme une base de Gröbner quelconque en une base minimale réduite.

Exercice 253. On donne l'idéal

$$\mathcal{I} = \langle f_1, f_2 \rangle = \langle xz - y^2, x^3 - z^2 \rangle$$

A-t-on que

$$f = -4x^2y^2z^2 + y^6 + 3z^5 \in \mathcal{I}?$$

On répondra en utilisant `myDivision` et le théorème 239. On confirmera sa réponse en testant avec les instructions `f in Ideal([f1, f2])`.

Premières applications

Très vite, les bases de Groebner ont trouvé un usage dans divers domaines applicatifs. En robotique par exemple, la position des différentes pièces d'un robot (cf. figure 11 pour un cas particulier) obéit à des systèmes polynômiaux (typiquement, des intersections de sphères). En électronique, la conception de puces de micro-processeurs repose

sur la simulation de courants dans différents circuits électriques qui obéissent à des systèmes d'équations différentielles que l'on peut traiter symboliquement comme des systèmes polynômiaux.

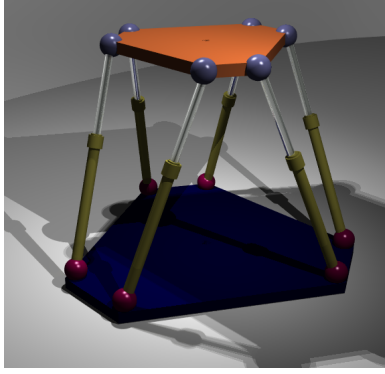


FIGURE 11: Schéma d'un hexapode (illustration Wikipédia). Pour une position donnée de la plateforme, la longueur de chaque bras possède une expression polynômiale (compliquée!).

Les bases de Gröbner jouissent de la propriété suivante :

Proposition 254. Soit t un entier compris entre 1 et $n - 1$. Si G est une base de Gröbner pour l'ordre lexicographique d'un idéal $\mathfrak{J} \subseteq \mathbb{K}[\mathbf{x}]$, alors les éléments de $G \cap \mathbb{K}[x_{t+1}, \dots, x_n]$ forment une base de Gröbner de l'idéal $\mathfrak{J} \cap \mathbb{K}[x_{t+1}, \dots, x_n]$ (appelé idéal d'élimination des t premières variables).

Résolution de système d'équations polynômiales

L'exemple d'utilisation de bases de Gröbner le plus naturel est l'étude de systèmes d'équations polynômiales, en particulier quand le système ne possède qu'un ensemble fini de solutions que l'on souhaite déterminer.

La proposition 254 montre qu'avec l'ordre lexicographique, on peut « triangulariser » le système.

Dans ce qui suit, $\mathcal{V}(\mathfrak{J})$ désigne l'ensemble des zéros des polynômes de \mathfrak{J} . On parle aussi de *variété algébrique*. Remarquez que parler des zéros d'un système de polynôme ou des zéros des polynômes de l'idéal engendré par ce système revient au même.

Exemple 255. On veut trouver les zéros $\mathcal{V}(\langle f_1, f_2, f_3 \rangle)$ du système d'équations polynômiales suivant dans \mathbb{F}_7 .

$$\begin{cases} f_1 &= x^3 + 2x^2y - x^2z + 4x^2 - xy^2 + xyz + 3xy + 5xz^2 + 2xz \\ &\quad + 5x - y^3 + 5y^2z + y^2 + yz^2 - yz + y + z^3 + 2z^2 + 3z + 1 \\ f_2 &= x^2 + 3xy + 2xz + 2x + 4y^2 + 3yz + 3y + z^2 + 2z + 6 \\ f_3 &= xz + 5x + 5yz + 4y + 5z^2 + 4z \end{cases}$$

On munit $\mathbb{F}_7[x, y, z]$ de l'ordre lexicographique. Une base de Gröbner de l'idéal engendré par f_1, f_2 et f_3 , calculée par l'instruction `Ideal(f1, f2, f3).groebner_basis()`,

renvoie le système d'équations

$$\begin{cases} g_1 &= x - y^5 - y^4 + 2y^3 + 3y^2 - y + 2z + 6 &= 0 \\ g_2 &= y^6 + 3y^4 - y^3 + y^2 + 5y + 5 &= 0 \\ g_3 &= y^3z + 5y^3 + y^2z^2 + 3y^2z + 4y^2 + 2yz + 3y + 5z^2 + 4z &= 0 \\ g_4 &= z^3 + z^2 + 4z + 1 &= 0 \end{cases}$$

Le polynôme g_4 ne dépend que de la variable z et se factorise (par les méthodes des précédents chapitres) dans $\mathbb{F}_7[z]$ en

$$g_4 = (z - 1)(z - 2)(z - 3).$$

On étudie 3 cas :

— Si z vaut 1, les polynômes

$$\begin{cases} g_2(x, y, 1) &= y^6 + 3y^4 - y^3 + y^2 - 2y - 2 \\ g_3(x, y, 1) &= -y^3 + y^2 + 5y + 2 \end{cases}$$

ne dépendent que de la variable y . Seul $y = 1$ est une racine commune. On reporte dans g_1 .

$$g_1(x, 1, 1) = x + 3.$$

On a trouvé un premier zéros qui est $(4, 1, 1)$.

— Si z vaut 2, les polynômes

$$\begin{cases} g_2(x, y, 2) &= y^6 + 3y^4 - y^3 + y^2 - 2y - 2 \\ g_3(x, y, 2) &= 0 \end{cases}$$

ne dépendent que de la variable y . Les racines dans \mathbb{F}_7 sont $y = -2$ et $y = 1$. On reporte dans g_1 .

$$g_1(x, -2, 2) = x + 3 \quad \text{et} \quad g_1(x, 1, 2) = x - 2$$

On a trouvé deux zéros qui sont $(4, -2, 2)$ et $(2, 1, 2)$.

— Si z vaut 3, les polynômes

$$\begin{cases} g_2(x, y, 3) &= y^6 + 3y^4 - y^3 + y^2 - 2y - 2 \\ g_3(x, y, 3) &= -y^3 + y^2 + 2y + 1 \end{cases}$$

ne dépendent que de la variable y . Seul $y = -2$ est une racine commune. On reporte dans g_1 .

$$g_1(x, -2, 3) = x - 2.$$

On a trouvé un dernier zéros qui est $(2, -2, 3)$.

En conclusion, le système possède 4 racines dans \mathbb{F}_7 .

Exercice 256. On considère le système de polynômes

$$\begin{cases} f(x, y) = (y^2 + 6)(x - 1) - y(x^2 + 1) = 0 \\ g(x, y) = (x^2 + 6)(y - 1) - x(y^2 + 1) = 0 \end{cases}$$

de $\mathbb{Q}[x, y]$. On choisit d'utiliser l'ordre lexicographique sur les monômes.

1. Donner une base de Gröbner de $\mathcal{I} = \langle f, g \rangle$. Que remarque-t-on au sujet du second polynôme obtenu?
2. En déduire un sur-ensemble de $\{y_0 \in \mathbb{C}; (x_0, y_0) \in \mathcal{V}(\mathcal{I})\}$.
3. En procédant par substitution, calculer l'ensemble des solutions du système sur \mathbb{C} .
4. Contrôler l'ensemble des solutions réelles en traçant les courbes $f = 0$ et $g = 0$ dans le plan. (Voir figure 12)

On pourra utiliser la commande `Ideal([f,g]).groebner_basis()`.

Utiliser `univariate_polynomial`, `roots` et `subs`.

Utiliser `implicit_plot`

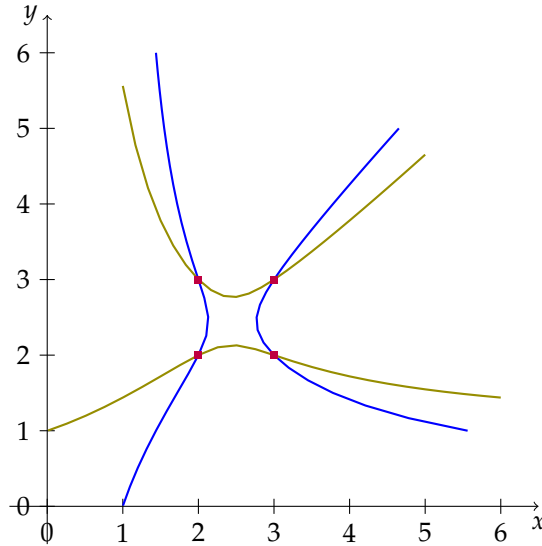


FIGURE 12: Représentation de $f = 0$ (bleu) et $g = 0$ (vert olive) et des points d'intersection (rouge).

Exercice 257. On cherche à déterminer l'ensemble des maxima de la fonction polynômiale $g = x^2y - 2xy + y + 1$ sur le cercle unité, d'équation $f = x^2 + y^2 - 1 = 0$. On sait que les minima ou maxima locaux satisfont aux conditions de Lagrange (ou de Karush-Kuhn-Tucker) : il existe $\lambda \in \mathbb{R}$ tel que $\nabla g = \lambda \cdot \nabla f$.

1. Quel système d'équation doit-on résoudre? On note \mathcal{I} l'idéal qu'il engendre.
2. Calculer une base de Gröbner pour l'ordre lexicographique et l'ordre opposé.
3. Résoudre le système.

4. Tracer le graphe de g sur le cercle (on pourra paramétrer le cercle par des fonctions trigonométriques bien connues afin de produire le graphique).

Pour représenter $f(t)$ en fonction de t entre t_{\min} et t_{\max} , utiliser `var('t')` pour créer une variable symbolique puis `plot(f(t), (t, tmin, tmax))`

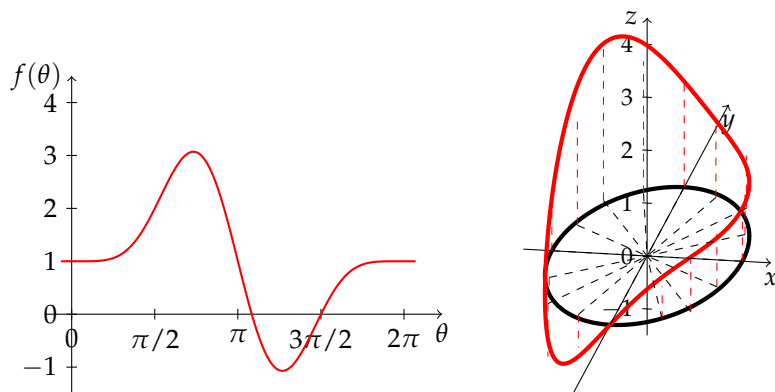


FIGURE 13: Représentation de $g = x^2y - 2xy + y + 1$ sur le cercle unité paramétré par $\theta \mapsto e^{i\theta}$.

Exercice 258. Déterminer les extrema de $f = x^3 + 2xyz - z^2$ sur la sphère unité de \mathbb{R}^3 .

Application 259. La *cryptographie à clé publique multi-variée* est l'une des candidates sérieuses dans la compétition actuelle pour standardiser la cryptographie post-quantique (i.e. la cryptographie capable de résister à l'apparition d'ordinateurs quantiques). Elle est basée sur la NP-difficulté de trouver les zéros d'un système général d'équations polynômiales multivariées. Plusieurs variantes sont en compétition actuellement, qui portent toutes sur des protocoles de signature (voir application 516) plutôt que des schémas de chiffrements. Ces variantes reposent sur le format suivant.

Génération de clés Alice choisit une famille de polynômes une famille $(p^{(k)}(\mathbf{x}))_{1 \leq k \leq m}$ de polynômes multivariés quadratiques.

$$p^{(k)}(x_1, \dots, x_n) = \sum_{1 \leq i < j \leq n} p_{i,j}^{(k)} x_i x_j + \sum_{1 \leq i \leq n} p_i^{(k)} x_i + p_0^{(k)} \in \mathbb{K}_q[x].$$

et deux applications affines $S : \mathbb{K}^m \rightarrow \mathbb{K}^m$ et $T : \mathbb{K}^n \rightarrow \mathbb{K}^n$.

Clé privée (S^{-1}, F^{-1}, T^{-1}) . Les polynômes $(p^{(k)}(\mathbf{x}))_{1 \leq k \leq m}$ sont choisis de sorte que l'application

$$F : \begin{cases} \mathbb{K}^n & \rightarrow & \mathbb{K}^m \\ \mathbf{x} & \mapsto & (p^{(k)}(\mathbf{x}))_{k \in \llbracket 1, m \rrbracket} \end{cases}$$

possède une structure qui la rende facilement inversible.

Clé publique $P = S \circ F \circ T$.

Signature Pour signer le message (ou son haché) $w \in \mathbb{K}^m$, Alice calcule $x = S^{-1}(w)$, $y = F^{-1}(x)$ et $z = T^{-1}(y)$. Elle publie z avec son message.

Déchiffrement Pour confirmer l'authenticité de la signature z , Bob calcule $P(z)$ et compare le résultat avec le haché du message.

La cryptographie multivariée est menacée par deux grandes familles de dangers : les attaques directes et les attaques structurelles. Les attaques directes consistent à chercher à résoudre les équations directement en utilisant des bases de Groebner. Les attaques structurelles tirent partie de la structure de l'application F quand elle est insuffisamment dissimulée dans la fonction P .

Exemple 260. Le cryptosystème multivarié *huile et vinaigre*, aujourd'hui cassé, a inspiré diverses propositions crédibles et actuellement en lice dans la compétition de standardisation de la cryptographie post-quantique. Il consiste à séparer les variables en deux catégories : les variables vinaigre $(x_i)_{i \in V}$ et les variables huile $(x_i)_{i \in H}$ puis ne considérer que des polynômes $p^{(k)}$ tels que $p_{i,j}^{(k)} = 0$ si $(i, j) \in H^2$. La trappe est la suivante : lorsqu'on donne des valeurs quelconques aux variables vinaigre $(x_i)_{i \in V}$, résoudre $F(\mathbf{x}) = 0$ devient un problème d'algèbre linéaire sur l'ensemble des variables huile.

Prenons pour illustrer $n = 4$, $V = \{1, 2\}$, $H = \{3, 4\}$, les polynômes

$$\begin{aligned}
 p^{(1)}(x_1, \dots, x_4) &= 4x_1^2 + x_1x_2 - 2x_1x_3 + 7x_1x_4 - x_1 - 5x_2^2 \\
 &\quad - 2x_2x_3 - 4x_2x_4 - 5x_2 - 3x_3 + 3x_4 + 3 \\
 p^{(2)}(x_1, \dots, x_4) &= -5x_1^2 - 8x_1x_2 - x_1x_3 - 4x_1x_4 + 8x_1 \\
 &\quad - 6x_2^2 - 4x_2x_3 - 2x_2x_4 - 2x_2 + 5x_4 + 9 \\
 S(y_1, y_2) &= \begin{pmatrix} 8y_1 & +6y_2 & -8 \\ 3y_1 & +9y_2 & +1 \end{pmatrix} \\
 T(\mathbf{x}) &= \begin{pmatrix} & 7x_2 & +15x_3 & +10x_4 & +3 \\ 16x_1 & +17x_2 & +3x_3 & +3x_4 & \\ 8x_1 & +15x_2 & +15x_3 & +5x_4 & +2 \\ 15x_1 & +10x_2 & +10x_3 & +8x_4 & +13 \end{pmatrix} \\
 F(x_1, \dots, x_4) &= S \circ F \circ T \\
 F^{(1)}(x_1, \dots, x_4) &= 14x_1^2 + 11x_1x_2 + 4x_1x_3 + 11x_1x_4 + 12x_1 \\
 &\quad + 8x_2^2 + 14x_2x_3 + x_2x_4 + 16x_2 + 16x_3^2 \\
 &\quad + 6x_3x_4 + 10x_3 + x_4^2 + 10x_4 + 2 \\
 F^{(2)}(x_1, \dots, x_4) &= 8x_1^2 + 3x_1x_2 + 3x_1x_3 + 13x_1x_4 \\
 &\quad + 2x_1 - x_2^2 + 7x_2x_3 + 2x_2x_4 + 16x_2 \\
 &\quad + 4x_3^2 + 17x_3 + 10x_4 + 10
 \end{aligned}$$

Supposons que Alice veuille signer le message de haché $m = (4, 7)$. Elle commence par calculer

$$S^{-1}(m) = (-5, -4).$$

Pour trouver une solution au système

$$F(x_1, x_2, x_3, x_4) = (-5, -4).$$

Elle choisit au hasard $x_1 =$ et $x_2 = 3$ qui fournit un système linéaire inversible :

$$\begin{cases} 8x_3 + 17x_4 + 6 &= -5 \\ 6x_3 + 14x_4 + 4 &= -4 \end{cases}$$

Elle en déduit un point $(1, 3, ,)$. On en déduit finalement

Manipulations algébriques

Exercice 261. On donne la courbe \mathcal{C} paramétrée par la variable t définie par

$$\begin{cases} x(t) = t^2 \\ y(t) = t^3 \\ z(t) = t^4 \end{cases}$$

On souhaite obtenir une équation implicite de \mathcal{C} . On se place dans $\mathbb{Q}[x, y, z, t]$. Donner une base de Gröbner de $\langle x - t^2, y - t^3, z - t^4 \rangle$ pour

un ordre bien choisi (donner des raisons heuristiques de ce choix). Utiliser les derniers polynômes de la base obtenue comme candidats à une équation implicite de \mathcal{C} . Vérifier à la main.

Exercice 262 (Ovale de Descarte). Dans le plan réel, on fixe les points O de coordonnées $(0,0)$ et Ω de coordonnées $(1,0)$. Donner une équation cartésienne et représenter avec `implicit_plot` le lieu des points M tels que $OM + 2\Omega M = 3$. [Indication : on introduira des variables représentant OM et ΩM et on les éliminera.]

Exercice 263. Écrire $\sin(\theta)^6$ comme un polynôme en $u(\theta) = \sin(\theta) + \cos(\theta)$ et en $v(\theta) = \sin(2\theta) + \cos(2\theta)$.

Utiliser la méthode `reduce`

Exercice 264. On peut utiliser les base de Gröbner pour calculer un ensemble de générateurs d'une intersection d'idéaux.

1. Soient I, J deux idéaux de $\mathbb{K}[\mathbf{x}]$. On note t une indéterminée supplémentaire. Montrer que

$$I \cap J = (tI + (1-t)J) \cap \mathbb{K}[\mathbf{x}].$$

2. On donne

$$I = \langle x, z \rangle \quad \text{et} \quad J = \langle y^2, x - yz \rangle$$

Calculer un ensemble de générateurs de $I \cap J$ (utiliser la proposition 254).

TP 7 : Applications choisies des bases de Gröbner

Buts : Pratiquer les calculs avec les bases de Gröbner. Apprendre ou réviser des notions de bases en géométrie algébrique (en vue du cours ACCQ205) et renforcer votre culture sur le sujet. Découvrir quelques applications amusantes d'approches algébriques en théorie des graphes

Travaux préparatoires : Cours et exercice

Évaluation du TP : Exercices 267 (points singuliers), 277 (valuation), 287 (enveloppe), 295 (coloration de graphe), 288 (preuve de théorèmes géométriques), 299 (programmation entière), 282 (surface de Clebsch).

Un peu de géométrie algébrique traditionnelle

Il est courant d'utiliser un polynôme ou une famille de polynômes pour définir des objets géométriques tels que des courbes, des surfaces, etc. À cause de nombreux cas dégénérés, les liens entre polynômes et objets géométriques correspondants sont compliqués à exposer. Ils seront abordés dans le cours ACCQ205.

Points réguliers, courbes lisses

Si $f \in \mathbb{K}[\mathbf{x}]$ est un polynôme à n variables, l'ensemble des zéros de f forme en général une *hypersurface* de \mathbb{K}^n et l'hyperplan tangent à cette courbe au point $M = (m_1, \dots, m_n) \in \mathbb{K}^n$ a pour équation

$$[(\nabla f)(M)] \cdot [x_1 - m_1, \dots, x_n - m_n] = \sum_{i=1}^n \frac{\partial f}{\partial x_i}(M) \cdot (x_i - m_i) = 0$$

L'intersection de plusieurs hypersurfaces s'appelle une *variété*, cette notion généralise l'idée de courbe (dimension 1) ou de surface (dimension 2).

Définition 265. Soit \mathcal{V} une variété de \mathbb{K}^n définie par l'intersection de t hypersurfaces

$$\begin{cases} f_1(\mathbf{x}) &= 0 \\ &\vdots \\ f_t(\mathbf{x}) &= 0 \end{cases}$$

On dit qu'un point M de \mathcal{V} est *régulier* si la famille $(\nabla f_i(M))_{i \leq t}$ est de rang $n - d$, où d est la dimension de \mathcal{V} au voisinage de M .

On parle de point *singulier* sinon.

Lorsque tous les points sont réguliers, on parle de *variété lisse*.

En langage courant, dire qu'un point est régulier signifie que l'espace tangent est bien défini. Pour une courbe plane \mathcal{C} définie par $f(x, y) = 0$, être lisse signifie que le système $\left\{ \frac{\partial f}{\partial x} = 0, \frac{\partial f}{\partial y} = 0 \right\}$ n'admet de pas de solution sur \mathcal{C} . Pour une courbe de l'espace \mathbb{K}^3 définie par $f_1(x, y, z) = f_2(x, y, z) = 0$, être lisse signifie que le produit vectoriel $(\nabla f_1) \wedge (\nabla f_2)$ ne s'annule jamais sur \mathcal{C} .

Exemple 266. On cherche les points singuliers de la strophoïde (fig. 14) d'équation

$$f_1(x, y) = -925x^3 + 3720x^2y - 4372x^2 - 5400xy^2 + 15828xy \\ - 10680x + 4320y^3 - 15768y^2 + 16560y - 4800.$$

On calcule les dérivées partielles

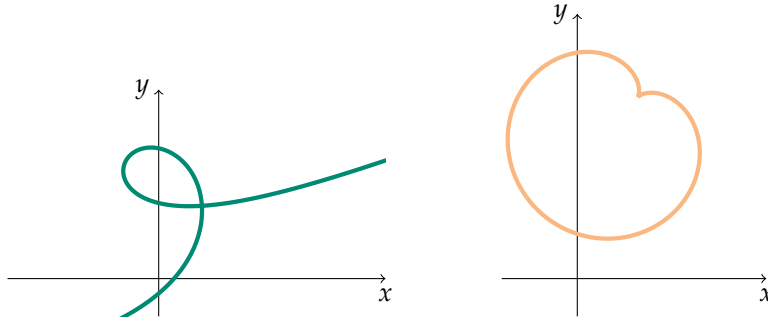


FIGURE 14: Représentation d'une strophoïde et d'une cardioïde

$$\frac{\partial f_1}{\partial x} = -2775x^2 + 7440xy - 8744x - 5400y^2 + 15828y - 10680$$

$$\frac{\partial f_1}{\partial y} = 3720x^2 - 10800xy + 15828x + 12960y^2 - 31536y + 16560$$

Pour déterminer les zéros du système $\left\{ f, \frac{\partial f_1}{\partial x}, \frac{\partial f_1}{\partial y} \right\}$, on exploite la proposition 254 (« triangularisation du système »). On introduit l'ordre lexicographique sur $\mathbb{Q}[x, y]$ et on calcule une base de Gröbner. On obtient le nouveau système

$$\begin{cases} x - 4/7 &= 0 \\ y - 26/21 &= 0 \end{cases}$$

qui fournit le point $\left(\frac{4}{7}, \frac{26}{21} \right)$. La courbe ne possède qu'un seul point singulier (qui est ici un point double).

Exercice 267. Point de rebroussement d'une cardoïde (fig. 14). Représenter avec `implicit_plot` la courbe de \mathbb{R}^2 d'équation

$$f_2(x, y) = 5x^4 - 10x^3 + 10x^2y^2 - 40x^2y + 40x^2 - 10xy^2 \\ + 40xy - 32x + 5y^4 - 40y^3 + 115y^2 - 136y + 48 = 0$$

et rechercher ses points singuliers.

Exercice 268. Pour chacun des polynômes f suivant, représenter la courbe de \mathbb{R}^2 d'équation $f(x, y) = 0$ et rechercher ses points singuliers.

1. Point multiple d'un quadrifolium (fig. 15)

$$f_3(x, y) = 83521x^6 - 589560x^5 + 250563x^4y^2 - 854862x^4y \\ + 2405547x^4 - 1179120x^3y^2 + 3868320x^3y - 5616560x^3 \\ + 250563x^2y^4 - 1709724x^2y^3 + 6467198x^2y^2 \\ - 11568340x^2y + 9201647x^2 - 589560xy^4 + 4177440xy^3 \\ - 12743920xy^2 + 18074720xy - 10247320x + 83521y^6 - 854862y^5 \\ + 3934935y^4 - 10447460y^3 + 17085215y^2 - 16009582y + 6534889$$

2. Point d'osculation et point double (fig 15)

$$f(x, y) = 289x^4 + 484x^3 + 578x^2y^2 - 312x^2y + 76x^2 + 860xy^2 \\ - 208xy - 144x + 289y^4 + 104y^3 + 264y^2 - 32y - 48$$

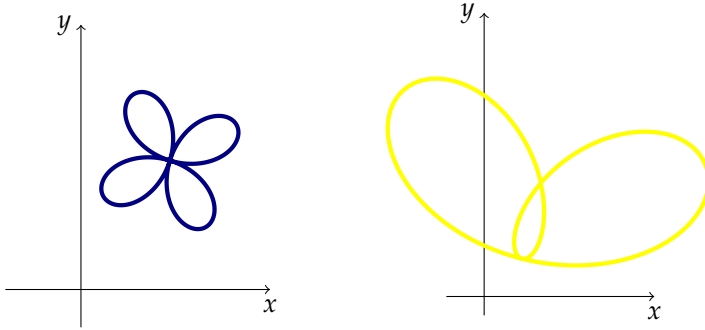


FIGURE 15: Représentation d'un quadrifolium et d'une courbe enlacée

3. Plusieurs points multiples

$$f_5(x, y) = 11x^7 - 7x^6y - 103x^6 + 8x^5y^2 + 55x^5y \\ + 399x^5 + 3x^4y^3 - 62x^4y^2 - 172x^4y - 825x^4 \\ - 10x^3y^4 + 2x^3y^3 + 178x^3y^2 + 271x^3y + 978x^3 \\ + 10x^2y^5 + 10x^2y^4 - 11x^2y^3 - 248x^2y^2 \\ - 223x^2y - 660x^2 - 30xy^5 + 40xy^4 - 26xy^3 \\ + 180xy^2 + 88xy + 232x + y^7 - 4y^6 + 26y^5 - 44y^4 \\ + 33y^3 - 56y^2 - 12y - 32$$

Valuation d'une fonction rationnelle en un point d'une courbe

Comme il est indésirable d'utiliser un polynôme plutôt qu'un autre pour définir une variété, nous introduisons les définitions suivantes.

Définition 269. On note $\mathfrak{I}(E)$ l'idéal des polynômes s'annulant sur un sous-ensemble E de \mathbb{K}^n :

$$\mathfrak{I}(E) = \{f(\mathbf{x}) \in \mathbb{K}[\mathbf{x}]; \forall M \in E \subseteq \mathbb{K}^n, f(M) = 0\}.$$

Réciproquement, on note, pour tout idéal $\mathfrak{I} \subseteq \mathbb{K}[\mathbf{x}]$,

$$\mathcal{V}(\mathfrak{I}) = \{M \in \mathbb{K}^n; \forall f \in \mathfrak{I}, f(M) = 0\}.$$

Remarque 270. Une *variété algébrique* est l'un de ces $\mathcal{V}(\mathfrak{I})$. On souhaiterait qu'idéaux et variétés algébriques se correspondent. C'est presque le cas : pour tout idéal \mathfrak{I} , on peut montrer que si un polynôme f s'annule sur $\mathcal{V}(\mathfrak{I})$, alors une certaine puissance de f appartient à \mathfrak{I} . Ce résultat s'appelle le théorème des zéros de Hilbert (ou *Nullstellenatz*).

Si $p \in \mathbb{K}[x]$ est un polynôme et $x_0 \in \mathbb{K}$ un point quelconque, on appelle *ordre d'annulation* ou *valuation* de p en x_0 , et on note $\text{ord}_{x_0}(p)$, le plus grand entier k tel que $(x - x_0)^k$ divise p . On remarquera que $\text{ord}_{x_0}(p)$ est aussi la dimension de l'espace vectoriel $\mathbb{K}[x]/\langle p, (x - x_0)^n \rangle$ où n est le degré de p . En effet,

$$\langle p, (x - x_0)^n \rangle = \langle \text{pgcd}(p, (x - x_0)^n) \rangle = \langle (x - x_0)^{\text{ord}_{x_0}(p)} \rangle.$$

Avec quelques précautions, il est possible d'étendre cette définition à des fonctions polynomiales ou rationnelles définies sur des courbes.

Définition 271. Soit \mathcal{C} une courbe algébrique lisse définie sur un corps \mathbb{K} par un certain idéal $\mathfrak{I} = \mathfrak{I}(\mathcal{C})$ engendré par les polynômes f_1, \dots, f_t . Soit $g \in \mathbb{K}[\mathbf{x}]/\mathfrak{I}$ une fonction définie sur la courbe et $M = (m_1, \dots, m_n)$ un point de \mathcal{C} . On considère $\mathbf{u} \in \mathbb{K}^n$ tel que

$$v(x) = u_1(x_1 - m_1) + \dots + u_n(x_n - m_n)$$

ne s'annule sur aucun point zéro commun à \mathfrak{I} et g en dehors de M . On note d le produit des degrés totaux $d = \deg g \cdot \deg f_1 \cdots \deg f_t$. On appelle *valuation* de g en M la dimension de l'espace vectoriel

$$(\mathbb{K}[\mathbf{x}]/\mathfrak{I})/\langle g, v(\mathbf{x})^d \rangle = \mathbb{K}[\mathbf{x}]/\langle f_1, \dots, f_t, g, v(\mathbf{x})^d \rangle$$

On appelle *valuation* de la fonction rationnelle g/h la différence

$$\text{ord}_M\left(\frac{g}{h}\right) = \text{ord}_M(g) - \text{ord}_M(h).$$

Remarque 272. On notera que si $g_1, g_2, h_1, h_2 \in \mathbb{K}[\mathbf{x}]$ vérifient

$$g_1 h_2 \equiv g_2 h_1 \pmod{\mathfrak{I}},$$

$\frac{g_1}{h_1}$ et $\frac{g_2}{h_2}$ définissent la même fonction rationnelle sur \mathcal{C} (on note par analogie $\mathbb{K}(\mathcal{C})$ leur ensemble) et que l'ordre en M ne dépend pas du représentant choisi.

D'autre part, on notera qu'un polynôme (ou une fonction régulière) g s'annule en M si et seulement si $\text{ord}_M(g) > 0$. On dit que g/h possède un pôle en M si $\text{ord}_M(\frac{g}{h}) < 0$.

Une définition plus classique consiste à fixer une fonction π dont l'ordre est 1 en M (on parle d'*uniformisante*). Alors la valuation en M d'une fonction rationnelle h est aussi le plus grand entier ℓ tel que $h(x)/\pi(x)^\ell$ soit défini et non nul en M .

Remarque 273. Le reste d'une division par une base de Gröbner G de \mathcal{I} se trouve dans la zone non-couverte par le diagramme en escalier de l'idéal engendré par les termes dominants de \mathcal{I} . Une base vectorielle de $\mathbb{K}[x]/\mathcal{I}$ est l'ensemble de ces monômes. Pour trouver la dimension de $\mathbb{K}[x]/\mathcal{I}$, il suffit de compter les points exclus du diagramme (algorithme 17).

Avec SageMath, utiliser `normal_basis()`.

Algorithme 17 : Dimension de $\mathbb{K}[x]/\mathcal{I}$

Entrées : Idéal propre \mathcal{I} tels que $\dim \mathbb{K}[x]/\mathcal{I} < +\infty$ engendré par une base de Gröbner $f_1, \dots, f_t \in \mathbb{K}[x]$

Sorties : Dimension de du \mathbb{K} -espace vectoriel
 $\dim \mathbb{K}[x]/\mathcal{I} < +\infty$

```

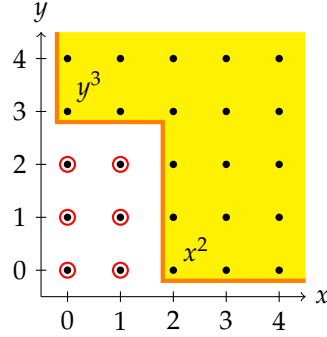
1   $A \leftarrow \{\text{mdeg}(\text{td}(f_j)); j \leq t\}$ 
2   $R \leftarrow \{(0, \dots, 0)\}$ 
3  pour  $i \in \llbracket 1, n \rrbracket$  faire
4       $S \leftarrow R$ 
5      tant que  $S \neq \emptyset$  faire
6           $t = (t_1, \dots, t_n) \leftarrow$  un élément de  $S$ 
7           $S \leftarrow S \setminus \{t\}$ 
8           $t \leftarrow (t_1, \dots, t_i + 1, \dots, t_n)$ 
9          tant que  $\forall \alpha \in A, \exists i \in \llbracket 1, n \rrbracket, t_i < \alpha_i$  faire
10              $R \leftarrow R \cup \{t\}$ 
11              $t \leftarrow (t_1, \dots, t_i + 1, \dots, t_n)$ 
12 retourner  $|R|$ 
```

Exemple 274. Cherchons les zéros de la fonction $g = x^2y - 2xy + y$ sur le cercle d'équation $f = x^2 + y^2 - 1$ (cf. exercice 257 & figure 12). La résolution du système $\{f = g = 0\}$ montre que les zéros de g sur le cercle se trouvent aux points $M_1 = (1, 0)$ et $M_{-1} = (-1, 0)$. Voici le calcul de leur multiplicité de deux manières distinctes.

1. *Calculs avec l'ordre degrevlex.* Une base de Gröbner de l'idéal $\langle f, g \rangle$ est

$$\{y^3 + 2xy - 2y, x^2 + y^2 - 1\}$$

dont les termes dominants sont y^3 et x^2 . Le diagramme en escalier de $\langle f, g \rangle$ est donc



L'algorithme 17 fournit

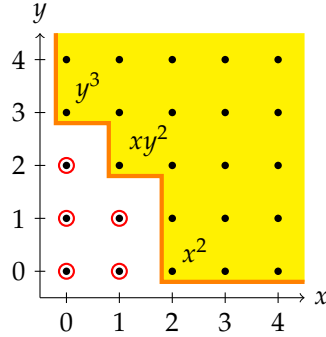
$$\{1, x, y, xy, y^2, xy^2\}$$

comme base de $\mathbb{Q}[x, y] / \langle f, g \rangle$ (R est représenté par les cercles rouges sur le diagramme). Ainsi g possède 6 zéros comptés avec multiplicité.

— Pour calculer l'ordre de g en M_1 , on peut utiliser $v_1(x, y) = x - 1$ (car $v_1(M_1) = 0$ et $v_1(M_{-1}) \neq 0$). Une base de Gröbner de $\langle f, g, v_1^6 \rangle$ est

$$\{xy^2 - 3y^2 - 4x + 4, y^3 + 2xy - 2y, x^2 + y^2 - 1\}$$

dont les termes dominants sont xy^2 , y^3 et x^2 et le diagramme en escalier de $\langle f, g, v_1^6 \rangle$ est



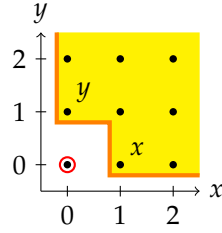
L'algorithme 17 fournit $\{1, x, y, xy, y^2\}$ comme base de $\mathbb{Q}[x, y] / \langle f, g, v_1^6 \rangle$.

Aussi, $\text{ord}_{M_1}(g) = 5$.

— Pour calculer l'ordre de g en M_{-1} , on peut utiliser $v_{-1}(x, y) = x + 1$. Une base de Gröbner de $\langle f, g, v_{-1}^6 \rangle$ est

$$\{x + 1, y\}$$

dont les termes dominants sont x , y et le diagramme en escalier de $\langle f, g, v_{-1}^6 \rangle$ est



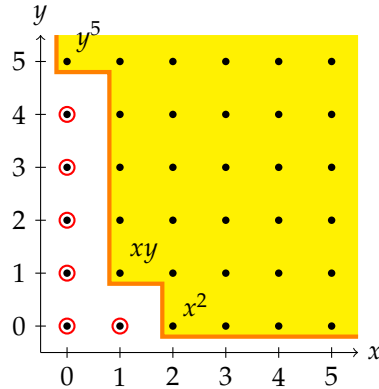
On en déduit que $\mathbb{Q}[x, y] / \langle f, g, v_1^6 \rangle$ admet pour base $\{1\}$ et que $\text{ord}_{M_{-1}}(g) = 1$.

On peut contrôler que $\text{ord}_{M_1}(g) + \text{ord}_{M_{-1}}(g) = 6$.

2. *Calculs avec l'ordre lexicographique.* Une base de Gröbner de l'idéal $\langle f, g \rangle$ est

$$\left\{ x^2 + y^2 - 1, xy + \frac{1}{2}y^3 - y, y^5 \right\}$$

dont les termes dominants sont x^2 et xy et y^5 . Le diagramme en escalier $\text{td}(\langle f, g \rangle)$ est donc



On en déduit que $\mathbb{Q}[x, y] / \langle f, g \rangle$ admet pour base alternative

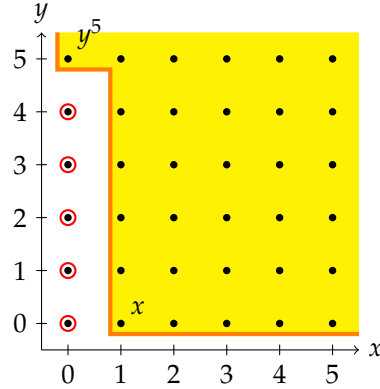
$$\left\{ 1, x, y, y^2, y^3, y^4 \right\}$$

et est bel et bien de dimension 6.

— Calcul de l'ordre de g en M_1 . On conserve $v_1(x, y) = x - 1$. Une base de Gröbner de $\langle f, g, v_1^6 \rangle$ est

$$\left\{ x + \frac{1}{8}y^4 + \frac{1}{2}y^2 - 1, y^5 \right\}$$

dont les termes dominants sont x et y^5 . Le diagramme en escalier $\text{td}(\langle f, g, v_1^6 \rangle)$ est donc



On en déduit que $\mathbb{Q}[x, y] / \langle f, g, v_1^6 \rangle$ admet pour base alternative $\{1, y, y^2, y^3, y^4\}$. Ceci confirme $\text{ord}_{M_1}(g) = 5$.

- Calcul de l'ordre de g en M_1 . On conserve $v_{-1}(x, y) = x + 1$. Une base de Gröbner de $\langle f, g, v_{-1}^6 \rangle$ est encore $\{x + 1, y\}$ qui confirme que $\text{ord}_{M_{-1}}(g) = 1$.

3. *Calculs plus traditionnels.* On joue avec l'expression de g

- Ordre en M_1 . On remarque que

$$\begin{aligned} g(x, y) &= y(x-1)^2 = y^5 \frac{(x-1)^2}{y^4} \\ &\equiv y^5 \frac{(x-1)^2}{(1-x^2)^2} = y^5 \frac{1}{(x+1)^2} \pmod{f} \end{aligned}$$

Or la droite $\{y = 0\}$ n'est pas tangente au cercle, donc $\pi(x, y) = y$ satisfait bien $\text{ord}_{M_1}(\pi) = 1$ (π est une uniformisante en M_1). Comme $\frac{1}{(x+1)^2}$ ne s'annule pas en M_1 , la factorisation ci-dessus suffit à montrer que $\text{ord}_{M_1}(g) = 5$.

- Ordre en M_{-1} . De même, $g = y(x-1)^2$ où $(x-1)^2$ ne s'annule pas en M_{-1} suffit à calculer que $\text{ord}_{M_{-1}}(g) = 1$.

Un peu d'intuition aurait pu nous prédire ces résultats. D'après le théorème de Bezout (cf. un cours de géométrie), le nombre de zéros d'une intersection de deux courbes est le produit des degrés (ici : $3 \times 2 = 6$). On voit sur la figure 12 que l'intersection au niveau du point M_{-1} se fait sans osculation : M_{-1} est donc d'ordre 1. Par différence, M_1 est d'ordre 5. On voit d'ailleurs sur la figure que les 2 courbes sont collées l'une à l'autre.

Exercice 275. Refaire l'exemple ci-dessus en utilisant l'ordre lexicographique inversé (dans lequel $x < y$).

Exemple 276. Continuons l'exemple 251. On considère

$$f_1 = x^2 - y, \quad f_2 = xy - z \quad \text{et} \quad \mathfrak{J} = \langle f_1, f_2 \rangle.$$

On souhaite étudier la fonction polynomiale $g = z^4 - xy$ sur la courbe $\mathcal{C} = \mathcal{V}(\mathcal{J})$.

La figure 16 représente le diagramme en escalier de $\text{td}(\mathcal{J}, g)$; les points rouges représentent R tel que calculé par l'algorithme 17. Le quotient $\mathbb{Q}[\mathbf{x}] / \langle f_1, f_2, g \rangle$ est de dimension 12 en tant que \mathbb{Q} -espace vectoriel et admet comme base

$$\{1, x, y, y^2, z, yz, y^2z, z^2, yz^2, y^2z^2, z^3, yz^3\}.$$

On sait donc déjà que g compte 12 zéros comptés avec leur multiplicité.

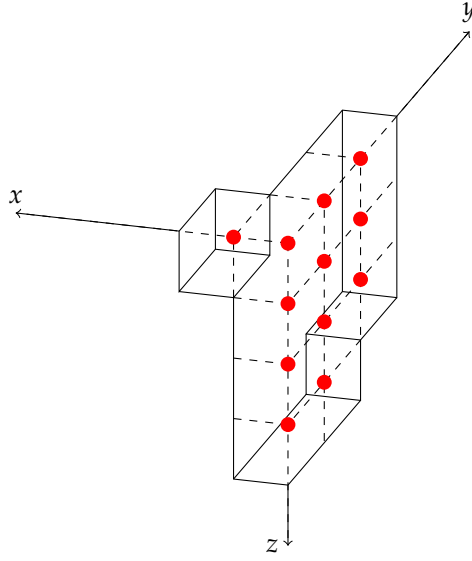


FIGURE 16: Monômes exclus des termes dominants de l'idéal engendré par $x^2 - y$, $xy - z$ et $z^4 + xy$.

Pour chercher les zéros de g , on peut noter que \mathcal{C} se paramètre via

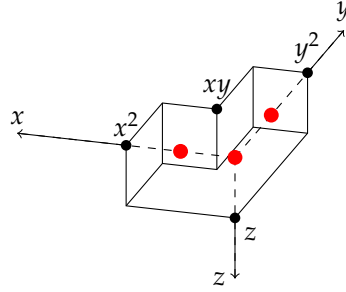
$$t \mapsto M_t = (t, t^2, t^3)$$

et qu'au point M_t , g prend la valeur

$$g(M_t) = t^{12} - t^3 = t^3 \prod_{\zeta \text{ tq. } \zeta^9=1} (t - \zeta).$$

Ainsi g possède 10 zéros : le point M_0 et les points M_ζ où ζ est l'une des 9 racines 9-ième de l'unité.

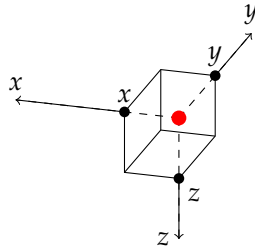
— Pour l'ordre de g en $M_0 = (0, 0, 0)$, on peut noter que $v_0(x, y, z) = x$ convient. Une base de Groebner de $\langle \mathcal{J}, g, v_0^8 \rangle$ est $\{x^2 - y, xy, y^2, z\}$. Le diagramme en escalier de $\text{td}(\langle \mathcal{J}, g, v_0^8 \rangle)$ est (avec en noir les générateurs et en rouge les monômes exclus)



si bien que $\mathbb{Q}[x]/\langle \mathcal{I}, g, v_0^8 \rangle$ admet pour base $\{1, x, y\}$. Donc

$$\text{ord}_{M_0}(g) = 3$$

— Pour l'ordre de g en $M_\zeta = (\zeta, \zeta^2, \zeta^3)$, on peut noter que $v_\zeta(x, y, z) = x - \zeta$ convient. Une base de Groebner de $\langle f_1, f_2, g, v_\zeta^8 \rangle$ est $\{x - \zeta, y - \zeta^2, z - \zeta^3\}$, avec pour diagramme en escalier



si bien que $\mathbb{Q}[x]/\langle f_1, f_2, g, v_\zeta^8 \rangle$ admet pour base $\{1\}$ et

$$\text{ord}_{M_\zeta}(g) = 1.$$

Conclusion : $\text{ord}_{M_0}(g) + \sum_{\zeta \text{ tq. } \zeta^9=1} \text{ord}_{M_\zeta}(g) = 3 + 9 \cdot 1 = 12$ comme annoncé.

Exercice 277. Soit \mathcal{Z} la courbe définie par l'intersection d'un cylindre à base circulaire et d'un cylindre à base parabolique (voir fig 17) d'équation

$$\begin{cases} x^2 + y^2 - 1 &= 0 \\ 5x - (z - 3)^2 &= 0 \end{cases}.$$

1. Que pouvez-vous dire par rapport à $\mathcal{Z}(\mathbb{R})$ de la courbe \mathcal{Z}_1 paramétrée, pour $\theta \in [0, \pi]$, par

$$\begin{cases} x(\theta) &= \sin \theta \\ y(\theta) &= \cos \theta \\ z(\theta) &= 3 + \sqrt{5 \sin \theta} \end{cases} ?$$

Représenter avec SageMath la courbe $\mathcal{Z}(\mathbb{R})$ tout entier.

2. Vérifier que \mathcal{Z} est lisse (en résolvant un certain système de d'équations). Contrôler avec les instructions

Voir `parametric_plot3d`.

Utiliser `show(Z1+Z2)` pour concatener deux courbes).

Essayer `vector(f1.gradient())`
`.cross_product(vector(f2.gradient()))`

```

A.<x,y,z> = AffineSpace(QQ, 3)
f1 = x^2+y^2-1
f2 = 5*x-(z-3)^2
Z = Curve([f1,f2],A)
Z.is_smooth()

```

3. Soit h la fonction rationnelle

$$h(x,y,z) = \frac{z^2 - 6z + 5}{x^2 - x + y^2} \in \mathbb{R}(\mathcal{Z}).$$

Calculer l'ensemble des zéros et des pôles de h sur la courbe \mathcal{Z} avec leur ordre. (On pourra avoir besoin du corps `QQ5.<rac5> = QuadraticField(5)` qui permet de définir $\sqrt{5}$ comme variable symbolique.)

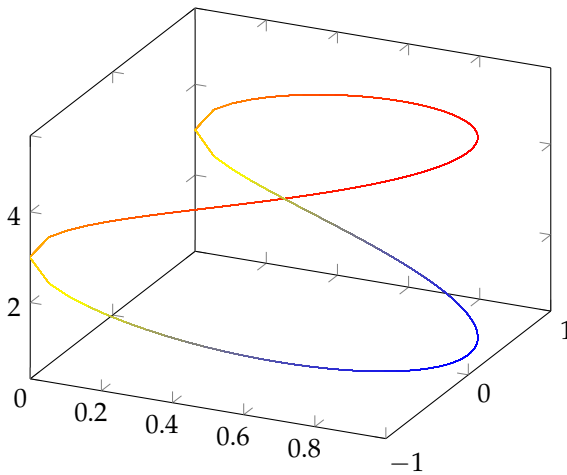


FIGURE 17: Intersection de deux cylindres

Remarque 278. On appelle *diviseur* d'une fonction rationnelle h l'ensemble des zéros et des pôles $(P_i)_{i \leq t}$ de h comptés avec leur multiplicité. Compte tenu du caractère additif d'un diviseur quand on multiplie deux fonctions, on note généralement un diviseur par une somme formelle

$$\operatorname{div}(h) = \operatorname{ord}_{P_1}(h)(P_1) + \cdots + \operatorname{ord}_{P_t}(h)(P_t).$$

Exercice 279. Soit \mathcal{E} la courbe définie par

$$f(x,y) = x^2 + x - y^3 \in \mathbb{F}_2[x,y]$$

1. Montrer que \mathcal{E} n'admet aucun point singulier, même sur une extension de \mathbb{F}_2 .
2. Comparer les fonctions rationnelles

$$h_1(x,y) = \frac{x^2}{y^3} \quad \text{et} \quad h_2(x,y) = \frac{x}{1+x} \in \mathbb{K}(\mathcal{E}).$$

3. Calculer $\text{ord}_M(h_1)$ et $\text{ord}_M(h_2)$ pour tout point $M \in \mathcal{E}(\overline{\mathbb{F}_2})$.
4. Déterminer $\sum_{M \in \mathcal{C}} \text{ord}_M(h_1)$.

Remarque 280. En général, il est toujours vrai pour n'importe quelle fonction rationnelle h que

$$\sum_{M \in \mathcal{C}(\overline{\mathbb{K}})} \text{ord}_M(h) = 0$$

lorsque \mathcal{C} est une courbe lisse et que la somme porte sur l'ensemble des points projectifs définis sur la clôture algébrique $\overline{\mathbb{K}}$ de \mathbb{K} .

Les 27 droites d'une cubique non singulière

Définition 281. Soit \mathbb{K} un corps, on note $\mathbb{P}^n(\mathbb{K})$ l'espace projectif de dimension n sur \mathbb{K} , c'est-à-dire l'ensemble

$$\mathbb{P}^n(\mathbb{K}) = \mathbb{K}^{n+1} / \sim$$

où \sim est la relation d'équivalence

$$\forall \mathbf{x}, \mathbf{y} \in \mathbb{K}^{n+1}, \quad \mathbf{x} \sim \mathbf{y} \text{ si } \exists \lambda \in \mathbb{K} \setminus \{0\}, \mathbf{x} = \lambda \mathbf{y}.$$

À tout ensemble algébrique (affine) de \mathbb{K}^n , défini comme le lieu des zéros d'une famille de polynômes $\{f_i(x_1, \dots, x_n) = 0\}_{i \leq t}$, on peut associer un ensemble projectif

$$\left\{ (X_0, \dots, X_n) \in \mathbb{P}^n(\mathbb{K}); \forall i \leq t, X_0^{\deg f_i} f_i(X_1/X_0, \dots, X_n/X_0) = 0 \right\}.$$

Ainsi, la droite affine d'équation $\alpha x + \beta y + \gamma = 0$ admet $\alpha X + \beta y + \gamma Z = 0$ comme équation projective.

Exercice 282 (Surface de Clebsch). On appelle *surface de Clebsch* l'ensemble des points $(X_0, X_1, X_2, X_3, X_4) \in \mathbb{R}^5$ tels que

$$\mathcal{C} : \begin{cases} X_0^3 + X_1^3 + X_2^3 + X_3^3 + X_4^3 &= 0 \\ X_0 + X_1 + X_2 + X_3 + X_4 &= 0 \end{cases}$$

1. En utilisant `implicitplot3d`, représenter avec SageMath la surface de \mathbb{R}^3 d'équation

$$x_1^3 + x_2^3 + x_3^3 + 1 - (x_1 + x_2 + x_3 + 1)^3 = 0$$

Quel lien faites-vous avec \mathcal{C} ?

2. Soit D la droite paramétrée par

$$\begin{cases} \mathbb{K} \cup \{\infty\} & \rightarrow & \mathbb{P}(\mathbb{R}^5) \\ t & \mapsto & (1 : t : a + dt : b + et : c + ft) \\ \infty & \mapsto & (0 : 1 : d : e : f) \end{cases}$$

À quelle condition D est-elle incluse dans \mathcal{C} ? [Indications : on substituera l'équation de D dans celle de \mathcal{C} et on résoudra système de six équations polynomiales en $\{a, b, c, d, e, f\}$.]

3. Dresser la liste de l'ensemble des droites (réelles) contenues dans \mathcal{C} .

Remarque 283. Le théorème de Cayley-Salmon (1849) affirme que toute surface cubique lisse contient exactement 27 droites complexes. La surface de Clebsch est un exemple dans lequel toutes ces droites sont réelles. Les 27 droites d'une cubique ainsi que leurs points d'intersections forment une configuration remarquable qui apparaît également dans d'autres domaines des mathématiques.

Applications divers des bases de Gröbner

Enveloppe d'une courbe

On suppose que l'on dispose d'une famille de courbe $\mathcal{C}_t = \mathcal{V}(f)$ où $f(x, y, t) \in \mathbb{R}[x, y, t]$.

Définition 284. On appelle *enveloppe* de la courbe l'ensemble des points de \mathbb{R}^2 tels que $f(x, y, t)$ et $\frac{\partial}{\partial t}f(x, y, t) = 0$ pour un certain t .

Remarque 285. À première vue, nous pouvons nous demander d'où provient l'étrange définition ci-dessus. Supposons que l'enveloppe admette une paramétrisation $t \mapsto M(t) = (\phi(t), \psi(t))$. On doit avoir d'une part que $M(t) \in \mathcal{C}_t$. D'autre part, nous notons que l'enveloppe est tangente à \mathcal{C}_t en $M(t)$. Mais alors $\nabla f_t(x, y) \perp M'(t)$, autrement dit

$$\frac{\partial f}{\partial x}\phi'(t) + \frac{\partial f}{\partial y}\psi'(t) = 0$$

Or $f(M(t), t) = 0$, donc $\frac{\partial f}{\partial x}\phi'(t) + \frac{\partial f}{\partial y}\psi'(t) + \frac{\partial f}{\partial t} = 0$, ce qui donne la seconde équation $\frac{\partial}{\partial t}f(x, y, t) = 0$ par différence.

Exemple 286. On note M_θ le point du cercle unité d'angle θ avec l'axe (Ox) et (D_t) la droite $(M_\theta M_{2\theta})$ avec $t = \tan(\theta/2)$. La droite (D_t) a pour équation cartésienne

$$g(t, x, y) = (-2t^4 + 6t^2)y + (-6t^3 + 2t)x + (-2t^3 - 2t) = 0$$

Pour en chercher l'enveloppe, on munit $\mathbb{Q}[t, x, y]$ de l'ordre lexicographique et on calcule une base de Gröbner du système $\left\{g, \frac{\partial g}{\partial t}\right\}$ (d'après la proposition 254, on s'attend à ce que la dernière équation ne dépende pas de t). On obtient effectivement

$$h = x^6 + 2x^4y^2 + x^2y^4 - 2x^5 - 4x^3y^2 - 2xy^4 + \frac{1}{3}x^4 + \frac{4}{3}x^2y^2 + y^4 + \frac{28}{27}x^3 + \frac{4}{3}xy^2 - \frac{1}{9}x^2 - \frac{2}{3}y^2 - \frac{2}{9}x - \frac{1}{27}$$

qui est indépendant de t et qui correspond à l'enveloppe des droites (voir figure 18).

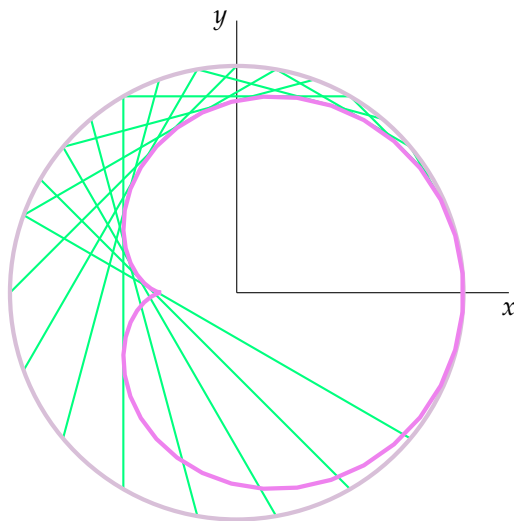


FIGURE 18: La cardioïde comme enveloppe de cordes d'un cercle

Exercice 287. On donne la famille de polynômes

$$f(x, y, t) = (x - t)^2 + (y + t^2 - 4t)^2 - 4 \in \mathbb{R}[x, y, t].$$

On note C_t la courbe d'équation $f(x, y, t) = 0$.

1. Identifier la famille $(C_t)_{t \in \mathbb{R}}$.
2. En calculant une base de Gröbner de l'idéal $\langle f, \frac{\partial}{\partial t} f(x, y, t) \rangle$ pour un ordre bien choisi de $\mathbb{R}[x, y, t]$, éliminer la variable t et retrouver un polynôme candidat décrivant l'enveloppe \mathcal{E} de la famille de courbes.
3. Représenter sur un même graphique C_t pour diverses valeurs de t et l'enveloppe \mathcal{E} .
4. Un robot expulse des billes rondes de rayon 2. Le vecteur-vitesse initial de chaque bille est $(1, 4)$; les billes évoluent selon les lois de la mécanique newtonienne. Le robot est placé sous un toit de pente 1 situé à 5 unités au dessus de lui. Les billes peuvent-elles librement évoluer ou tapent-elles le toit?

•

Preuve de théorèmes géométriques

Exercice 288. On considère un triangle formé des points $A(0, 0)$, $B(1, 0)$ et $C(u, v)$ de \mathbb{R}^2 . On note Δ_A , Δ_B , Δ_C les médianes issues de A , B et C .

1. Décrire les idéaux $\mathfrak{I}(\Delta_A)$, $\mathfrak{I}(\Delta_B)$ et $\mathfrak{I}(\Delta_C)$ de $\mathbb{R}[x, y]$.
2. Que peut-on dire de $\mathfrak{I}(\Delta_A) + \mathfrak{I}(\Delta_B)$ et de $\mathfrak{I}(\Delta_A \cap \Delta_B)$?
3. Montrer que $\mathfrak{I}(\Delta_C) \subseteq \mathfrak{I}(\Delta_A \cap \Delta_B)$.
4. Quel théorème classique de géométrie vient-on de redémontrer?

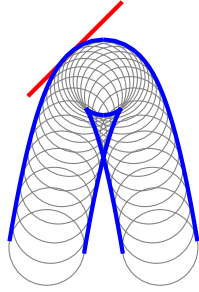


FIGURE 19: Enveloppe d'une famille de cercles parcourant une parabole.

Exercice 289. Soient A, B, C, D, E et F six points du plan. Montrer que les assertions suivantes peuvent toutes être exprimées par une ou plusieurs équations polynômiales.

1. (AB) est parallèle à (CD) ,
2. (AB) est perpendiculaire à (CD) ,
3. A, B et C sont colinéaires,
4. les distances AB et CD sont égales,
5. C appartient au cercle de centre A et rayon AB
6. C est le milieu de $[AB]$
7. les angles aigus \widehat{ABC} et \widehat{DEF} sont égaux.
8. la droite (BD) est une bissectrice de l'angle \widehat{ABC} .

Exercice 290 (Le cercle d'Apollonius). Soit ABC un triangle rectangle en A de coordonnées $A(0,0)$, $B(u_1,0)$ et $C(0,u_2)$. On note $P(x_1, x_2)$, $Q(0, x_3)$ et $R(x_4, 0)$ les milieux des segments opposés à A, B et C et $H(x_5, x_6)$ le pied de la hauteur issue de A .

1. Exprimer par 6 polynômes $(h_i)_{i \leq 6}$ les définitions de P, Q, R et H .
2. On introduit $O(x_7, x_8)$. Exprimer par deux équations h_7 et h_8 que O est le centre du cercle passant par P, Q et R .
3. Donner une équation g qui exprime que H appartient à ce même cercle.
4. A-t-on que $g \in \langle (h_i)_{i \leq 8} \rangle \subseteq \mathbb{Q}[u_1, u_2, x_1, \dots, x_8]$?
5. A-t-on que $g \in \langle (h_i)_{i \leq 8} \rangle \subseteq \mathbb{Q}(u_1, u_2)[x_1, \dots, x_8]$?⁹
(Ceci revient à se demander s'il existe un polynôme $c(u_1, u_2)$ non nul tel que $c \cdot g \in \langle (h_i)_{i \leq 8} \rangle \subseteq \mathbb{Q}[u_1, u_2, x_1, \dots, x_8]$ et permet d'éliminer des situations géométriques dégénérées, ici le cas $u_1 = u_2 = 0$.) Que peut-on conclure?
6. La figure ci-contre semble indiquer que A appartient aussi au même cercle. Qu'en pensez-vous?

9. On se placera dans $\text{PolU}.\langle u_1, u_2 \rangle = \text{PolynomialRing}(\mathbb{Q}, 2) \setminus \text{FracU} = \text{FracU}$

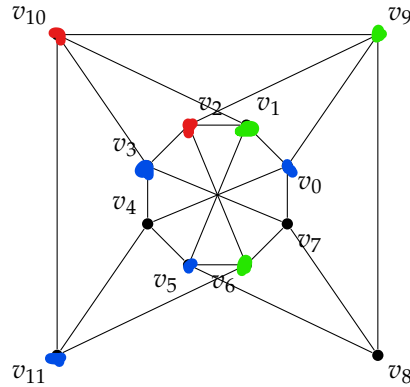


FIGURE 21: Graphe de Chao et Chen (1993)

```

G= Graph(12)
G.add_cycle(range(8))
G.add_edges([(i,i+4) for i in range(4) ])
G.add_edges([(8,5),(8,7),(9,0),(9,2),(10,1),
(10,3),(11,4),(11,6) ])
G.add_edges([(8,9),(9,10),(10,11)])
G.show()
G.coloring()

```

Combien de couleurs sont nécessaires pour colorer ce graphe ?

2. Calculer une base de Gröbner de l'idéal $\mathcal{I}_{G,3}$ pour l'ordre lexicographique $x_0 < x_1 < \dots < x_{11}$.

Indications :

```

MPol = PolynomialRing(QQ,12,'x',
order = 'invlex')
phi (v) = v^3-1
psi (u,v) = u^2+u*v+v^2
IG= Ideal(MPol, [phi(MPol.gen(v))
for v in G.vertices()
+ [psi(MPol.gen(u),MPol.gen(v))
for (u,v) in G.edges(labels=false)])

```

3. En inspectant la base de Gröbner obtenue, montrer que le graphe ne possède qu'une seule manière d'être coloré. Retrouver les résultats de la question 1.

Remarque 296. En réalité, le comportement de la base de Gröbner de l'exercice 295 est typique de graphes uniquement k -colorables. Il a été démontré que dans ce cas, la base obtenue est formée de n polyômes qui indiquent directement quel coloration adopter.

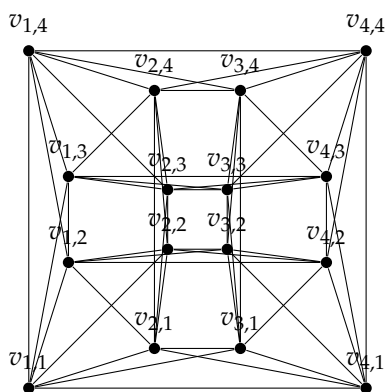
Exercice 297 (Sudoku). Un Sudoku 4×4 est un tableau de nombres 4×4 à valeurs dans $\mu_4 = \{\pm 1, \pm i\}$ tel que les 4 lignes, les 4 colonnes

et les 4 sous-tableaux 2×2 localisés dans un coin contiennent les 4 valeurs de μ_4 .

$x_{1,4}$	$x_{2,4}$	$x_{3,4}$	$x_{4,4}$
$x_{1,3}$	$x_{2,3}$	$x_{3,3}$	$x_{4,3}$
$x_{1,2}$	$x_{2,2}$	$x_{3,2}$	$x_{4,2}$
$x_{1,1}$	$x_{2,1}$	$x_{3,1}$	$x_{4,1}$

FIGURE 22: Sudoku 4×4

1. Exprimer le problème du Sudoku comme un problème de coloration d'un certain graphe que l'on notera S_4 .

FIGURE 23: Graphe S_4 du Sudoku 4×4

2. Coder dans `PolynomialRing(CC,4,var_array='x', order = 'invlex')` l'idéal $\mathcal{I}_{S_4,4}$.
3. Résoudre le problème suivant :

4			
	3	4	
	2	1	
			2

4. Combien de Sudoku auriez-vous pu résoudre de tête depuis que vous avez commencé cet exercice ?

Liens avec la programmation entière

Définition 298. On appelle *programme entier* tout problème d'optimisation de la forme

$$\min \langle \mathbf{c}, \mathbf{x} \rangle$$

$$\text{sujet à } \begin{cases} \mathbf{Ax} = \mathbf{b} \\ \mathbf{x} \in \mathbb{N}^n \end{cases}$$

où $\mathbf{A} \in \mathbb{Z}^{m \times n}$, $\mathbf{b} \in \mathbb{Z}^m$ et $\mathbf{c} \in \mathbb{Z}^n$.

Nous présentons sur un exemple comment certaines idées issues des bases de Gröbner ont été adaptées à la résolution de ce type de programme¹⁰.

Exercice 299. On souhaite rendre la monnaie pour \$1,17 à l'aide de pièces de 1¢ (*penny*), 5¢ (*nickel*), 10 ¢ (*dime*) et 25 ¢ (*quater*) en minimisant le nombre de pièces.

$$\begin{aligned} & \min p + n + d + q \\ & \text{sujet à } \begin{cases} p + 5n + 10d + 25q = 117 \\ (p, n, d, q) \in \mathbb{N}^4 \end{cases} \end{aligned}$$

1. Construire l'idéal $\mathfrak{J} = \langle P^5 - N, P^{10} - D, P^{25} - Q \rangle$ dans l'anneau $\mathbb{Q}[P, N, D, Q]$ et en calculer une base de Gröbner.
2. Réduire P^{117} modulo \mathfrak{J} pour un ordre monomial bien choisi. Que remarque-t-on ?

10. Arjeh M. Cohen, Hans Cuypers, and Hans Sterk, editors. *Some tapas of computer algebra*, volume 4 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, 1999

Exercices de révisions I

Tous les exercices peuvent être résolus à la main (sans SageMath).

Exercice 300. Soit M le \mathbb{Z} -module engendré par les vecteurs

$$\mathbf{m}_1 = \begin{pmatrix} 1 \\ 0 \\ 3 \end{pmatrix} \quad \text{et} \quad \mathbf{m}_2 = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}$$

et N le \mathbb{Z} -module engendré par les vecteurs

$$\mathbf{n}_1 = \begin{pmatrix} 13 \\ 7 \\ -10 \end{pmatrix}, \quad \mathbf{n}_2 = \begin{pmatrix} -1 \\ -2 \\ 2 \end{pmatrix} \quad \text{et} \quad \mathbf{n}_3 = \begin{pmatrix} 4 \\ 3 \\ -4 \end{pmatrix}.$$

Est-ce que $M \subseteq N$?

Exercice 301. Résoudre dans \mathbb{Z}^3 le système d'équations suivant

$$\begin{cases} 2x - 7y + 3z \equiv 0 \pmod{5} \\ x - 5y + 4z \equiv 1 \pmod{3} \\ 4x \quad \quad - 6z \equiv 2 \pmod{8} \end{cases}$$

Exercice 302. Soit N le module engendré par les colonnes de la matrice

$$A = \begin{pmatrix} -28 & 30 & -32 & 44 & -2 \\ 22 & -24 & 26 & -32 & 2 \\ 26 & -30 & 34 & -28 & 4 \end{pmatrix}$$

Donner le rang et les facteurs invariants de $M = \mathbb{Z}^3/N$? Donner un générateur de l'idéal annulateur de la partie de torsion de M .

Exercice 303. Soit Λ le réseau engendré par les vecteurs colonnes de la matrice

$$\begin{pmatrix} -3 & 1 & -2 \\ 1 & -1 & 1 \\ 2 & -2 & 3 \end{pmatrix}$$

Calculer une base LLL-réduite de Λ . Quels sont les minima successifs de Λ ? Nommer Λ .

Exercice 304. Soit f le polynôme

$$f = x^{24} + x^4 + 1 \in \mathbb{F}_{32}[x]$$

1. Quelle est la factorisation sans facteurs carrés de f ?
2. Montrer que f ne possède pas de racine dans \mathbb{F}_2 .
3. Montrer que f ne possède pas de facteur de degré 2 dans \mathbb{F}_2 .
4. Montrer que f ne possède pas de facteur de degré 3 dans \mathbb{F}_2 .
5. Que peut-on dire de la factorisation de f en tant que polynôme de $\mathbb{F}_2[x]$?
6. En déduire la factorisation de f dans $\mathbb{F}_{32}[x]$.

Exercice 305. Soient les polynômes

$$f = x^2 + y + 2, \quad g = y^3 + y + 1 \in \mathbb{F}_3[x, y].$$

1. Calculer une base de Groebner de l'idéal \mathcal{I} engendré par f et g pour l'ordre lexicographique (avec $x > y$).
2. Dessiner le diagramme en escalier des monômes dominants de \mathcal{I} .
3. Donner une base de $\mathbb{F}_3[x, y]/\mathcal{I}$ et sa dimension.
4. Donner les zéros de g sur la parabole d'équation $f = 0$ ainsi que leur multiplicité.

Deuxième partie

**ACCQ 203b : Algorithmes
pour l'arithmétique**

TP 8 : Primalité des entiers

Buts : Apprendre les algorithmes classiques de primalité.

Travaux préparatoires : Cours et exercices 314, 326 & 330.

Évaluation du TP : Exercices 318 (critère de Rabin-Miller), 319 (performances de Rabin-Miller), 327 (critère de Solovay-Strassen), 329 (comparaison entre Rabin-Miller et Solovay-Strassen), 332 (test de Lucas) & 333 (Test BPSW). .

Le contexte

Les techniques cryptographiques contemporaines font un usage intensif de nombres premiers. On pense bien sûr à RSA, mais aussi à de nombreux cryptosystèmes s'exécutant dans un groupe G dont l'ordre $|G|$ doit être aussi proche d'un nombre premier que possible (G peut par exemple être l'ensemble des points d'une courbe elliptique). Il a été crucial au cours du dernier quart du XX^e siècle de trouver des méthodes prouvant la primalité de grands entiers.

On dispose depuis longtemps de tests probabilistes qui permettent rapidement d'infirmer qu'un nombre est premier : on parle de *tests de composition*. Certains nombres composés peuvent leur échapper, on parle alors de *pseudo-premier*.

À l'inverse, on connaît peu d'algorithmes efficaces garantissant qu'un nombre est premier et ces tests ne sont pas de complexité polynomiale. Il a fallu attendre 2002 (test AKS) pour savoir que ce problème est décidable en temps polynomial, même si l'algorithme proposé n'est pas compétitif en pratique.

Dans l'industrie, on se contente souvent d'utiliser des nombres premiers probables. Dans tous les cas, s'il s'avère nécessaire de prouver qu'un nombre est réellement premier, on commence toujours par vérifier qu'il est premier probable.

Dans ce T.P., nous passons en revue les techniques préhistoriques de primalité. Malheureusement, les techniques modernes sont encore hors de notre portée : elles utilisent la théorie analytique des nombres ou les courbes elliptiques.

Exercice 306. Quel est le plus grand nombre premier connu ? Comparer la taille en bits d'un fichier contenant ce nombre avec celle du dernier clip vidéo de votre groupe musical favori.

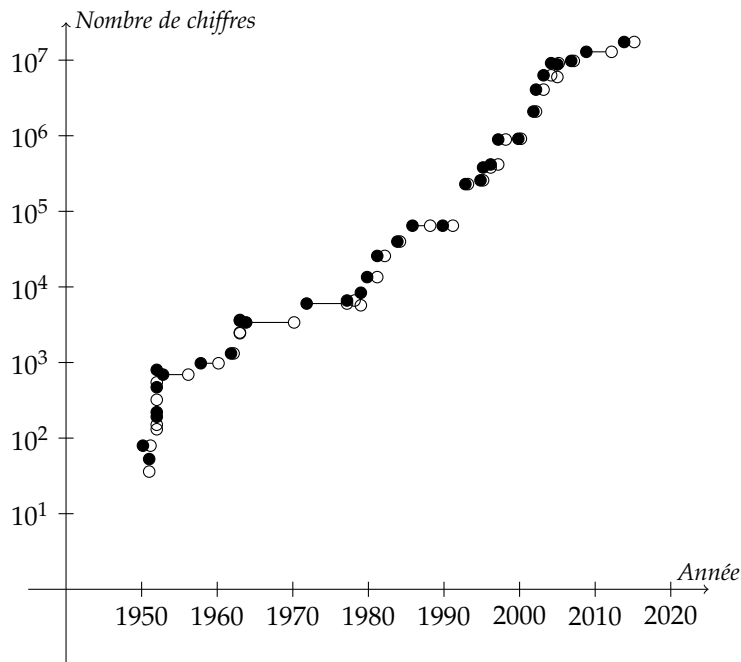


FIGURE 24: Plus grand premier connu au cours du temps

Tests de composition

Le test de Fermat et les nombres de Carmichael

Rappelons que \mathbb{F}_p^\times est un groupe d'ordre $p - 1$, ce qui a pour conséquence :

Théorème 307 (Petit théorème de Fermat). Soit p premier, alors, pour tout élément $a \in \mathbb{F}_p^\times$, $a^{p-1} = 1$.

On en tire un premier test :

Algorithme 18 : Test de composition de Fermat**Entrées** : Entier n **Sorties** : n « n'est pas premier » ou « peut être premier »

```

1 Choisir  $a \in \{2, \dots, n-1\}$ .
2 si  $g \leftarrow a \wedge n > 1$  alors
3   | retourner  $g$  est un facteur de  $n$ 
4 Calculer  $b = a^{n-1} \bmod n$ .
5 si  $b \neq 1$  alors
6   | retourner  $n$  composé
7 sinon
8   | retourner  $n$  peut être premier

```

Définition 308. Lorsque l'algorithme parvient à montrer que n est composé et que a est premier à n , on appelle a un *témoin de Fermat*. On appelle *nombre de Carmichael* tout nombre composé dépourvu de témoin de Fermat, c'est-à-dire un entier n tel que

$$\forall a \in (\mathbb{Z}/n\mathbb{Z})^\times, \quad a^{n-1} \equiv 1 \pmod{n}.$$

Malheureusement les nombres de Carmichael existent; on sait même depuis 1994 qu'il en existe une infinité.

- Exercice 309.** 1. Programmer avec SageMath une fonction `testF` effectuant le test de Fermat et renvoyant un booléen en sortie.
 2. Exhiber la liste `listCarmichael` les nombres de Carmichael ≤ 3000 .

- Exercice 310.** 1. Montrer qu'un entier n de Carmichael est forcément sans facteurs carrés et tel que $p-1$ divise $n-1$ pour tout diviseur premier p de n .
 2. Montrer qu'un nombre de Carmichael est forcément impair et possède au moins 3 diviseurs premiers.

Exercice 311. On suppose que $a \in \mathbb{Z}/n\mathbb{Z} \setminus \{0\}$ et que $a \wedge n > 1$. Montrer que $a^{n-1} \neq 1$. Que peut-on en déduire de la ligne 2 de l'algorithme ci-dessus?

Test de Rabin-Miller

Définition 312. On rappelle que la *valuation p -adique* d'un entier n est le plus grand entier v tel que p^v divise n . On appelle *partie première à p* le quotient n/p^v .

Proposition 313. Soit p un nombre premier. Soit v la valuation dyadique et m la partie première à 2 de $p-1$, alors pour tout $a \in \mathbb{F}_p^\times$

$$a^m = 1 \quad \text{ou} \quad \exists d \in \llbracket 0, v-1 \rrbracket, \quad a^{2^d m} = -1.$$

Exercice 314. Démontrer la proposition 313.

Définition 315. Soit n un entier impair. On note v et m les valuation et partie première à 2 de $n - 1$. Un élément $a \in (\mathbb{Z}/n\mathbb{Z})^\times$ tel que $a^m \not\equiv 1 \pmod{n}$ et pour tout $d \in \llbracket 0, v - 1 \rrbracket$, $a^{2^d m} \not\equiv -1 \pmod{n}$ s'appelle un *témoin de Rabin-Miller*. Dans le cas contraire, on dit que a est une base de pseudoprimauté pour n .

Exemple 316. Pour montrer que $n = 21$ est composé, nous avons calculé les puissances de a pour $a \in (\mathbb{Z}/n\mathbb{Z})^\times$. Avec $m = 5$ et d variant de 0 à 2,

a	a^m	a^{2m}	$a^{2^2 m}$	a	a^m	a^{2m}	$a^{2^2 m}$
-10	2	4	-5	1	1	1	1
-8	-8	1	1	2	10	-5	4
-5	4	-5	4	4	-5	4	-5
-4	5	4	-5	5	17	16	4
-2	10	-5	4	8	8	1	1
-1	-1	1	1	10	-2	4	-5

Ce tableau montre que ± 10 , ± 5 , ± 4 et ± 2 sont déjà des témoins de Fermat pour n . Ils sont *a fortiori* des témoins de Rabin-Miller. De plus, ± 8 sont des témoins de Rabin-Miller qui ne sont pas témoin de Fermat.

Corollaire 317. Soit $n \geq 10$ un entier impair composé, alors n possède au moins $\frac{3}{4}\varphi(n)$ témoins de Rabin-Miller (où φ est l'indicatrice d'Euler).

Démonstration. .

□

On en tire le test de composition suivant

Algorithme 19 : Test de composition de Rabin-Miller**Entrées** : Entier n **Sorties** : n « est composé » ou « peut être premier »

```

1 Choisir aléatoirement  $a \in \llbracket 2, n-1 \rrbracket$ .
2 Calculer  $v$  et  $m$  tels que  $n-1 = 2^v m$  avec  $m$  impair.
3 si  $g \leftarrow a \wedge n > 1$  alors
4   retourner  $g$  est un facteur de  $n$ 
5  $b \leftarrow a^m \bmod n$ 
6 si  $b = 1$  alors
7   retourner  $n$  peut être premier
8 pour  $i \in \llbracket 1, v \rrbracket$  faire
9   si  $b^2 = 1 \bmod n$  alors
10     $g \leftarrow (b+1) \wedge n$ 
11    si  $g = 1$  ou  $n$  alors
12      retourner  $n$  peut être premier
13    sinon
14      retourner  $g$  est un facteur de  $n$ 
15     $b \leftarrow b^2 \bmod n$ 
16 retourner «  $n$  est composé ».

```

- Exercice 318.** 1. Écrire une fonction `testRM` qui reprend l'algorithme de Rabin-Miller et renvoie un booléen en sortie.
2. Trouver un témoin `temoinRM561` de Rabin-Miller pour $n = 561$.

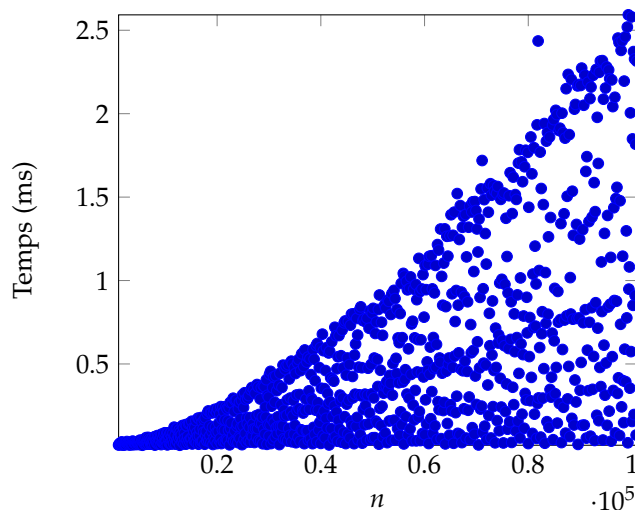


FIGURE 25: Temps d'exécution du test de Rabin-Miller.

- Exercice 319.** 1. Pour tout entier $10 \leq n \leq 500$, effectuer 100 tests de primalité de Rabin-Miller. Représenter par un diagramme en barre

Indication :

```

bar_chart( [sum( [testRM(n) for i in range(nbttests)] ) for n in range(10, 500)] )

```

la statistique de succès au test (cf. FIG. 26).

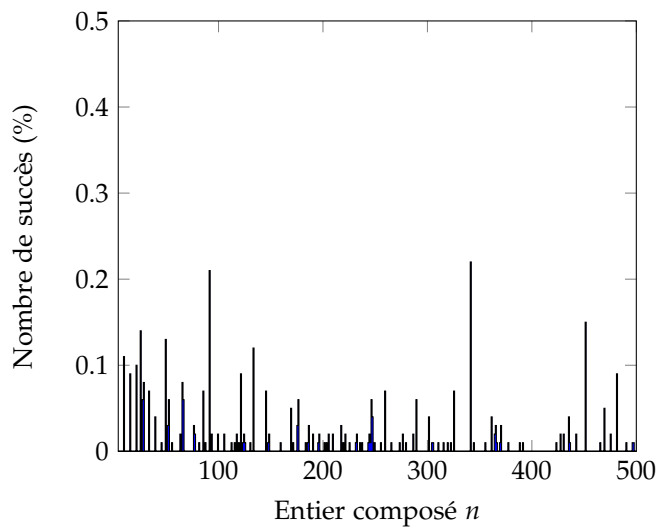


FIGURE 26: Statistique de fausse primalité pour le test de Rabin-Miller.

2. Sur votre échantillon, quels nombres composés sont-ils les plus fréquemment déclarés premiers ? Ont-ils une propriété particulière ?
3. On souhaite réduire à 2^{-50} la probabilité diagnostiquer premier un nombre composé. Comment modifier l'algorithme ? Faites quelques tests.
4. Etudier le temps d'exécution du nouvel algorithme en fonction de n . (cf. FIG. 27)

Indication :
`list_plot([timeit('testRM(n)', number=20, repeat=10) for n in range(1, 50000)], x_label='n', y_label='Temps (ms)')`

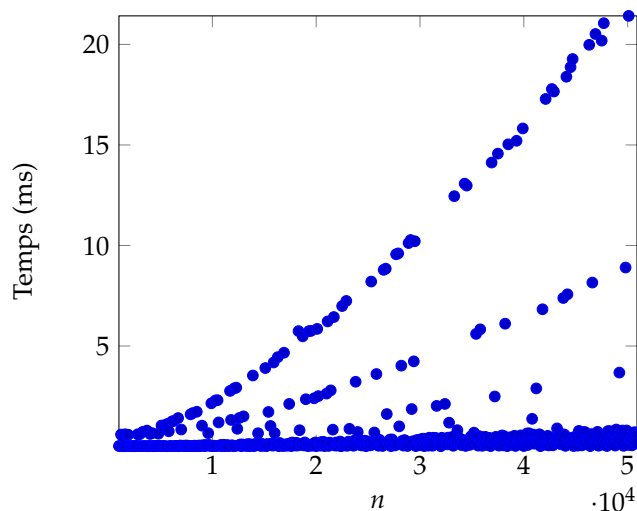


FIGURE 27: Temps d'exécution du test de Rabin-Miller itéré.

Test de Solovay-Strassen

Définition 320. Soit p premier et $a \in \mathbb{F}_p$. On appelle *symbole de Legendre*

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{si } a = 0 \\ 1 & \text{si } a \text{ est un carré de } \mathbb{F}_p \\ -1 & \text{si } a \text{ n'est pas un carré de } \mathbb{F}_p \end{cases}.$$

Définition 321. Soit m et n deux entiers, on appelle *symbole de Jacobi* le produit

$$\left(\frac{m}{p_1}\right) \cdots \left(\frac{m}{p_r}\right)$$

où les $(p_i)_{i \leq r}$ sont les facteurs premiers (éventuellement répétés) de n .

En notant

$$\varepsilon(n) = (-1)^{(n-1)/2}, \quad \omega(n) = (-1)^{(n^2-1)/8},$$

$$\vartheta(m, n) = (-1)^{(m-1)(n-1)/4},$$

les règles de réciprocité quadratique conduisent à :

Algorithme 20 : Symbole de Jacobi

Entrées : Entier m , entier $n \geq 3$ impair

Sorties : Symbole $\left(\frac{m}{n}\right)$

```

1 si  $m < 0$  alors
2   retourner  $\varepsilon(n) \cdot \text{Jacobi}(-m, n)$ 
3 sinon si  $m > 0$  pair alors
4   retourner  $\omega(n) \cdot \text{Jacobi}(m/2, n)$ 
5 sinon si  $m > 1$  impair alors
6   retourner  $\vartheta(m, n) \cdot \text{Jacobi}(n \bmod m, m)$ 
7 sinon si  $m = 1$  alors
8   retourner 1
9 sinon si  $m = 0$  alors
10  retourner 0

```

Exercice 322. 1. Coder une fonction `myJacobi(m,n)` calculant le symbole de Jacobi.

2. Vérifier que `myJacobi(m,n)` coïncide avec celle de SageMath pour tout couple $(m, n) \in \llbracket 3, 101 \rrbracket^2$.

Proposition 323. Soit n un entier impair, alors, n est premier si et seulement si pour tout $a \in (\mathbb{Z}/n\mathbb{Z})^\times$, $\left(\frac{a}{n}\right) \equiv a^{(n-1)/2} \pmod{n}$.

Définition 324. Un élément $a \in (\mathbb{Z}/n\mathbb{Z})^\times$ tel que $\left(\frac{a}{n}\right) \not\equiv a^{(n-1)/2} \pmod{n}$ s'appelle un *témoin de Solovay*.

Corollaire 325. Si $n \geq 3$ est un entier impair composé, il y a au moins $\varphi(n)/2$ témoins de Solovay (où φ est l'indicatrice d'Euler).

Exercice 326. Démontrer la proposition 323 et le corollaire 325. Indications :

1. Lorsque n est premier, étudier le morphisme $\phi : x \mapsto x^{(n-1)/2}$.
2. Pour la réciproque.
 - (a) Montrer que si n n'est pas premier, n est de Carmichael. En déduire que n est sans facteurs carrés.
 - (b) Soit $(p_i)_{i \leq s}$ les diviseurs premiers de n , montrer que

$$\forall (\alpha_i)_{i \leq s} \subseteq \prod_{i=1}^s \mathbb{F}_{p_i}^\times, \quad \alpha_1^{(n-1)/2} = \left(\frac{\alpha_1}{p_1}\right) \left(\frac{\alpha_2}{p_2}\right) \cdots \left(\frac{\alpha_s}{p_s}\right) \pmod{p_1}$$

et y voir une contradiction.

- (c) Étudier le morphisme $x \mapsto \left(\frac{x}{n}\right) x^{-(n-1)/2}$ de $(\mathbb{Z}/n\mathbb{Z})^\times$.

On peut donc construire le test suivant.

Algorithme 21 : Test de composition de Solovay-Strassen

Entrées : Entier n

Sorties : n « n'est pas premier » ou « peut être premier »

```

1 si  $n$  pair alors
2   retourner «  $n$  est composé »
3 Choisir  $a \in \{2, \dots, n-1\}$ .
4 si  $a \wedge n \neq 1$  alors
5   retourner «  $n$  est composé ».
6 si  $\text{Jacobi}(a, n) = a^{(n-1)/2}$  alors
7   retourner «  $n$  peut être premier ».
8 sinon
9   retourner «  $n$  est composé ».
```

Exercice 327. 1. Implémenter le test de Solovay-Strassen testSS (vous pouvez utiliser au choix votre propre symbole de Jacobi ou l'implémentation native de SageMath).

2. Pour tout entier $10 \leq n \leq 500$, effectuer 100 tests de primalité de Solovay-Strassen. Représenter par un diagramme en barre la statistique de succès au test (cf. FIG. 28).
3. Sur votre échantillon, quels nombres composés sont-ils les plus fréquemment déclarés premiers ? Ont-ils une propriété particulière ?
4. On souhaite réduire à 2^{-50} la probabilité diagnostiquer premier un nombre composé. Comment modifier l'algorithme ? Faites quelques tests.

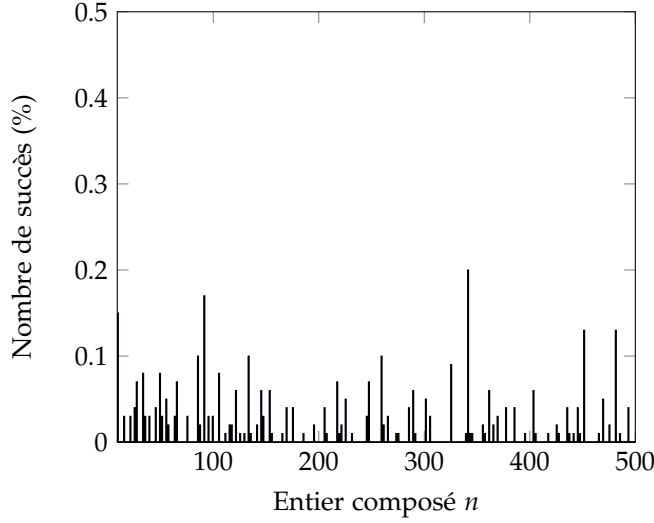


FIGURE 28: Statistique de fausse primalité pour le test de Solovay-Strassen.

Comparaison des tests de Rabin-Miller et de Solovay-Strassen

Proposition 328. Soit $n \geq 3$ un entier impair et $a \in (\mathbb{Z}/n\mathbb{Z})^\times$. Si a est un témoin de Solovay-Strassen, alors a est un témoin de Rabin-Miller.

Démonstration. Soient $v = v_2(n-1)$ et m les entiers tels que $n-1 = 2^v m$ et a un témoin de Rabin-Miller.

Cas $a^m \equiv 1 \pmod{n}$. Comme m est impair, on a d'une part, $\left(\frac{a}{n}\right) = \left(\frac{a}{n}\right)^m = \left(\frac{a^m}{n}\right) = \left(\frac{1}{n}\right) = 1$. D'autre part, $a^{(n-1)/2} = (a^m)^{2^{v-1}} = 1 \pmod{n}$.

Cas $a^{2^s m} \equiv -1 \pmod{n}$ avec $s \in \llbracket 0, v-1 \rrbracket$. D'une part,

$$a^{(n-1)/2} = (-1)^{2^{v-1-s}} = \begin{cases} -1 \pmod{n} & \text{si } s = v-1 \\ 1 \pmod{n} & \text{si } s < v-1 \end{cases}$$

Soit p un diviseur premier de n , on note $p-1 = 2^{v'} m'$ avec $v' = v_2(p-1)$ et m' impair. On a encore $a^{2^s m} \equiv -1 \pmod{p}$ et aussi $a^{2^s m m'} \equiv -1 \pmod{p}$. Par la proposition 323 et comme m est impair,

$$\left(\frac{a}{p}\right) \equiv a^{(p-1)/2} \equiv a^{m(p-1)/2} \equiv a^{2^{v'-1} m m'} \pmod{p}$$

Mais alors, forcément $v' - 1 \geq s$ et

$$\left(\frac{a}{p}\right) = 1 \text{ si } v' > s+1 \quad \text{et} \quad \left(\frac{a}{p}\right) = -1 \text{ si } v' = s+1.$$

Soit k le nombre de premiers p (comptés avec multiplicité) tels que $v_2(p-1) = s+1$. Alors $\left(\frac{a}{n}\right) = (-1)^k$. D'autre part, $n \equiv (1 + 2^{s+1})^k \equiv 1 + k2^{s+1} \pmod{2^{s+2}}$. Soit k est pair, alors $n \equiv 1 \pmod{2^{s+2}}$ et $s > v+1$; or $\left(\frac{a}{n}\right) = 1$; donc $a^{(n-1)/2} \equiv 1 \equiv \left(\frac{a}{n}\right)$. Soit k est impair, alors $n \not\equiv 1 \pmod{2^{s+2}}$ et $s = v+1$; or $\left(\frac{a}{n}\right) = -1$; donc $a^{(n-1)/2} \equiv -1 \equiv \left(\frac{a}{n}\right)$. \square

Exercice 329. Exhiber la liste `listRMvsSS` des couples (n, a) avec $n \leq 150$ tels que n soit composé, a soit un témoin de composition de Rabin-Miller pour n et a ne soit pas un témoin pour Solovay-Strassen.

Le test de Lucas

Soient p et q deux entiers tels que $\Delta = p^2 - 4q \neq 0$. On considère les suites linéaires récurrentes $\mathbf{u} = (u_k)_{k \in \mathbb{N}}$ et $\mathbf{v} = (v_k)_{k \in \mathbb{N}}$ définies par

$$\begin{cases} u_0 &= 0 \\ u_1 &= 1 \\ \forall k \in \mathbb{N}, u_{k+2} &= pu_{k+1} - qu_k \end{cases}$$

et

$$\begin{cases} v_0 &= 2 \\ v_1 &= p \\ \forall k \in \mathbb{N}, v_{k+2} &= pv_{k+1} - qv_k \end{cases}$$

On note ρ et σ les racines du polynôme caractéristique des deux suites $\chi(x) = x^2 - px + q$.

Exercice 330. 1. Donner une formule close pour u_k et v_k en fonction de ρ, σ et k .

2. En déduire que

$$\begin{cases} u_{2k} &= u_k v_k \\ v_{2k} &= v_k^2 - 2q^k \end{cases} \text{ et } \begin{cases} u_{2k+1} &= u_{k+1} v_k - q^k \\ v_{2k+1} &= v_{k+1} v_k - pq^k \end{cases}.$$

Théorème 331. Soient p et q deux entiers tels que $\Delta = p^2 - 4q \neq 0$. Soient $\mathbf{u} = (u_k)_{k \in \mathbb{N}}$ et $\mathbf{v} = (v_k)_{k \in \mathbb{N}}$ les suites définies ci-dessus. Soit n un nombre premier ne divisant pas $2q\Delta$. On pose $\varepsilon = n - \left(\frac{\Delta}{n}\right)$ (symbole de Jacobi) que l'on décompose en $\varepsilon = 2^t m$ avec m impair. Alors n divise u_m ou l'un des $v_{2^s m}$ pour $s \in \llbracket 0, t-1 \rrbracket$.

Démonstration. Nous nous plongeons modulo n et travaillons dans \mathbb{F}_{n^2} .

Si $\left(\frac{\Delta}{n}\right) = 1$, le polynôme $\chi(x) \in \mathbb{F}_n[x]$ est scindé. Ses racines ρ et σ appartiennent à \mathbb{F}_n et sont stables par le Frobenius : $\rho^n = \rho$ et $\sigma^n = \sigma$.

Si $\left(\frac{\Delta}{n}\right) = -1$, le polynôme $\chi(x) \in \mathbb{F}_n[x]$ est irréductible. Ses racines ρ et σ appartiennent à \mathbb{F}_{n^2} et sont échangées par le Frobenius : $\rho^n = \sigma$ et $\sigma^n = \rho$.

Dans tous les cas $(\rho/\sigma)^\varepsilon = 1 \in \mathbb{F}_{n^2}$ ce qui équivaut à

$$(\rho/\sigma)^m = 1 \text{ ou } \exists s \in \llbracket 0, t-1 \rrbracket, (\rho/\sigma)^{2^s m} = -1.$$

$$\Leftrightarrow \rho^m - \sigma^m = 0 \text{ ou } \exists s \in \llbracket 0, t-1 \rrbracket, \rho^{2^s m} + \sigma^{2^s m} = 0.$$

$$\Leftrightarrow u_m = 0 \in \mathbb{F}_{n^2} \text{ ou } \exists s \in \llbracket 0, t-1 \rrbracket, v_{2^s m} = 0 \in \mathbb{F}_{n^2}.$$

□

Algorithme 22 : Test de Lucas

Entrées : Entier n , couple (p, q) **Sorties :** n « est composé » ou « peut être premier »

```

1  $\Delta \leftarrow p^2 - 4q$ 
2  $g \leftarrow n \wedge 2q\Delta$ 
3 si  $1 < g < n$  alors
4   | retourner  $g$  est un facteur de  $n$ 
5 sinon si  $g = n$  alors
6   | retourner mauvais choix de  $(p, q)$ 
7 Décomposer  $n - \left(\frac{\Delta}{n}\right)$  en  $2^t m$  avec  $m$  impair
8  $g \leftarrow n \wedge u_m$ 
9 si  $1 < g < n$  alors
10  | retourner  $g$  est un facteur de  $n$ 
11 sinon si  $g = n$  alors
12  | retourner  $n$  est pseudo premier
13 pour  $s \in \llbracket 0, t-1 \rrbracket$  faire
14   |  $g \leftarrow n \wedge v_{2^s m}$ 
15   | si  $1 < g < n$  alors
16   |   | retourner  $g$  est un facteur de  $n$ 
17   | sinon si  $g = n$  alors
18   |   | retourner  $n$  est pseudo premier
19 retourner  $n$  est composé

```

Exercice 332. Implémenter le test de Lucas `testL` et le vérifier sur quelques exemples.

Le test BPSW

Ce test est dû à Robert Baillie, Carl Pomerance, John Selfridge et Samuel Wagstaff (c. 1980). Il revient à combiner certains des tests déjà présentés. À ce jour, on ne connaît pas de nombre composé passant le test BPSW. Le premier contre-exemple possède probablement plus de 10000 bits. On peut en tout cas l'utiliser avec certitude pour des entiers de moins de 64 bits.

Algorithme 23 : Test de BPSW**Entrées :** Entier n **Sorties :** n « est composé » ou « peut être premier »

```

1 si le test de Rabin-Miller avec  $a = 2$  échoue alors
2   retourner  $n$  est composé
3 Chercher le premier terme  $\Delta$  de la suite  $\left((-1)^k(2k+5)\right)_{k \in \mathbb{N}}$  tel
   que  $\left(\frac{\Delta}{n}\right) = -1$ .
4 Poser  $p = 1$  et  $q = (1 - \Delta)/4$ .
5 si le test de Lucas avec  $(p, q)$  échoue alors
6   retourner  $n$  est composé
7 retourner  $n$  est BPSW-premier

```

Exercice 333. Implémenter le test BPSW `testBPSW` et vérifier qu'il est correct sur les 1000 premiers entiers.

*Tests de primalité**Le crible d'Eratosthène*

Le crible d'Eratosthène est certainement la plus célèbre des manières de générer une liste de nombre premiers.

Algorithme 24 : Crible d'Eratosthène**Entrées :** Entier B **Sorties :** Liste des nombres premiers $\leq B$

```

1  $T \leftarrow [\top]^B$ 
2  $d \leftarrow 1$ 
3 tant que  $d \leq \sqrt{B}$  faire
4    $d \leftarrow d + 1$ 
5   si  $T[d]$  alors
6     pour  $i \in \llbracket 2d, 3d, \dots, B \rrbracket$  faire
7        $T[i] \leftarrow \perp$ 
8 retourner  $\{i \in \llbracket 2, B \rrbracket \text{ tels que } T[i] = \top\}$ 

```

Exercice 334. 1. Comment tester rapidement si $d \leq \sqrt{B}$?
 2. Écrire une fonction `cribleEratosthene` codant le crible décrit ci-dessus et comparer votre résultat avec une liste générée par Sage-Math.

Le test $n - 1$ de Pocklington-Lehmer

Proposition 335 (Critère de Lehmer-Pocklington). Soit $n \geq 2$ un entier dont on connaît une factorisation partielle de $n - 1$, i.e. on dispose d'entiers

f et u tels que $n - 1 = fu$ et de l'ensemble des diviseurs premiers de f . On suppose que pour tout diviseur premier q de f , il existe un entier a_q tel que

$$a_q^{n-1} \equiv 1 \pmod{n} \quad \text{et} \quad \left(a_q^{(n-1)/q} - 1\right) \wedge n = 1.$$

Alors tout facteur premier de n est congru à 1 modulo f . Si de plus, $u \leq f + 1$, alors n est premier.

Démonstration. Soit q un diviseur premier de n et v sa valuation dans $n - 1$. L'existence de a_q montre que $(\mathbb{Z}/q\mathbb{Z})^\times$ contient un élément $b_q = a_q^{(n-1)/p^v}$ d'ordre q^v . Donc q^v divise $q - 1$ qui est l'ordre du groupe $(\mathbb{Z}/p\mathbb{Z})^\times$. En combinant, $p - 1$ est encore divisible par f . En particulier, $p > u$. Si $u \leq f + 1$, $n = 1 + fu < (f + 1)^2 \leq p^2$. Mais alors, tout facteur premier de n est $> \sqrt{n}$, et n est premier. \square

Proposition 336. Soit n un entier premier et p un diviseur premier de $n - 1$. Il y a $(n - 1)(1 - 1/q)$ éléments $a \in \mathbb{Z}/n\mathbb{Z}$ tels que

$$a^{n-1} \equiv 1 \pmod{n} \quad \text{et} \quad \left(a_q^{(n-1)/q} - 1\right) \wedge n = 1.$$

Démonstration. Comme n est premier, $(\mathbb{Z}/n\mathbb{Z})^\times$ est cyclique d'ordre $n - 1$. L'application $a \mapsto a^{(n-1)/q}$ est donc une surjection vers le groupe des q racines q -ièmes de l'unité. Il y a donc $\frac{n-1}{q}$ éléments dans le noyau. \square

Le théorème suivant permet d'améliorer la zone de performance du critère de Pocklington-Lehmer.

Théorème 337 (Brillhart, Lehmer, Selfridge). Soit $n \geq 2$ un entier dont on connaît une factorisation partielle de $n - 1$, i.e. on dispose d'entiers f et u tels que $n - 1 = fu$ et de l'ensemble des diviseurs premiers de f . On suppose que $n^{1/3} \leq f < n^{1/2}$, que pour tout diviseur premier q de f , il existe un entier a_q tel que

$$a_q^{n-1} \equiv 1 \pmod{n} \quad \text{et} \quad \left(a_q^{(n-1)/q} - 1\right) \wedge n = 1$$

Soient $c_1, c_2 \in \llbracket 0, f - 1 \rrbracket$ tels que $n = c_2 f^2 + c_1 f + 1$. Alors n est premier si et seulement si $c_1^2 - 4c_2$ n'est pas un carré.

Démonstration. Supposons n composé. Un facteur p de n doit vérifier $p \equiv 1 \pmod{f}$, donc être $> n^{1/3}$: n est donc le produit de deux premiers p et q , avec disons

$$p = af + 1 \quad q = bf + 1, \quad a \leq b.$$

On doit avoir

$$c_2 f^2 + c_1 f + 1 = n = abf^2 + (a + b)f + 1$$

Montrons que $c_2 = ab$ et $c_1 = a + b$, car alors $c_1^2 - 4c_2 = (a - b)^2$ est bien un carré. Notons que $abf^2 < n \leq f^3$, donc $ab \leq f - 1$. Il s'en suit que ou $ab \leq f - 1$ ou $a = 1$ et $b = f - 1$. Ce dernier cas conduit à $n = (f + 1)((f - 1)f + 1) = f^3 - 1$ ce qui contredit $f \geq n^{1/3}$. Donc ab et $a + b$ sont deux entiers $< f$. Par unicité de la décomposition en base f , $c_2 = ab$ et $c_1 = a + b$.

Réciproquement, si $c_1^2 - 4c_2 = u^2$, alors $n = \left(\frac{c_1+u}{2} + 1\right) \left(\frac{c_1-u}{2} + 1\right)$ qui est une factorisation non-triviale. \square

Exercice 338. 1. Décrivez et programmez un algorithme qui utilise les théorèmes ci-dessus afin de prouver la primalité d'un nombre.

2. Que pensez-vous du nombre $2^{89} - 1$?

3. Utiliser `is_prime()` pour montrer que $2^{607} - 1$ est premier. Parvenez-vous à le montrer avec votre algorithme ?

Remarque 339. Grappiller suffisamment de facteurs de $n - 1$ pour appliquer le critère de Pocklington-Lehmer peut s'avérer difficile. L'un des tests de primalité modernes, le test ECPP (Elliptic Curve Primality Proving) d'Atkin et Morain, reprend ce critère avec une courbe elliptique variable au lieu de $n - 1$, ce qui permet, en jouant sur le choix de la courbe, de trouver plus facilement beaucoup de petits facteurs.

Exercice 340 (Test de Pépin). On note $F_n = 2^{2^n} + 1$ le n -ième nombre de Fermat. Les nombres F_n sont premiers pour $n \in \llbracket 0, 4 \rrbracket$; trouver d'autres nombres premiers de Fermat est un problème ouvert. Montrer que F_n est premier si et seulement si $3^{(F_n-1)/2} \equiv -1 \pmod{F_n}$.

Le test de APR-CL

Le test APR-CL est dû à Adleman, Pomerance et Rumely (1983) et a été amélioré par Cohen et Lenstra. Il est basé sur des sommes de Gauß. Sa complexité est $O((\log n)^{c \log \log \log n})$.

Ce test est trop compliqué pour vous être détaillé. En voici les ingrédients. Soit χ un caractère de Dirichlet modulo q , c'est-à-dire un homomorphisme $(\mathbb{Z}/q\mathbb{Z})^\times \rightarrow \mathbb{C}$. On note $\zeta_q = \exp(2i\pi/q)$ une racine primitive q -ième de l'unité dans \mathbb{C} . On appelle *somme de Gauß* la somme

$$\tau(\chi) = \sum_{x \in (\mathbb{Z}/q\mathbb{Z})^\times} \chi(x) \zeta_q^x.$$

Proposition 341. Soit n un entier premier et χ un caractère d'ordre premier p avec $p|q - 1$. On a la congruence

$$\tau(\chi)^n \equiv \chi(n)^{-n} \tau(\chi^n) \pmod{n}.$$

Réciproquement, si cette congruence est satisfaite pour suffisamment de caractères χ et si d'autres conditions facilement vérifiables sont satisfaites, alors n est premier.

Le test de Agrawal-Kayal-Saxena

Le test de Agrawal, Kayal et Saxena a permis (en 2002 seulement) de montrer que décider la primalité d'un entier appartient à la classe **P** des problèmes décidables en temps polynomial. Même si sa complexité a été améliorée depuis sa publication, il n'est pas utilisé en pratique car son temps de calcul reste trop long par rapport aux algorithmes précédents.

Il est issu de l'observation suivante :

Proposition 342. *Un entier n est premier si et seulement si*

$$g(x^n) = g(x)^n \quad (13)$$

pour tout polynôme de la forme $g(x) = x + a \in \mathbb{Z}/n\mathbb{Z}[x]$ (a étant une constante).

On assouplit ce critère par deux idées supplémentaires : d'une part, il est suffisant de tester les s premières valeurs de a pour s logarithmique en n ; d'autre part, on peut écraser le calcul en travaillant modulo $x^r - 1$ pour un certain petit r , et alors, n est une puissance d'un nombre premier.

Pour tout entier r , on note $\omega_r(n)$ l'ordre multiplicatif de n dans $\mathbb{Z}/r\mathbb{Z}$.

Théorème 343. *Soit n un entier. Soit r tel que $\omega_r(n) \geq 4 \log^2 n$ et $s = \lfloor 2\sqrt{\varphi(r)} \log n \rfloor$. On suppose que*

1. *n ne possède pas de facteur premier inférieur à r ,*
2. *$g(x^n) = g(x)^n \pmod{n, x^r - 1}$ pour tout $g(x) = x + a$ et $a \in \llbracket 1, s \rrbracket$.*

Alors n est une puissance d'un nombre premier.

Démonstration. Pour un certain nombre premier $q \mid \omega_r(n)$, $n^{\omega_r(n)/q} \not\equiv 1 \pmod{r}$. Par conséquent, n possède un diviseur premier p tel que $n^{\omega_r(n)/q} \not\equiv 1 \pmod{r}$. En particulier, $\omega_r(p) \geq q$.

Nous laissons au lecteur le soin de vérifier aussi que

$$n^{2\lfloor \sqrt{r} \rfloor} \leq \binom{q+s+1}{s} \quad (14)$$

Nous introduisons

$$\mathcal{T} = \{e \in \mathbb{N}; g(x^e) = g(x)^e \in \mathbb{F}_p[x]/\langle x^r - 1 \rangle \text{ pour } g(x) = x + a, a \in \llbracket 1, s \rrbracket\}$$

Par hypothèse, n appartient à \mathcal{T} . Mais p appartient aussi à \mathcal{T} , et par stabilité par la multiplication, \mathcal{T} contient le monoïde $\{n^i p^j; i, j \in \mathbb{N}^2\}$. Par le lemme des tiroirs dans $\mathbb{Z}/r\mathbb{Z}$, il existe deux couples d'indices $0 \leq i, j \leq \lfloor \sqrt{r} \rfloor$ tels que $t = n^i p^j$ et $u = n^{i'} p^{j'}$ vérifient $t \equiv u \pmod{r}$. Notons pour plus tard que

$$|t - u| \leq n^{2\lfloor \sqrt{r} \rfloor} \quad (15)$$

Dans l'immédiat, $x^t = x^u \in \mathbb{F}_p[x]/\langle x^r - 1 \rangle$ et par conséquent pour tout $a \in \llbracket 1, s \rrbracket$

$$(x+a)^t = x^t + a = x^u + a = (x+a)^u \in \mathbb{F}_p[x]/\langle x^r - 1 \rangle.$$

Soit $h \in \mathbb{F}_p[x]$ le polynôme annulateur d'une racine r -ième de l'unité sur \mathbb{F}_p : h est un diviseur de $x^r - 1$ de degré $\omega_r(p)$. Donc $\deg h \geq q$.

Il est encore vrai que pour tout $a \in \llbracket 1, s \rrbracket$,

$$(x+a)^t = (x+a)^u \in \mathbb{F}_p[x]/\langle h \rangle,$$

relation qui s'étend à tout le groupe multiplicatif G engendré par les $x+a$. On note que G satisfait

$$\binom{q+s+1}{s} \leq |G| \quad (16)$$

car G contient les $\prod_{a \in A} (x+a)$ avec $A \subseteq \llbracket 1, s \rrbracket$ et $|A| \leq q \leq \deg h$.

On peut identifier $\mathbb{F}_p[x]/\langle h \rangle$ au corps à $p^{\omega_r(p)}$. Il s'en suit que G est un groupe cyclique et possède des éléments d'ordre $|G|$. Notons g l'un d'entre eux. Avec les équations (14), (15) et (16), il vient que $|t-u| \leq |G|$. Or $g^t = g^u$. Donc $t = u$. Autrement dit $n^i p^j = n^{i'} p^{j'}$. Comme $(i, j) \neq (i', j')$, n est une puissance de p . \square

Une estimation de théorie analytique des nombres certifie que l'on peut trouver r avec $r \leq \lceil 16 \log^5 n \rceil$ et garantit que l'algorithme suivant est bien en temps polynomial.

Algorithme 25 : Test de primalité AKS

Entrées : Entier n

Sorties : n « est premier » ou « est composé »

```

1 pour  $b \in \llbracket 2, \lfloor \log_2 n \rfloor \rrbracket$  faire
2   si  $n$  est une puissance  $b$ -ième alors
3     retourner  $n$  est composé
4 Calculer le plus petit entier  $r$  tel que  $\omega_r(n) > (\log n)^2$ 
5 pour  $a \in \llbracket 1, r \rrbracket$  faire
6   si  $a \wedge n > 1$  alors
7     retourner  $n$  est composé
8 si  $n \leq r$  alors
9   retourner  $n$  est premier
10 pour  $a \in \llbracket 1, \lfloor 2\sqrt{\varphi(r)} \log n \rfloor \rrbracket$  faire
11   si  $(x+a)^n \not\equiv x^n + a \pmod{n, x^r - 1}$  alors
12     retourner  $n$  est composé
13 retourner  $n$  est premier
```

Exercice 344. Implémenter l'algorithme AKS.

TP 9 : Factorisation des entiers

Buts : Découvrir quelques techniques de factorisation des entiers

Travaux préparatoires : Exercices 353 (paradoxe des anniversaires), 362 (algorithme de Tonelli) & 377 (crible quadratique)

Évaluation du TP : Exercices 348 (divisions successives), 351 (factorisation d'un nombre B -friable), 355 (ρ de Pollard), 358 ($p-1$ de Pollard) & 379 (crible quadratique). Pour chaque algorithme de factorisation, faites des tests et comparez les temps d'exécution

Contexte

Factoriser un entier est comme terrasser un dragon : la tâche est difficile et il n'est pas déshonorant d'échouer ! Ce que nous appelons algorithme dans ce TP sont des méthodes qui, quand elles fonctionnent, fournissent une factorisation. Mais elles ne marchent pas toujours !

Comme pour les tests de primalité (cf. TP précédent), nous ne serons pas en mesure de présenter les techniques modernes de factorisation d'entiers. À ce jour, l'algorithme le plus performant est le *crible algébrique* (ou crible général de corps de nombres) dont la complexité est

$$\mathcal{L}_n\left(\frac{1}{3}, \sqrt[3]{\frac{64}{9}}\right) = O\left\{\exp\left[\left(\frac{64}{9}\log n\right)^{\frac{1}{3}}(\log\log n)^{\frac{2}{3}}\right]\right\}.$$

Typiquement, un entier de 150 chiffres décimaux peut être factorisé par cet algorithme. Un autre algorithme important de factorisation est l'algorithme ECM (*Elliptic Curve Factorisation Method*) dû à Lenstra (voir algorithme 37).

Les logiciels de calcul actuels combinent en général différents algorithmes de factorisation qui sont choisis en fonction de la taille de l'entier à factoriser et de celle des facteurs recherchés.

Exercice 345. On note, pour $c > 0$ et $\alpha \in [0, 1]$,

$$\mathcal{L}_n(\alpha, c) = O\left\{\exp\left[c(\log n)^\alpha(\log\log n)^{1-\alpha}\right]\right\}.$$

Comment qualifie-t-on un algorithme dont le temps d'exécution est $\mathcal{L}(0, c)$, $\mathcal{L}(0, 1)$, $\mathcal{L}(0, 2)$ ou $\mathcal{L}(1, c)$ sur une instance de taille $\log n$?

Exercice 346. Qu'est ce que le défi de factorisation RSA (*RSA factoring challenge*)? À ce jour, de combien de bits est composé le plus grand nombre de ce défi à avoir été factorisé?

Remarque 347. Dans ce qui suit, nous montrons souvent comment trouver un facteur non trivial g de n . Pour factoriser n totalement, il convient de répéter la procédure jusqu'à factorisation complète.

Les divisions successives

Pour factoriser un petit nombre (ce critère étant bien sûr subjectif), il est tout à fait acceptable de se contenter d'essayer les diviseurs successifs.

Algorithme 26 : Divisions successives

Entrées : Entier n

Sorties : Liste des diviseurs premiers de n

```

1  $\mathcal{F} \leftarrow \emptyset$ 
2  $d = 2$ 
3 tant que  $d^2 \leq n$  faire
4   tant que  $d|n$  faire
5      $n \leftarrow n/d$ 
6      $\mathcal{F} \leftarrow \mathcal{F} \cup \{d\}$ 
7    $d \leftarrow d + 1$ 
8 si  $n \neq 1$  alors
9    $\mathcal{F} \leftarrow \mathcal{F} \cup \{n\}$ 
10 retourner  $\mathcal{F}$ 
```

Exercice 348. 1. Coder l'algorithme de divisions successives.

2. Que pensez-vous de la ligne $d \leftarrow d + 1$?

3. Plus généralement, soient p_1, p_2, \dots, p_k les k premiers entiers premiers. On se propose d'accélérer l'algorithme en testant d'abord la divisibilité par p_1, \dots, p_k et puis en remplaçant dans l'algorithme la ligne $d \leftarrow d + 1$ par une ligne idoine (faisant décrire à d les valeurs premières à $p_1 p_2 \cdots p_k$). Quelle proportion de diviseurs teste-t-on de la sorte?

4. Implémenter le nouvel algorithme pour $k = 2$.

Définition 349. On dit qu'un nombre n est *B-friable* (en anglais *B-smooth*) si tout diviseur premier de n est inférieur à B .

Exemple 350. L'entier 60 est 5-friable mais 61 est loin de l'être.

Exercice 351. On suppose connue la liste P des nombres premiers $\leq B$. Adapter l'algorithme de division successive pour factoriser un nombre *B-friable*.

La méthode ρ de Pollard

Soit n un entier à factoriser. L'idée de la méthode de Pollard (1975) est de générer une suite récurrente dans $\mathbb{Z}/n\mathbb{Z}$ par

$$\forall k \in \mathbb{N}, \quad x_{k+1} = f(x_k) \pmod{n}$$

où f est une fonction polynomiale. Si p divise n , on a encore $x_{k+1} \equiv f(x_k) \pmod{p}$. Comme $\mathbb{Z}/p\mathbb{Z}$ est fini, la suite $(x_k \pmod{p})_{k \in \mathbb{N}}$ est ultimement périodique (représentation en forme de la lettre ρ , cf. fig. 29), autrement dit, il existe des entiers m et t tels que

$$\forall k \geq m, \quad x_{k+t} \equiv x_k \pmod{p}$$

Mais alors, sauf si $x_{k+t} = x_k$ exactement, $(x_{k+t} - x_k) \wedge n$ fournit un facteur de n . De plus, si f se comporte comme une fonction aléatoire de $\mathbb{Z}/p\mathbb{Z}$ dans lui-même, t est de l'ordre de $O(\sqrt{p})$ par le lemme du paradoxe des anniversaires.

Lemme 352. Soit $\mathbf{x} = (x_k)_{k \in \mathbb{N}}$ une suite de variables aléatoires i.i.d. selon la loi uniforme dans un ensemble à p éléments. Soit s le plus petit indice tel que $x_s = x_k$ pour un certain $k < s$ (s est aussi une v.a.). Alors $\mathbb{E}[s] = O(\sqrt{p})$.

Exercice 353. On se propose de démontrer dans le lemme ci-dessus que

$$\mathbb{E}[s] \leq 2 + \frac{\sqrt{2\pi p}}{2}.$$

1. Montrer que $\mathbb{P}[s \geq k] = \prod_{i=0}^{k-1} (1 - i/p)$.
2. Montrer que $\mathbb{P}[s \geq k] \leq e^{-\frac{(k-1)^2}{2p}}$ en utilisant la convexité de l'exponentielle.
3. Montrer que $\mathbb{E}[s] = \sum_{k=1}^{\infty} \mathbb{P}[s \geq k]$.
4. On rappelle que $\int_0^{\infty} e^{-x^2} = \frac{\sqrt{\pi}}{2}$. Montrer que

$$\mathbb{E}[s] \leq 2 + \frac{\sqrt{2\pi p}}{2}.$$

Afin de trouver t , nous pouvons recourir à l'algorithme du lièvre et de la tortue de Floyd (1967) qui consiste à tester successivement $(x_{2k} - x_k) \wedge n$ pour $k \in \mathbb{N}$. D'autres raffinements sont possibles (méthode de Brent).

Le choix de f relève d'un peu de magie noire : $f(x) = x^2 + c$ avec $c = 1$ fonctionne bien. On a en tout cas intérêt à choisir un polynôme non linéaire rapide à calculer ¹¹.

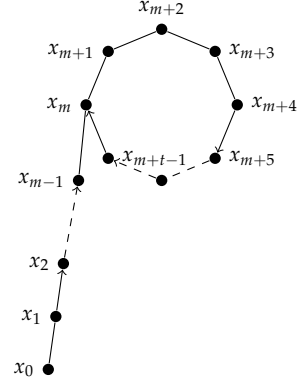


FIGURE 29: Forme en ρ de la suite $(x_k)_{k \in \mathbb{N}} \subseteq \mathbb{Z}/p\mathbb{Z}$

¹¹. Il vaut mieux éviter $c = -2$ car $(y + 1/y)^2 - 2 = y^2 + 1/y^2$.

Algorithme 27 : Méthode ρ de Pollard**Entrées** : Entier n **Sorties** : Diviseur propre de n ou « échec »

```

1 Choisir  $x \in \mathbb{Z}/n\mathbb{Z}$  aléatoirement.
2  $y \leftarrow x; \quad f \leftarrow (x \mapsto x^2 + 1)$ 
3 répéter
4    $x \leftarrow f(x); \quad y \leftarrow f(f(y)); \quad g \leftarrow (x - y) \wedge n$ 
5 jusqu'à  $g > 1$ ;
6 si  $g = n$  alors
7   retourner Echec
8 sinon
9   retourner  $g$  est un facteur

```

Exemple 354. On souhaite factoriser l'entier $n = 222763$. On initialise la suite avec $x_0 = 735$. On obtient les valeurs suivantes

i	$x = x_i$	$y = x_{2i}$	$(x - y) \wedge n$
1	94700	97147	1
2	97147	153699	1
3	185115	7148	1
4	153699	92619	1
5	34741	32049	673

ce qui révèle la factorisation $n = 331 \cdot 673$ après seulement 5 étapes. De façon générale, l'algorithme produit un résultat après un nombre d'itération compris entre 1 et 32. La figure 30 montre la distribution de fréquence d'un germe x_0 en fonction du nombre d'itération pour factoriser $n = 222763$ avec la méthode ρ de Pollard. Il y a par exemple 1120 valeurs initiales de x_0 qui conduisent à 5 itérations et en moyenne il faut 14,4 itérations pour y parvenir.

Exercice 355. Écrire une fonction `myPollardrho` qui programme l'algorithme ρ de Pollard et renvoie un facteur (1 en cas d'échec).

Méthode $p - 1$ de Pollard

Définition 356. On dit qu'un nombre est *B-ultrafriable* (en anglais *B-supersmooth*) si tout diviseur de la forme p^ν (avec p premier) est inférieur à B .

Exemple 357. L'entier $792 = 2^3 3^2 11$ est 11-ultrafriable car 2^3 , 3^2 et 11 sont ≤ 11 .

L'algorithme $p - 1$ est un algorithme inventé par John Pollard en 1974. Il repose sur l'idée suivante. Soit n un entier à factoriser et p l'un de ses facteurs premiers. Supposons que $p - 1$ divise un certain entier

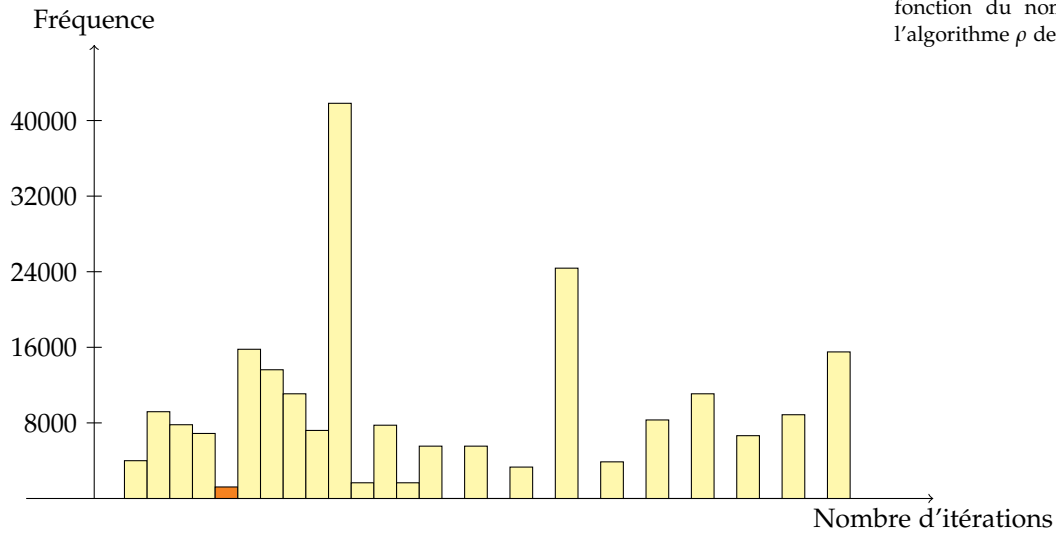


FIGURE 30: Fréquence du germe x_0 en fonction du nombre d'itérations dans l'algorithme ρ de Pollard

m . Alors, par le petit théorème de Fermat, pour tout entier a , $a^m \equiv 1 \pmod{p}$. Donc le $\text{pgcd}(a^m - 1, n)$ n'est pas trivial et peut fournir un diviseur de n .

Il nous reste à décider d'une valeur de m . Nous choisissons le ppcm de l'ensemble des entiers $\{1, 2, 3, \dots, b\}$ pour une certaine borne b . Ce choix de m permet d'attraper les facteurs premiers p de n tels que $p - 1$ est b -ultrafriable.

On obtient l'algorithme suivant :

Algorithme 28 : Méthode $p - 1$ de Pollard

Entrées : Entier n , borne de friabilité b

Sorties : Diviseur propre de n ou « échec »

```

1  $a \leftarrow$  Élément aléatoire de  $\llbracket 2, n - 1 \rrbracket$ .
2 si  $g = a \wedge n > 1$  alors
3   retourner  $g$ 
4 pour  $p$  entier premier  $\leq b$  faire
5   Calculer le plus grand entier  $\alpha_p$  tel que  $p^{\alpha_p} \leq b$ 
6   pour  $j \in \llbracket 1, \alpha_p \rrbracket$  faire
7      $a \leftarrow a^{p^j} \pmod{n}$ 
8 si  $1 < g = (a - 1) \wedge n < n$  alors
9   retourner  $g$ 
10 sinon
11   retourner Échec

```

Bilan : soit n un entier que nous cherchons à factoriser. On suppose que n se décompose en un produit $n = pq$ où p et q sont deux entiers premiers entre eux. Alors, par le lemme chinois,

$$(\mathbb{Z}/n\mathbb{Z})^\times = (\mathbb{Z}/p\mathbb{Z})^\times \times (\mathbb{Z}/q\mathbb{Z})^\times.$$

La méthode $p-1$ de Pollard révèle le facteur p lorsque l'on a obtenu dans l'algorithme les éléments $a \in (\mathbb{Z}/n\mathbb{Z})^\times$ et m tels que

$$a^m = 1 \pmod{p} \quad a^m \neq 1 \pmod{q}. \quad (17)$$

Exercice 358. Écrire une fonction `myPollardpml` qui programme l'algorithme $p-1$ de Pollard. [Indication : on peut itérer sur des nombres premiers avec `for p in primes(b):`]

Exemple 359. On fixe $p = 83$, $q = 107$ et $r = 149$. On pose $n = 1323269 = 83 \cdot 107 \cdot 149$. On choisit $b = 35$. On a

$$\begin{aligned} m &= \text{ppcm}\{1, 2, 3, \dots, 35\} \\ &= 2^6 \cdot 3^4 \cdot 5^3 \cdot 7^2 \cdot 11^2 \cdot 13^2 \cdot 17^2 \cdot 19^2 \cdot 23^2 \cdot 29^2 \cdot 31^2 \\ &= 28961647344853795740168000 \end{aligned}$$

Mais

$$\begin{aligned} (\mathbb{Z}/n\mathbb{Z})^\times &= (\mathbb{Z}/p\mathbb{Z})^\times \oplus (\mathbb{Z}/q\mathbb{Z})^\times \oplus (\mathbb{Z}/r\mathbb{Z})^\times \\ &= (\mathbb{Z}/82\mathbb{Z}) \oplus (\mathbb{Z}/106\mathbb{Z}) \oplus (\mathbb{Z}/148\mathbb{Z}) \end{aligned}$$

Notons que $p-1 = 2 \cdot 41$, $q-1 = 2 \cdot 53$ et $r-1 = 2^2 \cdot 37$. Aussi, lorsqu'on calcule a^m , on obtient en général un élément d'ordre 41 modulo p , d'ordre 53 modulo q et d'ordre 37 modulo r . Il est donc très peu probable que le pgcd $(a^m - 1 \wedge n)$ soit non trivial (en vérité cette probabilité est $\frac{1}{41} \frac{52}{53} \frac{36}{37} + \frac{40}{41} \frac{1}{53} \frac{36}{37} + \frac{40}{41} \frac{52}{53} \frac{1}{37} + \frac{1}{41} \frac{1}{53} \frac{36}{37} + \frac{1}{41} \frac{52}{53} \frac{1}{37} + \frac{40}{41} \frac{1}{53} \frac{1}{37}$ ou environ 6.86%). La méthode $p-1$ de Pollard échoue. .

Exercice 360. On fixe une borne de friabilité b . Construire quelques petits nombres premiers p tels que $p-1$ ne soit pas b -friable. Vérifier que leur produit ne peut jamais être factorisé par la méthode $p-1$ de Pollard.

Remarque 361. La méthode $p-1$ de Pollard a été revisitée par Lenstra et sa méthode de factorisation par les courbes elliptiques (voir algorithme 37). En deux mots, elle consiste à préférer travailler dans un groupe de points d'une courbe elliptique plutôt que de travailler dans le groupe $G = \mathbb{Z}/n\mathbb{Z}$ en espérant comme ici que « $G \pmod{p}$ » soit un groupe d'ordre friable.

Une méthode de crible

Racine carrée dans \mathbb{F}_p et algorithme de Tonelli

Nous savons déjà déterminer rapidement si a possède une racine carrée dans \mathbb{F}_p , autrement dit l'équation $x^2 \equiv a \pmod{p}$ possède des solutions dans \mathbb{F}_p : pour cela il suffit de calculer le symbole de Legendre en utilisant les règles de réciprocité quadratique.

Voici à présent une méthode pour calculer effectivement une racine carrée qui remonte à Gauß et à Tonelli (1891).

Algorithme 29 : Racine carrée dans \mathbb{F}_p (Tonelli)

Entrées : Premier impair p et entier a tel que $\left(\frac{a}{p}\right) = 1$
Sorties : Solution à $x^2 \equiv a \pmod{p}$

```

1 si  $p \equiv 3 \pmod{4}$  alors
2    $x \leftarrow a^{(p+1)/4} \pmod{p}$ 
3 sinon si  $p \equiv 5 \pmod{8}$  alors
4    $x \leftarrow a^{(p+3)/8} \pmod{p}$ 
5    $c \leftarrow x^2 \pmod{p}$ 
6   si  $c \neq a \pmod{p}$  alors
7      $x \leftarrow x2^{(p-1)/4} \pmod{p}$ 
8 sinon
9   Trouver  $d \in \llbracket 2, p-1 \rrbracket$  tel que  $\left(\frac{d}{p}\right) = -1$ 
10  Décomposer  $p-1 = 2^s t$  avec  $t$  impair.
11   $A \leftarrow a^t \pmod{p}$ 
12   $D \leftarrow d^t \pmod{p}$ 
13   $m = 0$ 
14  pour  $i \in \llbracket 0, s-1 \rrbracket$  faire
15    si  $(AD^m)^{2^{s-1-i}} \equiv -1 \pmod{p}$  alors
16       $m \leftarrow m + 2^i$ 
17   $x \leftarrow a^{(t+1)/2} D^{m/2} \pmod{p}$ 
18 retourner  $x$ 
```

Exercice 362. Soit p un premier impair, a et $d \in \mathbb{F}_p$ tels que $\left(\frac{a}{p}\right) = 1$ et $\left(\frac{d}{p}\right) = -1$.

- 1.(a) Que vaut $a^{(p-1)/2}$? (Utiliser la proposition 323).
 (b) A quelle condition $(p+1)/2$ est-il pair? Montrer que $a^{(p+1)/4}$ est une racine de a .
2. On note s et t tels que $p-1 = 2^s t$ avec t impair. On note $A = a^t \in \mathbb{F}_p$ et $D = d^t \in \mathbb{F}_p$.
 (a) Quel est l'ordre de D^{-1} dans \mathbb{F}_p^\times ?

- (b) Montrer que l'ordre de A divise 2^{s-1} . En déduire qu'il existe un entier $\mu \in \llbracket 0, 2^{s-1} - 1 \rrbracket$ tel que $A \equiv D^{-2\mu} \pmod{p}$.
- (c) En déduire que $a^{(t+1)/2} D^\mu$ est une racine carrée de a .
3. Montrer que la variable m de l'algorithme est égale à l'entier μ et que l'on a $AD^m \equiv 1 \pmod{p}$ en fin de boucle « Pour ».
4. Justifier l'algorithme 29.

Exercice 363. 1. Programmer l'algorithme de Tonelli `myTonelli`.

2. Vérifier que `myTonelli(a,p)` renvoie une racine carrée de a pour tout premier $3 \leq p \leq 500$ et tout $a \in \mathbb{F}_p$.

Remarque 364. À ce jour, on ne sait pas trouver d'entier d tel que $\left(\frac{d}{p}\right) = -1$ en temps polynomial. On se contente donc de faire une recherche probabiliste. Sous l'hypothèse de Riemann généralisée, on pourrait montrer qu'un tel d existe avec $d < 2 \ln p$.

Le relèvement de Hensel et racine carrée dans $\mathbb{Z}/p^\alpha\mathbb{Z}$

Supposons à présent que l'on veuille chercher des racines carrées dans $\mathbb{Z}/p^\alpha\mathbb{Z}$ plutôt que dans $\mathbb{Z}/p\mathbb{Z}$ (pour $\alpha \geq 2$). Une technique très générale de manipulation des polynômes permet d'accomplir cette tâche.

Proposition 365 (Lemme d'Hensel). Soient p un premier, α un entier, $f \in \mathbb{Z}[x]$ un polynôme et $r \in \mathbb{Z}$ tel que $f(r) \equiv 0 \pmod{p^\alpha}$. On suppose que $f'(r) \not\equiv 0 \pmod{p}$. Alors, en notant $x = f(r)/p^\alpha$, $z = [f'(r)]^{-1} \pmod{p^\alpha}$,

$$f(r') \equiv 0 \pmod{p^{2\alpha}} \text{ où } r' = r - p^\alpha xz$$

Remarque 366. Cette proposition est en fait un analogue p -adique de la méthode de Newton pour la recherche approchée d'une racine d'une fonction.

Démonstration. On a pour tout y

$$f(r + p^\alpha y) \equiv f(r) + p^\alpha y f'(r) \pmod{p^{2\alpha}}$$

Donc

$$f(r + p^\alpha y) = 0 \Leftrightarrow y = -f(r)p^{-\alpha}[f'(r)]^{-1} \pmod{p^\alpha}$$

□

Algorithme 30 : Relèvement de Hensel**Entrées** : Polynôme $f \in \mathbb{Z}[x]$, entier r tel que $f(r) \equiv 0 \pmod{p}$.Entier k **Sorties** : Entier r tel que $f(r) \equiv 0 \pmod{p^{2^k}}$ 1 **pour** $i \in \llbracket 0, k-1 \rrbracket$ **faire**2 $x \leftarrow f(r)p^{-2^i} \pmod{p^{2^i}}$ 3 $z \leftarrow [f'(r)]^{-1} \pmod{p^{2^i}}$ 4 $y \leftarrow -xz \pmod{p^{2^i}}$ 5 $r \leftarrow r + yp^{2^i}$ 6 **retourner** r

Exemple 367. On cherche une racine de $x^2 - 2886$ dans $\mathbb{Z}/5^5\mathbb{Z}$. Par l'algorithme de Tonelli, on commence obtenir les deux racines ± 1 dans $\mathbb{Z}/5\mathbb{Z}$. Des applications successives du relèvement d'Hensel donnent ± 6 comme racine dans $\mathbb{Z}/5^2\mathbb{Z}$, ± 69 dans $\mathbb{Z}/5^3\mathbb{Z}$ et ± 144306 dans $\mathbb{Z}/5^4\mathbb{Z}$. Ainsi $\pm 144306 \equiv \pm 556 \pmod{5^5}$ sont les deux racines cherchées.

Exercice 368. Soit $a \in \mathbb{Z}/p^\alpha\mathbb{Z}$ (p premier impair et $\alpha \geq 1$) tel que $\left(\frac{a}{p}\right) = 1$.

1. Combien y-a-t-il de racine carrées de a dans $\mathbb{Z}/p^\alpha\mathbb{Z}$?
2. Construire un algorithme qui renvoie l'une des racines carrées de a dans $\mathbb{Z}/p^\alpha\mathbb{Z}$.

Remarque 369 (Racine de a si a est divisible par p). Dans le cas où $a \in \mathbb{Z}/p^\alpha\mathbb{Z}$ est tel que $a \equiv 0 \pmod{p}$, a admet une racine carrée si et seulement si sa p -valuation est paire et que la partie première à p est un carré modulo p .

Exemple 370. On cherche la racine de 100 dans $\mathbb{Z}/125\mathbb{Z}$. Comme 100 est divisible par 5^2 , résoudre l'équation $x^2 \equiv 100 \pmod{125}$ revient à résoudre (avec $x = 5x'$) l'équation $x'^2 \equiv 4 \pmod{5}$. Les solutions sont donc $x = \pm 5 \times 2$. Par contre $30 = 5 \times 6$ n'admet pas de racine dans $\mathbb{Z}/125\mathbb{Z}$ car la 5-valuation de 30 est impaire.

Remarque 371 (Racine carré modulo 2^α pour $\alpha \geq 3$). Si $a \in \mathbb{Z}/2^\alpha\mathbb{Z}$ avec $\alpha \geq 3$ et a impair, il faut que $a \equiv 1 \pmod{8}$ pour que l'équation $x^2 \equiv a \pmod{2^\alpha}$ admette des solutions; il y a alors exactement 4 solutions. On les calcule en commençant en remontant une solution depuis $\mathbb{Z}/8\mathbb{Z}$ où les 4 solutions sont ± 1 et ± 3 . En particulier, ± 1 et $\pm(2^{\alpha-1} - 1)$ sont les 4 racines de l'unité. Si a est pair, on se réfère à la remarque 369.

Exemple 372. On cherche la racine de $a = 41$ modulo 64. Il y a 4 solutions modulo 8 : ± 1 et ± 3 . Montrons comment relever la solution 1 jusqu'au module $2^6 = 64$.

- On résoud l'équation $(4\epsilon + 1)^2 \equiv 41 \pmod{64}$, ce qui équivaut à $16\epsilon^2 + 8\epsilon \equiv 40$. En regardant modulo 16, on obtient $8\epsilon \equiv 8 \pmod{16}$ ce qui équivaut à $\epsilon \equiv 1 \pmod{2}$.
- On résoud ensuite l'équation $(8\epsilon' + 5)^2 \equiv 41 \pmod{64}$, ce qui équivaut à $64\epsilon'^2 + 80\epsilon' \equiv 16 \pmod{64}$ ou encore $5\epsilon' \equiv 1$. Donc $\epsilon' \equiv 1 \pmod{2}$.
- On résoud ensuite l'équation $(16\epsilon'' + 13)^2 \equiv 41 \pmod{64}$, ce qui équivaut à $32\epsilon'' \equiv 0 \pmod{64}$. Donc $\epsilon'' \equiv 0 \pmod{2}$.

Les 3 autres racines carrées de a , à savoir, 19 et, se calculent de façon analogue à partir de -1 , de 3 et de -3 . On obtient -13 et $\pm 19 \pmod{64}$ (que l'on aurait aussi pu obtenir par multiplication par les 3 racines carrées de l'unité non triviales).

Cribler un polynôme

La méthode du crible peut être utilisée tant pour chercher des nombres premiers que pour identifier des nombres B -friables : dans ce second cas, au lieu d'initialiser un tableau avec des booléens, on utilise des entiers initialisés à 1 et on multiplie par un premier p chaque case à chaque fois que son indice est divisible par une puissance de p .

Exemple 373. Supposons que l'on cherche l'ensemble des entiers 3-friables ≤ 15 . On commence avec

$$F = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$$

On cherche les multiples de 2, puis de 4 et 8 en multipliant par 2 les cases du tableau 1 fois sur 2, puis 1 fois sur 4 et enfin 1 fois sur 8

$$F = [1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2]$$

$$F = [1, 2, 1, 4, 1, 2, 1, 4, 1, 2, 1, 4, 1, 2]$$

$$F = [1, 2, 1, 4, 1, 2, 1, 8, 1, 2, 1, 4, 1, 2]$$

On cherche les multiples de 3, puis de 9 en multipliant par 3 les cases du tableau 1 fois sur 3, puis 1 fois sur 9.

$$F = [1, 2, 3, 4, 1, 6, 1, 8, 3, 2, 1, 12, 1, 2, 3]$$

$$F = [1, 2, 3, 4, 1, 6, 1, 8, 9, 2, 1, 12, 1, 2, 3]$$

On voit que $F[k] = k$ pour $k = 1, 2, 3, 4, 6, 8, 9, 12$ ce qui fournit la liste cherchée.

La méthode du crible peut être adaptée à la recherche tant de nombres premiers que de nombres friables dans une suite $(f(k))_{1 \leq k \leq K}$ générée par un polynôme f . En effet, la propriété r divise $f(k)$ pour r et k entier revient à dire que k est une racine de f dans $\mathbb{Z}/r\mathbb{Z}$. On peut donc

chercher l'ensemble des racines $(\rho_i)_{i \leq d}$ de f dans $\mathbb{Z}/r\mathbb{Z}$ puis cribler dans la liste $\llbracket 1, K \rrbracket$ les indices congrus à l'un des ρ_i .

Nous montrons comment faire pour reconnaître les nombres B -friables.

Algorithme 31 : Crible quadratique basique

Entrées : Entier K , polynôme f , borne B

Sorties : Suite des indices $k \leq K$ tels que $(f(k))$ est B -friables

```

1  $F \leftarrow [1]^K$ .
2 Générer la liste  $P$  des premiers  $\leq B$ .
3 pour  $p \in P$  et  $\alpha$  tel que  $p^\alpha \leq \max_{k \in \llbracket 1, K \rrbracket} f(k)$  faire
4   pour  $\rho$  racine de  $f(x)$  modulo  $p^\alpha$  faire
5     pour  $k \in \llbracket \rho, \rho + p^\alpha, \dots, K \rrbracket$  faire
6        $F[k] \leftarrow F[k] \cdot p$ .
7 retourner  $\{k \in \llbracket 1, K \rrbracket; f(k) = F[k]\}$ 
```

Exemple 374. On considère le polynôme $f(x) = (x + 54)^2 - 2886$ et on souhaite trouver les valeurs 11-friables dans la suite

$$(f(0), f(1), \dots) =$$

$$[30, 139, 250, 363, 478, 595, 714, 835, 958, 1083, 1210, 1339, 1470, 1603, \dots]$$

Initialement, le tableau de crible est

$$[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$$

Les racines de f dans $\mathbb{Z}/p^\alpha\mathbb{Z}$ sont $-54 \pm \sqrt{2886}$ (où la racine peut être calculée avec les algorithmes de Tonelli et de Hensel).

Pour $p = 2$, on obtient une unique racine $f(0) = 0 \pmod{2}$. Le tableau commence par

$$[2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1].$$

Il n'y a aucune racine modulo une puissance plus grande de 2 (en effet, 2886 n'est divisible qu'une seule fois par 2, voir remarque 369).

Pour $p = 3$, on obtient de même une unique racine $f(0) = 0 \pmod{3}$ et aucune autre racine pour une puissance plus grande (en effet, 2886 n'est divisible qu'une seule fois par 3, voir remarque 369). Le tableau passe à

$$[6, 1, 2, 3, 2, 1, 6, 1, 2, 3, 2, 1, 6, 1].$$

Pour $p = 5$, on obtient deux racines $f(0) = f(2) = 0 \pmod{5}$. Le tableau passe à

$$[30, 1, 10, 3, 2, 5, 6, 5, 2, 3, 10, 1, 30, 1].$$

Puis ces racines se relèvent en $f(15) = f(2) = 0 \pmod{25}$. Le tableau passe à

$$[30, 1, 50, 3, 2, 5, 6, 5, 2, 3, 10, 1, 30, 1].$$

On a encore $f(15) = f(2) = 0 \pmod{5^3}$, d'où

$$[30, 1, 250, 3, 2, 5, 6, 5, 2, 3, 10, 1, 30, 1].$$

Et enfin $f(15) = f(502) = 0 \pmod{5^4}$ ce qui ne change plus le tableau. Comme 5^5 dépasse $\max_{k \in \llbracket 0, 14 \rrbracket} f(k)$, nous passons au premier p suivant.

Pour $p = 5$, on obtient deux racines $f(5) = f(6) = 0 \pmod{7}$.

$$[30, 1, 250, 3, 2, 35, 42, 5, 2, 3, 10, 1, 210, 7].$$

etc. On finit avec le tableau

$$[30, 1, 250, 363, 2, 35, 42, 5, 2, 3, 1210, 1, 1470, 7].$$

Ce qui permet de conclure que $f(0), f(2), f(3), f(10), f(12)$ sont les seuls nombres 11-friables de la liste.

Exercice 375. On considère n un entier et on cherche les entiers B -friables parmi les K premiers termes de la suite $(x^2 - n)_{x \geq \lceil \sqrt{n} \rceil} \subseteq \mathbb{Z}$.

1. Comment adapter l'algorithme ci-dessus au cas spécifié ?
2. Programmer votre solution.

Remarque 376. En réalité, pour augmenter la vitesse de calcul, on préférera travailler avec un tableau F dans lequel on additionne des termes $\lceil \log p \rceil$ plutôt que d'effectuer des multiplications. Même en travaillant avec des flottants de basse précision et quitte à tester par division successives une seconde fois les nombres B -friables identifiés par le crible, la vitesse de calcul s'en trouve améliorée : d'une part on manipule des entiers plus petits, d'autre part additionner est toujours beaucoup plus rapide que multiplier.

Le crible quadratique, version basique

Faute de disposer du bagage en théorie des nombres algébrique pour présenter le crible général de corps de nombres (ou crible algébrique), nous présentons l'un de ses ancêtres : le crible quadratique.

L'idée de base du crible quadratique est de trouver des entiers y et z distincts tels que

$$y^2 \equiv z^2 \pmod{n} \Leftrightarrow (y - z)(y + z) \equiv 0 \pmod{n}. \quad (18)$$

Dans le cas où $y \not\equiv \pm z \pmod{n}$, le pgcd $(y - z) \wedge n$ fournit un facteur de n .

Exercice 377. 1. Soit n un entier impair divisible par exactement k diviseurs premiers. Montrer que l'équation $a^2 \equiv 1 \pmod{n}$ possède 2^k solutions.

2. Montrer que si $a \not\equiv \pm 1 \pmod n$ et satisfait $a^2 \equiv 1 \pmod n$, alors $(a-1) \wedge n$ est un diviseur non trivial de n .
3. Estimer la probabilité de calculer un facteur non trivial de n lorsque l'on dispose de deux entiers x et y premiers à n , tirés aléatoirement tels que $x^2 \equiv y^2 \pmod n$.

L'idée de base du crible quadratique est de générer une série de relations de congruence de la forme

$$x_i^2 \equiv a_i \pmod n \quad (x_i, a_i \in \mathbb{Z} \text{ distincts})$$

en tirant x_i au hasard et d'en sélectionner certaines judicieusement pour que $\prod a_i$ soit un carré, disons y^2 . Mais alors, en posant $z = \prod x_i$, on obtient

$$z^2 \equiv y^2 \pmod n \Leftrightarrow (z-y)(z+y) \equiv 0 \pmod n.$$

Exemple 378. Supposons que nous voulions factoriser $n = 2886$. Nous avons

$$\begin{cases} x_0 = 54^2 = 2916 \equiv 30 \pmod n \\ x_1 = 55^2 = 3025 \equiv 139 \pmod n \\ x_2 = 56^2 = 3136 \equiv 250 \pmod n \\ x_3 = 57^2 = 3249 \equiv 363 \pmod n \\ \vdots \end{cases}$$

Nous fixons la base de friabilité à $\{2, 3, 5, 7, 11\}$ et étudions cette suite (voir exemple 374). Il vient

$$\begin{cases} a_0 = 30 &= 2 \cdot 3 \cdot 5 \\ a_1 = 139 &= 139 \\ a_2 = 250 &= 2 \cdot 5^3 \\ a_3 = 363 &= 3 \cdot 11^2 \end{cases}$$

Comme a_1 n'est pas friable, nous le rejetons. Les exposants de a_0, a_2, a_3 sont

$$\mathbf{e}_0 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} 1 \\ 0 \\ 3 \\ 0 \\ 0 \end{pmatrix} \quad \text{et} \quad \mathbf{e}_3 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 2 \end{pmatrix}.$$

Pour trouver un carré, on résoud sur \mathbb{F}_2 le système à 3 inconnues et 5 équations

$$\kappa_0 \mathbf{e}_0 + \kappa_2 \mathbf{e}_2 + \kappa_3 \mathbf{e}_3 \equiv 0 \pmod 2.$$

Or, par chance, \mathbf{e}_0 , \mathbf{e}_2 et \mathbf{e}_3 sont liés sur \mathbb{F}_2 : en prenant $\kappa_0 = 1$, $\kappa_2 = 1$

et $\kappa_3 = 1$, on arrive à

$$\begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 3 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 4 \\ 0 \\ 2 \end{pmatrix} \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \in \mathbb{F}_2^5.$$

L'entier $30 \cdot 250 \cdot 363$ est un carré, en l'occurrence celui de

$$y = \pm 2^{(\kappa_0 + \kappa_2)/2} \cdot 3^{(\kappa_0 + \kappa_3)/2} \cdot 5^{(\kappa_0 + 3\kappa_2)/2} \cdot 11^{\kappa_3} = \pm 2 \cdot 3 \cdot 5^2 \cdot 11 = 1650.$$

Comme $z = 54 \cdot 56 \cdot 57 = 172368 \equiv 2094 \pmod{n}$, nous avons donc trouvé que

$$2094^2 \equiv 1650^2 \pmod{n}$$

Or $(2094 - 1650) \wedge n = 13$. Nous en déduisons la factorisation

$$n = 13 \cdot 222.$$

Pour trouver de tels x_i , nous énumérons $x_0 = \lceil \sqrt{n} \rceil$, $x_1 = \lceil \sqrt{n} \rceil + 1$, $x_2 = \lceil \sqrt{n} \rceil + 2$, etc. Pour déterminer un produit carré, nous ne gardons que les x_i tels que le reste a_i de la division de x_i^2 par n est B -friable pour un certain B (et qui sont donc facilement factorisables). Si nous notons p_1, \dots, p_m l'ensemble des nombres premiers $\leq B$, nous avons factorisé

$$\begin{cases} x_0^2 \equiv a_0 = p_1^{e_{1,0}} p_2^{e_{2,0}} \cdots p_m^{e_{m,0}} \\ x_1^2 \equiv a_1 = p_1^{e_{1,1}} p_2^{e_{2,1}} \cdots p_m^{e_{m,1}} \\ \vdots \\ x_k^2 \equiv a_k = p_1^{e_{1,k}} p_2^{e_{2,k}} \cdots p_m^{e_{m,k}} \end{cases}$$

Étant donné $I \subseteq \llbracket 0, k \rrbracket$, savoir si $z = \prod_{i \in I} x_i = p_1^{\sum_{i \in I} e_{1,i}} \cdots p_m^{\sum_{i \in I} e_{m,i}}$ est un carré revient à vérifier que

$$\begin{cases} \sum_{i \in I} e_{1,i} \equiv 0 \pmod{2} \\ \vdots \\ \sum_{i \in I} e_{m,i} \equiv 0 \pmod{2} \end{cases}$$

Aussi, trouver $I \subseteq \llbracket 0, k \rrbracket$ tel que z est un carré revient à trouver une relation de dépendance linéaire dans la famille $\mathcal{E} = \{\mathbf{e}_0, \dots, \mathbf{e}_m\}$ des vecteurs $\mathbf{e}_i = (e_{j,i})_{j \leq m} \in \mathbb{F}_2^m$.

On note

$$L(n) = e^{\sqrt{\ln n \ln \ln n}}.$$

qui nous servira de borne de friabilité. Cette borne est un compromis qui résulte d'estimations sur la répartition des nombres friables. Cependant, le choix d'une bonne borne de friabilité est plus un art qu'une science : avec une borne trop petite, l'algorithme ne parvient pas à trouver un facteur, avec une borne trop généreuse, l'algorithme s'exécute en un temps trop long.

Algorithme 32 : Crible quadratique basique**Entrées :** Entier n impair qui n'est pas une puissance**Sorties :** Diviseur propre de n

```

1  $B \leftarrow \lceil L(n)^{1/2} \rceil$ 
2 Calculer la liste  $p_1, \dots, p_m$  des premiers  $p \leq B$ 
3 Rechercher les  $m + 1$  premiers termes de la suite  $(a = x^2 \bmod n)_{x \geq \lceil \sqrt{n} \rceil}$  qui soient  $B$ -friables. Stocker  $(x, a)$  dans  $S$ .
4 pour  $(x, a) \in S$  faire
5    $\lfloor$  Factoriser  $a$  en  $a = p_1^{e_{1,x}} p_2^{e_{2,x}} \dots p_m^{e_{m,x}}$ 
6 Construire la matrice  $((e_{j,x}))_{1 \leq j \leq m, (x,a) \in S}$  dans  $\mathbb{F}_2^{m \times (m+1)}$ 
7 Chercher un vecteur du noyau et appeler  $K$  son support.
8  $z \leftarrow \prod_{(x,a) \in K} x \bmod n$ 
9  $y \leftarrow \sqrt{\prod_{(x,a) \in K} a} = p_1^{\frac{1}{2} \sum_{(x,a) \in K} e_{1,x}} p_2^{\frac{1}{2} \sum_{(x,a) \in K} e_{2,x}} \dots p_m^{\frac{1}{2} \sum_{(x,a) \in K} e_{m,x}} \bmod n$ 
10  $d \leftarrow (z - y) \wedge n$ 
11 retourner  $d$ 

```

Exercice 379. Écrire un programme `cribleQuadratique` qui implémente l'algorithme du crible quadratique. (Pour implémenter les lignes 3–5 du pseudo-code de l'algorithme, vous pouvez utiliser `prime_divisor` ou adapter l'exercice 351 ou l'exercice 375 .).

Remarque 380. La limitation du crible quadratique provient du fait que dans la suite des termes $a = x^2 \bmod n$, il y a potentiellement de gros facteurs premiers. On perdrait moins de temps à travailler directement avec des nombres dont on s'attend à ce qu'ils possèdent de petits facteurs. C'est ce que fait l'algorithme du crible général, qui utilise les entiers d'un *corps de nombre*, c'est-à-dire une extension de corps de \mathbb{Q} de degré fini.

TP 10 : Invariants de similitude et LFSR

Buts : Acheter l'étude des modules sur un anneau principal. Comprendre comment fonctionne la théorie spectrale sur les matrices. Comprendre ce qu'est un LFSR

Travaux préparatoires : Relire le TP 7. Cours et exercices 385 (décomposition d'un module) 391 (composantes primaires), 392 (énumération de modules), 393 (énumération de modules),

Évaluation du TP : Exercices 401 (invariants de similitude), 408 (forme de Frobenius d'une matrice), 416 (forme de Jordan d'une matrice), 455 (Berlekamp-Massey), 440 (orbites d'un LFSR), 437 (période d'un LFSR), 452 (séries génératrices).

Dans ce TP, A désigne un anneau principal.

Composantes primaires d'un A -module

Comme au TP 7, A désigne un anneau principal. Nous fixons une fois pour toute un système de représentants \mathcal{P} de l'ensemble des éléments irréductibles (ou premiers¹²) de A .

Exemple 381. Si $A = \mathbb{Z}$, \mathcal{P} correspond à l'ensemble $\{2, 3, 5, 7, 11, \dots\}$ habituel.

Définition 382. Soient A un anneau principal et $a \in A$, on appelle p -valuation de a , et on note $v_p(a)$, le plus grand entier $\alpha \in \mathbb{N}$ tel que

$$p^\alpha | a.$$

Proposition 383. Si A un anneau principal et $a \in A$, alors la décomposition de a en facteurs premiers est

$$a = u \prod_{p \in \mathcal{P}} p^{v_p(a)}.$$

Rappelons le théorème suivant :

Théorème 384 (Lemme chinois). Soient A un anneau principal et $d \in A$ un élément quelconque, alors on a l'isomorphisme d'anneaux

$$A/dA \simeq \bigoplus_{p \in \mathcal{P}} A/p^{v_p(d)}A$$

12. Un élément irréductible dans un anneau intègre est un élément qui n'est ni inversible, ni produit de deux éléments non inversibles ; un élément premier p est un élément non inversible tel que $p|ab$ implique $p|a$ ou $p|b$. Dans un anneau factoriel (notamment les anneaux principaux), ces deux notions coïncident.

Exercice 385. Les A -modules M suivants peuvent-ils être décomposés en somme de deux sous-modules propres? Énumérer toutes les possibilités.

1. $A = \mathbb{Z}, M = A/496125A$;
2. $A = \mathbb{F}_2[x], M = A/(x^3 + x^2 + x + 1)A$;
3. $A = \mathbb{F}_7[x], M = A/(x^4 + 3x^3 + 4x + 1)A$.

[Indications : $496125 = 3^4 \cdot 5^3 \cdot 7^2$, $x^3 + x^2 + x + 1 = (x + 1)^3$ et $x^4 + 3x^3 + 4x + 1 = (x - 2)(x - 3)(x^2 + x - 1)$.]

Nous avons vu qu'un module de type fini M est toujours de la forme (cf théorème 74)

$$M \simeq A^r \oplus \bigoplus_{i=1}^s A/d_i A$$

Nous pouvons continuer à décomposer chaque facteur $A/d_i A$ par le lemme chinois.

Définition 386. Avec les notations ci-dessus et pour $p \in \mathcal{P}$, on appelle *composante p -primaire* le facteur

$$M(p) = \bigoplus_{i=1}^s A/p^{v_p(d_i)} A.$$

La suite croissante $(v_p(d_i))_{i \leq s}$ s'appelle le *type* de $M(p)$.

Remarque 387. La composante p -primaire est en réalité l'ensemble des éléments x de M tels que $p^r x = 0$ pour un certain r . Ceci assure le fait que la composante p -primaire de M est intrinsèque : elle ne dépend pas de la décomposition utilisée pour la calculer.

Théorème 388. Si A est un anneau principal et M est un A -module de torsion, alors

$$M = \bigoplus_{p \in \mathcal{P}} M(p).$$

Exemple 389. Soit $M = \mathbb{Z}^2 \oplus \mathbb{Z}/5\mathbb{Z} \oplus \mathbb{Z}/10\mathbb{Z} \oplus \mathbb{Z}/150\mathbb{Z}$ donné par son rang 2 et ses facteurs invariants $5|10|150$. Alors

$$M(2) = (\mathbb{Z}/2\mathbb{Z})^2$$

est de type $(1, 1)$,

$$M(3) = (\mathbb{Z}/3\mathbb{Z})$$

est de type (1) ,

$$M(5) = (\mathbb{Z}/5\mathbb{Z})^2 \oplus (\mathbb{Z}/25\mathbb{Z})$$

est de type $(1, 1, 2)$, et

$$M_{tors} = M(2) \oplus M(3) \oplus M(5).$$

Nous pouvons terminer par un résultat d'unicité.

Théorème 390. Soit M un module de type fini sur un anneau principal A et soit

$$M \simeq A^r \oplus \bigoplus_{i=1}^s A/d_i A$$

la décomposition garantie par le théorème 87. Alors r , s et les idéaux $d_i A$ sont uniques.

De même, le type de chaque composante primaire de M est unique.

Démonstration. La partie A^r correspond au quotient M/M_{tors} : donc s est bien unique. On peut supposer que M est de torsion. Compte tenu du théorème 388, il suffit de contrôler que l'écriture des composantes p -primaires sous la forme

$$M(p) = \bigoplus_{i=1}^s A/p^{r_i} A$$

est unique (en ordonnant les exposants $r_1 \leq r_2 \leq \dots \leq r_s$). Or si $M = A/p^r A$, alors

$$p^k M / p^{k+1} M = \begin{cases} A/pA & \text{si } k < r \\ 0 & \text{si } k \geq r \end{cases}$$

Mais A/pA est un corps. Donc $p^k M / p^{k+1} M$ est un A/pA espace vectoriel de dimension $|\{i \in \mathbb{N}; r_i > k\}|$. La dimension de cet espace étant intrinsèque à M , la suite $(r_i)_{i \in \mathbb{N}}$ est bien unique. \square

Exercice 391. Soit M le module défini à l'exercice 80. Préciser les composantes primaires de M et leur type.

Exercice 392. Énumérer tous les \mathbb{Z} -modules de cardinal 6436343.

Exercice 393. Énumérer tous les \mathbb{Z} -modules de cardinal 360 et donner leurs facteurs invariants.

Exercice 394. On appelle *nombre de partitions* $p(t)$ de l'entier t le nombre de façons distinctes de décomposer t en somme. Les premières valeurs sont les suivantes :

t	$p(t)$	Décompositions
1	1	1
2	2	2 ou 1 + 1
3	3	3, 2 + 1 ou 1 + 1 + 1
4	5	4, 3 + 1, 2 + 2, 2 + 1 + 1 ou 1 + 1 + 1 + 1

Combien y-a-t-il, à isomorphisme près, de \mathbb{Z} -modules distincts de cardinal n où la décomposition en facteurs premiers de n est $n = \prod_i p_i^{\alpha_i}$?

Réduction des endomorphismes et invariants de similitude

Nous approfondissons dans cette section l'exemple 22 : E désigne un \mathbb{K} -espace vectoriel sur un certain corps \mathbb{K} , u désigne un endomorphisme E , nous munissons E d'une structure de module sur l'anneau $\mathbb{K}[x]$ grâce à la multiplication scalaire définie par :

$$\cdot : \begin{cases} \mathbb{K}[x] \times E & \rightarrow E \\ (p(x), \mathbf{v}) & \mapsto [p(u)](\mathbf{v}) \end{cases}$$

Exemple 395. Choisissons $E = \mathbb{R}^2$ et u l'endomorphisme de matrice

$$u = \begin{pmatrix} 3 & 7 \\ -4 & 5 \end{pmatrix}$$

dans la base canonique. Le produit du vecteur $\mathbf{v} = \begin{pmatrix} 2 \\ 6 \end{pmatrix}$ par le polynôme $p(x) = x^2 + 2$ conduit aux calculs suivants :

$$\begin{aligned} p(u) &= \begin{pmatrix} 3 & 7 \\ -4 & 5 \end{pmatrix}^2 + 2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} -19 & 56 \\ -32 & -3 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \\ &= \begin{pmatrix} -17 & 56 \\ -32 & -1 \end{pmatrix} \end{aligned}$$

si bien que

$$p \cdot \mathbf{v} = \begin{pmatrix} -17 & 56 \\ -32 & -1 \end{pmatrix} \begin{pmatrix} 2 \\ 6 \end{pmatrix} = \begin{pmatrix} 302 \\ -70 \end{pmatrix}.$$

Les invariants de similitude d'un endomorphisme

Toute base vectorielle $\mathcal{B} = (\mathbf{v}_i)_{i \leq n}$ de E reste une famille génératrice de E en tant que $\mathbb{K}[x]$ -module. De plus,

Lemme 396. Soit $\mathcal{B} = (\mathbf{v}_i)_{i \leq n}$ une base vectorielle de E , $((u_{i,j}))_{1 \leq i,j \leq n}$ la matrice de u dans \mathcal{B} , $\mathcal{E} = (\mathbf{e}_i)_{i \leq n}$ la base canonique de $\mathbb{K}[x]^n$ et ψ la surjection

$$\psi : \begin{cases} \mathbb{K}[x]^n & \rightarrow E \\ \mathbf{e}_i & \mapsto \mathbf{v}_i \end{cases}$$

alors $\ker \psi$ admet pour base la famille $\mathcal{F} = (\mathbf{f}_j)_{j \leq n}$ où

$$\mathbf{f}_j = x \cdot \mathbf{e}_j - \sum_{i=1}^n u_{i,j} \mathbf{e}_i \in \mathbb{K}[x]^n.$$

Démonstration. Un calcul prouve que $\mathbf{f}_j \in \ker \psi$. Montrons ensuite que tout élément \mathbf{z} de $\ker \psi$ se décompose sur \mathcal{F} . Notons

$$\mathbf{z} = \sum_{j=1}^n b_j(x) \mathbf{e}_j, \quad b_j(x) \in \mathbb{K}[x]$$

une décomposition de \mathbf{z} dans \mathcal{E} . Par l'identité $x \cdot \mathbf{e}_j = \mathbf{f}_j - \sum_i u_{i,j} \mathbf{e}_i$, on sépare les constantes et obtient une expression de la forme

$$\mathbf{z} = \sum_{i=1}^n g_i(x) \mathbf{f}_i + c_i \mathbf{e}_i$$

où $g_i(x) \in \mathbb{K}[x]$ et $c_i \in \mathbb{K}$. Comme $\psi(\mathbf{z}) = 0$,

$$\psi \left(\sum_{i=1}^n c_i \mathbf{e}_i \right) = \sum_{i=1}^n c_i \mathbf{v}_i = 0.$$

Mais alors $c_i = 0$, car \mathcal{B} est une base vectorielle de E .

Supposons enfin que \mathcal{F} ne soit pas libre mais lié par la relation

$$\begin{aligned} \sum_{i=1}^n b_i(x) \mathbf{f}_i &= 0 \\ \Leftrightarrow \sum_{i=1}^n b_i(x) x \mathbf{e}_i &= \sum_{i,j=1}^n b_i(x) u_{i,j} \mathbf{e}_j \end{aligned}$$

Alors, \mathcal{E} étant une base,

$$b_i(x) \mathbf{x} = \sum_j b_j(x) u_{i,j}$$

ce qui conduit à une contradiction sur les degrés.

□

Avec les notations du lemme, E est donc isomorphe au quotient $\mathbb{K}[x]^n / \ker \psi$. Comme $\ker \psi$ est libre de dimension n , E est bien de torsion. On note enfin que le déterminant de la famille $(\mathbf{f}_i)_{i \leq n}$ est le polynôme caractéristique χ_u de l'endomorphisme u . D'après l'exercice 95, c'est un exposant de E . Ainsi :

Théorème 397 (Hamilton-Cayley). *Le $\mathbb{K}[x]$ -module E est un module de torsion dont le polynôme caractéristique $\chi_u(x)$ est un exposant.*

Définition 398. On appelle *invariants de similitude* de u les facteurs invariants de E en tant que $\mathbb{K}[x]$ -module (voir remarque 88) et *polynôme minimal* de u le générateur unitaire de son idéal annulateur.

Remarque 399. Si l'on réinterprète le lemme, les invariants de similitude $(p_1 | p_2 | \dots | p_s)$ se calculent simplement en appliquant l'algorithme de forme normale de Smith à une matrice de $u - x \cdot \text{id} \in \mathbb{K}[x]^{n \times n}$.

Remarque 400. On déduit de la décomposition que le *polynôme minimal* de u est $\pi_u(x) = p_s(x)$ et que le *polynôme caractéristique* de u est $\chi_u(x) = \det(x \cdot \text{id} - u) = p_1 p_2 \cdots p_s(x)$.

Exercice 401. Pour chacune des matrices \mathbf{M} sur \mathbb{Q} suivantes, répondre aux questions.

1. Calculer en utilisant SageMath la forme normale de Smith de $(\mathbf{M} - x\mathbf{I}_5)$. On veillera à définir M sur \mathbb{Q} et à définir x avec `PolQQ.<x>=PolynomialRing(QQ)`.
2. En déduire les invariants de similitudes (p_1, p_2, \dots, p_s) de M .
3. Comparer le résultat avec celui de la fonction SageMath *ad hoc* (à savoir `M.rational_form(format='invariants')`).
4. Calculer le polynôme minimal de M avec `M.minimal_polynomial`. Comparer avec p_s .
5. Calculer avec SageMath $p_s(\mathbf{M})$.
6. Calculer le polynôme caractéristique de M avec `M.characteristic_polynomial`. Comparer avec le produit $p_1 p_2 \cdots p_s$.

$$\mathbf{M}_1 = \begin{pmatrix} -1841 & -10363 & 22304 & 108021 & -243809 \\ 1366 & 7695 & -16535 & -80130 & 180869 \\ -1072 & -6088 & 13069 & 63408 & -143144 \\ 506 & 2951 & -6298 & -30700 & 69343 \\ 82 & 502 & -1061 & -5214 & 11788 \end{pmatrix} \in \mathbb{Q}^{5 \times 5}$$

$$\mathbf{M}_2 = \begin{pmatrix} 570 & 1652 & -8251 & 3807 & 34007 \\ -178 & -522 & 2666 & -1196 & -10988 \\ 540 & 1573 & -7866 & 3622 & 32430 \\ -42 & -118 & 580 & -275 & -2387 \\ 135 & 393 & -1967 & 905 & 8109 \end{pmatrix} \in \mathbb{Q}^{5 \times 5}$$

$$\mathbf{M}_3 = \begin{pmatrix} 64 & -300 & 924 & -228 & 3168 \\ -80 & 404 & -1232 & 304 & -4224 \\ 35 & -175 & 543 & -133 & 1848 \\ -15 & 75 & -231 & 61 & -792 \\ -20 & 100 & -308 & 76 & -1052 \end{pmatrix} \in \mathbb{Q}^{5 \times 5}$$

Exercice 402. Soit T un endomorphisme du \mathbb{k} -espace vectoriel E . Montrer que les trois assertions suivantes sont équivalentes :

1. L'anneau $\mathbb{k}[T]$ est intègre.
2. L'anneau $\mathbb{k}[T]$ est un corps.
3. le polynôme minimal $\pi_T(x) \in \mathbb{k}[x]$ de T est irréductible.

Endomorphismes cycliques et décomposition de Frobenius

Définition 403. Soit $p(x) = x^d + \sum_{i=0}^{d-1} a_i x^i \in \mathbb{K}[x]$ un polynôme unitaire, on appelle *matrice compagnon* de p la matrice

$$\mathbf{C}(p) = \begin{pmatrix} 0 & \dots & \dots & \dots & -a_0 \\ 1 & 0 & & & -a_1 \\ 0 & 1 & 0 & & -a_2 \\ \vdots & \ddots & \ddots & & \vdots \\ 0 & \dots & 0 & 1 & -a_{d-1} \end{pmatrix}.$$

Son polynôme caractéristique est $\det(x \cdot \text{Id} - u) = p(x)$.

Proposition 404. Soit E un espace vectoriel de dimension d et u un endomorphisme de E . Sont équivalentes les assertions suivantes :

1. E est un $\mathbb{K}[x]$ -module cyclique, i.e. il existe un polynôme unitaire $p(x) = x^d + \sum_{i=0}^{d-1} a_i x^i \in \mathbb{K}[x]$ tel que $E \simeq \mathbb{K}[x]/p(x)\mathbb{K}[x]$
2. Il existe une base vectorielle de E dans laquelle l'endomorphisme u admet la matrice compagnon $\mathbf{C}(p)$ pour matrice.
3. Il existe un vecteur $\mathbf{v} \in E$ tel que $\mathbf{v}, u(\mathbf{v}), \dots, u^{d-1}(\mathbf{v})$ forme une base vectorielle de E et des scalaires $(a_i)_{0 \leq i < n} \subseteq A^n$ tels que

$$u^d(\mathbf{v}) = -a_0\mathbf{v} - a_1u(\mathbf{v}) - \dots - a_{d-1}u^{d-1}(\mathbf{v}).$$

Démonstration. 1. \Rightarrow 3. : Supposons que E est isomorphe à $\mathbb{K}[x]/p(x)\mathbb{K}[x]$.

Nous savons que $\mathcal{B} = (1, x, x^2, \dots, x^{d-1})$ est une base vectorielle du quotient $\mathbb{K}[x]/p(x)\mathbb{K}[x]$. Nous notons \mathbf{v} le vecteur de E identifié au polynôme constant 1. Alors, $\mathbf{v}, u(\mathbf{v}), \dots, u^{d-1}(\mathbf{v})$, qui est l'image de \mathcal{B} dans E , est une base de E . Enfin, comme p annule u , $u^d(\mathbf{v}) + a_{d-1}u^{d-1}(\mathbf{v}) + \dots + a_1u(\mathbf{v}) + a_0\mathbf{v} = 0$ comme voulu.

3. \Rightarrow 2. Nous exprimons l'endomorphisme u dans la base $\mathbf{v}, u(\mathbf{v}), \dots, u^{d-1}(\mathbf{v})$ et obtenons directement la matrice compagnon souhaitée.

2. \Rightarrow 1. En notant \mathbf{v} le premier vecteur de la base utilisée, on remarque que $p(x) \cdot \mathbf{v} = 0$ et qu'aucun polynôme de degré inférieur ne peut annuler \mathbf{v} . Donc le sous-module engendré par \mathbf{v} est isomorphe à $\mathbb{K}[x]/p(x)\mathbb{K}[x]$.

Pour des raisons de dimension d'espaces vectoriel, il s'agit de E tout entier. \square

Exercice 405. Soit u l'endomorphisme de \mathbb{C}^3 dont la matrice suit. Dans chacun des cas suivants, dire si le $\mathbb{C}[x]$ -module induit est cyclique ou non.

1.

$$u : \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & i \end{pmatrix},$$

2.

$$u : \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix},$$

3.

$$u : \begin{pmatrix} -1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Nous pouvons ainsi reformuler le corolaire 87.

Théorème 406 (Décomposition de Frobenius). Soit $(p_1|p_2|\dots|p_s)$ les invariants de similitude de u ¹³, alors il existe un changement de base de E tel

13. Avec SageMath, utiliser `rational_form`

que u ait pour matrice.

$$\begin{pmatrix} \mathbf{C}(p_1) & & & \\ & \mathbf{C}(p_2) & & \\ & & \ddots & \\ & & & \mathbf{C}(p_s) \end{pmatrix}.$$

Remarque 407. La suite $p_1|p_2|\cdots|p_s$ détermine uniquement la classe d'équivalence u pour la relation de similitude (dans ce contexte on l'appelle suite des *invariants de similitude*). Par ailleurs, ceci permet de prouver que si deux matrices sont semblables sur une extension de corps, elle le sont déjà sur le corps de base.

Nous avons vu (cf. remarque 63) que la théorie des modules nous permettait de classer les classes d'équivalence des matrices et des endomorphismes de A -modules. Pour la relation de *similitude* nous ne sommes capable d'y répondre que pour des matrices à valeurs dans un corps ou des endomorphismes d'espaces vectoriels car nous avons besoin d'étudier des $\mathbb{K}[x]$ -modules, qui ne sont principaux que lorsque \mathbb{K} est un corps.

Exercice 408. Quelle est la forme de Frobenius des matrices de l'exercice 401 ? Comparer avec `M.rational_form()`.

Exercice 409. Déterminer la décomposition de Frobenius de chaque classe de similitude d'un endomorphisme de $E = (\mathbb{F}_2)^4$.

Exercice 410. Les matrices ci-dessous sont elles semblables sur le corps \mathbb{Q} ? (Même question sur \mathbb{C}).

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 6 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -4 \\ 0 & 0 & 0 & 0 & 0 & 1 & 4 \end{pmatrix}, \mathbf{B} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix},$$

$$\mathbf{C} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}.$$

Exercice 411. Montrer que toutes les matrices $n \times n$ sur \mathbb{K} dont le polynôme caractéristique est $\chi(x) \in \mathbb{K}[x]$ sont semblables si et seulement

si $\chi(x)$ ne possède pas de facteurs multiples dans sa factorisation dans $\mathbb{K}[x]$.

Exercice 412. Montrer que, sur un corps \mathbb{K} ,

1. deux matrices 2×2 sont semblables si et seulement si elles ont même polynôme minimal.
2. deux matrices 3×3 sont semblables si et seulement si elles ont même polynôme minimal et même polynôme caractéristique.

Composantes primaires et décomposition de Jordan

Nous pouvons aussi nous intéresser à la décomposition en composantes primaires. On obtient dans ce cas la décomposition de Jordan. Pour simplifier la présentation, nous supposons que \mathbb{K} est algébriquement clos.

Définition 413. Soit r un entier et $\lambda \in \mathbb{K}$ un scalaire. On appelle *matrice de Jordan* la matrice

$$\mathbf{J}_{r,\lambda} = \begin{pmatrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \lambda & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & & \ddots & \ddots & 1 \\ 0 & \cdots & \cdots & 0 & \lambda \end{pmatrix} \in \mathbb{K}^{r \times r} \quad (19)$$

Proposition 414. La suite des invariants de similitude de $\mathbf{J}_{r,\lambda}$ est de longueur 1 et égale $((x - \lambda)^r)$.

Théorème 415 (Décomposition de Jordan). Il existe¹⁴ une suite de scalaires $(\lambda_i)_{i \leq t} \in \overline{\mathbb{K}}$ (éventuellement répétés) et une suite d'entiers $(r_i)_{i \leq t}$ tels que u admette

14. Avec SageMath, utiliser `jordan_form()`

$$\begin{pmatrix} \mathbf{J}_{r_1,\lambda_1} & 0 & \cdots & 0 \\ 0 & \mathbf{J}_{r_2,\lambda_2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{J}_{r_t,\lambda_t} \end{pmatrix}$$

pour matrice dans une certaine base.

Démonstration. Comme le corps est algébriquement clos, les polynômes irréductibles sont de degré 1. Nous considérons la décomposition en composantes primaires du $\mathbb{K}[x]$ -module E . Elle est de la forme

$$E \simeq \bigoplus_{i=1}^t \mathbb{K}[x] / \langle (x - \lambda_i)^{r_i} \rangle$$

où les λ_i sont éventuellement répétés. Il suffit dès lors d'exploiter la remarque. \square

- Exercice 416.** 1. Factoriser les invariants de similitude des matrices de l'exercice 401 et en déduire leur forme de Jordan.
2. Vérifier que `M.jordan_form()` renvoie le résultat que vous aviez anticipé.
3. Lesquelles des matrices sont diagonalisables ? Lesquelles ne le sont pas ?

Exercice 417. Soit M le module défini par l'exemple 22. Montrer que M est cyclique ssi le polynôme caractéristique de u coïncide avec le polynôme minimal de u .

Exercice 418. Donner toutes les formes de Jordan et leurs invariants de similitude de matrices possibles dont le polynôme caractéristique est $(x-7)^2(x+11)^3$.

Exercice 419. Donner toutes les formes de Jordan de matrices 8×8 de polynôme minimal $x^2(x-1)^3$. Préciser leurs invariants de similitude.

Exercice 420. On note α une racine de $x^3 + x + 1 \in \mathbb{F}_2[x]$ et β une racine de $x^4 + x + 1 \in \mathbb{F}_2[x]$. Reprendre l'exercice 409 et donner pour chaque endomorphisme sa réduite de Jordan.

Exercice 421. Montrer que les matrices suivantes de $(\mathbb{F}_p)^{p \times p}$ sont semblables :

$$\mathbf{M}_1 = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & & 0 & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{pmatrix} \quad \mathbf{M}_2 = \begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \ddots & \vdots \\ \vdots & \ddots & 1 & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & \cdots & 0 & 1 \end{pmatrix}.$$

Exercice 422. Montrer qu'un bloc de Jordan est semblable à sa transposée. En déduire qu'une matrice est toujours semblable à sa transposée.

Proposition 423 (Exponentiation des blocs de Jordan). Soient k, r des entiers et λ un scalaire, alors

$$\mathbf{J}_{r,\lambda}^k = \begin{pmatrix} \lambda^k & \binom{k}{1}\lambda^{k-1} & \cdots & \binom{k}{\ell}\lambda^{k-\ell} & \cdots & \binom{k}{r-1}\lambda^{k-r+1} \\ 0 & \lambda^k & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \binom{k}{\ell}\lambda^{k-\ell} \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \binom{k}{1}\lambda^{k-1} \\ 0 & \cdots & \cdots & \cdots & 0 & \lambda^k \end{pmatrix} \quad (20)$$

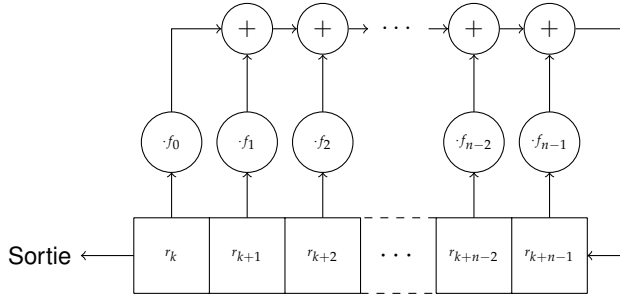
Suites récurrentes linéaires et LFSR

Définitions

Mathématiquement, l'objet que nous allons étudier n'est rien d'autre qu'une *suite récurrente linéaire*. L'usage de ces suites est très répandu dans la mesure chaque terme peut être calculé par le dispositif électronique suivant.

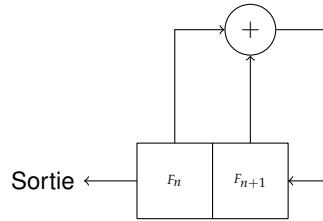
Définition 424. Soient \mathbb{F} un corps, $(f_i)_{0 \leq i < n} \in \mathbb{F}^n$ des éléments, appelés *coefficients de connexion*, et $(r_i)_{0 \leq i < n} \in \mathbb{F}^n$ des éléments, appelés *germe*. On suppose que $f_0 \neq 0$. Un *registre à décalage à rétroaction linéaire*¹⁵ à valeur dans \mathbb{F} (en anglais *linear feedback shift register*, d'où l'acronyme LFSR) est un dispositif électronique formé de n cellules synchronisées par une horloge extérieure, de n portes de multiplication et de $n - 1$ portes d'additions organisés la forme suivante :

15. Avec SageMath,
 $\mathbb{F}_2 = \text{FiniteField}(2)$
 $\text{germe} = [\mathbb{F}_2(x) \text{ for } x \text{ in } [0, 1, 0, 1]]$
 $\text{relation} = [\mathbb{F}_2(x) \text{ for } x \text{ in } [1, 0, 0, 1]]$
 $\text{lfsr_sequence}(\text{relation}, \text{germe}, 112)$
 permet de coder un LFSR (ici sur \mathbb{F}_2).



À chaque impulsion de l'horloge, les contenus du registre sont décalés d'une case vers la gauche, le nouveau coefficient étant calculé par le circuit. On note $(r_k)_{k \in \mathbb{N}} \subseteq \mathbb{F}$ la suite produite par le registre à décalage à rétroaction linéaire.

Exemple 425. La *suite de Fibonacci* $(F_n)_{n \in \mathbb{N}}$ est engendrée par le registre à décalage à rétroaction linéaire suivant avec le germe $(F_0, F_1) = (0, 1)$:



Les valeurs de la sortie du registre à décalage à rétroaction linéaire forment une suite récurrente linéaire définie par le germe et par la relation

$$\forall k \in \mathbb{N}, \quad r_{k+n} = f_0 r_k + f_1 r_{k+1} + \cdots + f_{n-1} r_{k+n-1}. \quad (21)$$

Définition 426. Étant donnée la suite récurrente linéaire définie par l'équation 21 et produite par le registres à décalage à rétroaction linéaire de la définition 424, on appelle *polynôme caractéristique* de la suite ou *polynôme de rétroaction* du LFSR le polynôme

$$\chi(x) = x^n - f_{n-1}x^{n-1} - \dots - f_1x - f_0 \in \mathbb{F}[x]$$

et *polynôme de connection* le polynôme réciproque

$$c(x) = x^n \cdot \chi(1/x) = 1 - f_{n-1}x - \dots - f_1x^{n-1} - f_0x^n \in \mathbb{F}[x].$$

Proposition 427. En notant

$$\mathbf{v}_k = \begin{pmatrix} r_k \\ r_{k+1} \\ \vdots \\ r_{k+n-1} \end{pmatrix} \in \mathbb{K}^n \quad \text{et} \quad \mathbf{C} = \begin{pmatrix} 0 & 1 & & \\ 0 & \ddots & \ddots & \\ & & 0 & 1 \\ f_0 & f_1 & & f_{n-1} \end{pmatrix},$$

la suite $(\mathbf{v}_k)_{k \in \mathbb{N}}$ obtenue est une suite géométrique de raison \mathbf{C} .

$$\forall k \in \mathbb{N}, \quad \mathbf{v}_k = \mathbf{C}^k \mathbf{v}_0. \quad (22)$$

Remarque 428. En général, un registre à décalage à rétroaction linéaire est défini sur \mathbb{F}_2 . Les additions sont alors des portes XOR et les multiplications reviennent à cabler ou non les sorties du registre.

Théorème 429. Soient $\lambda_1, \dots, \lambda_m$ les racines distinctes du polynôme caractéristique χ d'une suite linéaire récurrente $(r_k)_{k \in \mathbb{N}}$ et α_i leur multiplicité. Alors il existe des polynômes $A_i(x) \in \mathbb{F}[x]$ de degré $\deg A_i < \alpha_i$ tels que

$$\forall k \in \mathbb{N}, \quad r_k = A_1(k)\lambda_1^k + \dots + A_m(k)\lambda_m^k.$$

Démonstration. D'après l'équation 22, la suite du contenu du registre est une suite géométrique de raison $\mathbf{C} = \mathbf{C}(\chi)'$ (transposée de la matrice compagnon de χ). La forme de Jordan de \mathbf{C} est $\mathbf{J} = \text{diag}(J_{\alpha_i, \lambda_i})$. Soit $\mathbf{U} \in \text{GL}_n(\mathbb{F})$ telle que $\mathbf{C} = \mathbf{U}\mathbf{J}\mathbf{U}^{-1}$. On a alors $\mathbf{v}_k = \mathbf{U}\mathbf{J}^k\mathbf{U}^{-1}\mathbf{v}_0$. La proposition 423 permet de conclure. \square

Remarque 430 (Problème de Skolem). Aussi étonnant que cela puisse paraître, alors que l'on dispose d'une expression du terme général d'une suite linéaire récurrente, on ne sait pas si déterminer si la suite s'annule à un certain rang ou si la suite reste positive est un problème décidable ou non (i.e. s'il existe un algorithme qui répond à la question). Ce problème, dit *problème de Skolem*, et ses variantes ont pourtant une portée très large (par ex. en théorie des nombres pour de l'approximation diophantienne, en biologie pour l'extinction d'une population en biologie, en probabilité pour atteignabilité dans des chaînes de Markov, en calculs quantiques pour des problèmes de seuils dans des automates quantiques, etc).

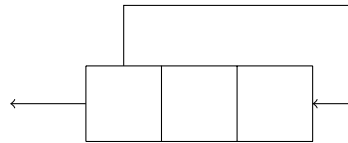
Périodicité et LFSR maximaux

Les registres à décalage à rétroaction linéaire sur des corps finis produisent forcément des suites périodiques puisque le registre ne peut parcourir qu'un nombre fini d'états.

Attention. Dans ce qui suit nous parlerons successivement de la période du polynôme de rétroaction puis de la période d'une suite.

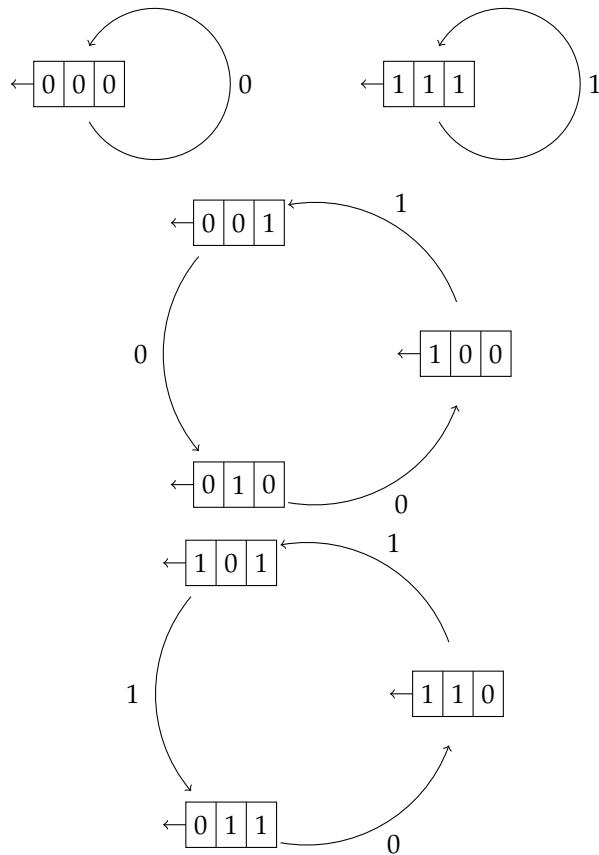
Exemple 431. Nous étudions dans cet exemple le fonctionnement dans le détail de tous les registres à décalage à rétroaction linéaire à 3 cellules sur \mathbb{F}_2 . Il y en a 4.

1. Ce premier LFSR

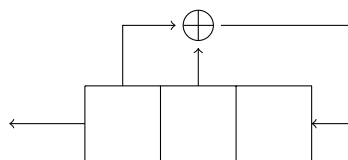


a pour polynôme de rétroaction $\chi_1 = x^3 + 1 = (x + 1)(x^2 + x + 1) \in \mathbb{F}_2[x]$.

Il engendre ultimement 4 suites différentes : $[0]^\infty$, $[1]^\infty$, $[1, 0, 0]^\infty$ et $[1, 1, 0]^\infty$ dont les périodes respectives sont 1, 1, 3 et 3 qui correspondent au contenu du registre suivant.

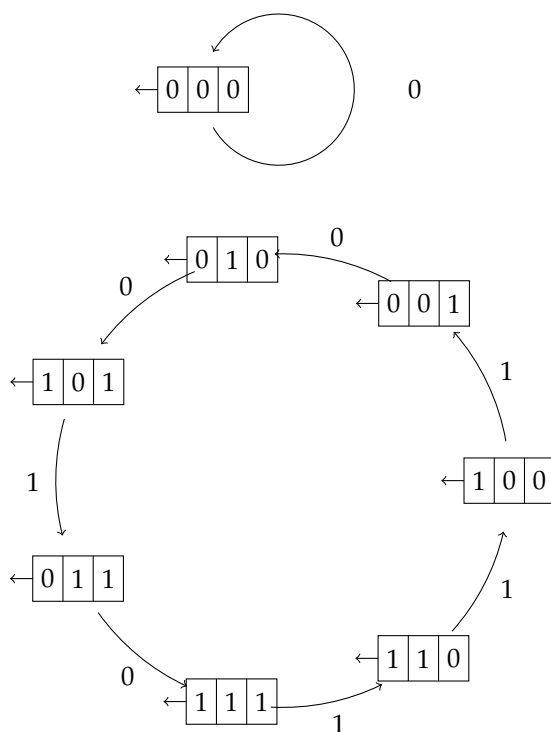


2. Ce deuxième LFSR

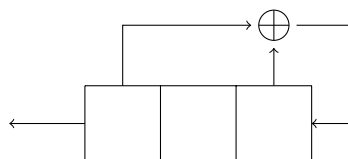


a pour polynôme de rétroaction $\chi_2 = x^3 + x + 1 \in \mathbb{F}_2[x]$.

Il engendre ultimement 2 suites différentes : $[0]^\infty$, $[1, 0, 0, 1, 0, 1, 1, 0]^\infty$, dont les périodes respectives sont 1 et 7 qui correspondent au contenu du registre suivant.

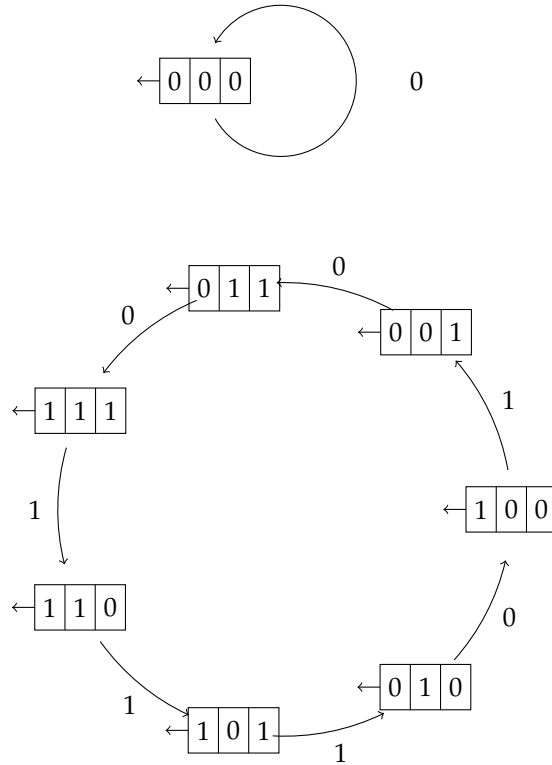


3. Ce troisième LFSR

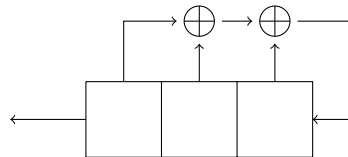


a pour polynôme de rétroaction $\chi_3 = x^3 + x^2 + 1 \in \mathbb{F}_2[x]$.

Il engendre ultimement 2 suites différentes : $[0]^\infty$, $[1, 0, 0, 1, 1, 1, 0]^\infty$, dont les périodes respectives sont 1 et 7 qui correspondent au contenu du registre suivant.

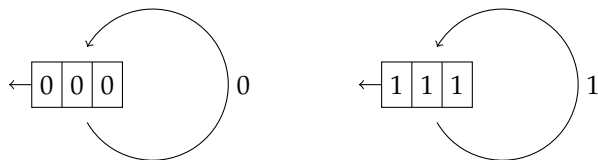


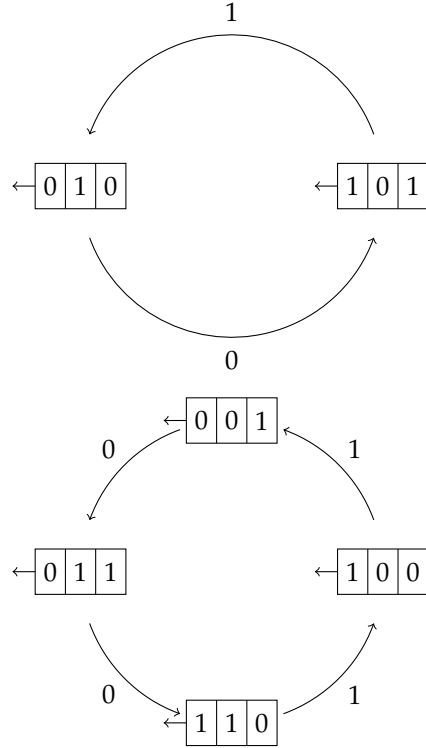
4. Ce quatrième LFSR



a pour polynôme de rétroaction $\chi_4 = x^3 + x^2 + x + 1 = (x + 1)^3 \in \mathbb{F}_2[x]$.

Il engendre ultimement 4 suites différentes : $[0]^\infty$, $[1]^\infty$, $[10]^\infty$ et $[1001]^\infty$ dont les périodes respectives sont 1, 1, 2 et 4 qui correspondent au contenu du registre suivant.





L'exemple permet de constater qu'avec un même nombre de registres, il est possible d'engendrer des suites de périodes variables.

Définition 432. Nous noterons

$$m_1(t_1) \oplus m_2(t_2) \oplus \cdots \oplus m_\ell(t_\ell)$$

le fait qu'un registre à décalage à rétroaction linéaire produise m_i orbites de période t_i pour $i \leq \ell$.

Exemple 433. En reprenant l'exemple 431, les périodes du premier LFSR se notent :

$$2(1) \oplus 2(3)$$

celles du second

$$1(1) \oplus 1(7)$$

celles du troisième

$$(1) \oplus (7)$$

et celles du dernier

$$2(1) \oplus (2) \oplus (4).$$

La matrice \mathbf{C} est la matrice de l'endomorphisme M_x de multiplication par x dans $\mathbb{K}[x]/(\chi(x))$

$$M_x : \begin{cases} \mathbb{K}[x]/(\chi(x)) & \rightarrow & \mathbb{K}[x]/(\chi(x)) \\ p(x) & \mapsto & x \cdot p(x) \end{cases}$$

exprimée dans la base $\mathcal{B} = (1, x, \dots, x^{n-1})$. Aussi, en notant $v_0(x) = \sum_{i=0}^{n-1} r_i x^i$, \mathbf{v}_k correspond aux coordonnées de $x^n v_0(x)$ cette même base \mathcal{B} .

Définition 434. On appelle *période* du polynôme χ l'ordre de \mathbf{C} ou encore le plus petit entier t tel que $\chi(x) \mid x^t - 1$. Par convention, la période est $+\infty$ si t n'existe pas.

Exemple 435. Les polynômes de l'exemple 431 ont pour période :

i	χ_i	t
1	$x^3 + 1$	3
2	$x^3 + x + 1$	7
3	$x^3 + x^2 + 1$	7
4	$x^3 + x^2 + x + 1$	4

Remarque 436. Lorsqu'on connaît une borne n sur l'exposant d'un groupe G , on peut rechercher l'ordre d'un élément g de ce groupe plus rapidement que par une recherche naïve par l'algorithme « pas de bébé, pas de géant » de Shanks : On pose $m = \lceil \sqrt{n} \rceil$ et on cherche $0 \leq i, j \leq m$ tels que $g^{im+j} = 1_G$. Pour cela, on stocke $(g^j)_{0 \leq j < m}$ dans une table et on teste successivement si $(g^{-m})^i$ appartient à cette table (pour $1 \leq i < m$).

Algorithme 33 : Algorithme « Baby step, giant step »

Entrées : Élément g d'un groupe G d'exposant $\leq n$

Sorties : Ordre de g

```

1  $m \leftarrow \lceil \sqrt{n} \rceil$ 
2  $T \leftarrow \emptyset$ 
3  $e \leftarrow 1$ 
4 pour  $j \in \llbracket 0, m-1 \rrbracket$  faire
5    $T \leftarrow T \cup \{e\}$ 
6    $e \leftarrow e \cdot g$ 
7   si  $j > 0$  et  $e = 1_G$  alors
8     retourner  $j$ 
9  $h \leftarrow g^{-m}$ 
10  $\gamma \leftarrow h$ 
11 pour  $i \in \llbracket 1, m-1 \rrbracket$  faire
12   pour  $j \in \llbracket 0, m-1 \rrbracket$  faire
13     si  $\gamma = T[j]$  alors
14       retourner  $im + j$ 
15    $\gamma \leftarrow \gamma \cdot h$ 
```

Exercice 437. 1. Quelle borne peut-on choisir sur l'exposant du groupe multiplicatif de $\mathbb{F}_q[x] / \langle \chi(x) \rangle$?

2. Adapter l'algorithme pas de bébé, pas de géant au calcul de la période d'un polynôme. On pourra travailler dans `PolynomialRing(FiniteField(q)).quotient(chi)`.
3. Quelle est la période des polynômes suivant ?
 - (a) $\chi_1 = 1 + x + x^2 + x^3 + x^4 \in \mathbb{F}_2[x]$,
 - (b) $\chi_2 = 1 + x + x^2 + x^4 \in \mathbb{F}_2[x]$,
 - (c) $\chi_3 = 1 + x^3 + x^6 \in \mathbb{F}_2[x]$,
 - (d) $\chi_4 = 3 + x^2 \in \mathbb{F}_5[x]$,
 - (e) $\chi_5 = 3 + 3x + x^2 \in \mathbb{F}_5[x]$.

Proposition 438 (Périodicité). Soit $(r_k)_{k \in \mathbb{N}}$ une suite récurrente linéaire d'ordre n de polynôme caractéristique χ . On appelle t la période du polynôme χ . Alors

1. Quelque soit le germe \mathbf{v}_0 , la suite des nombres produits $(r_k)_{k \in \mathbb{N}}$ est périodique de période divisant t ,
2. Quelque soit le germe \mathbf{v}_0 , la suite des états du registre $(\mathbf{v}_k)_{k \in \mathbb{N}}$ est périodique de période divisant t ,

De plus, il existe un germe tel que le LFSR produit une suite périodique de période t .

Exemple 439. Le tableau suivant illustre la proposition avec les LFSR de l'exemple 431.

i	χ_i	Période de χ_i	Période des suites
1	$x^3 + 1$	3	1, 3
2	$x^3 + x + 1$	7	1, 7
3	$x^3 + x^2 + 1$	7	1, 7
4	$x^3 + x^2 + x + 1$	4	1, 2, 4

Exercice 440. 1. Ecrire un programme `periode` calculant la période d'un LFSR en fonction de son germe et de ses coefficients de rétroaction.

2. Pour chaque polynôme χ de l'exercice 437, simuler le LFSR de polynôme de rétroaction χ sur l'ensemble \mathbb{F}_q^n des germes possibles. Regrouper les germes par orbites identiques.

Exercice 441. Exprimer les résultats de l'exercice 440 dans la notation de la définition 432.

Proposition 442. Sur le corps fini \mathbb{F}_q , un LFSR de polynôme de rétroaction $\chi = \psi^\ell$, où ψ est irréductible, génère l'ensemble des orbites

$$1(1) \oplus \frac{q^d - 1}{t_1}(t_1) \oplus \frac{q^d(q^d - 1)}{t_2}(t_2) \oplus \dots \oplus \frac{q^{(\ell-1)d}(q^d - 1)}{t_\ell}(t_\ell)$$

où $d = \deg \psi$ et t_i est la période de ψ^i .

Utiliser `cartesian_product_iterator` sur $[F]^n$ pour construire la puissance n -ième de \mathbb{F} .
On pourra créer une structure de donnée « Union-Find » avec `DisjointSet` et joindre deux germes avec la méthode `union`.

Exemple 443. On peut illustrer la proposition ainsi

χ	Période de χ	Période des suites
$x + 1$	1	$2(1)$
$(x + 1)^2$	2	$2(1) \oplus (2)$
$(x + 1)^3$	4	$2(1) \oplus (2) \oplus (4)$
$x^2 + x + 1$	3	$(1) \oplus (3)$
$x^3 + x + 1$	7	$(1) \oplus (7)$
$x^3 + x^2 + 1$	7	$(1) \oplus (7)$

Proposition 444. Soient ψ et ψ' deux polynômes premiers entre eux dont les ensembles d'orbites sont

$$m_1(t_1) \oplus m_2(t_2) \oplus \cdots \oplus m_\ell(t_\ell) \quad \text{et}$$

$$m'_1(t'_1) \oplus m'_2(t'_2) \oplus \cdots \oplus m'_{\ell'}(t'_{\ell'}),$$

alors l'ensemble des orbites de $\psi\psi'$ est

$$\bigoplus_{i,i'} m_i m_{i'} \operatorname{pgcd}(t_i, t_{i'}) \left(\operatorname{ppcm}(t_i, t_{i'}) \right)$$

Exemple 445. Le LFSR de polynôme de rétroaction $\chi = x^3 + 1 = (x + 1)(x^2 + x + 1)$ génère des suites de périodes

$$2(1) \oplus 2(3).$$

Exercice 446. Vérifier les propositions ci-dessus en étudiant les périodes de χ_1^2 et de $\chi_1\chi_2$ (où les polynômes sont tirés de l'exercice 437).

Dans le cadre d'applications cryptographiques, on souhaite construire des suites à valeur dans un corps fini aussi aléatoires que possible, donc de grande période.

Définition 447. On dit qu'un LSFR de longueur n à valeur dans \mathbb{F}_q est *maximal* s'il produit une suite de période $q^n - 1$ quelque soit le germe non nul utilisé.

Théorème 448. Un LFSR sur un corps fini \mathbb{F}_q de polynôme caractéristique $\chi \in \mathbb{F}_q[x]$ est maximal si et seulement si χ est un polynôme primitif de $\mathbb{F}_q[x]$.

Démonstration. Compte tenu de la proposition 438, un LFSR est maximal si et seulement si x est d'ordre $q^n - 1$ dans $\mathbb{F}[x]/(\chi(x))$ ce qui est précisément la définition d'un polynôme primitif. \square

Exercice 449. Quelle est la période maximale d'un LFSR de longueur $n = 4$ et à valeur dans \mathbb{F}_2 ? Dessiner tous les LFSR maximaux.

L'algorithme de Berlekamp-Massey

Définition 450. Soit $r = (r_k)_{k \in \mathbb{N}}$ une suite récurrente linéaire d'ordre n . On appelle *série génératrice* de la suite r la série formelle

$$R(X) = \sum_{k \in \mathbb{N}} r_k x^k \in \mathbb{F}[[x]].$$

Proposition 451. La série génératrice d'une suite récurrente linéaire est une fraction rationnelle de la forme

$$R(X) = \frac{p(x)}{c(x)} \in \mathbb{F}(x),$$

où p un polynôme de degré $< n$ dépendant linéairement du germe \mathbf{v}_0 et c le polynôme de connexion (voir définition 426).

Démonstration. La relation (21) entre les coefficients de la série équivaut à dire que tous les termes de degré $> n$ de

$$R(x) - f_0 x^n R(x) + f_1 x^{n-1} R(x) + \cdots + f_{n-1} x R(x)$$

sont nuls, autrement dit que $c(x) \cdot R(x)$ est un polynôme p de degré $< n$. \square

Exercice 452. Quelle suite possède la série génératrice suivante. Dessiner le LFSR correspondant.

Utiliser par exemple `PowerSeriesRing`.

1. $R(x) = \frac{x}{1+x^2} \in \mathbb{F}_2[[x]],$
2. $R(x) = \frac{x^2}{1+x^2} \in \mathbb{F}_2[[x]]$ (voyez-vous un lien avec la suite précédente?),
3. $R(x) = \frac{1}{1+x+x^2} \in \mathbb{F}_2[[x]],$
4. $R(x) = \frac{1+x}{1-x^3} \in \mathbb{F}_2[[x]]$

Étant donné une suite linéaire récurrente $r = (r_k)_{k \in \mathbb{N}}$ d'ordre $\leq n$ et à valeur dans un corps quelconque \mathbb{F} , il est possible de retrouver le LFSR minimal ayant produit la suite r grâce à l'algorithme de Berlekamp-Massey et en utilisant seulement les $2n$ premiers termes de la suite. Il est par conséquent impératif de combiner plusieurs LFSR de manière non-linéaire en vue d'applications cryptographiques.

L'algorithme fonctionne ainsi : on suppose que $c(x) = c_0 + c_1 x + \cdots + c_\ell x^\ell$ est le polynôme de connexion d'un LFSR capable de produire la suite (r_0, \dots, r_k) et on cherche à ajuster $c(x)$ jusqu'à ce qu'il produise réellement la suite $(r_0, \dots, r_k, r_{k+1})$

Algorithme 34 : Algorithme de Berlekamp-Massey**Entrées :** Suite $\mathbf{r} = (r_0, r_1, \dots, r_{n-1})$ de n termes dans \mathbb{F}_q **Sorties :** Polynôme de connexion $c(x) \in \mathbb{F}_q[x]$ et longueur ℓ du plus court LFSR générant \mathbf{r}

```

1  $c(x) = 1 \in \mathbb{F}_q[x], \ell = 0,$ 
2  $c^*(x) = 1 \in \mathbb{F}_q[x], d^* = 1 \in \mathbb{F}_q, m = -1.$ 
3 pour  $k \in \llbracket 0, n-1 \rrbracket$  faire
4   Calculer la discrédance :  $d \leftarrow r_k + \sum_{i=1}^{\ell} c_i r_{k-i} \in \mathbb{F}_q$ 
5   si  $d \neq 0$  alors
6      $t(x) \leftarrow c(x)$ 
7      $c(x) \leftarrow c(x) - d \cdot (d^*)^{-1} \cdot c^*(x) \cdot x^{k-m}$ 
8     si  $\ell \leq k/2$  alors
9        $\ell \leftarrow k+1-\ell$ 
10       $c^*(x) \leftarrow t(x)$ 
11       $d^* \leftarrow d$ 
12       $m \leftarrow k$ 
13 retourner  $\langle c(x), \ell \rangle$ 

```

Théorème 453. L'algorithme de Berlekamp-Massey (algorithme 34) renvoie le polynôme de connexion du plus court LFSR produisant la suite \mathbf{r} .

Démonstration. Montrons par récurrence sur k que, à l'issue de l'itération k dans la boucle « for », $(c(x), \ell)$ décrit un LFSR capable de produire la suite $(r_0, \dots, r_k, r_{k+1})$. On peut s'assurer que c'est bien le cas pour $k = -1$ en phase d'initialisation.

Nous notons $c^{(k)}(x)$, $\ell^{(k)}$ et $d^{(k)}$ les valeurs de c , ℓ et d à l'entrée dans la boucle. Par hypothèse, nous savons que,

$$\forall j \in \llbracket \ell^{(k)} + 1, k \rrbracket, \quad r_j = - \sum_{i=1}^{\ell_k} c_i r_{j-i}.$$

Si la discrédance $d^{(k+1)} = r_k + \sum_{i=1}^{\ell} c_i^{(k)} r_{k-i}$ est nulle, le même LFSR produit bien la suite $(r_0, \dots, r_k, r_{k+1})$. Sinon, on cherche à ajuster $c(x)$ en choisissant c de la forme

$$c^{(k+1)}(x) = c^{(k)}(x) + \lambda \cdot x^s \cdot c^{(t-1)}(x)$$

où λ est une constante, t le dernier rang tel que la discrédance $d^{(t)}$ soit non nulle et s un entier qui permet de faire glisser $c^{(t-1)}(x)$ à l'endroit dans la suite r où la discrédance $d^{(t)}$ était non nulle. Avec ce polynôme, la nouvelle discrédance est

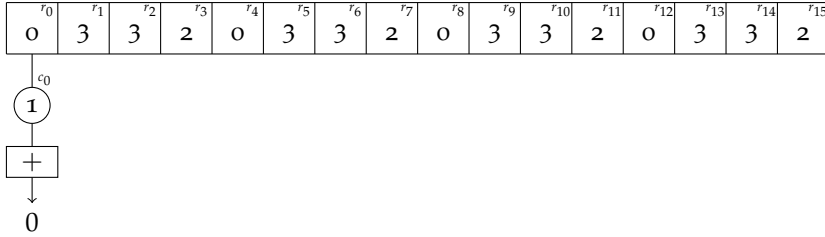
$$d^{(k+1)'} = d^{(k+1)} + \lambda d^{(m)}$$

pourvu que l'on prenne $s = k+1-t$. Il est clair que le choix $\lambda = d^{(k+1)} \cdot (d^{(m)})^{-1}$ nous amène à nos fins.

Nous admettons qu'il s'agit du plus court. \square

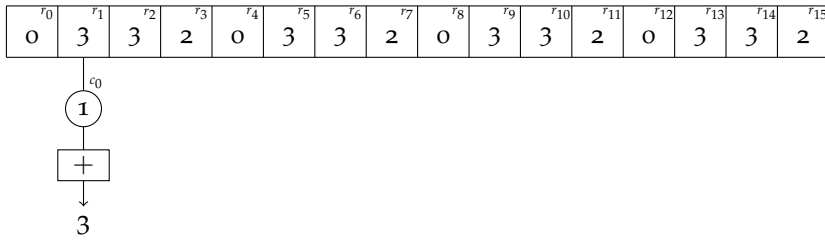
Exemple 454. Soit la suite $r = [0, 3, 3, 2, 0, 3, 3, 2, 0, 3, 3, 2, 0, 3, 3, 2]$ de \mathbb{F}_5 à étudier. On initialise à $c(x) = 1$. Voici alors le détail du fonctionnement de l'algorithme.

— Cas $k = 0$. Le polynôme $c(x)$ vaut 1 en entrée de boucle.



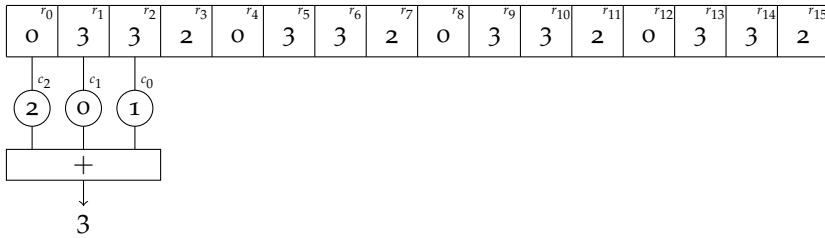
La discrédance étant nulle, il n'y a rien à faire.

— Cas $k = 1$. Le polynôme $c(x)$ vaut 1 en entrée de boucle.



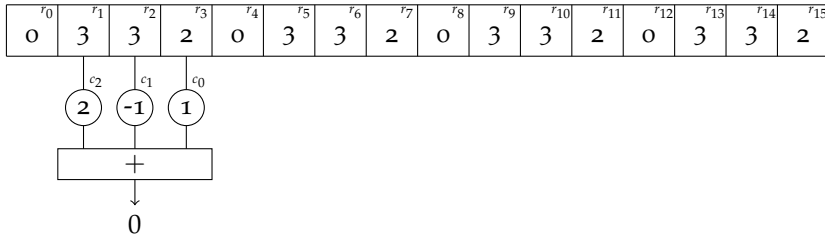
La discrédance est 3. On retranche $3 \cdot x \cdot x$ à $c(x)$. Le polynôme $c(x)$ vaut désormais $2x^2 + 1$. Par ailleurs, $c^*(x)$ vaut 1 et $d^* = 3$.

— Cas $k = 2$. Le polynôme $c(x)$ vaut $2x^2 + 1$ en entrée de boucle.



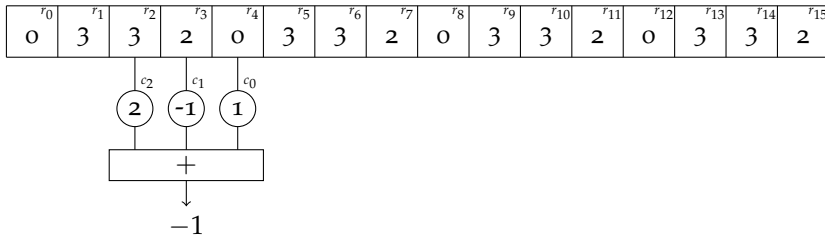
La discrédance est 3. On retranche $3 \cdot 3^{-1} \cdot x$ à c . Le polynôme $c(x)$ vaut désormais $2x^2 - x + 1$. Par ailleurs, $c^*(x)$ vaut 1 et $d^* = 3$.

— Cas $k = 3$. Le polynôme $c(x)$ vaut $2x^2 - x + 1$ en entrée de boucle.



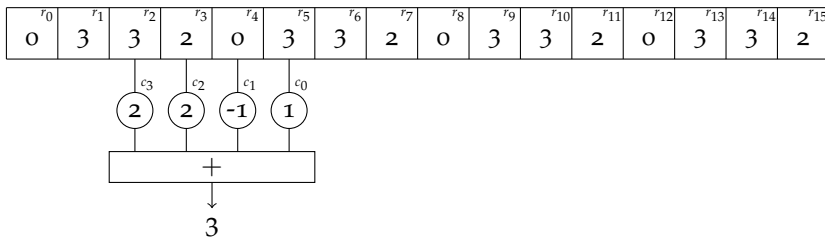
La discr pance  tant nulle, il n'y a rien   faire.

— Cas $k = 4$. Le polyn me $c(x)$ vaut $2x^2 - x + 1$ en entr e de boucle.



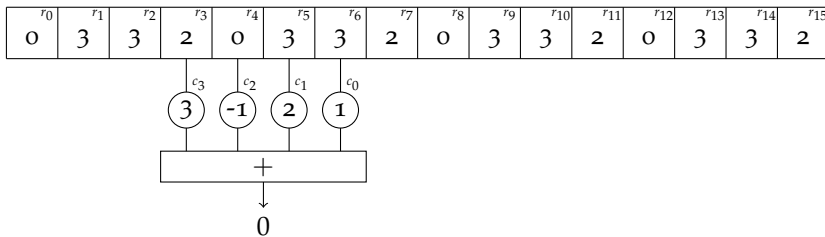
La discr pance est -1 . On retranche $3 \cdot x^2 \cdot x$   c . Le polyn me $c(x)$ vaut desormais $2x^3 + 2x^2 + 4x + 1$. Par ailleurs, $c^*(x)$ vaut $2x^2 + 4x + 1$ et $d^* = 4$.

— Cas $k = 5$. Le polyn me $c(x)$ vaut $2x^3 + 2x^2 - x + 1$ en entr e de boucle.



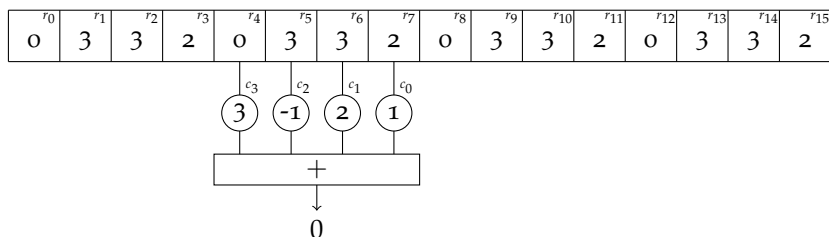
La discr pance est 3. On retranche $2 \cdot x (2x^2 - x + 1)$   c . Le polyn me $c(x)$ vaut desormais $3x^3 + 4x^2 + 2x + 1$. Par ailleurs, $c^*(x)$ vaut $2x^2 - x + 1$ et $d^* = 4$.

— Cas $k = 6$. Le polyn me $c(x)$ vaut $3x^3 - x^2 + 2x + 1$ en entr e de boucle.



La discrédance étant nulle, il n'y a rien à faire.

- Cas $k = 7$. Le polynôme $c(x)$ vaut $3x^3 - x^2 + 2x + 1$ en entrée de boucle.



La discrédance étant nulle, il n'y a rien à faire.

- Cas $k = 8$ et suivants. La discrédance restant toujours nulle, la valeur de c n'évolue plus.

Le polynôme de connexion est donc $3x^3 - x^2 + 2x + 1$.

Exercice 455. 1. Écrire une implémentation `myBerlekampMassey` de l'algorithme 34 de Berlekamp-Massey. On pourra tester son algorithme en comparant avec la fonction `berlekamp_massey` de SageMath. Par exemple avec le code

Avec SageMath, $p[i]$ désigne le coefficient de x^i si $p = \sum_i p_i x^i$ est un polynôme.

```
Fq = FiniteField(q)
for _ in range(100):
    t=randint(1,10)
    r = [Fq.random_element() for _ in range(t)]
    r = r+r+r+r
    p1 = berlekamp_massey(r)
    p2 = myBerlekampMassey(r)
    if p1.reverse() - p2 != 0:
        print "Erreur"
```

2. Trouver le plus petit LFSR qui engendre la suite $[101100001]^\infty$ sur \mathbb{F}_2 .

Cryptographie symétrique à base de LFSR

Définition 456. La *cryptographie symétrique* permet à deux interlocuteurs, Alice et Bob, de s'échanger des messages sans qu'aucun espion (Ève) ne puisse accéder à ce message. Elle se fait selon le protocole suivant, résumé à la figure 31.

Etape 0 (Génération de clés) Alice et Bob conviennent d'un secret commun s qui leur permet de générer une clé $K = \text{GenClé}(s)$.

Etape 1 (Chiffrement) Alice chiffre son *clair* m avec K . Elle envoie le *chiffré* $c = \text{Chiffrement}(m, K)$ à Bob.

Etape 2 (Déchiffrement) Bob déchiffre le chiffré c d'Alice avec la même clé K . Il obtient le clair $m = \text{Déchiffrement}(c, K)$.

Attaque passive Ève écoute toutes les transmissions effectuées entre Alice et Bob et connaît les trois fonctions employées par Alice et Bob.

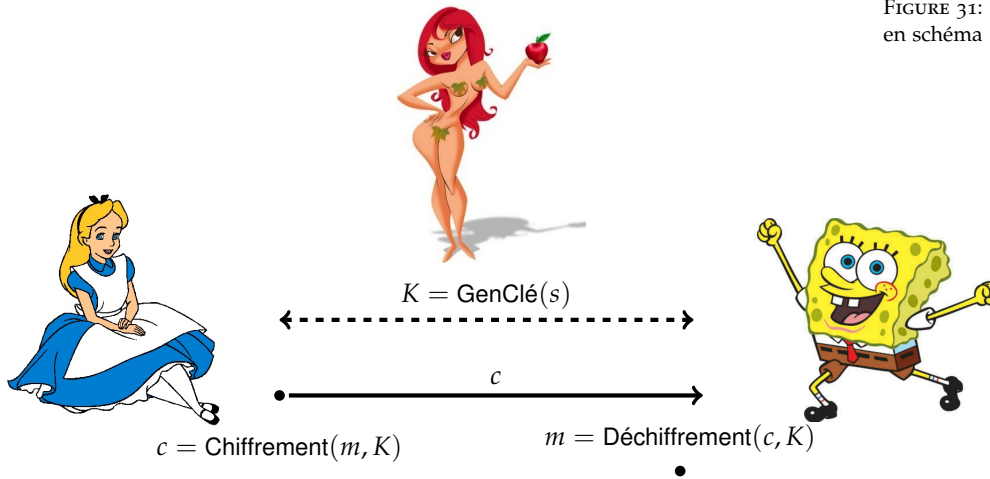


FIGURE 31: La cryptographie symétrique en schéma

La cryptographie symétrique est la forme la plus ancienne de cryptographie (voir définition 513 pour la version asymétrique).

Alice et Bob peuvent par exemple utiliser le protocole de Diffie-Hellman pour convenir de leur secret commun.

Définition 457. Le *chiffrement par flot* consiste à masquer un message $(m_k)_{k \in \mathbb{N}}$ à l'aide d'une *suite chiffrante* $(r_k)_{k \in \mathbb{N}}$ en utilisant l'opération XOR :

Génération de clé Mécanisme de génération d'une suite de bits $(r_k)_{k \in \mathbb{N}}$ appelée *suite chiffrante*.

Secret partagé Données d'initialisation de la suite chiffrante.

Chiffrement Alice calcule et envoie le chiffré

$$\forall k \in \mathbb{N}, \quad c_k = m_k \oplus r_k.$$

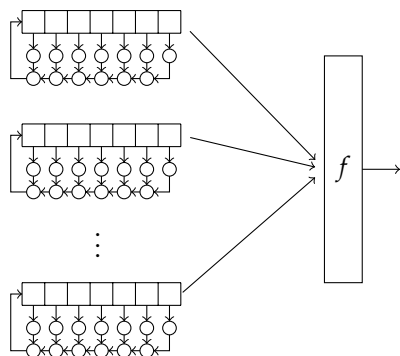
Déchiffrement Bob retrouve le clair avec

$$\forall k \in \mathbb{N}, \quad m_k = c_k \oplus r_k.$$

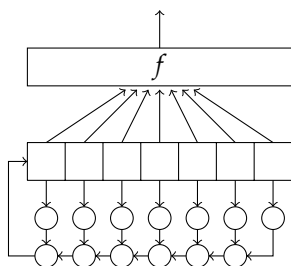
On peut être tenté d'utiliser un LFSR, dont la rétroaction reste secrète et forme la clé, afin de générer une suite chiffrante. Ceci est dangereux. L'algorithme de Berlekamp-Massey montre que $2n$ bits consécutifs de la suite chiffrante (où n est le nombre de cellules du registre) permettent de retrouver toute la suite. Un tel cryptosystème serait donc extrêmement vulnérable aux attaques à clairs connus.

Trois méthodes circonviennent les registres à décalage à rétroaction linéaire afin de générer des suites chiffrantes robustes aux attaques :

- la combinaison de n registres à décalage à rétroaction linéaire distincts par une fonction booléenne non-linéaire f ,



- le filtrage interne des cellules d'un même registre par une fonction booléenne non-linéaire f ,



- et le contrôle des horloges (l'horloge du LFSR est avancée ou non selon l'état du système).

Exemple 458 ($A_5/1$). Le système $A_5/1$ est utilisé dans les communications GSM. Il a été développé vers 1987 et a fuité vers 1994. Il combine 3 registres à décalage à rétroaction linéaire de longueur 19, 22 et 23 dont on XOR les sorties pour produire un bit. Chaque LFSR admet une cellule distinguée. L'horloge de chaque LFSR s'actionne ou non selon que son bit distingué est majoritaire ou non parmi les trois bits distingués.

Exemple 459 (Eo). Le système Eo est utilisé pour les communications Bluetooth. Il combine 4 registres à décalage à rétroaction linéaire de longueur 25, 31, 33 et 39. La fonction de combinaison des 4 LFSR accède également à une mémoire de 4 bits, qui est aussi mise à jour à chaque étape.

Exemple 460 (Snow). Le système SNOW fait partie des standards ISO et a été utilisé pour la téléphonie 3G. Il recourt à un registre à décalage à rétroaction linéaire de longueur 16.

Exemple 461. Une compétition, eSTREAM, a été organisée entre 2004 et 2008 afin de déterminer de nouveaux algorithmes de chiffrement par flot. Parmi les lauréats, le système *Grain* utilise des registres à décalage à rétroaction linéaire.

TP 11 : Codes correcteurs d'erreurs algébriques

Buts : Retravailler les concepts de géométrie algébrique dans le contexte du codage correcteur d'erreurs.

Travaux préparatoires : Cours et exercice 502 (questions 1-3,5-6 à la main)

Évaluation du TP : Exercices 465 (code de Hamming), 477 (décodage de Berlekamp-Massey), 482 (décodage en liste), 502 (code algébrique), 472 (bornes), 510 (codes battant GV).

Contexte

Les codes correcteurs d'erreurs linéaires (rappels)

Définition 462. Étant donné deux mots $\mathbf{c} = c_1c_2 \dots c_n$ et $\mathbf{c}' = c'_1c'_2 \dots c'_n$ de longueur n sur un alphabet quelconque, la *distance de Hamming* est

$$d(\mathbf{c}, \mathbf{c}') = |\{i \in \llbracket 1, n \rrbracket; c_i \neq c'_i\}| \quad (23)$$

et le *poids de Hamming* est $w(\mathbf{c}) = d(\mathbf{c}, \mathbf{0})$ (si l'alphabet contient 0).

Définition 463. Un $[n, k, d]_q$ -code correcteur d'erreur linéaire est un sous-espace vectoriel de \mathbb{F}_q^n tel que pour tout couple de mots $\mathbf{c} \neq \mathbf{c}' \in C$, $d(\mathbf{c}, \mathbf{c}') \geq d$.

On appelle $R = \frac{k}{n}$ le *rendement* de C et $\delta = \frac{d}{n}$ la *distance minimale relative*.

Remarque 464. Pour définir un code quelconque, on omet la condition de sous-espace vectoriel dans la définition ci-dessus. Quand le code est linéaire, il suffit de vérifier que pour tout $\mathbf{c} \in C$ non nul, $w(\mathbf{c}) \geq d$.

Le code C peut être décrit par une matrice \mathbf{G} de taille $k \times n$, dite *matrice génératrice*, qui est une matrice dont les lignes engendrent C . Il peut aussi être décrit par une matrice \mathbf{H} de taille $(n - k) \times n$, dite *matrice de contrôle*, qui est une matrice dont les lignes engendrent l'orthogonal C^\perp de C . Un mot de code \mathbf{c} est caractérisé par l'équation $\mathbf{H}\mathbf{c} = \mathbf{0}$. On appelle encore *syndrôme* le vecteur $\mathbf{H}\mathbf{c}$.

Exercice 465. Le code de Hamming C est un $[7, 4, d]_2$ -code que l'on peut obtenir avec SageMath via `C = codes.HammingCode(GF(2), 3)`. Donner une base de C (avec `generator_matrix()`), une matrice de contrôle et sa distance minimale (avec `minimum_distance()`).

Si l'on souhaite transmettre un mot m de k lettres par un certain canal, on utilise une *fonction d'encodage* : $\phi : \mathbb{F}_q^k \rightarrow C$. Cette fonction se présente sous la forme d'un isomorphisme de \mathbb{F}_q -espaces vectoriels et l'on transmet $\mathbf{c} = \phi(\mathbf{m})$ plutôt que \mathbf{m} . Le *décodage par maximum de vraisemblance* revient à trouver le mot de C (s'il existe) le plus proche du message reçu.

Le décodage devient *non-ambigu* si l'on se contente de rechercher (s'il existe) un mot du code $\mathbf{c} \in C$ à distance $< \frac{d}{2}$ du message reçu. Dans ce cas, le mot \mathbf{c} est unique; on parle de *décodage unique* (par opposition au *décodage en liste* qui consiste à chercher tous les mots $\mathbf{c} \in C$ à distance $e \geq d/2$ du mot reçu). De plus, pourvu que moins de $\frac{d}{2}$ symboles de \mathbf{c} mutent au cours de la transmission, l'existence de \mathbf{c} est assurée.

Ainsi, utiliser C permet d'envoyer R symboles par usage du canal, mais le récepteur peut tolérer un taux d'erreur de $\delta/2$: la communication est moins efficace mais plus fiable.

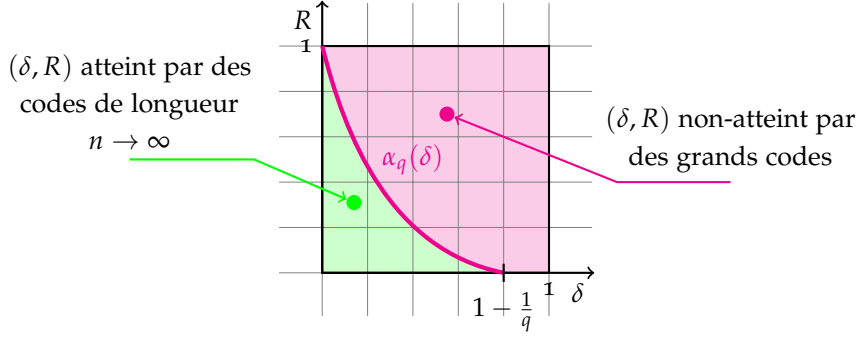
Remarque 466. Le décodage par maximum de vraisemblance (*maximum likelihood decoding* ou MLD) d'un code linéaire est l'analogue pour les codes du problème du vecteur le plus proche, CVP, pour les réseaux. Comme pour les réseaux, MLD est un problème NP-complet.

Bornes sur les codes

SEUIL ENTRE RENDEMENT ET DISTANCE MINIMALE RELATIVE Dans la pratique, on cherche à transmettre des messages aussi longs que voulu. Nous raisonnerons donc sur R et δ pour n asymptotiquement grand.

Il paraît intuitivement clair qu'un bon rendement (i.e. une communication rapide) se fait au dépens d'une grande distance minimale relative (i.e. bonne tolérance aux erreurs) et *vice versa*. Pour toute distance minimale relative $\delta \in \mathbb{Q}$, il existe un rendement optimal seuil $\alpha_q(\delta)$ tel qu'il existe une infinité de codes de distance relative δ et de rendement R' ssi $R' \leq \alpha_q(\delta)$. De plus, on peut démontrer que $\alpha_q(\delta)$ s'étend en une fonction continue sur $[0, 1]$. Le *graal* d'un théoricien des codes est de construire une famille infinie explicite de codes réalisant $R = \alpha_q(\delta)$. Malheureusement à ce jour, on ne connaît pas la valeur de $\alpha_q(\delta)$ en général, mais seulement des encadrements.

BORNE INFÉRIEURE SUR $\alpha_q(\delta)$

FIGURE 32: Allure de la fonction α_q

Définition 467. On appelle *fonction entropie q -aire* la fonction

$$H_q(x) = \begin{cases} 0 & \text{si } x = 0 \\ x \log_q(q-1) - x \log_q x - (1-x) \log_q(1-x) & \text{si } 0 < x \leq 1 - \frac{1}{q} \end{cases} \quad (24)$$

Fait 468. Une boule de Hamming de rayon rn dans \mathbb{F}_q^n a pour volume $q^{nH_q(r)+o(n)}$ lorsque $r \leq 1 - 1/q$.

Proposition 469 (Borne inférieure de Gilbert et Varshamov). Pour tout $0 \leq \delta \leq 1 - \frac{1}{q}$, on a

$$\alpha_q(\delta) \geq 1 - H_q(\delta) \quad (25)$$

Démonstration. (Idée). Un code linéaire tiré au hasard vérifie cette inégalité avec une probabilité strictement positive. \square

BORNES SUPÉRIEURES SUR $\alpha_q(\delta)$

Proposition 470 (Borne supérieure de Singleton). Un $[n, k, d]_q$ -code vérifie toujours $d + k \leq n + 1$

Démonstration. Soit $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ une matrice de contrôle (i.e. $\mathbf{c} \in C$ ssi $\mathbf{H}\mathbf{c} = 0$). La matrice \mathbf{H} est de rang $n - k$. S'il existe un mot $\mathbf{c} \in C$ de poids $\leq n - k$, comme $\mathbf{H}\mathbf{c} = 0$, moins de $n - k$ colonnes de \mathbf{H} sont liées, ce qui contredit le rang de \mathbf{H} . \square

Les codes atteignant la borne de Singleton sont dits *maximum distance separable* (MDS). Les codes de Reed-Solomon sont MDS.

On a encore les bornes suivantes

Théorème 471. Nous notons $\theta = 1 - \frac{1}{q}$. Alors

1. Borne supérieure de Plotkin : si $0 \leq \delta \leq \theta$

$$\alpha_q(\delta) \leq 1 - \frac{\delta}{\theta} \quad (26)$$

De plus, $\alpha_q(\delta) = 0$ sinon.

2. Borne supérieure de Hamming

$$\alpha_q(\delta) \leq 1 - H_q(\delta/2) \quad (27)$$

3. La borne supérieure de MRRW (McEliece, Rodemich, Rumsey et Welch)

$$\alpha_q(\delta) \leq H_q \left(\frac{1}{q} \left(q - 1 - (q - 2)\delta - 2\sqrt{(q - 1)\delta(1 - \delta)} \right) \right) \quad (28)$$

Démonstration. (Idée des méthodes employées). Pour des démonstrations complètes, consulter¹⁶ par exemple.

16.

1. Double comptage sur les lettres des mots du code.
2. Méthode de pavage de l'espace par des boules disjointes.
3. Programmation linéaire.

□

BILAN ENTRE LES BORNES La figure 35 reproduit quelques bornes inférieure et supérieure connues (pour la clarté de l'esquisse, toutes ne sont pas représentées). Elle met en évidence une *terra incognita*, zone dans laquelle on ne sait pas s'il est possible de construire de codes de longueur arbitrairement grande ou si cet espoir est vain.

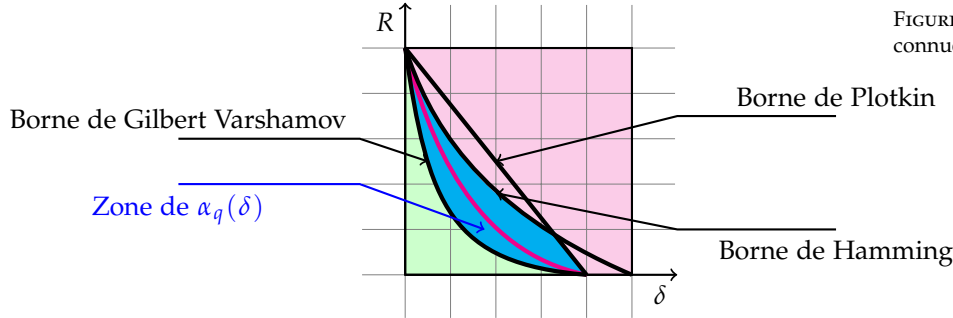


FIGURE 33: Résumé de quelques bornes connues sur les codes

Exercice 472. Représenter sur une même figure la borne inférieure (borne d'existence) de Gilbert-Varshamov et les bornes supérieures (inexistence) de Plotkin, de Hamming et de MRRW pour différentes valeurs de q .

On pourra utiliser `plot([f,g,h], (x,0,1))` pour représenter les fonctions f , g et h sur l'intervalle $[0, 1]$.

Nous parviendrons dans la partie suivante à grignoter cette zone grise par le bas à l'aide de codes géométriques (ou codes de Goppa) pour $q \geq 49$. Dans le cas le plus important, le cas $q = 2$, aucune amélioration par rapport à la borne de Gilbert-Varshamov n'est connue. Ceci signifie qu'asymptotiquement on ne sait pas mieux faire qu'un code tiré au hasard.

Les codes de Reed-Solomon

Présentation

Les codes de Reed-Solomon (1960) sont le prototype des codes algébriques. Ils ont été utilisés et continuent de l'être dans de nombreuses applications, seuls ou combinés à d'autres techniques de codage : transmissions spatiales (mission *Voyager* et suivantes), disques compacts ou autres formes de stockage, codes barre 2D, etc.

Définition 473. Soient x_0, \dots, x_{n-1} les $n = q - 1$ éléments non-nuls de \mathbb{F}_q et $\mathcal{L}_{<k} = \mathbb{F}_q[x]_{<k}$ l'espace des polynômes de degré $< k$. On appelle *code de Reed-Solomon* le $[n, k, n - k + 1]_q$ -code

$$RS_{(k,q)} = \{(f(x_0), \dots, f(x_{n-1})), f \in \mathcal{L}_{<k}\}. \quad (29)$$

Démonstration. Un polynôme de degré $< k$ possède $< k$ racines. Ainsi les coordonnées d'un mot de code ne peuvent s'annuler que $k - 1$ fois au maximum. La distance minimale du code est donc $\geq n - k + 1$. \square

Les codes de Reed-Solomon atteignent par conséquent la borne de Singleton (ils sont MDS) : ce sont les meilleurs possibles en terme de compromis entre rendement R et distance minimale relative δ . Un regret cependant : pour un alphabet fixé, leur longueur est limitée à $q - 1$ et ne peut être agrandie.

Remarque 474. Une matrice génératrice de $RS_{(k,q)}$ est donnée par la matrice $k \times n$ de Vandermonde

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_0 & x_1 & \dots & x_{n-1} \\ \vdots & & & \\ x_0^{k-1} & x_1^{k-1} & \dots & x_{n-1}^{k-1} \end{pmatrix}$$

Comme $\sum_{i=0}^{n-1} x_i^\ell = 0$ pour $\ell \neq 0$ (on aura reconnu la somme des racines n -ièmes de l'unité), une matrice de contrôle de $RS_{(k,q)}$ est

$$\mathbf{H} = \begin{pmatrix} x_0 & x_1 & \dots & x_{n-1} \\ x_0^2 & x_1^2 & \dots & x_{n-1}^2 \\ \vdots & & & \\ x_0^{n-k} & x_1^{n-k} & \dots & x_{n-1}^{n-k} \end{pmatrix}$$

Remarque 475. Les codes de Reed-Solomon sont typiques de codes par évaluation. On se donne un \mathbb{F}_q -espace vectoriel de fonctions \mathcal{L} et d'un sous-ensemble S de leur domaine de définition. Les mots de code sont les $|S|$ -upplets $c_f = (f(x))_{x \in S}$ pour $f \in \mathcal{L}$.

Décodage des codes de Reed-Solomon

Un résultat de 2004 montre que, même restreint aux codes de Reed-Solomon, le décodage par maximum de vraisemblance (MLD) reste NP-difficile.

Cependant, plusieurs algorithmes remarquables sont associés aux codes de Reed-Solomon : en particulier l'*algorithme de Berlekamp-Massey* (qui sert aussi à retrouver le plus court registre à décalage à rétroaction linéaire —LFSR en anglais— ou le polynôme minimal d'une suite linéaire récurrente), l'*algorithme de Welch-Berlekamp* et l'*algorithme de Guruswami-Sudan* de décodage en listes. Tous ces algorithmes sont basés sur des techniques de manipulation de polynômes (interpolation, factorisation).

L'algorithme de Welch-Berlekamp permet de décoder jusqu'à $\frac{n-k+1}{2}$ erreurs en temps $\mathcal{O}(n^3)$ ¹⁷.

17. La complexité polynomiale de cet algorithme n'entre pas en contradiction avec le résultat de NP-complétude de MLD puisqu'on ne décode que jusqu'à une taille d'erreur fixée à l'avance.

Décodage de Berlekamp-Massey

Dans ce qui suit, nous choisissons un élément primitif α de \mathbb{F}_q et supposons que la suite $(x_i)_{0 \leq i \leq n-1}$ a été obtenue en prenant $x_i = \alpha^i$ pour tout $i \in \llbracket 0, n-1 \rrbracket$. Soient \mathbf{H} la matrice de contrôle du code définie à la remarque 474 et

$$\begin{aligned} \mathbf{c} &= (c_0, \dots, c_{n-1}), \text{ le mot de code envoyé,} \\ \mathbf{r} &= (r_0, \dots, r_{n-1}), \text{ le mot reçu,} \\ \mathbf{e} = \mathbf{r} - \mathbf{c} &= (e_0, \dots, e_{n-1}), \text{ le vecteur d'erreur.} \end{aligned}$$

Le syndrome $\mathbf{s} = \mathbf{H}\mathbf{r}$ se calcule selon l'expression

$$\forall j \in \llbracket 1, n-k \rrbracket, \quad s_j = \sum_{i=0}^{n-1} r_i \alpha^{ij}.$$

Nous ne connaissons par le vecteur d'erreur \mathbf{e} . Mais comme $\mathbf{H}\mathbf{c} = 0$, le syndrôme vérifie $\mathbf{s} = \mathbf{H}\mathbf{r} = \mathbf{H}\mathbf{r} - \mathbf{H}\mathbf{c} = \mathbf{H}\mathbf{e}$ et donc

$$\forall j \in \llbracket 1, n-k \rrbracket, \quad s_j = \sum_{i=0}^{n-1} e_i \alpha^{ij}.$$

Introduisons le polynôme $e(x) = e_0 + e_1x + \dots + e_{n-1}x^{n-1}$, nous avons ainsi :

$$\forall j \in \llbracket 1, n-k \rrbracket, \quad s_j = e(\alpha^j). \quad (30)$$

On fixe les notations

$$\begin{aligned}
 M &= \{i \in \llbracket 0, n-1 \rrbracket; e_i \neq 0\} \text{ la position des erreurs,} \\
 \varepsilon &= |M| \text{ nombre d'erreurs,} \\
 \sigma(x) &= \prod_{i \in M} (1 - \alpha^i x) \text{ le polynôme localisateur des erreurs,} \\
 \omega(x) &= \sum_{i \in M} e_i \alpha^i x \prod_{j \in M \setminus \{i\}} (1 - \alpha^j x).
 \end{aligned}$$

Bien entendu, *a priori*, nous ne pouvons pas construire ces objets puisque nous ne connaissons pas le vecteur d'erreurs. Cependant, nous notons que les racines de σ révèlent les erreurs : une erreur se produit en position i si et seulement si x_i est racine de $\sigma(x)$. Nous remarquons également que

$$\begin{aligned}
 \frac{\omega(x)}{\sigma(x)} &= \sum_{i \in M} \frac{e_i \alpha^i x}{1 - \alpha^i x} \\
 &= \sum_{i \in M} e_i \sum_{\ell=1}^{\infty} (\alpha^i x)^\ell \\
 &= \sum_{\ell=1}^{\infty} \left(\sum_{i \in M} e_i \alpha^{\ell i} \right) x^\ell \\
 &= \sum_{\ell=1}^{\infty} e(\alpha^\ell) \cdot x^\ell.
 \end{aligned}$$

Cette équation (identique à celle de la proposition 451) est exactement celle que sait résoudre l'algorithme de Berlekamp-Massey (voir algorithme 34). Le polynôme $\sigma(x)$ étant de degré ε , nous pouvons retrouver $\sigma(x)$ à partir des 2ε premiers termes de la série. Grâce à l'équation 30, nous connaissons $e(\alpha^\ell) = s_\ell$ pour $\ell \in \llbracket 1, n-k \rrbracket$ et pouvons ainsi décoder jusqu'à $\frac{n-k}{2}$ erreurs.

Exemple 476. On considère $q = 11$ et $k = 5$. Nous choisissons $\alpha = 6$ (qui est bien un élément primitif de \mathbb{F}_{11}). La matrice du $[10, 5]_{11}$ -code $RS_{5,11}$ est alors

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 6 & 3 & 7 & 9 & 10 & 5 & 8 & 4 & 2 \\ 1 & 3 & 9 & 5 & 4 & 1 & 3 & 9 & 5 & 4 \\ 1 & 7 & 5 & 2 & 3 & 10 & 4 & 6 & 9 & 8 \\ 1 & 9 & 4 & 3 & 5 & 1 & 9 & 4 & 3 & 5 \end{pmatrix}$$

que l'on obtient avec SageMath avec l'instruction

```
G = ( matrix([[z^j for z in [alpha^i
for i in range(n)] ] for j in range(k)]))
```

et la matrice de contrôle

$$\mathbf{H} = \begin{pmatrix} 1 & 6 & 3 & 7 & 9 & 10 & 5 & 8 & 4 & 2 \\ 1 & 3 & 9 & 5 & 4 & 1 & 3 & 9 & 5 & 4 \\ 1 & 7 & 5 & 2 & 3 & 10 & 4 & 6 & 9 & 8 \\ 1 & 9 & 4 & 3 & 5 & 1 & 9 & 4 & 3 & 5 \\ 1 & 10 & 1 & 10 & 1 & 10 & 1 & 10 & 1 & 10 \end{pmatrix}$$

et l'on a bien $\mathbf{GH}^\top = 0$.

Disons que l'on envoie

$$\mathbf{c} = \mathbf{mG} = (2, 1, 5, 4, 0, 4, 2, 0, 7, 7)$$

avec $\mathbf{m} = (1, -1, 2, 0, 0)$ et que l'erreur soit

$$\mathbf{e} = (-2, -1, 0, 0, 0, 0, 0, 0, 0, 0),$$

si bien que le récepteur reçoit

$$\mathbf{r} = (0, 0, 5, 4, 0, 4, 2, 0, 7, 7).$$

Nous commençons par déterminer le vecteur syndrome, qui est ici :

$$\mathbf{s} = \mathbf{Hr} = (3, 6, 2, 0, 10)$$

Nous avons donc besoin de retrouver le polynôme de connection de la série entière $3x + 6x^2 + 2x^3 + 10x^5 + O(x^6)$. Avec `berlekamp_massey([3, 6, 2, 0]) . reverse()` de SageMath, nous obtenons $\sigma(x) = 6x^2 + 4x + 1$ qui se factorise en $(1 - x)(1 - 6x)$. Nous reconnaissons α^0 et α^1 . Aussi, nous avons déterminé que les positions 0 et 1 du message reçu sont corrompues.

En résumé : la position des erreurs est $M = \{0, 1\}$. Le nombre d'erreurs est $\epsilon = |M| = 2$. Le polynôme localisateur des erreurs est $(1 - x)(1 - 6x)$. Une dernière étape d'algèbre linéaire permet d'obtenir \mathbf{m} (on résout le système $\mathbf{r} = \mathbf{mG}$ dont on a retiré les colonnes 0 et 1).

Exercice 477. On donne $q = 13$, $k = 5$, $\alpha = 6 \in \mathbb{F}_{13}$. Un message a été encodé avec le $[12, 5]_{13}$ -code $RS_{5,13}$ et on a reçu le vecteur

$$\mathbf{r} = (5, 2, 9, 9, 1, 11, 7, 5, 7, 4, 3, 1)$$

1. Calculer la matrice génératrice \mathbf{G} et la matrice de contrôle \mathbf{H} (selon la définition de la remarque 474).
2. Calculer le syndrome $\mathbf{s} = \mathbf{Hr}$.
3. Calculer le polynôme localisateur d'erreurs $\sigma(x)$.
4. Calculer la position des erreurs M .
5. Retrouver le message \mathbf{m} envoyé.

Décodage en liste

Un résultat de Sudan (1997) a révolutionné la pratique des codes correcteurs d'erreurs en remettant au goût du jour le décodage en liste. Ce type de décodage consiste à renvoyer, étant donné un message reçu \mathbf{r} , non plus un unique mot de code à distance $< \frac{d_{\min}}{2}$ mais une liste de mots de codes possibles \mathbf{c} tels que $w(\mathbf{r} - \mathbf{c}) \leq t$ où t est une borne sur le nombre d'erreurs fixée à l'avance. L'algorithme de Guruswami-Sudan, qui s'applique au code de Reed-Solomon, permet de franchir la barrière de correction des algorithmes de décodage unique avec certaines valeurs $t > \frac{n-k+1}{2}$.

Précisément, étant donné un message reçu \mathbf{r} , nous voulons résoudre un problème d'interpolation polynômiale sur les n couples $(x_i, r_i)_{0 \leq i \leq n-1}$ et cherchons l'ensemble des polynômes

$$f \in \mathcal{L}_{<k} \text{ tel que } |\{i \in \llbracket 0, n-1 \rrbracket, f(x_i) = r_i\}| > n - t.$$

Définition 478. Soit $Q(x, y) = \sum_{i,j} q_{i,j} x^i y^j \in \mathbb{F}_q[x, y]$ un polynôme. Nous appelons *degré pondéré par $(1, k-1)$* de Q :

$$\deg_{1,k-1}(Q(x, y)) = \max_{q_{i,j} \neq 0} i + (k-1)j. \quad (31)$$

Théorème 479 (Sudan). Soit $Q(x, y)$ un polynôme de $\mathbb{F}_q[x, y]$ de degré $\deg_{1,k-1} Q < n - t$ qui satisfait $Q(x_i, r_i) = 0$ pour tout $i \in \llbracket 0, n-1 \rrbracket$. Alors, pour tout polynôme $f \in \mathcal{L}_{<k}$, si le mot de code $\mathbf{c} = (f(x_0), \dots, f(x_{n-1}))$ est à distance de Hamming $\leq t$ du mot reçu \mathbf{r} , alors $y - f(x)$ est un facteur de $Q(x, y)$.

Démonstration. Soit f un tel polynôme. Par hypothèse $f(x_i) = r_i$ pour au moins $n - t$ valeurs de i , autrement dit $Q(x_i, f(x_i)) = 0$. Mais

$$\deg Q(x, f(x)) \leq \deg_{1,k-1} Q(x, y) < n - t$$

Comme un polynôme univarié ne peut avoir plus de zéros que son degré, $Q(x, f(x))$ est le polynôme nul.

Voyons pour un temps Q comme un polynôme en la variable y sur l'anneau $\mathbb{F}_q[x]$. Nous venons de dire que $y = f(x)$ est une racine de Q , ce qui montre (écrire la division euclidienne pour s'en convaincre) que Q est divisible par $y - f(x)$. \square

On peut déduire du théorème de Sudan l'algorithme suivant.

1. Chercher un polynôme $Q(x, y) \in \mathbb{F}_q[x, y]$ de degré $\deg_{1,k-1} Q < n - t$ tel que $Q(x_i, r_i) = 0$ pour tout $i \in \llbracket 0, n-1 \rrbracket$.
2. Trouver tous les facteurs de Q de la forme $y - f(x)$.
3. Renvoyer f quand f est de degré $< k$ et coïncide avec r au moins $n - t$ fois.

Pour s'assurer de l'existence du polynôme Q , nous raisonnons sur la dimension de l'espace vectoriel dans lequel nous le cherchons.

Lemme 480. *Soit d un entier. L'ensemble des polynômes $Q \in \mathbb{F}_q[x, y]$ tels que $\deg_{1, k-1} Q \leq d$ forme un espace vectoriel de dimension $\geq \frac{(d+1)(d+2)}{2(k-1)}$.*

Démonstration. On évalue la somme

$$\begin{aligned}
 & \sum_{j=0}^{\lfloor \frac{d}{k-1} \rfloor} \sum_{i=0}^{d-(k-1)k} 1 \\
 &= \sum_{j=0}^{\lfloor \frac{d}{k-1} \rfloor} d - (k-1)j + 1 \\
 &= (d+1) \left(\left\lfloor \frac{d}{k-1} \right\rfloor + 1 \right) - \frac{k-1}{2} \left(\left\lfloor \frac{d}{k-1} \right\rfloor + 1 \right) \left\lfloor \frac{d}{k-1} \right\rfloor \\
 &\geq \frac{(d+1)(d+2)}{2(k-1)}.
 \end{aligned}$$

□

L'existence d'un polynôme Q satisfaisant les hypothèses du théorème de Sudan est assurée si $\frac{(d+1)(d+2)}{2(k-1)} > n$ avec $d \leftarrow n - t - 1$. Choisir $t < n - \sqrt{2(k-1)n}$. Compte tenu du degré de Q , nous savons de plus que qu'il y a au plus $\sqrt{2n/(k-1)}$ mots de code possibles.

Pour trouver les facteurs $y - f(x)$ de Q , nous pouvons fixer un polynôme irréductible $\psi(x)$ de degré $\geq k$ (par exemple le polynôme $\psi(x) = x^{q-1} - \gamma$ où γ est un élément primitif de \mathbb{F}_q , voir exercice 182). Soit θ une racine de ψ . Le problème initial revient à trouver les racines $f(\theta)$ de $Q(\theta, y)$ dans l'extension de corps $\mathbb{F}_q[\theta] \simeq \mathbb{F}_{q^{\deg \psi}}$ (voir exercice 193).

Exemple 481. On travaille sur \mathbb{F}_{17} avec $n = p - 1 = 16$ et $k = 3$. On a fixé $\alpha = 6$ comme générateur de \mathbb{F}_{17} . La matrice génératrice du $[16, 3]_{17}$ -code $RS_{3,17}$ est

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 6 & 2 & 12 & 4 & 7 & 8 & 14 & 16 & 11 & 15 & 5 & 13 & 10 & 9 & 3 \\ 1 & 2 & 4 & 8 & 16 & 15 & 13 & 9 & 1 & 2 & 4 & 8 & 16 & 15 & 13 & 9 \end{pmatrix}$$

La distance minimale du code $RS_{3,17}$ est $16 + 1 - 3 = 14$, donc le décodage unique¹⁸ peut se faire jusqu'à une distance $6 = \lfloor \frac{14-1}{2} \rfloor$ du code. L'algorithme de Sudan permet quant à lui de trouver une liste de mots de codes jusqu'à une distance¹⁹ 8. On souhaite décoder le vecteur reçu

$$\mathbf{r} = (15 \ 16 \ 7 \ 9 \ 16 \ 6 \ 15 \ 0 \ 7 \ 7 \ 4 \ 8 \ 9 \ 5 \ 3 \ 6).$$

18. par exemple avec Berlekamp-Massey

19. en réalité $\lfloor 16 - \sqrt{2 \cdot (3-1) \cdot 16} \rfloor$

On cherche comme indiqué un polynôme Q tel que $\deg_{1,2} Q < n - \sqrt{2(k-1)n} = 8$. Les monômes $x^i y^j$ possibles de Q sont :

$$[1, y, y^2, y^3, x, xy, xy^2, xy^3, x^2, x^2 y, x^2 y^2, x^3, x^3 y, x^3 y^2, x^4, x^4 y, x^5, x^5 y, x^6, x^7]$$

car leurs exposants (i, j) satisfont $i, j \geq 0$ et $i + 2j < 8$. Le polynôme Q est un élément du noyau de la matrice (en utilisant le même ordre des monômes et en évaluant en $x = \alpha^i$ et $y = r_i$ pour obtenir la ligne i)

$$\left(\begin{array}{cc|cccccccccccccccccccccccc} & & 1 & y & y^2 & y^3 & x & xy & xy^2 & xy^3 & x^2 & x^2 y & x^2 y^2 & x^3 & x^3 y & x^3 y^2 & x^4 & x^4 y & x^5 & x^5 y & x^6 & x^7 \\ x \leftarrow \alpha^0 = 1 & y \leftarrow r_0 = 15 & 1 & 15 & 4 & 9 & 1 & 15 & 4 & 9 & 1 & 15 & 4 & 1 & 15 & 4 & 1 & 15 & 1 & 15 & 1 & 1 \\ x \leftarrow \alpha^1 = 6 & y \leftarrow r_1 = 16 & 1 & 16 & 1 & 16 & 6 & 11 & 6 & 11 & 2 & 15 & 2 & 12 & 5 & 12 & 4 & 13 & 7 & 10 & 8 & 14 \\ x \leftarrow \alpha^2 = 2 & y \leftarrow r_2 = 7 & 1 & 7 & 15 & 3 & 2 & 14 & 13 & 6 & 4 & 11 & 9 & 8 & 5 & 1 & 16 & 10 & 15 & 3 & 13 & 9 \\ x \leftarrow \alpha^3 = 12 & y \leftarrow r_3 = 9 & 1 & 9 & 13 & 15 & 12 & 6 & 3 & 10 & 8 & 4 & 2 & 11 & 14 & 7 & 13 & 15 & 3 & 10 & 2 & 7 \\ x \leftarrow \alpha^4 = 4 & y \leftarrow r_4 = 16 & 1 & 16 & 1 & 16 & 4 & 13 & 4 & 13 & 16 & 1 & 16 & 13 & 4 & 13 & 1 & 16 & 4 & 13 & 16 & 13 \\ x \leftarrow \alpha^5 = 7 & y \leftarrow r_5 = 6 & 1 & 6 & 2 & 12 & 7 & 8 & 14 & 16 & 15 & 5 & 13 & 3 & 1 & 6 & 4 & 7 & 11 & 15 & 9 & 12 \\ x \leftarrow \alpha^6 = 8 & y \leftarrow r_6 = 15 & 1 & 15 & 4 & 9 & 8 & 1 & 15 & 4 & 13 & 8 & 1 & 2 & 13 & 8 & 16 & 2 & 9 & 16 & 4 & 15 \\ x \leftarrow \alpha^7 = 14 & y \leftarrow r_7 = 0 & 1 & 0 & 0 & 0 & 14 & 0 & 0 & 0 & 9 & 0 & 0 & 7 & 0 & 0 & 13 & 0 & 12 & 0 & 15 & 6 \\ x \leftarrow \alpha^8 = 16 & y \leftarrow r_8 = 7 & 1 & 7 & 15 & 3 & 16 & 10 & 2 & 14 & 1 & 7 & 15 & 16 & 10 & 2 & 1 & 7 & 16 & 10 & 1 & 16 \\ x \leftarrow \alpha^9 = 11 & y \leftarrow r_9 = 7 & 1 & 7 & 15 & 3 & 11 & 9 & 12 & 16 & 2 & 14 & 13 & 5 & 1 & 7 & 4 & 11 & 10 & 2 & 8 & 3 \\ x \leftarrow \alpha^{10} = 15 & y \leftarrow r_{10} = 4 & 1 & 4 & 16 & 13 & 15 & 9 & 2 & 8 & 4 & 16 & 13 & 9 & 2 & 8 & 16 & 13 & 2 & 8 & 13 & 8 \\ x \leftarrow \alpha^{11} = 5 & y \leftarrow r_{11} = 8 & 1 & 8 & 13 & 2 & 5 & 6 & 14 & 10 & 8 & 13 & 2 & 6 & 14 & 10 & 13 & 2 & 14 & 10 & 2 & 10 \\ x \leftarrow \alpha^{12} = 13 & y \leftarrow r_{12} = 9 & 1 & 9 & 13 & 15 & 13 & 15 & 16 & 8 & 16 & 8 & 4 & 4 & 2 & 1 & 1 & 9 & 13 & 15 & 16 & 4 \\ x \leftarrow \alpha^{13} = 10 & y \leftarrow r_{13} = 5 & 1 & 5 & 8 & 6 & 10 & 16 & 12 & 9 & 15 & 7 & 1 & 14 & 2 & 10 & 4 & 3 & 6 & 13 & 9 & 5 \\ x \leftarrow \alpha^{14} = 9 & y \leftarrow r_{14} = 3 & 1 & 3 & 9 & 10 & 9 & 10 & 13 & 5 & 13 & 5 & 15 & 15 & 11 & 16 & 16 & 14 & 8 & 7 & 4 & 2 \\ x \leftarrow \alpha^{15} = 3 & y \leftarrow r_{15} = 6 & 1 & 6 & 2 & 12 & 3 & 1 & 6 & 2 & 9 & 3 & 1 & 10 & 9 & 3 & 13 & 10 & 5 & 13 & 15 & 11 \end{array} \right)$$

Ce noyau est de dimension 5. Une de ses bases correspond aux 5 polynômes

$$12x^7 + 6x^5 y + 4x^5 + 5x^4 y + 9x^3 y^2 + 11x^4 + 12x^3 y + 5x^2 y^2 + 5xy^3 + 6x^3 + 12x^2 y + 5xy^2 + 3y^3 + 6y^2 + 1$$

$$3x^7 + 11x^6 + x^5 y + 2x^5 + 7x^4 y + 8x^3 y^2 + 4x^4 + 9x^3 y + 2xy^3 + 14x^3 + 7x^2 y + 5xy^2 + 10y^3 + 15y^2 + y$$

$$9x^7 + 9x^6 + 5x^5 y + 2x^5 + 7x^4 y + x^3 y^2 + 15x^4 + 11x^3 y + 6x^2 y^2 + 3xy^3 + 15x^3 + 8x^2 y + 6xy^2 + x$$

$$8x^7 + x^6 + 9x^5 y + 11x^5 + 3x^4 y + 8x^3 y^2 + 5x^4 + 4x^3 y + 2x^2 y^2 + 10xy^3 + 9x^3 + 10x^2 y + 15xy^2 + xy$$

$$4x^7 + 8x^6 + 5x^5 y - x^5 + 12x^4 y + 11x^3 y^2 - x^4 + 10x^3 y + 10x^2 y^2 + 6x^3 + 15x^2 y + x^2$$

dont les factorisations respectives sont

$$(x+4) \cdot \underbrace{(9x^2 + 12x + y + 13)}_{y-f_2(x)} \cdot \underbrace{(10x^2 + 12x + y + 14)}_{y-f_1(x)} \cdot (10x^2 + 14x + y + 9)$$

$$\underbrace{(9x^2 + 12x + y + 13)}_{y-f_2(x)} \cdot \underbrace{(10x^2 + 12x + y + 14)}_{y-f_1(x)} \cdot (2x^3 + xy + 5y)$$

$$x \cdot (4x^2 + 12x + y + 9) \cdot \underbrace{(9x^2 + 12x + y + 13)}_{y-f_2(x)} \cdot \underbrace{(10x^2 + 12x + y + 14)}_{y-f_1(x)}$$

$$x \cdot (9x^2 + y) \cdot \underbrace{(9x^2 + 12x + y + 13)}_{y-f_2(x)} \cdot \underbrace{(10x^2 + 12x + y + 14)}_{y-f_1(x)}$$

$$(x+4) \cdot x^2 \cdot \underbrace{(9x^2 + 12x + y + 13)}_{y-f_2(x)} \cdot \underbrace{(10x^2 + 12x + y + 14)}_{y-f_1(x)}$$

Chacune (une seule suffirait) de ces factorisations révèlent deux polynômes

$$f_1(x) = 7x^2 + 5x + 3 \quad \text{et} \quad f_2(x) = 8x^2 + 5x + 4$$

dont l'évaluation conduit aux mots de code

$$\begin{aligned} \mathbf{c}_1 &= \left(f(\alpha^0) \ f(\alpha^1) \ f(\alpha^2) \ f(\alpha^3) \ \dots \ f(\alpha^{15}) \right) \\ &= \left(0 \ 16 \ 12 \ 9 \ 16 \ 6 \ 12 \ 10 \ 7 \ 7 \ 9 \ 8 \ 10 \ 4 \ 0 \ 6 \right) \end{aligned}$$

et

$$\mathbf{c}_2 = \left(15 \ 13 \ 7 \ 0 \ 16 \ 7 \ 15 \ 0 \ 5 \ 4 \ 4 \ 16 \ 10 \ 5 \ 3 \ 13 \right)$$

qui se trouvent chacun à distance de Hamming 8 de \mathbf{r} et dans certains cas d'autres polynômes dont l'évaluation conduit à des mots de code éloignés de \mathbf{r} . Le décodage par maximum de vraisemblance serait délicat, puisque la distance entre \mathbf{r} et le code $RS_{3,17}$ est atteinte par plusieurs mots. La situation est résumée par la figure 34.

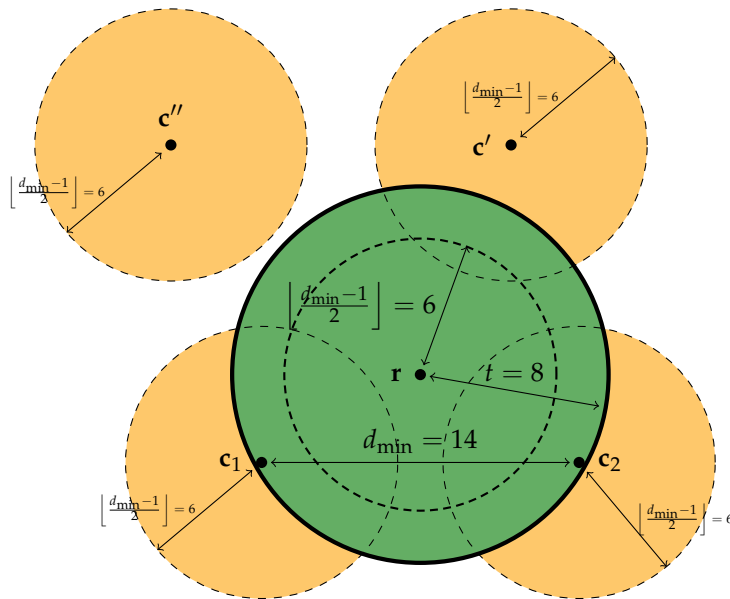


FIGURE 34: Illustration de l'exemple : le mot reçu \mathbf{r} est trop éloigné du code pour que le décodage unique fonctionne (zone orange), cependant le décodage en liste de Sudan permet de trouver une liste de 2 mots.

Exercice 482. On travaille sur \mathbb{F}_{23} dont un élément primitif est $\alpha = 14$ et le $[22, 3]_{23}$ -code de Reed-Solomon $RS_{3,23}$.

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 14 & 12 & 7 & 6 & 15 & 3 & 19 & 13 & 21 & 18 & 22 & 9 & 11 & 16 & 17 & 8 & 20 & 4 & 10 & 2 & 5 \\ 1 & 12 & 6 & 3 & 13 & 18 & 9 & 16 & 8 & 4 & 2 & 1 & 12 & 6 & 3 & 13 & 18 & 9 & 16 & 8 & 4 & 2 \end{pmatrix}$$

1. Jusqu'à quelle distance peut-on faire du décodage unique ?

2. Quel rayon de décodage l'algorithme de Sudan permet-il d'atteindre ?
3. On a encodé un message de 3 lettres de $A - W$ par un triplet \mathbf{m} d'éléments de \mathbb{F}_{23} via la fonction $A \mapsto 0, B \mapsto 1, C \mapsto 2$, etc.²⁰ Ensuite, \mathbf{m} a été transmis à l'aide du code de Reed-Solomon $RS_{3,23}$ ($\mathbf{m} \mapsto \mathbf{mG}$). Vous recevez le mot

20. On peut utiliser `chr` et `ord` pour passer d'un caractère à son numéro ASCII et vice-versa.

$\mathbf{r} = (12, 18, 15, 22, 17, 5, 14, 21, 17, 4, 13, 8, 4, 10, 15, 11, 22, 12, 13, 9, 14, 12)$

- (a) À quelle liste de mots de codes conduit l'exécution de l'algorithme de Sudan ?
- (b) Quels sont les messages correspondants ?
- (c) À quel mot de code et quel message conduit le décodage par maximum de vraisemblance ?

Les codes géométriques

Les codes géométriques algébriques ou codes de Goppa ont été introduit par Valery Denisovich Goppa au début des années 80. Ils reprennent les « bons » ingrédients des codes de Reed-Solomon et ont permis à Tsfasman, Vladut et Zink de battre la borne asymptotique de Gilbert-Varshamov en 1982 par une suite de codes géométriques. De plus, les méthodes de décodage et notamment le décodage en liste ont pu leur être étendues.

L'idée est la suivante. Nous allons continuer à « évaluer des polynômes de degré borné » (car alors le poids des mots non-nuls n'est pas trop petit) sur un objet plus grand : une courbe algébrique au lieu d'une partie de la droite \mathbb{F}_q .

Rappels de géométrie algébrique

CONCEPTS FONDAMENTAUX Dans ce qui suit, on se donne une courbe projective lisse \mathcal{X} définie sur \mathbb{F}_q . Nous distinguerons les points $\mathcal{X}(\mathbb{F}_q)$ de la courbe définis sur \mathbb{F}_q des points $\mathcal{X}(\overline{\mathbb{F}_q})$ définis sur $\overline{\mathbb{F}_q}$. Quand nous parlons de courbe plane, il s'agit tout bonnement de l'ensemble

$$\mathcal{X}(\mathbb{F}_q) = \left\{ (X_0 : Y_0 : Z_0) \in \mathbb{P}^2(\mathbb{F}_q); F(X_0, Y_0, Z_0) = 0 \right\}$$

où F est un *polynôme homogène* de $\mathbb{F}_q[X, Y, Z]$. De plus, un point de la forme $(x_0 : y_0 : 1)$ est dit *affine* (il est racine du polynôme $f(x, y) = F(x, y, 1)$) tandis qu'un point de la forme $(x_0 : y_0 : 0)$ est dit à *l'infini* par rapport au plan $Z = 1$.

On note $\mathbb{F}_q(\mathcal{X})$ l'espace des fonctions rationnelles sur \mathcal{X} , c'est-à-dire l'ensemble des fonctions régulières sur un ouvert de Zariski non vide

(qui est alors dense) de \mathcal{X} . Pour une courbe plane, il s'agit de

$$\mathbb{F}_q(\mathcal{X}) = \left\{ \frac{g(X, Y, Z)}{h(X, Y, Z)}; g, h \in \mathbb{F}_q[X, Y, Z] \text{ homogènes et de même degré} \right\} \cup \{0\} / \sim$$

où \sim est la relation d'équivalence $g/h \sim g'/h'$ ssi $gh' - g'h$ est un multiple de F .

Remarque 483. Nous adoptons ici une approche opposée à celle la section 3 de ²¹ : nous définissons la courbe puis introduisons son corps de fonction plutôt que l'inverse.

21. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

Fait 484. Si $f \in \mathbb{F}_q(\mathcal{X})$, on peut construire l'ordre $\text{ord}_P(f)$ de f en un point P . (Grossièrement, il s'agit de compter la multiplicité de P en tant que zéro ou pôle de f , cf définition 271 ou section 3.3 de *ibidem*.)

Dans le cas d'une courbe plane, pour calculer l'ordre de $\frac{g}{h} \in \mathbb{F}_q(\mathcal{X})$ en un point affine $P = (x_0, y_0)$ de la courbe $\{f(x, y) = 0\}$, il suffit de calculer

$$\begin{aligned} \text{ord}_P\left(\frac{g}{h}\right) &= \dim(\mathbb{F}_q[[x, y]] / \langle f(x + x_0, y + y_0), g(x + x_0, y + y_0) \rangle) \\ &\quad - \dim(\mathbb{F}_q[[x, y]] / \langle f(x + x_0, y + y_0), h(x + x_0, y + y_0) \rangle). \end{aligned}$$

où $\mathbb{F}_q[[x, y]]$ est l'anneau des séries formelles en x et en y . (En fait, $\mathbb{F}_q[[x, y]] / \langle f(x + x_0, y + y_0) \rangle$ correspond à l'anneau de valuation décrit à la section 3.2 de *ibidem*.) Alternativement, si v est une fonction affine qui s'annule en (x_0, y_0) mais pas en un autre zéro commun à f , g et h ,

$$\begin{aligned} \text{ord}_P\left(\frac{g}{h}\right) &= \dim(\mathbb{F}_q[[x, y]] / \langle f, g, v^{\deg f \deg g} \rangle) \\ &\quad - \dim(\mathbb{F}_q[[x, y]] / \langle f, h, v^{\deg f \deg h} \rangle) \end{aligned}$$

qui peut s'évaluer comme on l'a vu à l'aide de bases de Gröbner.

Définition 485. Étant donné une courbe algébrique projective et lisse \mathcal{X} définie sur \mathbb{F}_q , on appelle *diviseur* de \mathcal{X} une \mathbb{Z} -combinaison linéaire (formelle) finie des points de \mathcal{X}

$$D = \sum_{Q \in \mathcal{X}} n_Q(Q) \quad (32)$$

telle que $n_Q = n_{\sigma(Q)}$ pour σ dans le groupe de Galois. Le *degré* de D est l'entier $\deg D = \sum_{Q \in \mathcal{X}} n_Q$; le *support* de D est l'ensemble de points $\text{Supp}(D) = \{Q \in \mathcal{X}; n_Q \neq 0\}$.

En particulier, si f est une fonction rationnelle de $\mathbb{F}_q(\mathcal{X})$, on peut considérer le *diviseur principal* associé à f :

$$\text{div}(f) = \sum_{Q \in \mathcal{X}} \text{ord}_Q(f) (Q) \quad (33)$$

Il est de degré 0 (voir 3.7.3 de *ibidem*).

Définition 486. On appelle *espace des fonctions rationnelles associées à D* le \mathbb{F}_q -espace vectoriel suivant :

$$\mathcal{L}(D) = \{f \in \mathbb{F}_q(\mathcal{X}); \operatorname{div}(f) + D \geq 0\} \cup \{0\} \quad (34)$$

Nous rappelons la proposition 3.8.2(i.b) et le théorème 3.10.2 avec le corollaire 3.10.3 de *op. cit.*.

Proposition 487. Si $\deg D < 0$, alors $\mathcal{L}(D) = \{0\}$.

Théorème 488 (Riemann-Roch). Soient \mathcal{X} une courbe projective lisse de genre g sur \mathbb{F}_q et D un diviseur de \mathcal{X} , alors $\dim \mathcal{L}(D) \geq \deg D + 1 - g$ avec égalité si $\deg D > 2g - 2$.

Nous citons par ailleurs le résultat suivant qui donne facilement le genre d'une courbe plane. (Attention : il faut tout de même vérifier les hypothèses de lissité)

Théorème 489 (Formule de Plücker). Soit \mathcal{X} la courbe projective irréductible lisse définie par le polynôme homogène $F \in \mathbb{F}_q[X, Y, Z]$ de degré d . Alors le genre de \mathcal{X} est

$$g = \frac{(d-1)(d-2)}{2}. \quad (35)$$

QUELQUES OUTILS POUR MANIPULER DES COURBES

Irréductibilité : Les courbes employées ici sont toutes supposées irréductibles (voir 3.1.8 de ²²); voici un critère simple pour le vérifier.

Théorème 490 (Critère d'Eisenstein). Soient A un anneau intègre, $p(y)$ un polynôme de $A[y]$

$$p(y) = a_m y^m + a_{m-1} y^{m-1} + \cdots + a_1 y + a_0,$$

I un idéal premier de A tel que

$$a_m \notin I, \quad \forall i \in [0, m-1], a_i \in I, \quad a_0 \notin I^2.$$

Alors, $p(y)$ est un polynôme irréductible de $A[y]$.

Démonstration. Supposons que $p(y)$ admette une factorisation sous la forme $p(y) = g(y)h(y)$ avec les notations

$$\begin{cases} g(y) &= g_d y^d + \cdots + g_1 y + g_0 \\ h(y) &= h_{m-d} y^{m-d} + \cdots + h_1 y + h_0 \end{cases} ,$$

où $1 \leq d \leq m-1$. Par réduction modulo I , il ne reste que le monôme dominant de p :

$$p(y) = g(y)h(y) = a_m y^m \pmod{I}.$$

22. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

Mais l'anneau quotient A/I est intègre, donc le monôme $a_my^m \in (A/I)[y]$ ne peut se factoriser que sous la forme d'un produit de monômes de $(A/I)[y]$. Donc

$$\begin{cases} g(y) & \equiv by^d \pmod{I} \\ h(y) & \equiv cy^{m-d} \pmod{I} \end{cases}$$

En particulier, les termes g_0 et h_0 sont nuls modulo I . Mais $a_0 = g_0h_0$. Donc $a_0 \in I^2$ ce qui était exclu. \square

Corollaire 491. Soit $f(x, y) = \sum_{i=0}^m f_i(x)y^i$ un polynôme tel que $f_0(x), \dots, f_m(x)$ sont premiers entre eux. On suppose qu'il existe un polynôme $p(x)$ irréductible divisant $f_0(x), \dots, f_{m-1}(x)$ mais pas $f_m(x)$ tel que $p^2(x)$ ne divise pas $f_0(x)$. Alors $f(x, y)$ est irréductible.

Exemple 492. Soit $f(x, y) = y^2 + x^2 + x^4 \in \mathbb{F}_2[x]$. On serait tenté d'appliquer le critère d'Eisenstein avec $p(x) = x^2$ qui n'est pas irréductible. Mais le polynôme n'est pas irréductible car $f(x, y) = (x + y + x^2)^2$.

Intersection de deux courbes :

Théorème 493 (Bézout). Soient deux courbes \mathcal{X} et \mathcal{X}' projectives planes n'ayant pas de composante commune, alors le nombre total d'intersections (sur la clôture algébrique) entre \mathcal{X} et \mathcal{X}' comptées avec leur multiplicité vaut le produit des degrés de \mathcal{X} et \mathcal{X}' .

Exemple 494. L'intersection d'une droite avec une conique (cercle, ellipse, hyperbole, parabole) compte deux points. L'intersection entre deux coniques compte quatre points.

Cardinalité de courbes : Le théorème de base quand on compte les points d'une courbe est le suivant. Il est donné sans preuve.

Théorème 495 (Hasse-Weil). Soit \mathcal{X} une courbe projective lisse de genre g sur \mathbb{F}_q et $N = |\mathcal{X}(\mathbb{F}_q)|$, alors

$$|N - (q + 1)| \leq 2g\sqrt{q}. \quad (36)$$

Remarque 496. En fait, on a même $|N - (q + 1)| \leq g[2\sqrt{q}]$ (théorème de Serre).

Les codes

CONSTRUCTION GÉNÉRALE

Définition 497. Soient \mathcal{X} une courbe projective lisse sur \mathbb{F}_q , $S = \{P_1, \dots, P_n\}$ un ensemble de points distincts de \mathcal{X} définis sur \mathbb{F}_q , D un diviseur de

\mathcal{X} . On suppose que S et $\text{Supp}(D)$ sont disjoints. On appelle *code géométrique* associé à \mathcal{X} , S et D (ou *code de Goppa*) le code

$$\mathcal{G}_{(\mathcal{X}, S, D)} = \{(f(P_1), \dots, f(P_n)), f \in \mathcal{L}(D)\}. \quad (37)$$

Tout comme les codes de Reed-Solomon, le code $\mathcal{G}_{(\mathcal{X}, S, D)}$ est un code par évaluation.

Théorème 498. On note D_S le diviseur $D_S = (P_1) + \dots + (P_n)$. Le code $\mathcal{G}_{(\mathcal{X}, S, D)}$ est de longueur n , de dimension $k = \dim \mathcal{L}(D) - \dim \mathcal{L}(D - D_S)$ et de distance minimale $d \geq n - \deg D$.

En particulier, si $2g - 2 < \deg D < n$, on a $k = \deg D + 1 - g$.

Remarque 499. On appelle parfois $d' = n - \deg D$ la *distance assignée*.

Démonstration. 1. La longueur de \mathcal{X} est claire.

2. Notons φ l'application linéaire $\varphi : f \in \mathcal{L}(D) \mapsto (f(P_1), \dots, f(P_n)) \in \mathbb{F}_q^n$. Quel est le noyau de φ ? On a $f \in \ker \varphi$ ssi $f(P_i) = 0$ pour tout $1 \leq i \leq n$ ou encore ssi $\text{ord}_{P_i}(f) \geq 1$. Ceci revient à dire que $f \in \mathcal{L}(-D_S)$. Nous en déduisons, comme D et D_S ont des supports disjoints, que $\ker \varphi = \mathcal{L}(D - D_S)$. Il s'ensuit que $k = \dim \mathcal{L}(D) - \dim \mathcal{L}(D - D_S)$ par le théorème du rang.
3. Concernant la distance minimale, donnons-nous $f \in \mathcal{L}(D)$ non nul tel que $c = \varphi(f)$ est un mot de code de poids d . Sans perte de généralité, on peut supposer les dernières coordonnées de c sont nulles : $f(P_{d+1}) = \dots = f(P_n) = 0$. Cela signifie que $\text{ord}_{P_i}(f) \geq 1$ pour $d+1 \leq i \leq n$. Notons D_d le diviseur $D_d = (P_{d+1}) + \dots + (P_n)$. Comme D et D_d possèdent des supports disjoints, on a que $\text{div}(f) + D - D_d \geq 0$, autrement dit, $f \in \mathcal{L}(D - D_d)$. Comme $\mathcal{L}(D - D_d)$ est non-nul, d'après la proposition 487, $\deg(D - D_d) \geq 0$, ce qui équivaut à $d \geq n - \deg D$.
4. À présent si $\deg D < n$, alors $\deg(D - D_S) < 0$. Mais alors, d'après la proposition 487, $\mathcal{L}(D - D_S) = 0$. D'autre part, le théorème de Riemann-Roch indique dans le cas $2g - 2 < \deg D$ que $\dim \mathcal{L}(D) = \deg D + 1 - g$

□

Remarque 500. On déduit du théorème que si $2g - 2 < \deg D < n$, on a

$$k + d \geq n + 1 - g. \quad (38)$$

Pour $g = 0$, c'est la borne de Singleton; plus g est grand, plus le code s'éloigne de cette borne.

Exemple 501 (Codes de Reed-Solomon). Soit $\mathcal{X} = \mathbb{P}^1$ la droite projective sur \mathbb{F}_q (voir la section 3.4 de ²³). C'est une courbe de genre 0 (d'après 3.10.4 de *ibidem*. ou via la formule de Plücker par exemple).

²³. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

La droite $\mathcal{X}(\mathbb{F}_q)$ contient les q points affines $P_a = (a : 1)$ pour $a \in \mathbb{F}_q$ et le point à l'infini $P_\infty = (1 : 0)$; la droite $\mathcal{X}(\overline{\mathbb{F}_q})$ contient en plus les points $P_a = (a : 1)$ pour $a \in \overline{\mathbb{F}_q}$.

Nous choisissons $D = (k-1)(P_\infty)$. Alors l'espace $\mathcal{L}(D)$ est formé des éléments de $\mathbb{F}_q(\mathbb{P}^1)$, autrement dit des fractions rationnelles $f = \frac{g}{h}$, telles que $\text{ord}_{P_a}(f) \geq 0$ pour tout $a \in \overline{\mathbb{F}_q}$ et $\text{ord}_{P_\infty}(f) = \deg h - \deg g \geq k-1$. La première condition montre que f n'a pas de pôles : donc $g = 1$ et f est un polynôme, la seconde que $\deg f < k$.

Nous choisissons enfin $S = \{P_a; a \in \mathbb{F}_q^\times\}$. Nous avons reconstruit les codes de Reed-Solomon.

CODES ISSUES DE COURBES PLANES

Courbes elliptiques : L'exemple le plus simple de construction de codes provient de courbes elliptiques (les courbes de genre 1). D'après la remarque 500, nous savons déjà qu'elle peuvent fournir de bons codes.

Prenons la courbe \mathcal{E} d'équation (affine)

$$f(x, y) = x^2 + x - y^3 \in \mathbb{F}_2[x, y]$$

ou d'équation projective

$$F(X, Y, Z) = X^2Z + XZ^2 - Y^3 \in \mathbb{F}_2[X, Y, Z].$$

On emploie le critère d'Eisenstein, avec $f_3(x) = 1$, $f_1(x) = f_2(x) = 0$, $f_0(x) = x^2 + x$ et $p(x) = x$, pour s'assurer que cette équation définit une courbe irréductible. On pourrait aussi voir à la main qu'une factorisation de la forme $x^2 + x - y^3 = (x - t_1(y))(x - t_2(y))$ conduit à une contradiction.

On a

$$\frac{\partial F}{\partial X} = Z^2, \quad \frac{\partial F}{\partial Y} = Y^2 \text{ et } \frac{\partial F}{\partial Z} = X^2,$$

qui ne s'annulent pas simultanément sur $\mathcal{X}(\overline{\mathbb{F}_2})$. La courbe est donc lisse. Le degré est $d = 3$; elle est bien de genre $g = \frac{(d-1)(d-2)}{2} = 1$ d'après la formule de Plücker (théorème 489).

Sur \mathbb{F}_2 , la courbe possède un point à l'infini et deux points affines

$$P_\infty = (1 : 0 : 0), \quad P_1 = (0 : 0 : 1) \quad \text{et} \quad P_2 = (1 : 0 : 1),$$

ce qui est peu pour construire un code intéressant.

Sur $\mathbb{F}_4 = \{0, 1, \alpha, \alpha^2\}$ où $\alpha^2 + \alpha + 1 = 0$, la courbe possède alors 9 points, les trois points précédents ainsi que

$$P_3 = (\alpha : 1 : 1) \quad P_4 = (\alpha : \alpha : 1) \quad P_5 = (\alpha : \alpha^2 : 1)$$

$$P_6 = (\alpha^2 : 1 : 1) \quad P_7 = (\alpha^2 : \alpha^2 : 1) \quad P_8 = (\alpha^2 : \alpha : 1)$$

Nous avons obtenu ces points en résolvant successivement les équations $F(X, Y, 0) = 0$, $f(1, y) = 0$ puis $f(\alpha, y, 1)$ et $f(\alpha^2, y, 1)$. Noter qu'une racine $(\alpha : y_0 : 1)$ de $f(\alpha, y, 1)$ donne directement par conjugaison que $(\alpha^2 : y_0^2 : 1)$ est racine de $f(\alpha^2, y, 1)$ et vice versa. La borne de Weil (théorème 495) indique que $||\mathcal{X}(\mathbb{F}_4)| - (4 + 1)| \leq 2 \cdot 1 \cdot \sqrt{4}$, soit $|\mathcal{X}(\mathbb{F}_4)| \leq 9$: nous ne pouvons rêver de courbe plus fournie pour appliquer notre méthode.

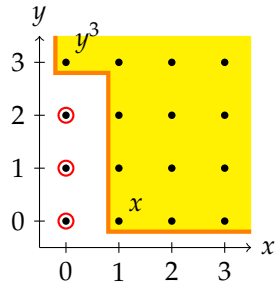
Dans le but de construire un code, prenons comme diviseur $D = r \cdot (P_\infty)$ et $S = \{P_i, 1 \leq i \leq 8\}$. Nous savons immédiatement par le théorème 498 qu'il existe des codes de paramètre $[8, r, n - r]_4$. Nous vérifions sur le site <http://www.codetables.de/> : ce sont les meilleurs codes possibles pour $2 \leq r \leq 6$

Déterminons une base de $\mathcal{L}(D)$ pour en déduire une matrice génératrice du code et cherchons là sous la forme de monômes $x^i y^j$. Calculons les diviseurs principaux tels que $\text{div}(X/Z)$ et $\text{div}(Y/Z)$, i.e. la différence entre zéros et pôles comptés avec multiplicité. Il nous faut d'abord trouver les intersections de $\{X = 0\}$, $\{Y = 0\}$ et $\{Z = 0\}$ avec \mathcal{X} avec leur multiplicité²⁴.

— On a $\{X = 0\} \cap \mathcal{X} = \{X = 0 \wedge X^2 Z + X Z^2 - Y^3 = 0\} = \{X = Y = 0\} = P_1$. De plus la multiplicité de P_1 est (qui peut se calculer en prenant par exemple $v = x$) vaut

$$\dim \mathbb{F}_q[x, y] / \langle x, x^2 + x - y^3, v^3 \rangle = 3.$$

En effet, une base de Gröbner de $\langle x, x^2 + x - y^3, x^3 \rangle$ est $\{y^3, x\}$ et le diagramme en escalier des termes dominants de cette base compte 3 termes.



(Et accessoirement, le théorème de Bézout est bien vérifié).

— Ensuite, $\{Y = 0\} \cap \mathcal{X} = \{Y = 0 \wedge X^2 Z + X Z^2 - Y^3 = 0\} = \{Y = 0 \wedge X Z(X + Z) = 0\} = \{Y = 0 \wedge X = 0\} \cup \{Y = 0 \wedge Z = 0\} \cup \{Y = 0 \wedge X + Z = 0\} = \{P_1, P_\infty, P_2\}$.

La multiplicité de P_1 peut se calculer avec $v = x$:

$$\dim \mathbb{F}_q[x, y] / \langle y, x^2 + x - y^3, v^3 \rangle = 1$$

car une base de Gröbner de $\langle x, x^2 + x - y^3, x^3 \rangle$ est $\{x, y\}$.

24. Il vaut mieux travailler en projectif directement, i.e. sur $\frac{X}{Z}$ au lieu de x , faute de quoi on oublierait ce qui se passe à l'infini.

La multiplicité de P_2 peut se calculer avec $v = x - 1$:

$$\dim \mathbb{F}_q[x, y] / \langle y, x^2 + x - y^3, v^3 \rangle = 1$$

car une base de Gröbner de $\langle x, x^2 + x - y^3, (x - 1)^3 \rangle$ est $\{x - 1, y\}$.

La multiplicité de P_∞ peut se calculer avec $v = z$:

$$\dim \mathbb{F}_q[y, z] / \langle y, z + z^2 - y^3, v^3 \rangle = 1$$

car une base de Gröbner de $\langle y, z + z^2 - y^3, z^3 \rangle$ est $\{y, z\}$.

— Enfin, $\{Z = 0\} \cap \mathcal{X} = \{Z = 0 \wedge X^2Z + XZ^2 - Y^3 = 0\} = \{Z = Y = 0\} = P_\infty$. La multiplicité de P_∞ est $\dim k[y, z] / \langle z, z + z^2 + y^3, z^3 \rangle = 3$.

On en déduit que

$$\operatorname{div}(X/Z) = 3(P_1) - 3(P_\infty)$$

$$\text{et } \operatorname{div}(Y/Z) = (P_1) + (P_2) - 2(P_\infty).$$

(Vérification : le degré vaut chaque fois 0, ce qui est le cas pour un diviseur principal).

Revenons à la détermination d'une base de $\mathcal{L}(D) = \mathcal{L}(r \cdot P_\infty)$. On sait par le théorème de Riemann-Roch que cet espace est de dimension $\deg D + 1 - g = r$ (car on est dans le cas $\deg D = r > 2g - 2 = 0$). On calcule que

$$\operatorname{div} \left(\frac{X^i Y^j}{Z^{i+j}} \right) = (3i + j) \cdot (P_1) + j \cdot (P_2) - (3i + 2j) \cdot (P_\infty)$$

par multiplicativité des ordres (encore une fois on peut vérifier que le degré est nul car il s'agit d'un diviseur principal).

Par ailleurs, on vérifie que $1, y, x, y^2, xy, y^3, xy^2$ forment des fonctions linéairement indépendantes. En effet, les ordres en P_∞ de ces fonctions sont échelonnés.

	1	y	x	y ²	xy	y ³	xy ²
$-\operatorname{ord}_{P_\infty}$	0	2	3	4	5	6	7

Mais l'ordre vérifie $\operatorname{ord}(f + g) \geq \min(\operatorname{ord}(f), \operatorname{ord}(g))$ (cf. prop 3.2.3(ii) de ²⁵), donc aucune de ces fonctions ne peut être une combinaison linéaire des précédentes. Le tableau 1 donne une liste de bases possibles pour $\mathcal{L}(D) = \mathcal{L}(r \cdot P_\infty)$.

Nous pouvons déduire les matrices génératrices des codes constuites des premières lignes du tableau d'évaluation (cf. table 2).

Courbes hermitiennes : On considère la courbe \mathcal{H} définie sur \mathbb{F}_{q^2} par l'équation affine

$$f(x, y) = x^q + x - y^{q+1} \in \mathbb{F}_{q^2}[x, y]$$

²⁵. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

TABLE 1: Bases de $\mathcal{L}(r \cdot P_\infty)$

r	Base de $\mathcal{L}(r \cdot P_\infty)$
1	1
2	$1, y$
3	$1, y, x$
4	$1, y, x, y^2$
5	$1, y, x, y^2, xy$
6	$1, y, x, y^2, xy, y^3$
7	$1, y, x, y^2, xy, y^3, xy^2$

TABLE 2: Évaluations et matrice génératrice

Évaluation de	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
1	1	1	1	1	1	1	1	1
y	0	0	1	α	α^2	1	α^2	α
x	0	1	α	α	α	α^2	α^2	α^2
y^2	0	0	1	α^2	α	1	α	α^2
xy	0	0	α	α^2	1	α^2	α	1
y^3	0	0	1	1	1	1	1	1
xy^2	0	0	α	1	α^2	α^2	1	α

ou l'équation projective

$$F(X, Y, Z) = X^q Z + XZ^q - Y^{q+1} \in \mathbb{F}_{q^2}[X, Y, Z].$$

Le critère d'Eisenstein (avec $p(x) = x$) montre que la courbe est irréductible. Par ailleurs,

$$\frac{\partial F}{\partial X} = Z^q, \quad \frac{\partial F}{\partial Y} = -Y^q \text{ et } \frac{\partial F}{\partial Z} = X^q$$

ne s'annulent pas simultanément, donc la courbe est lisse. D'après la formule de Plücker (théorème 489), la courbe est de genre $g = \frac{q(q-1)}{2}$.

En réalité, on peut remarquer que $f(x, y) = \text{Trace}_{\mathbb{F}_{q^2}/\mathbb{F}_q}(x) - \mathcal{N}_{\mathbb{F}_{q^2}/\mathbb{F}_q}(y)$, ce qui nous aide à compter les points de $\mathcal{H}(\mathbb{F}_{q^2})$. Comme la trace est un morphisme surjectif sur \mathbb{F}_q et que la norme est à valeur dans \mathbb{F}_q , pour tout $y_0 \in \mathbb{F}_{q^2}$, l'équation $f(x, y_0)$ admet $|\ker(\text{Trace})| = q$ racines. Il y a donc q^3 points affines. Il n'y a qu'un seul point à l'infini $P_\infty = (1 : 0 : 0)$. Au total, \mathcal{H} compte $q^3 + 1$ points sur \mathbb{F}_{q^2} . La borne de Weil (cf. théorème 495) prévoit que

$$|\mathcal{H}(\mathbb{F}_{q^2})| \leq q^2 + 1 + 2 \frac{q(q-1)}{2} \sqrt{q^2} = q^3 + 1.$$

Le nombre de points de la courbe est maximal.

On choisit $D = r \cdot (P_\infty)$ et $S = \mathcal{H}(\mathbb{F}_{q^2}) \setminus \{P_\infty\}$. D'après le théorème 498, on peut construire des codes géométriques de paramètre

$$\left[q^3, r+1 - \frac{q(q-1)}{2}, q^3 - \frac{q(q-1)}{2} \right]_{q^2}$$

pour $2g-2 < r < n$. Montrons que l'on peut préciser une base sous la forme de monômes $x^i y^j$.

Commençons par vérifier les conditions d'appartenance de $x^i y^j$ à $\mathcal{L}(rP_\infty)$ et calculons pour cela $\text{div}(x) = \text{div}\left(\frac{X}{Z}\right)$ et $\text{div}\left(\frac{Y}{Z}\right)$. Il nous faut déterminer la différence des zéros et pôles comptés avec leur multiplicité de chaque fraction rationnelle.

— On a $\{X = 0 \wedge X^q Z + XZ^q + Y^{q+1} = 0\} = \{X = 0 \wedge Y = 0\} = \{P_0 = (0 : 0 : 1)\}$. De plus la multiplicité de ce zéro est (en prenant $v = x$)

$$\dim \mathbb{F}_p[x, y] / \langle x, x^q + x - y^{q+1}, x^{q+1} \rangle = \dim \mathbb{F}_p[x, y] / \langle x, y^{q+1} \rangle = q+1.$$

— On a $\{Y = 0 \wedge X^q Z + XZ^q + Y^{q+1} = 0\} = \{Y = 0 \wedge XZ(X^{q-1} + Z^{q-1}) = 0\} = \{P_0, P_\infty\} \cup \{P_\xi : \xi^{q-1} = -1\}$ où $P_a = (a : 0 : 1)$. Cela forme $q+1$ points distincts (on cherche $\xi^{q-1} = -1$ dans la clôture algébrique). À cause du théorème de Bézout, ils doivent tous être de multiplicité 1.

— On a $\{Z = 0 \wedge X^q Z + XZ^q + Y^{q+1} = 0\} = \{Z = 0 \wedge Y = 0\} = \{P_\infty\}$. Pour calculer la multiplicité, il est commode de se placer dans le plan affine $X = 1$ autour de $(y = 0, z = 0)$. En prenant $v = y$, on obtient $\dim \mathbb{F}_p[y, z] / \langle z, z + z^q - y^{q+1}, y^{q+1} \rangle = \dim \mathbb{F}_p[x, y] / \langle z, y^{q+1} \rangle = q+1$.

En conclusion, $\text{div}\left(\frac{X}{Z}\right) = (q+1)P_0 - (q+1)P_\infty$ et $\text{div}\left(\frac{Y}{Z}\right) = P_0 + \sum_{\xi^{q-1}=-1} (P_\xi) - q(P_\infty)$. Donc

$$\text{div}(x^i y^j) = ((q+1)i + j) \cdot (P_0) + j \sum_{\xi^{q-1}=-1} (P_\xi) - ((q+1)i + qj) \cdot (P_\infty)$$

On considère la suite

	1	y	x	y ²	xy	x ²	y ³	y ² x	yx ²	x ³
—ord _{P₀}	0	q	q+1	2q	2q+1	2q+2	3q	3q+1	3q+2	3q+3
	...	x ^k	x ^{k-1} y	...	x ^{k-q+1} x ^{q-1}	...				
—ord _{P_∞}	...	kq	kq+1	...	(k+1)q-1					

Les ordres des points sont échelonnés et on vérifie que $A = \{(i, j) \in \mathbb{N}^2; (q+1)i + qj \leq r \wedge i \leq q-1\}$ a pour cardinal $r+1 - \frac{q(q-1)}{2}$. On en déduit que $\{x^i y^j; (i, j) \in A\}$ forme une base de $\mathcal{L}(rP_\infty)$. Ceci permet de construire les matrices génératrices des codes promis en évaluant la suite des polynômes sur l'ensemble des points.

Exercice 502. On travaillera dans cet exercice avec $\mathbb{F}_4 = \mathbb{F}_2[\alpha]/(\alpha^2 + \alpha + 1)$, $\mathbb{F}_8 = \mathbb{F}_2[\beta]/(\beta^3 + \beta + 1)$ et $\mathbb{F}_{16} = \mathbb{F}_2[\gamma]/(\gamma^4 + \gamma + 1)$. Soit \mathcal{E} la courbe d'équation affine

$$f(x, y) = y^2 + y + x^3 + x + 1 \in \mathbb{F}_2[x, y].$$

1. Donner une équation projective $F(X, Y, Z) = 0$ avec $F(X, Y, Z) \in \mathbb{F}_2[X, Y, Z]$.
2. Montrer que \mathcal{E} est irréductible, lisse et calculer le genre de \mathcal{E} .
3. Déterminer les points de \mathcal{E} sur \mathbb{F}_2 , \mathbb{F}_4 , \mathbb{F}_8 et \mathbb{F}_{16} . Comparer le nombre de points avec la borne de Weil.
4. Contrôler vos calculs avec SageMath.
5. Déterminer $\text{div}(f)$ pour les fonctions rationnelles $\theta_{\alpha, \beta}(x, y) = x^\alpha y^\beta$ avec $\alpha, \beta \in \mathbb{N}$.
6. Soit P_∞ l'unique point à l'infini de \mathcal{E} et $n = |\mathcal{E}(\mathbb{F}_{16})| - 1$. Trouver une base de $\mathcal{L}(rP_\infty)$ pour $0 \leq r \leq n$.
7. Construire un code $[4, r, 4 - r]_4$, $[12, r, 12 - r]_8$ et $[24, r, 24 - r]_{16}$ pour différentes valeurs de r .
8. Vérifier les paramètres des codes avec SageMath, comparer distance assignée et distance minimale.
9. Comparer les paramètres des codes obtenus avec les tables de www.codetables.de.

Exercice 503 (Quartique de Klein). On appelle *quartique de Klein* la courbe \mathcal{K} d'équation affine dans $\mathbb{F}_q[x, y]$

$$f(x, y) = x^3y + y^3 + x = 0$$

On pose $P_1 = (1 : 0 : 0)$, $P_2 = (0 : 1 : 0)$, $P_3 = (0 : 0 : 1)$.

1. Montrer que f est irréductible. Qu'en déduire de \mathcal{K} ?
2. Montrer que \mathcal{K} est lisse. Calculer le genre de \mathcal{K} .
3. Compter le nombre de points sur \mathbb{F}_q pour $q = 2, 4, 8$. Pour construire \mathbb{F}_8 , on pourra utiliser après justification un élément α tel que $\alpha^3 + \alpha + 1 = 0$. Comparer avec la borne de Weil et la borne de Serre.
4. Calculer $\text{div}(x)$ et $\text{div}(y)$.
5. Donner un base de $\mathcal{L}(r \cdot P_3)$ pour $r \geq 0$ formées de fonctions de la forme $\frac{y^i}{x^j}$.
6. En déduire la base d'un code de Goppa avec $S = \mathcal{K}(\mathbb{F}_8) \setminus \{P_3\}$, $D = r \cdot (P_3)$ et en donner les paramètres.

Exercice 504 (Courbe de Fermat). On appelle *courbe de Fermat*²⁶ la courbe plane \mathcal{F}_9 définie sur \mathbb{F}_2 par

$$F(X, Y, Z) = X^9 + Y^9 + Z^9 = 0.$$

On pose $R_a = (1 : a : 0)$.

26. Le grand théorème de Fermat affirme que les courbes en général $X^m + Y^m = Z^m$ ne possèdent pas de points non triviaux sur \mathbb{Q} pour $m \geq 3$.

1. Montrer que \mathcal{F}_9 irréductible, lisse et calculer son genre.
2. Rechercher les points à l'infini. On notera D_0 leur somme.
3. Vérifier que \mathbb{F}_8 est un sous-corps de \mathbb{F}_{64} . Montrer que l'application $\mathbb{F}_8 \rightarrow \mathbb{F}_8, x \mapsto x^9$ est une bijection. Montrer que l'application $\mathbb{F}_{64}^\times \rightarrow \mathbb{F}_8^\times, x \mapsto x^9$ est une surjection et que chaque pré-image compte 9 éléments.
4. Calculer $|\mathcal{F}_9(\mathbb{F}_2)|, |\mathcal{F}_9(\mathbb{F}_8)|$ et $|\mathcal{F}_9(\mathbb{F}_{64})|$. Comparer avec la borne de Weil. Vérifier vos calculs avec SageMath.
5. Soit le diviseur $D_0 = \sum_{\zeta^9=1} R_\zeta$. Quels sont les paramètres du code de Goppa associé à $r \cdot D_0$ et $S = \mathcal{F}_9(\mathbb{F}_{64}) \setminus \text{Supp}(D_0)$? Montrer que l'ensemble $\{x^i y^j; i+j \leq r \text{ et } i \leq 8\}$ est une base de $\mathcal{L}(rD_0)$.

La borne de Tsfasman-Vladut-Zink

Repartons de la remarque 500. Nous avons vu que pour un $[n, k, d]_q$ -code géométrique construit à partir d'une courbe de genre g , on a l'inégalité $k + d \geq n + 1 - g$, soit encore, en notant R le rendement et δ la distance minimale relative, un code vérifiant :

$$R + \delta \geq 1 - \frac{g-1}{n}.$$

Quel genre de quotient $\frac{g-1}{n}$ peut-on obtenir quand n grandit? Un petit quotient serait idéal pour battre la borne asymptotique de Gilbert-Varshamov.

Définition 505. Soit les deux nombres suivants

$$N_q(g) = \max \{ |\mathcal{X}(\mathbb{F}_q)|; \mathcal{X} \text{ est une courbe sur } \mathbb{F}_q \text{ de genre } g \} \quad (39)$$

et

$$A(q) = \limsup_{g \rightarrow +\infty} \frac{N_q(g)}{g}. \quad (40)$$

Une famille de courbes $(\mathcal{X}_i)_i$ de genre g_i et de cardinal n_i sur \mathbb{F}_q s'approchant de la limite $\frac{n_i}{g_i} \rightarrow A(q)$ permet de construire une famille de codes géométriques $\mathcal{G}_{(\mathcal{X}_i, S_i, D_i)}$ par le choix arbitraire d'un diviseur $D_i = r_i(Q_i)$ et de l'ensemble de points $S = \mathcal{X}(\mathbb{F}_q) \setminus \{Q_i\}$ où Q_i est un point quelconque. Si l'on note R et δ la limite du rendement et de la distance minimale relative de cette suite de codes,

$$R + \delta \geq 1 - \frac{1}{A(q)}. \quad (41)$$

Il est hors de nos possibilités d'étudier $A(q)$. Remarquons aussi que si l'on se limite à des courbes planaires, $|\mathcal{X}(\mathbb{F}_q)| \leq |\mathbb{P}^2(\mathbb{F}_q)| = q^2 + q + 1$: nous ne parviendrons même pas à montrer que $A(q) > 0$ avec elles. Nous pourrions aussi observer que les bornes de Weil du théorème

495 sont trop faibles pour calculer $A(q)$. Du côté positif, nous nous contentons de citer les résultats suivants. Commençons par préciser ce que l'on peut attendre de mieux.

Théorème 506 (Drinfeld-Vladut 83). *Pour toute puissance de premier q , on a*

$$A(q) \leq \sqrt{q} - 1. \quad (42)$$

Par ailleurs, certaines valeurs de $A(q)$ sont précisées par les théorèmes suivants (le second est en cours de publication).

Théorème 507 (Ihara 79). *Soit $q = p^n$ une puissance d'un premier p telle que n est pair, alors*

$$A(q) \geq \sqrt{q} - 1. \quad (43)$$

En combinant ce théorème avec le précédent, nous avons en réalité un cas d'égalité.

Théorème 508 (Garcia-Stichtenoth-Bassa-Beelen 12). *Soit $q = p^n$ une puissance d'un premier p telle que $n = 2m + 1 \geq 3$ est impair, alors*

$$A(q) \geq \frac{2(p^{m+1} - 1)}{p + 1 + \epsilon} \text{ avec } \epsilon = \frac{p - 1}{p^m - 1}. \quad (44)$$

En conséquence, nous pouvons montrer qu'il existe des codes géométriques meilleurs que la borne de Gilbert-Varshamov dans les cas suivants.

Théorème 509. *Pour tout $q \geq 49$ non premier, sauf peut-être pour $q = 125$, il existe des codes géométriques arbitrairement longs meilleurs que la borne asymptotique de Gilbert-Varshamov.*

Ce résultat est dû pour partie à Tsfasman-Vladut-Zink en 1982 qui ont été les premiers à remarquer que l'on peut dépasser la borne de Gilbert-Varshamov. Philosophiquement, cela signifie que des codes structurés permettent de faire mieux que des codes tirés au hasard.)

Démonstration. Les théorèmes 507 ou 508 montrent que pour des valeurs de q telles que dans les hypothèses du théorème, la droite d'équation $R + \delta = 1 - \frac{1}{A(q)}$ coupe deux fois la courbe d'équation $R = 1 - H_q(\delta)$ pour $0 < \delta < 1$.

□

Exercice 510. Reprendre l'exercice 472 lorsque $q = 49$ en ajoutant la droite d'équation $R + \delta = 1 - \frac{1}{\sqrt{q}-1}$. Que peut-on noter ?

Remarque 511. Le cas le plus intéressant compte tenu des applications reste le cas $q = 2$. Dans ce cas, on ne sait pas s'il est possible d'améliorer la borne de Gilbert-Varshamov.

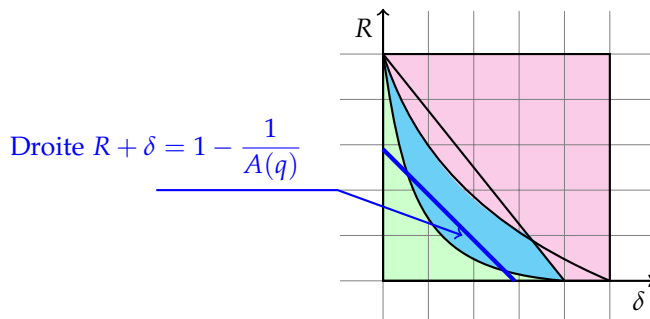


FIGURE 35: Codes algébriques et borne de Gilbert-Varshamov

Application à la cryptographie

Dans la compétition organisée par le N.I.S.T. pour la standardisation de la cryptographie post-quantique, les codes correcteurs d'erreur occupent une place de choix. Le cryptosystème le plus emblématique de cette famille de primitives cryptographique est le *chiffrement de McEliece*, inventé en 1978. À ce jour, ce système est peu utilisé à cause de la très grande taille de clé qu'il demande.

Voici les fonctions de ce cryptosystème.

Génération de clés Alice tire au hasard un $[n, k]_2$ -code correcteur d'erreur de matrice génératrice \mathbf{G} qui possède un algorithme de décodage jusqu'à t erreurs. Alice tire au hasard une matrice $\mathbf{S} \in \text{GL}_k(\mathbb{F}_2)$ et une matrice de permutation \mathbf{P} . Alice calcule $\hat{\mathbf{G}} = \mathbf{SGP}$.

Clé privée $(\mathbf{S}, \mathbf{G}, \mathbf{P})$

Clé publique $(\hat{\mathbf{G}}, t)$

Chiffrement Bob veut envoyer $\mathbf{m} \in \mathbb{F}_2^k$. Il génère un mot $\mathbf{z} \in \mathbb{F}_2^n$ de poids t . Il transmet le mot de code bruité $\mathbf{c} = \mathbf{m}\hat{\mathbf{G}} + \mathbf{z}$.

Déchiffrement Alice retrouve \mathbf{m} en calculant $\mathbf{c}' = \mathbf{c}\mathbf{P}^{-1}$, en décodant \mathbf{c}' vers \mathbf{m}' , puis en calculant $\mathbf{m} = \mathbf{m}'\mathbf{S}^{-1}$.

Le cryptosystème de McEliece s'emploie souvent avec des codes de Goppa (codes géométriques), pour lesquels on peut généraliser les méthodes de décodage vues dans le cas particulier des codes de Reed-Solomon.

Exemple 512. On illustre ci-dessous un échange de message entre Alice et Bob par le système de McEliece.

```

Fp = GF(11)
n, k = 10, 5
C = codes.GeneralizedReedSolomonCode(
    Fp.list()[1:n+1], k)
G = C.generator_matrix() # cle prive
E = C.encoder() # prive
d = C.minimum_distance()
D_alice = C.decoder('BerlekampWelch')
S = matrix(Fp,[[4, 8, 7, 10, 1],
               [6, 9, 2, 1, 6],
               [8, 0, 4, 5, 5],
               [0, 2, 7, 10, 9],
               [5, 1, 7, 6, 8]]) # cle privee
P = matrix(Fp,
            Permutation([5,9,1,6,2,7,4,8,10,3]
            ).to_matrix()) # cle privee
GG = S*G*P # cle publique
m = random_vector(Fp,k) # message de Bob
chan = channels.StaticErrorRateChannel(
    F^n, d//2-1)
c = chan.transmit(CC.encode(m)) # transmis par Bob
mm = vector(D.decode_to_message(
    c*P^(-1)))*S^(-1) #decodage par Alice
m == mm # verification

```


TP 12 : Cryptanalyse et cryptographie à base de réseaux

Buts : Appliquer l'algorithme LLL dans le cadre d'attaques cryptographiques classiques, construire des cryptosystèmes à base de réseaux.

Travaux préparatoires : Relire le TP sur les réseaux euclidiens et le cours du jour.

Évaluation du TP : Exercices 522 (sac à dos), 531 (attaque de Wiener), 526 (méthode de Coppersmith), 527 (messages stéréotypés), 542 (Babai), 546 (cryptosystème GGH).

Très vite après la découverte de l'algorithme LLL, des applications fulgurantes en cryptanalyse ont été obtenues. Elles ont valu aux réseaux une réputation sulfureuse qui n'a été brisée qu'au tournant des années 2000. Aujourd'hui, les réseaux sont énormément étudiés par les cryptographes pour deux raisons. D'une part, la difficulté des problèmes liés aux réseaux en font un objet de choix pour construire une cryptographie qui résisterait à des ordinateurs quantiques. D'autre part, c'est grâce aux réseaux que l'on a pu construire des mécanismes de chiffrement homomorphe, autrement dit des mécanismes tels que la fonction de chiffrement commute avec les opérations usuelles (addition, multiplication, etc).

Depuis 2017, le NIST (National Institute of Standards and Technology) américain a lancé une compétition afin de définir les futurs standards de cryptographie postquantique (voir <https://csrc.nist.gov/Projects/Post-Quantum-Cryptography>). En avril 2018, il restait 64 propositions de protocoles encore en course. Parmi eux, 26 étaient basés sur des réseaux euclidiens et 19 sur des codes correcteurs, objets que l'on voit souvent comme des analogues finis des réseaux.

Réseaux et cryptanalyse

Cryptographie asymétrique

Définition 513. La *cryptographie asymétrique* permet à un interlocuteur (Bob) d'envoyer un message à un destinataire (Alice) sans qu'aucun

espion (Ève) ne puisse accéder à ce message. Elle se fait selon le protocole suivant, résumé à la figure 38.

Etape 0 (Génération de clés) Alice génère aléatoirement une paire de clés $(K_a, K_b) = \text{GenClé}()$, appelées *clé privée* et *clé publique*. Elle diffuse la clé publique K_b .

Etape 1 (Chiffrement) Bob chiffre son *clair* m avec K_b . Il envoie le *chiffré* $c = \text{Chiffrement}(m, K_b)$ à Alice.

Etape 2 (Déchiffrement) Alice déchiffre le chiffré c de Bob avec K_a . Elle obtient le clair $m = \text{Déchiffrement}(c, K_a)$.

Attaque passive Ève écoute toutes les transmissions effectuées entre Alice et Bob et connaît les trois fonctions employées par Alice et Bob.

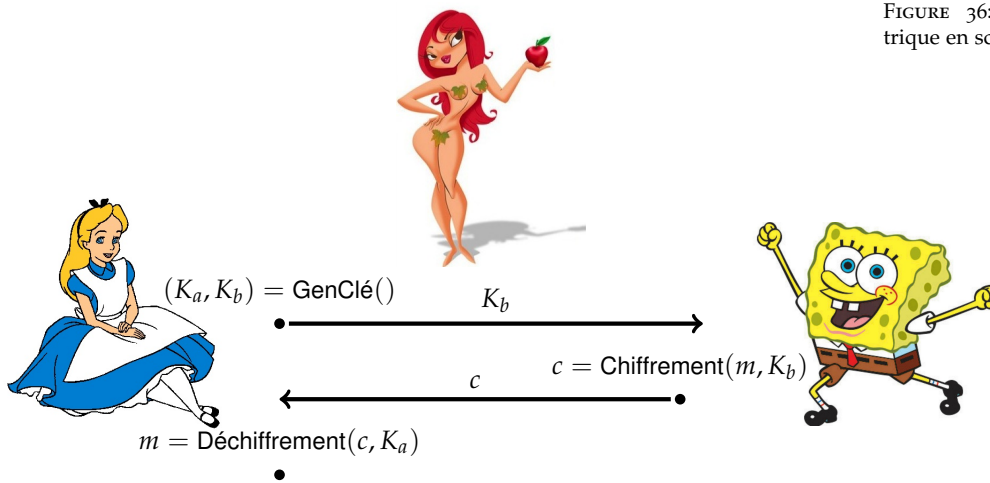


FIGURE 36: La cryptographie asymétrique en schéma

Remarque 514. On parle de chiffement par une fonction à trappe : Alice possède une information secrète lui permettant d'inverser la fonction de chiffement plus facilement que son adversaire.

Exemple 515 (RSA). L'algorithme de cryptographie asymétrique le plus connu est le chiffement RSA. Nous en rappelons les fonctions.

Génération de clés Alice tire au hasard deux nombres premiers p et q . Elle pose $N = pq$. Elle construit un couple d'entiers (e, s) tels que

$$e \cdot s \equiv 1 \pmod{\phi(N) = (p-1)(q-1)}$$

Clé privée (N, s)

Clé publique (N, e)

Chiffement Bob transforme $m \in \mathbb{Z}/N\mathbb{Z}$ en $c = m^e \pmod{N}$.

Déchiffement Alice retrouve m en calculant $m = c^s \pmod{N}$.

À ce jour, des clés de taille 2048 bits sont considérées comme obso-
lètes. L'ANSSI recommande des clés de taille supérieur à 3072 bits²⁷.

Application 516. Les primitives de chiffrement et de déchiffrement
permettent aussi de construire des schémas de *signature numérique*.

Génération de clés Alice et Bob conviennent d'un cryptosys-
tème de chiffrement asymétrique et d'une fonction de ha-
chage H . Alice génère une clé de chiffrement K_b et une
clé de déchiffrement K_a . Alice rend publique la clé K_b . La
clé K_a reste secrète.

Signature Afin d'authentifier un document m , Alice calcule
une signature $s = \text{Déchiffrement}(H(m), K_a)$. Alice trans-
met m et s à Bob.

Vérification Bob calcule $H(m)$ et $\text{Chiffrement}(s, K_b)$ et s'as-
sure que les résultats sont égaux.

Attaque La fonction Déchiffrement étant difficile à calculer
sans connaître la clé privée K_a , Mallory, un attaquant ma-
licieux, ne peut pas se faire passer pour Alice quand elle
envoie un message m' à Bob sauf si elle est capable de
casser le cryptosystème employé.

27. ANSSI. Guide de sélection d'algo-
rithmes cryptographiques. Technical Re-
port ANSSI-PA-079, Agence nationale de
la sécurité des systèmes d'information,
51, boulevard de La Tour-Maubourg,
75700 PARIS 07, 8 mars 2021. Version
1.0

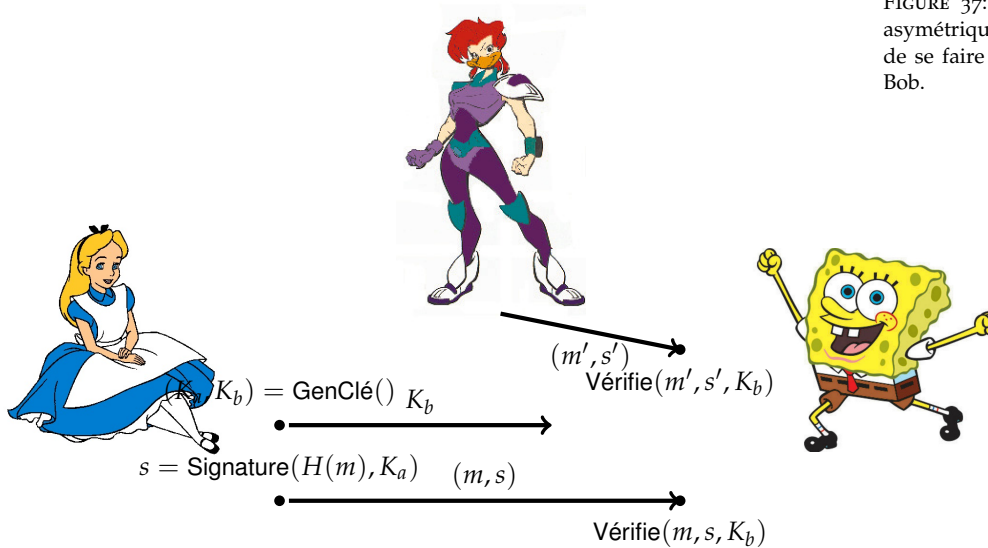


FIGURE 37: La signature cryptographie
asymétrique en schéma : Mallory essaie
de se faire passer pour Alice auprès de
Bob.

Problème du sac à dos et attaque de Lagarias et Odlyzko

Le cryptosystème du sac à dos de Merkle et Hellman date de 1978. Il
est avec RSA (publié la même année) l'un des premiers cryptosystème

asymétrique et semblait à ses débuts promis à un avenir radieux en raison de calculs moins volumineux.

LE PROBLÈME DE LA SOUS-SOMME

La sécurité supposée du cryptosystème et son nom reposent sur le problème NP-complet suivant, qui dérive du problème du sac à dos (*knapsack problem*), standard d'optimisation combinatoire :

Problème 517 (Sous-somme). Soient $\mathbf{a} = (a_1, \dots, a_n)$ une suite d'entiers positifs distincts appelés poids et s un entier obtenu comme somme de certains des poids, trouver un vecteur $\mathbf{x} = (x_i)_{1 \leq i \leq n} \in \{0, 1\}^n$ (appelé vecteur de contenu) tel que $s = \sum_{i=1}^n x_i a_i$.

Le terme sac à dos provient de l'idée que l'on remplit autant que possible un sac à dos de contenance s avec des objets de poids a_i .

Définition 518. Une suite $\mathbf{b} = (b_i)_{1 \leq i \leq n} \subseteq \mathbb{R}$ est dite *supercroissante* lorsque

$$\forall 1 \leq i < n, \quad b_{i+1} > \sum_{j=1}^i b_j.$$

Proposition 519. Lorsqu'une suite $\mathbf{b} = (b_i)_{1 \leq i \leq n}$ est supercroissante, le problème de la sous-somme peut être résolu par l'algorithme glouton suivant. À partir de $s = \sum_{i=1}^n x_i b_i$, on retranche b_n à s si cela est possible pour en déduire x_n et on répète avec l'entier $s' = s - x_n b_n$ et la suite $\mathbf{b}' = (b_1, \dots, b_{n-1})$.

DESCRIPTION DU CRYPTOSYSTÈME DE MERKLE ET HELLMAN Les primitives sont les suivantes.

Génération de clés Alice sélectionne une suite $\mathbf{b} = (b_i)_{1 \leq i \leq n}$ supercroissante. Afin de cacher au public la suite \mathbf{b} , Alice tire au hasard deux entiers $m > \sum_{i=1}^n b_i$ et k tels que $k \wedge m = 1$ et une permutation π de $[n]$. Elle calcule $\mathbf{a} = (a_i)_{1 \leq i \leq n}$ avec $a_i = k b_{\pi(i)} \bmod m$.

Clé privée Alice conserve comme clé privée la suite \mathbf{b} , les entiers k et m ainsi que la permutation π .

Clé publique Elle publie la clé publique \mathbf{a} .

Chiffrement Bob veut transmettre le vecteur $\mathbf{x} = (x_1, \dots, x_n) \in \{0, 1\}^n$ et envoie pour cela $s = \sum_{i=1}^n x_i a_i$.

Déchiffrement Pour déchiffrer le message reçu s , Alice calcule $(k^{-1}s \bmod m) = \sum_i x_i b_{\pi(i)}$. Comme m est suffisamment grand, calculer dans \mathbb{Z} ou modulo m revient à manipuler les mêmes entiers. Elle peut donc en déduire $(x_{\pi^{-1}(1)}, \dots, x_{\pi^{-1}(n)})$ par l'algorithme glouton, puis finalement \mathbf{x} en réordonnant les coordonnées.

ATTAQUE DE LAGARIAS ET ODLYZKO (85)

Soient \mathbf{v} le vecteur et \mathcal{L} le réseau engendré par la base \mathbf{B} suivants :

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & 1 & 0 \\ a_1 & a_2 & \cdots & a_n & -s \end{pmatrix} \quad \mathbf{v} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ 0 \end{pmatrix}$$

où $\mathbf{x} = (x_i)_i$ est le message clair envoyé par Bob. Il est clair que \mathbf{v} appartient au réseau car $\mathbf{v} = \mathbf{B}\mathbf{v}$. Mais notons que sa norme, $\|\mathbf{v}\| \leq \sqrt{n}$, est sensiblement courte que les vecteurs de la base, puisque les a_i sont typiquement de plutôt grands nombres. Un adversaire (Ève) qui applique l'algorithme LLL à la base \mathbf{B} a donc de grande chances d'obtenir comme premier vecteur de la base LLL réduite le vecteur \mathbf{v} . Les travaux de Lagarias et Odlyzko montrent que sous certaines hypothèses sur a la probabilité de cet événement devient écrasante quand n grandit.

Exercice 520. Montrer que toute instance du problème de sous-somme (problème 517) peut être (polynomialement) réduite au cas où le pgcd des poids est 1.

Exercice 521. On donne la suite suivante

$$a = (1, 3, 5, 11, 24, 46, 95, 203)$$

et l'entier $s = 127$. Décomposer s comme une somme d'éléments de a .

Exercice 522. Dans cet exercice, on s'intéresse au cryptosystème du sac à dos.

1. On donne la clé publique $b = [356, 278, 417, 27, 132, 464, 521]$ et le chiffré $s = 1287$. Retrouver le message x à l'aide de SageMath.
2. Composer votre propre cryptosystème (on prendra $n = 8$) et publier votre clé publique. Envoyer un message de votre choix à votre voisin de droite (soit un entier compris entre 0 et 127, soit une chaîne de caractères représentés par leur code ascii). Déchiffrer le chiffré que vous a envoyé votre voisin de gauche. Retrouver le clair à partir du chiffré que votre voisin de droite a envoyé à son voisin de droite.

Petites racines de polynômes et attaques sur RSA

Beaucoup de chercheurs ont proposé diverses attaques *ad hoc* contre RSA que l'on peut abstraire aujourd'hui en une méthode générale d'obtention de petites racines d'un polynôme basée sur une réduction LLL.

LA MÉTHODE DE COPPERSMITH

Pour montrer l'intérêt d'une telle approche, voyons que pour factoriser disons $N = pq$ un produit de deux premiers, il suffit de trouver toutes les racines entières du polynôme $f(x, y) = N - xy \in \mathbb{Z}/N\mathbb{Z}[x, y]$ pour $|x|$ ou $|y| \leq N^{1/2}$: il n'est pas nécessaire de considérer $N^{1/2} < x \leq N$.

Théorème 523 (Coppersmith). *Soit N un entier et*

$$f(x) = x^d + a_{d-1}x^{d-1} + \cdots + a_0 \in \mathbb{Z}/N\mathbb{Z}[x]$$

un polynôme unitaire de degré d . Alors il est possible de calculer en temps polynomial toute racine entière $|x| \leq B$ telle que $f(x) = 0 \pmod{N}$ pour $B = O(N^{1/d})$.

Remarque 524. On peut rendre ce théorème plus précis au sujet de B au prix d'hypothèses compliquées à énoncer.

Démonstration. (Esquisse). Rappelons d'abord que la recherche de racines d'un polynôme de $\mathbb{Z}[x]$ est un problème résolu par des méthodes classiques (voir les techniques du chapitre 7). Si les coefficients de f étaient assez petits, pour que

$$\sum_{i=0}^n |a_i| B^i < N \quad (45)$$

nous pourrions nous contenter de chercher les racines de f en tant que polynôme de $\mathbb{Z}[x]$ et ne retenir que celles $\leq B$. Rien ne garantit cependant a priori que les coefficients de f soient petits.

Cela dit, posons $g_{i,j}(x) = x^j N^i f^{m-i}(x)$ et observons que

$$f(x_0) = 0 \pmod{N} \quad \text{ssi} \quad g_{i,j}(x_0) = 0 \pmod{N^m}$$

pour un certain m fixé. Notre espoir est désormais que l'une des combinaisons linéaires

$$h(x) = \sum_{i,j} v_{i,j} \cdot g_{i,j}(x)$$

de ces polynômes satisfasse l'égalité (semblable à l'équation 45)

$$\sum_i |h_i| B^i \leq N^m$$

et que les petites racines dans \mathbb{Z} de $h(x)$ soient alors exactement les petites racines de $f(x)$ modulo N . Si l'on identifie le polynôme $p(x) = p_0 + p_1x + \cdots + p_kx^k$ au vecteur $(p_0, p_1, \dots, p_k, 0, \dots, 0)$, il revient au même de rechercher un vecteur court par le réseau engendré par $g_{i,j}(Bx)$. Par exemple pour $m = 1$, $0 \leq i \leq m$ et $0 \leq j \leq d \deg f$, le réseau a

pour base les colonnes de la matrice :

$$\begin{pmatrix} x^0 & g_{1,0} & g_{1,1} & \cdots & g_{1,d-1} & g_{0,0} & g_{0,1} & \cdots & g_{0,d-1} \\ x^1 & 0 & BN^m & \ddots & \vdots & Ba_1 & Ba_0 & \ddots & \vdots \\ & \vdots & \ddots & \ddots & 0 & \vdots & \vdots & \ddots & 0 \\ x^{d-1} & \vdots & & \ddots & B^{d-1}N^m & B^{d-1}a_{d-1} & \vdots & & B^{d-1}a_0 \\ x^d & \vdots & & & 0 & B^d & B^da_{d-1} & & \vdots \\ x^{d+1} & \vdots & & & \vdots & 0 & B^{d+1} & \ddots & \vdots \\ & \vdots & & & \vdots & \vdots & \ddots & \ddots & B^{2d-2}a_{d-1} \\ x^{2d-1} & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & B^{2d-1} \end{pmatrix} \quad (46)$$

Or l'algorithme LLL permet de trouver un vecteur \mathbf{v} d'un réseau \mathcal{L} tel que $\|\mathbf{v}\| \leq O\left((\text{disc } \mathcal{L})^{1/\dim \mathcal{L}}\right)$, ce qui suffit à trouver un polynôme h satisfaisant. \square

Algorithme 35 : Méthode de Coppersmith

Entrées : Polynôme $f \in \mathbb{Z}[x]$ unitaire de degré d .

Sorties : Toute racine $|x| \leq O(N^{1/d})$ de $f(x) = 0 \pmod{N}$.

- 1 $m \leftarrow \left\lceil \frac{\log N}{d} \right\rceil$ ou valeur précisée manuellement
 - 2 $B \leftarrow \left\lceil \frac{1}{2^e} N^{1/d} \right\rceil$ ou valeur précisée manuellement
 - 3 Construire la matrice \mathbf{M} (voir équation (46)) dont les vecteurs colonnes représentent les vecteurs coefficients des polynômes $g_{i,j}(Bx)$ où $g_{i,j}(x) = x^j N^i f^{m-i}(x)$ pour $0 \leq i \leq m$ et $0 \leq j \leq id$.
 - 4 Appliquer l'algorithme LLL aux colonnes de \mathbf{M} .
 - 5 Construire le polynôme $h(x)$ tel que le premier vecteur de la réduction de \mathbf{M} représente $h(Bx)$.
 - 6 Calculer l'ensemble R des racines dans \mathbb{Z} de $h(x)$
 - 7 Filter $R \leftarrow \{x \in R; |x| \leq B, f(x) = 0 \pmod{N}\}$
 - 8 **retourner** R
-

Dans la ligne 3 de l'algorithme 35, on pourrait aussi faire le choix de considérer tous les $g_{i,j}(x)$ dont le degré ne dépasse pas une certaine borne.

Exemple 525. Fixons $N = 143$, $B = 6$ et le polynôme

$$f(x) = x^2 - 27x - 33 \in \mathbb{Z}[x]$$

et cherchons les racines $x_0 \in \mathbb{Z}$ telles que $|x_0| \leq B$ et $f(x_0) \equiv 0 \pmod{N}$. On commence par noter que les racines réelles de f sont

$\frac{27 \pm \sqrt{861}}{2}$. Ainsi, factoriser f dans \mathbb{Z} ne nous est d'aucune aide.

On a

$$f^2(x) = x^4 - 54x^3 + 663x^2 + 1782x + 1089 \in \mathbb{Z}[x]$$

Si x_0 est une racine de f modulo N , alors x_0 est aussi une racine des polynômes suivants modulo N^2 :

$$g_{2,0}(x) = N^2, g_{2,1}(x) = N^2 \cdot x, g_{2,2}(x) = N^2 \cdot x^2,$$

$$g_{2,3}(x) = N^2 \cdot x^3, g_{2,4}(x) = N^2 \cdot x^4, g_{1,0}(x) = N \cdot f(x),$$

$$g_{1,1}(x) = x \cdot N \cdot f(x), g_{1,2}(x) = x^2 \cdot N \cdot f(x), g_{0,0}(x) = f^2(x).$$

Afin de trouver une combinaison linéaire de ces polynômes qui satisfait l'équation 45, nous cherchons un vecteur court dans le réseau engendré par les colonnes de la matrice

$$\begin{pmatrix} & g_{2,0}(Bx) & g_{2,1}(Bx) & g_{2,2}(Bx) & g_{2,3}(Bx) & g_{2,4}(Bx) & g_{1,0}(Bx) & g_{1,1}(Bx) & g_{1,2}(Bx) & g_{0,0}(Bx) \\ 1 & 20449 & 0 & 0 & 0 & 0 & -4719 & 0 & 0 & 1089 \\ x & 0 & 122694 & 0 & 0 & 0 & -23166 & -28314 & 0 & 10692 \\ x^2 & 0 & 0 & 736164 & 0 & 0 & 5148 & -138996 & -169884 & 23868 \\ x^3 & 0 & 0 & 0 & 4416984 & 0 & 0 & 30888 & -833976 & -11664 \\ x^4 & 0 & 0 & 0 & 0 & 26501904 & 0 & 0 & 185328 & 1296 \end{pmatrix}$$

Nous extrayons une \mathbb{Z} -base de cette matrice (en utilisant la forme normale de Hermite) puis nous appliquons l'algorithme de réduction LLL. Nous obtenons la base

$$\begin{pmatrix} -2420 & 1815 & -7744 & -2299 & 8470 \\ -8316 & -4488 & -5676 & -14850 & -9504 \\ 6084 & 4212 & -10764 & -936 & -8892 \\ 3456 & -8208 & -4104 & -3456 & 7560 \\ 6480 & 7776 & 3888 & -6480 & 2592 \end{pmatrix}$$

dont le premier vecteur fournit (en divisant par B^i le i -ème coefficient) le polynôme

$$\begin{aligned} h(x) &= \frac{6480}{6^4}x^4 + \frac{3456}{6^3}x^3 + \frac{6084}{6^2}x^2 - \frac{8316}{6}x - 2420 \\ &= 5x^4 + 16x^3 + 169x^2 - 1386x - 2420. \end{aligned}$$

Nous recherchons les racines de h dans \mathbb{Z} (selon techniques du TP 5) et obtenons la racine entière $x_0 = 5$. Nous pouvons vérifier que $f(5) = -N \equiv 0 \pmod{N}$.

Exercice 526. Coder la méthode de Coppersmith (algorithme 35) avec SageMath (on supposera f dans `Pol.<x> = PolynomialRing(ZZ)`, e désigne `exp(1)`).

BITS DE POIDS FORTS ET MESSAGE STÉRÉOTYPÉ AVEC RSA

La méthode de Coppersmith rend RSA vulnérable si l'on connaît les bits de poids forts de m . Ceci peut se produire notamment si un attaquant sait par avance qu'un message possède une certaine entête (exemple : « Mon général, [...], bien respectueusement. » dans un rapport militaire)

Attaque : Étant connu N, e, m^e et \tilde{m} tel que $|m - \tilde{m}| < N^{1/e}$, retrouver m . On applique la méthode de Coppersmith au polynôme $f(x) = (\tilde{m} + x)^e - m^e \in \mathbb{Z}/N\mathbb{Z}[x]$. Si r est une racine, $m = \tilde{m} + r$ était le message envoyé

Exercice 527. La clé publique RSA de Alice est $N = 42564360034887861127$ et $e = 3$. Bob envoie quotiennement à Alice le « mot de passe du jour » qui est une suite de deux caractères par un message de la forme "jj/mm:?" selon les lignes SageMath suivantes :

```
bin=BinaryStrings()
N = 42564360034887861127
e = 3
text="jj/mm:XX"
m = mod( ZZ(str(bin.encoding(text)), base=2), N)
c=m^e
```

Alice décode le message par les lignes :

```
mm=ZZ(c^s).str(2)
mm = '0'*(8*ceil(len(mm)/8)-len(mm))+mm
bin(mm).decoding()
```

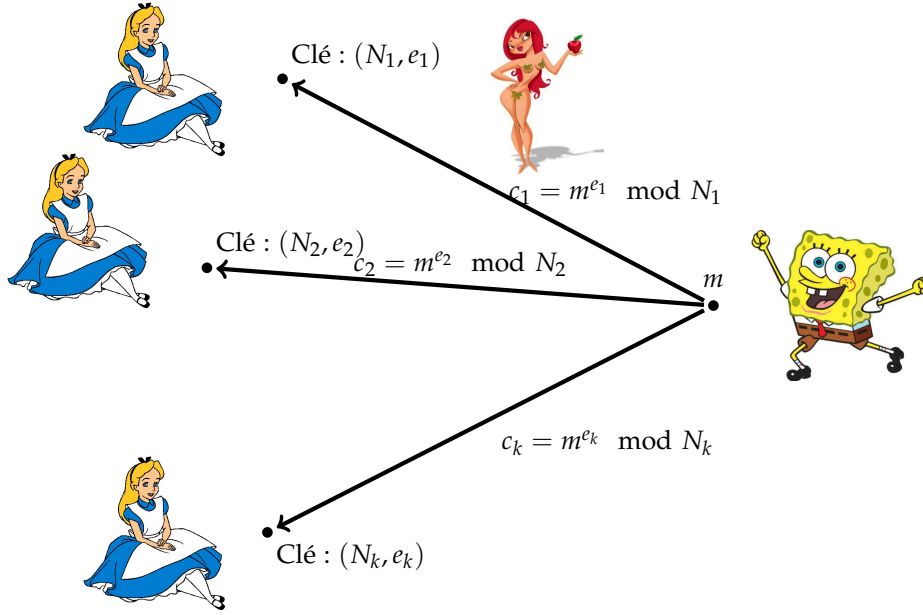
où s est sa clé privée. Le 8 juin, Eve intercepte le message $c = 12843085802751039909$. Que se passe-t-il ce jour là ?

ENVOI MULTIPLE ET ATTAQUE DE HÅSTAD SUR RSA Nous nous plaçons dans la situation suivante.

Scénario Bob envoie le même message plusieurs fois (disons à Alice, Alicia, Alix, Adélaïde, etc.). Chacune des filles publie sa clé RSA $(N_i, e_i)_{1 \leq i \leq k}$. Ève intercepte tous les chiffrés $c_i = m^{e_i} \bmod N_i$ envoyés par Bob.

Attaque Nous supposons que les entiers $(N_i)_{1 \leq i \leq k}$ sont premiers entre eux deux à deux (faute de quoi RSA serait cassé). Notons $N = \prod_{i=1}^k N_i$ leur produit. Si par malheur, tous les exposants étaient identiques égaux à e , le théorème des restes chinois nous permettrait de calculer $m^e \bmod N$. Mais pour k suffisamment grand, N devient si grand que

FIGURE 38: Envoi multiple



$m^e = (m^e \bmod N)$ et l'on peut retrouver m par une vulgaire racine e -ième.

Ce type d'attaque reste encore possible même si les e_i sont distincts. Dans un cadre plus général, soit α_i un entier (calculé par le théorème des restes chinois toujours) tel que $\alpha_i = 1 \bmod N_i$ et $\alpha_i = 0 \bmod N_j$ pour $j \neq i$. Soit

$$h(x) = \sum_{i=1}^k \alpha_i (x^{e_i} - c_i)^{ppcm((e_j)_j)/e_i} \bmod N$$

(polynôme calculable par Eve). Le polynôme h est unitaire de degré $e = \max_i e_i$. De plus, $h(m) = 0 \bmod N$. Mais alors, si $m < N^{1/e}$, Eve peut retrouver m par la méthode de Coppersmith. Ceci est en particulier le cas si $m \leq \min N_i$ (condition qui semble naturelle, puisque Bob ne peut pas envoyer plus d'information de tout façon) et si $k \geq e = \max_i e_i$. La parade pour Bob est néanmoins d'ajouter du bruit à m , de sorte que m et N soit du même ordre de grandeur.

ATTAQUE SUR RSA DE PETIT EXPOSANT

L'attaque du cryptosystème RSA suivante est due à Wiener. Elle fonctionne lorsque la clé privée est relativement petite : avec les notations de l'exemple 515, lorsque $|s| \leq N^{1/4}$. Nous cherchons à résoudre l'équation RSA

$$e \cdot s \equiv 1 \bmod \phi(N)$$

soit encore trouver s et k tels que

$$e \cdot s + k(p + q - 1) - 1 = kN. \quad (47)$$

Ceci revient à dire que le polynôme bivarié $f(x, y) = ex + y \in \mathbb{Z}[x, y]$ possède le couple

$$(x_0, y_0) = (s, k(p + q - 1) - 1)$$

comme racine modulo N . On peut noter que $k < s$. De plus, dans le cas où p et q sont de tailles proches, on a $p, q \simeq \sqrt{N}$. Donc $x_0 = O(N^{1/4})$ et $y_0 = O(N^{3/4})$. Le produit $x_0 y_0$ reste $O(N)$, qui permet d'envisager de résoudre l'équation par une méthode heuristique.

Pour trouver les racines « petites » de f , considérons les vecteurs

$$\mathbf{b}_1 = \begin{pmatrix} eX \\ Y \end{pmatrix} \text{ et } \mathbf{b}_2 = \begin{pmatrix} NX \\ 0 \end{pmatrix} \text{ avec } X = \lfloor N^{1/4} \rfloor \text{ et } Y = \lfloor N^{3/4} \rfloor$$

et le réseau Λ qu'il engendre. Nous voyons que $\|\mathbf{b}_1\| = \|\mathbf{b}_2\| = O(N^{5/4})$. Le réseau Λ contient aussi le vecteur

$$s\mathbf{b}_1 - k\mathbf{b}_2 = \begin{pmatrix} seX - kNX \\ sY \end{pmatrix} = \begin{pmatrix} (1 - k(p + q - 1))X \\ sY \end{pmatrix},$$

qui est de norme $O(N^{3/4})$, ce qui est relativement court dans Λ . L'algorithme LLL permet de retrouver ce vecteur (en tant que premier vecteur de la base LLL-réduite). On découvre $|s|, |k|$ et par suite $p + q$ et finalement p et q .

Remarque 528. Cette présentation peut être rendue rigoureuse. La présentation originale de cette attaque utilisait des fractions continues plutôt que des réseaux de rang 2.

Remarque 529. Avec une autre formulation de l'équation RSA que celle de la formule de l'équation (47) et en s'inspirant de la méthode de Coppersmith, d'autres auteurs ont formulé une attaque généralisant celle de Wiener pour $s \leq N^{1-1/\sqrt{2}}$.

Exemple 530. Alice a publié la clé $N = 24585612803$ et $e = 10878450977$. Pour retrouver la clé privée d'Alice, Ève calcule $X = 395$ et $Y = 62088444$. Elle construit ensuite le réseau Λ de base

$$\mathbf{b}_1 = \begin{pmatrix} 4296988135915 \\ 62088444 \end{pmatrix} \text{ et } \mathbf{b}_2 = \begin{pmatrix} 9711317057185 \\ 0 \end{pmatrix}.$$

Le premier vecteur de la base LLL-réduite de Λ est

$$\pm \begin{pmatrix} -6193500855 \\ 7015994172 \end{pmatrix} = \pm 113\mathbf{b}_1 \mp 50\mathbf{b}_2.$$

Ève en déduit que s pourrait valoir 113, $k = 50$. Alors on aurait $\sigma = (kN - es + 1)/k + 1 = p + q = 313596$. Elle peut résoudre l'équation $z^2 - \sigma z + N$ pour obtenir $p = 156797$ et $q = 156799$.

Exercice 531. Alice s'apprête à publier une clé publique RSA mais hésite entre les clés :

$$N = 65946239999, \quad e = 22022476093$$

et

$$N = 65946239999, \quad e = 10865199773.$$

Laquelle de ces clés lui conseillez-vous ?

Complexité algorithmique et cryptosystèmes

Différents problèmes algorithmiques peuvent être associés aux réseaux. Nous les citons pour des réseaux sur \mathbb{Z} afin de donner du sens à leurs questions de complexité (il est difficile d'exprimer un réel quelconque par une suite finie de bits). Commençons par les faciles.

Problème 532 (Appartenance). Etant donné une base $\mathbf{B} \in \mathbb{Z}^{m \times n}$ de \mathcal{L} et un vecteur $\mathbf{v} \in \mathbb{R}^m$, décider si $\mathbf{v} \in \mathcal{L}$.

Ce problème revient à résoudre le système linéaire $\mathbf{B}\mathbf{x} = \mathbf{v}$ et observer si \mathbf{x} est entier ou non.

Problème 533 (Inclusion). Etant donné des bases $\mathbf{B}_1 \in \mathbb{Z}^{m \times n}$ et $\mathbf{B}_2 \in \mathbb{Z}^{m \times n}$ de \mathcal{L}_1 et \mathcal{L}_2 , décider si $\mathcal{L}_1 \subseteq \mathcal{L}_2$.

Ceci revient à vérifier que les vecteurs de \mathbf{B}_1 appartiennent à \mathcal{L}_2 .

Problème 534 (Equivalence). Etant donné des bases $\mathbf{B}_1 \in \mathbb{Z}^{m \times n}$ et $\mathbf{B}_2 \in \mathbb{Z}^{m \times n}$ de \mathcal{L}_1 et \mathcal{L}_2 , décider si $\mathcal{L}_1 = \mathcal{L}_2$.

Ceci revient à vérifier $\mathcal{L}_1 \subseteq \mathcal{L}_2$ et $\mathcal{L}_2 \subseteq \mathcal{L}_1$.

Les problèmes du vecteur le plus court et du vecteur le plus proche

Nous décrivons les deux principaux problèmes liés aux réseaux : SVP ou Shortest vector problem et CVP ou Closest vector problem. Comme souvent, ils apparaissent sous trois formes : recherche, optimisation et décision.

Problème 535 (Shortest vector problem). Donné sous trois variantes :
 SearchSVP : Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} , trouver $\mathbf{v} \in \mathcal{L}$ tel que $\|\mathbf{v}\| = \lambda_1(\mathcal{L})$.
 OptimisationSVP : Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} , trouver $\lambda_1(\mathcal{L})$.
 DecisionalSVP : Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} et un rationnel $r \in \mathbb{Q}$, déterminer si $\lambda_1(\mathcal{L}) \leq r$.

Les trois problèmes sont de difficulté équivalente.

On peut aussi être intéressé par les approximations de SVP à un facteur γ près.

Problème 536 (Shortest vector problem approché). Donné sous trois variantes :

SearchSVP $_{\gamma}$: Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} , trouver $\mathbf{v} \in \mathcal{L}$ tel que $\|\mathbf{v}\| \leq \gamma \cdot \lambda_1(\mathcal{L})$.

OptimisationSVP $_{\gamma}$: Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} , trouver d tel que $\lambda_1(\mathcal{L}) \leq \gamma \leq d \cdot \lambda_1(\mathcal{L})$.

GapSVP $_{\gamma}$: Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} et un rationnel $r \in \mathbb{Q}$, déterminer si $\lambda_1(\mathcal{L}) \leq r$ ou si $\lambda_1(\mathcal{L}) > \gamma \cdot r$.

Ce dernier cas est un problème avec promesse : certaines entrées sont illégales si elles ne respectent pas la promesse d'être dans l'un des deux cas de sortie.

De la même manière on peut s'intéresser au problème du vecteur le plus proche :

Problème 537 (Shortest vector problem approché). Donné sous trois variantes :

SearchCVP $_{\gamma}$: Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} et un vecteur $\mathbf{t} \in \mathbb{Z}^m$, trouver $\mathbf{v} \in \mathcal{L}$ tel que $\|\mathbf{t} - \mathbf{v}\| \leq \gamma \min_{x \in \mathcal{L}} \|\mathbf{t} - x\|$.

OptimisationCVP $_{\gamma}$: Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} , trouver $d \in \mathbb{Q}$ tel que $\min_{x \in \mathcal{L}} \|\mathbf{t} - x\| \leq d \leq \gamma \min_{x \in \mathcal{L}} \|\mathbf{t} - x\|$.

GapCVP $_{\gamma}$: Etant donné $\mathbf{B} \in \mathbb{Z}^{m \times n}$ une base de \mathcal{L} et un rationnel $r \in \mathbb{Q}$, déterminer si $\min_{x \in \mathcal{L}} \|\mathbf{t} - x\| \leq r$ ou si $\min_{x \in \mathcal{L}} \|\mathbf{t} - x\| > \gamma \cdot r$.

La figure 39 résume l'évolution de la complexité de SVP $_{\gamma}$ et CVP $_{\gamma}$ en fonction de γ .

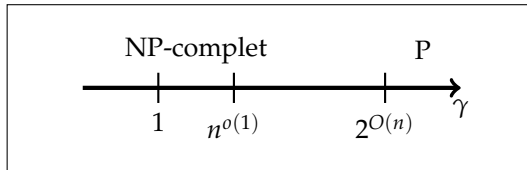


FIGURE 39: Complexité de SVP $_{\gamma}$ et CVP $_{\gamma}$ en fonction de γ

Nous ne donnerons pas de preuve concernant la NP complétude de SVP ou CVP. Concernant l'autre extrémité de la flèche, l'algorithme LLL répond au problème SVP $_{\gamma}$ pour γ exponentiel. Pour CVP $_{\gamma}$, nous verrons ci-dessous un algorithme. Mentionnons enfin que certaines classes intermédiaires ont été identifiées entre $\gamma = \sqrt{n}$ et $\gamma = 2^{O(n)}$ mais les citer nous entraînerait trop loin dans la théorie de la complexité.

Approximation de CVP

Nous précisons comment CVP peut être approché en temps polynomial.

Algorithme 36 : Algorithme du plan le plus proche (Babai)

Entrées : Base quelconque \mathbf{B} de \mathcal{L} , vecteur \mathbf{t}

Sorties : Vecteur $\mathbf{x} \in \mathcal{L}$ tel que

$$\|\mathbf{x} - \mathbf{t}\| \leq \gamma \min_{\mathbf{y} \in \mathcal{L}} \|\mathbf{t} - \mathbf{y}\| \text{ avec } \gamma = 2^{n/2}$$

1 Rechercher une base LLL -réduite de \mathcal{L}

2 $\mathbf{b} \leftarrow \mathbf{t}$

3 **pour** $j = n$ à 1 **faire**

4 $u_j \leftarrow \langle \mathbf{b}, \mathbf{b}_j^* \rangle / \langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle$

5 $\mathbf{b} \leftarrow \mathbf{b} - \text{Arrondi}(u_j) \mathbf{b}_j$

6 **retourner** $\mathbf{x} = \sum_{j=1}^n \text{Arrondi}(u_j) \mathbf{b}_j = \mathbf{t} - \mathbf{b}$

COMMENT COMPRENDRE CET ALGORITHME ? Si la base du réseau était orthogonale, il suffirait d'arrondir chaque coordonnée à l'entier le plus proche. Puisque la base B est LLL -réduite, elle très proche de la base orthogonale de Gram-Schmidt. L'algorithme utilise la base orthogonale $\mathbf{b}_1^*, \dots, \mathbf{b}_n^*$ et il revient à trouver un vecteur du réseau $\mathbf{x} \in \mathcal{L}$ tel que le reste $\mathbf{t} - \mathbf{x}$ se trouve dans la boîte $\prod_{i=1}^n \left[-\frac{\mathbf{b}_i^*}{2}, +\frac{\mathbf{b}_i^*}{2}\right]$.

Une autre façon de voir les choses (qui justifie le nom de cet algorithme) est de voir l'étape de la boucle comme la recherche récursive du plan $u\mathbf{b}_i^* + \text{Vect}(\mathbf{b}_1, \dots, \mathbf{b}_{i-1})$ le plus proche du point courant quand u varie dans \mathbb{Z} . Ainsi sur la figure (40), le plan bleu est le plus proche de \mathbf{t} , puis la droite verte de \mathbf{t}' , etc jusqu'au point \mathbf{y} .

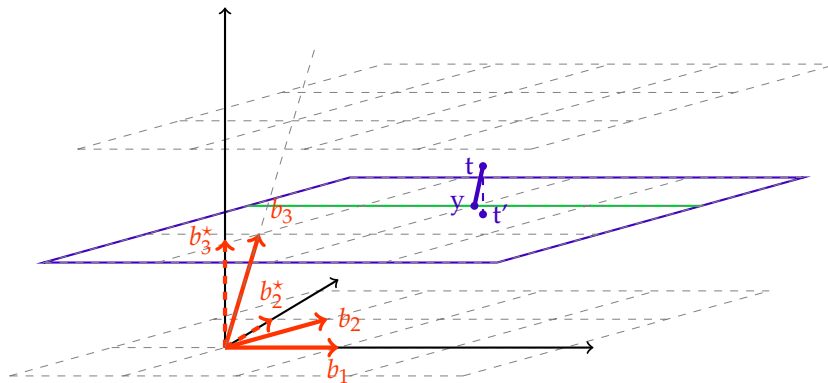


FIGURE 40: Illustration de l'algorithme de Babai

Remarque 538. L'algorithme se trompe dès la dimension 2. La figure

41 montre le bon découpage à droite (cellules de Voronoï) et le découpage par l'algorithme de Babai à gauche.

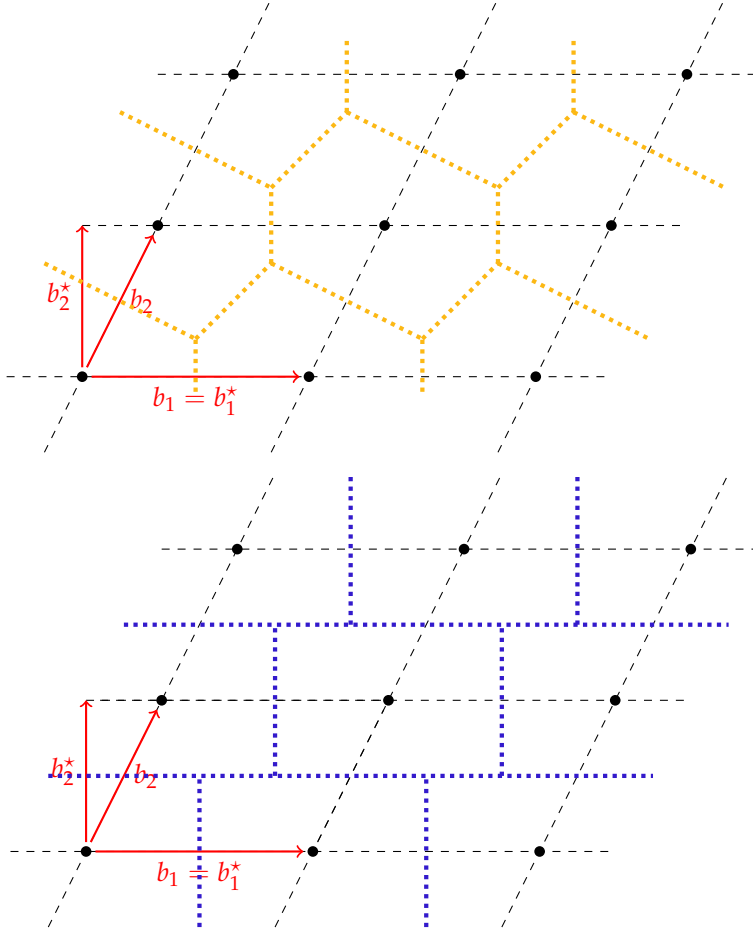


FIGURE 41: Cellules de Voronoï contre cellules de Babai

PERFORMANCES DE L'ALGORITHME

Lemme 539. Si $\mathbf{t} \in \text{Vect}(\mathbf{B})$, le résultat \mathbf{x} de l'algorithme de Babai satisfait

$$\|\mathbf{x} - \mathbf{t}\|^2 \leq \frac{1}{4} \sum_{i=1}^n \|\mathbf{b}_i^*\|^2. \quad (48)$$

Démonstration. Le vecteur \mathbf{b} est le vecteur « reste » qui mesure l'écart entre le point du réseau \mathbf{x} et le point initial \mathbf{t} . Après la j -ième étape, le vecteur \mathbf{b} satisfait $\left| \frac{\langle \mathbf{b}, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \right| \leq \frac{1}{2}$ et sa composante selon \mathbf{b}_j^* n'est plus modifiée. Donc à la fin de l'algorithme,

$$\mathbf{b} \in \prod_{i=1}^n \left[-\frac{\mathbf{b}_i^*}{2}, +\frac{\mathbf{b}_i^*}{2} \right]$$

Autrement dit, l'algorithme renvoie un point \mathbf{x} proche de \mathbf{t} tel que

$$\|\mathbf{x} - \mathbf{t}\|^2 \leq \frac{1}{4} \sum_{i=1}^n \|\mathbf{b}_i^*\|^2.$$

□

Lemme 540. Si $\mathbf{t} \in \text{Vect}(\mathbf{B})$, le résultat \mathbf{x} de l'algorithme de Babai satisfait

$$\|\mathbf{x} - \mathbf{t}\| \leq \frac{2^{n/2}}{2} \|\mathbf{b}_n^*\|. \quad (49)$$

Démonstration. La condition de Lovász donne

$$\forall 1 \leq i < n, \quad \|\mathbf{b}_i^*\|^2 \leq 2\|\mathbf{b}_{i+1}^*\|^2.$$

Donc, en reprennant l'égalité du lemme 539 et en itérant la précédente inégalité,

$$\|\mathbf{x} - \mathbf{t}\|^2 \leq \frac{1}{4} \sum_{i=1}^n 2^{n-i} \|\mathbf{b}_n^*\|^2$$

Ainsi

$$\|\mathbf{x} - \mathbf{t}\| \leq \frac{2^{n/2}}{2} \|\mathbf{b}_n^*\|.$$

□

Théorème 541. L'algorithme de Babai renvoie une approximation du vecteur le plus proche de facteur $2^{n/2}$.

Démonstration. Nous raisonnons par récurrence sur le rang n du réseau. Si ce rang est 1, le résultat de l'algorithme est exact.

Nous supposons que $n \geq 2$ et notons $\mathbf{y} = \text{CVP}(\mathbf{t}) \in \mathcal{L}$ le vecteur le plus proche de \mathbf{t} . Nous décomposons $\mathbf{y} = \sum_{i=1}^n y_i \mathbf{b}_i$. Deux cas peuvent se produire :

Cas 1 Soit $\lfloor u_n \rfloor = y_n$, autrement dit la « bonne couche » a été choisie à la première étape. On note $\mathbf{y}' = \sum_{i=1}^{n-1} y_i \mathbf{b}_i$, $\mathbf{x}' = \mathbf{x} - y_n \mathbf{b}_n$ et $\hat{\mathbf{t}}$ la projection orthogonale de $\mathbf{t} - y_n \mathbf{b}_n$ sur l'espace engendré par les $n-1$ premiers vecteurs. (Voir figure 42).

On observe que que $\hat{\mathbf{t}}$ et $\mathbf{t}' = \mathbf{t} - y_n \mathbf{b}_n$ ont à la fois même vecteur le plus proche (à savoir \mathbf{y}') et même résultat pour l'algorithme de Babai (à savoir \mathbf{x}'). Notons ce vecteur sur l'espace engendré par les $n-1$ premiers vecteurs.

On a, par hypothèse de récurrence,

$$\|\hat{\mathbf{t}} - \mathbf{x}'\|^2 \leq 2^{n-1} \|\hat{\mathbf{t}} - \mathbf{y}'\|^2$$

soit encore

$$\underbrace{\|\mathbf{t}' - \mathbf{x}'\|^2}_{=\|\mathbf{t} - \mathbf{x}\|} - \|\mathbf{t}' - \hat{\mathbf{t}}\|^2 \leq 2^{n-1} \left(\underbrace{\|\mathbf{t}' - \mathbf{y}'\|^2}_{=\|\mathbf{t} - \mathbf{y}\|} - \|\mathbf{t}' - \hat{\mathbf{t}}\|^2 \right)$$

ce qui permet de conclure comme voulu.

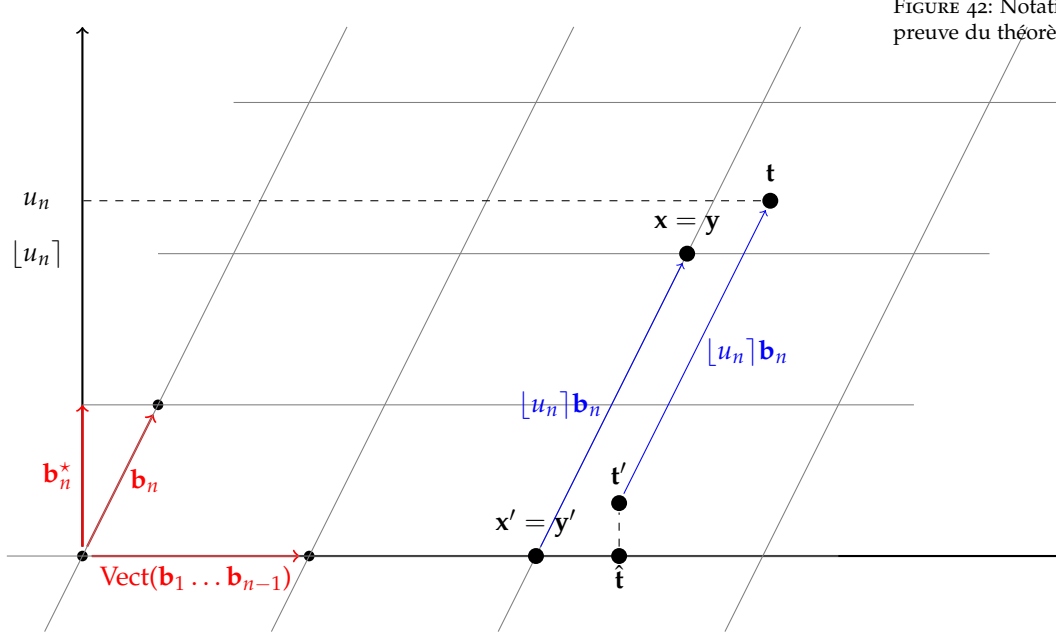


FIGURE 42: Notations pour le cas 1 de la preuve du théorème

Cas 2 Soit $\lfloor u_n \rfloor \neq y_n$, autrement dit une « mauvaise couche » a été choisie à la première étape. Nous savons que

$$\text{dist}(\mathbf{t}, \lfloor u_n \rfloor \mathbf{b}_n + \text{Vect}(\mathbf{b}_1, \dots, \mathbf{b}_{n-1})) \leq \frac{1}{2} \|\mathbf{b}_n^*\|.$$

Donc,

$$\|\mathbf{t} - \mathbf{y}\| \geq \frac{1}{2} \|\mathbf{b}_n^*\|.$$

En reprenant le lemme 540, il vient

$$\|\mathbf{x} - \mathbf{t}\| \leq \frac{2^{n/2}}{2} \|\mathbf{b}_n^*\| \leq 2^{n/2} \|\mathbf{t} - \mathbf{y}\|$$

comme voulu.

Si $\|\mathbf{t} - \mathbf{y}\|$ est supérieur à $\frac{1}{2} \|\mathbf{b}_n^*\|$, le théorème est établi. \square

Exercice 542. 1. Coder l'algorithme de Babai avec SageMath. On supposera que la base donnée en entrée est LLL-réduite.

2. Soient $B = (\mathbf{b}_1, \dots, \mathbf{b}_n)$ une base d'un réseau \mathcal{L} et $B^* = (\mathbf{b}_1^*, \dots, \mathbf{b}_n^*)$ la base orthogonale obtenue par le procédé de Gram-Schmidt. On appelle *parallélépipède orthogonalisé*

$$\mathcal{P} = \left\{ \sum_{i=1}^n x_i \mathbf{b}_i^*; 0 \leq x_i \leq 1 \right\}.$$

Étant donné un vecteur $\mathbf{v} \in \mathbb{R}^n$, proposer un algorithme calculant un vecteur $\mathbf{w} \in \mathcal{P}$ tel que $\mathbf{v} - \mathbf{w} \in \mathcal{L}$.

La cryptographie à base de réseaux

L'utilisation des réseaux en cryptographie a été motivée par des premiers travaux de Ajtai et Dwork (autour des années 1995) dans lesquels ils proposent un cryptosystème à clé publique dont la sécurité repose sur des problèmes de réseau pour une certaine classe de réseaux. De plus, ils montrent que casser leur système en moyenne (càd sur un réseau d'une certaine dimension tiré au hasard dans sa classe) est aussi difficile que de résoudre SVP pour tout réseau (d'une autre dimension dépendant de la première). Cette réduction de la complexité en moyenne à la complexité dans le pire des cas est remarquable pour la cryptographie. Si l'on pense à la factorisation d'entier, certes le problème est dur mais 50% des entiers sont pairs. Elle a motivé des recherches actives dans ce domaine.

Plus, précisément, étant donnée $\mathbf{A} \in \mathbb{Z}^{n \times m}$ une matrice à coefficients entiers et q un entier, on définit les deux réseaux suivants

$$\Lambda_q(\mathbf{A}) = \{\mathbf{y} \in \mathbb{Z}^m; \mathbf{y} \equiv \mathbf{A}^t \mathbf{t} \pmod{q} \text{ pour } \mathbf{t} \in \mathbb{Z}^n\}$$

$$\Lambda_q^\perp(\mathbf{A}) = \{\mathbf{y} \in \mathbb{Z}^m; \mathbf{A}\mathbf{y} \equiv 0 \pmod{q}\}$$

Ces deux réseaux satisfont $q\mathbb{Z}^m \subseteq \Lambda \subseteq \mathbb{Z}^m$. On parle de réseau q -aire. Ajtai a montré (en 1996) que pour certains paramètres q , n et m bien choisis, trouver le vecteur le plus court dans $\Lambda_q(\mathbf{A})$ quand \mathbf{A} est choisi uniformément dans $\mathbb{Z}_q^{n \times m}$ est aussi difficile que de résoudre des instances du problème SVP $_\gamma$ dans le *pire des cas*.

À ce jour, la cryptographie à base de réseaux est peu utilisée. Le principal grief concerne la taille des clés à employer comparée à celle des cryptosystèmes standards de même niveau de sécurité. Néanmoins, d'aucuns pensent que réseaux offrent de belles perspectives au cas où les ordinateurs *quantiques* parviennent à balayer les paradigmes de la cryptographie classique. Par ailleurs, la cryptographie à base de réseaux a permis de développer ces toutes dernières années des méthodes de *chiffrement homomorphe*, c'est-à-dire des méthodes permettant d'effectuer des opérations (additions, multiplications, comparaisons) sur les chiffrés qui commutent avec les fonctions de chiffrement et de déchiffrement.

LE SYSTÈME GGH (GOLDREICH-GOLDWASSER-HALEVI)

Le schéma GGH exploite, de la façon probablement la plus simple, la théorie des réseaux en vue d'application cryptographique : il se base sur l'idée que, quelque soit la base d'un réseau, générer un vecteur proche d'un vecteur du réseau est facile (il suffit de choisir un point du réseau et y ajouter une petite perturbation) mais retrouver le vecteur le plus proche est difficile à moins que la base ne possède des propriétés spéciales.

Definition 543 (Matrice HNF). On appelle *forme normale d'Hermite* d'une matrice \mathbf{A} à coefficients entiers l'unique matrice $\mathbf{H} = \mathbf{AU}$ où \mathbf{U} est unimodulaire telle que \mathbf{H} est triangulaire inférieure à coefficients positifs sur la diagonale ($h_{i,i} > 0$) et à diagonale dominante ($|h_{i,j}| < h_{i,i}$). On parle aussi de matrice échelonnée.

Génération de clés Alice choisit une base quasi-orthogonale \mathbf{B} d'un réseau et calcule la forme normale d'Hermite de $\mathbf{H} = \mathbf{BU}$ de \mathbf{B} .

Clé privée \mathbf{B}

Clé publique \mathbf{H}

Chiffrement Bob chiffre son message m par $c = \mathbf{H}m + e$ où e est un vecteur de perturbation de petite taille.

Déchiffrement Alice déchiffre $c = \mathbf{BU}m + e$ par l'algorithme de Babai en utilisant sa base privée \mathbf{B} : obtient $m' = \mathbf{U}m$, puis retrouve m en multipliant m' par \mathbf{U}^{-1} .

Remarque 544. Choisir de publier une base HNF revient à donner le moins d'information possible concernant \mathcal{L} , puisque tout adversaire peut calculer cette base efficacement à partir d'une base quelconque.

Remarque 545. Cet algorithme est présenté à des fins pédagogiques. En pratique, il demande d'utiliser des tailles de clés très grandes (par exemple $n \geq 300$ ne serait-ce que pour éviter que LLL permette de retrouver \mathbf{B} trop facilement) et les attaques connues suffisent à le casser pour des instances de taille raisonnable.

Exercice 546. On utilise dans cet exercice le cryptosystème de Goldreich, Goldwasser et Halevi.

1. On donne la clé privée $\mathbf{B} = 10\mathbf{I}_{40} + \mathbf{E}$ avec

```
E = matrix(ZZ,40, {(32, 32): 1, (22, 13): -2, (23, 8): -3,
(15, 31): -1, (22, 37): -1, (13, 5): -4, (38, 20): 2, (4, 12): 3,
(19, 22): -2, (15, 5): 2, (11, 32): -1, (11, 10): 3,
(1, 11): -4, (12, 33): 1, (0, 15): 1, (33, 17): 1,
(7, 19): -1, (11, 1): -2, (7, 27): 3, (19, 32): -4, (22, 10): 2,
(31, 39): -4, (34, 9): 2, (36, 17): 2, (18, 17): 1, (14, 6): -2,
(23, 14): 3, (23, 34): 2, (12, 11): -3, (0, 21): -3, (27, 22): -2,
(4, 29): -3, (23, 5): 1, (4, 6): -2, (24, 7): 2, (5, 38): -2,
(33, 13): -1, (9, 35): 3, (18, 36): 1, (22, 5): 1, (24, 25): 3,
(34, 31): 2, (6, 34): -3, (23, 33): -4, (20, 37): -1,
(38, 12): 2, (33, 0): -1, (4, 32): 3})
```

et l'on reçoit le chiffré c :

```
c = vector([-2, 0, 2, 0, 0, 1, -1, -1, -3, 0, 0, 2, -1, 13,
7, 2, 0, 2, 27, 2, 1, 17, -2, 899, 50, 15, 11, 1098, 7,
2, -1, 10, -1, 2, 156, 15, 42, 8, 525748584, 37])
```

Retrouver le message envoyé à l'aide de l'algorithme de Babai (NB : m a été encodé par une fonction décrite ci-dessous).

2. Pourrait-on utiliser la base \mathbf{H} seule pour retrouver m ? Et en utilisant LLL?

```
def StringToAscii(text):
    return [ZZ(ord(s)) for s in list(text)]
def AsciiToString(listascii):
    return "".join([chr(a) for a in listascii])
bin = BinaryStrings()
def StringToInts(text):
    return [ZZ(str(t), base=2) for t in list(bin.encoding(text))]
def IntsToString(listints):
    return bin("".join([str(t) for t in listints])).decoding()
```

NTRU

Le cryptosystème NTRU a été inventé en 1996 par Hoffstein, Pipher et Silverman. Il consiste à dissimuler le message \mathbf{m} en lui ajoutant un point \mathbf{z} du réseau engendré par des vecteurs colonnes de la matrice

$$\mathbf{H} = \begin{pmatrix} h_0 & h_{n-1} & h_{n-2} & \dots & h_1 \\ h_1 & h_0 & h_{n-1} & & h_2 \\ h_2 & h_1 & h_0 & & h_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{n-1} & h_{n-2} & h_{n-3} & \ddots & h_0 \end{pmatrix}.$$

D'une part, à cause de la structure cyclique des coefficients de la base du réseau, transmettre une base du réseau est seulement linéaire en la dimension et non pas quadratique. D'autre part, retrouver \mathbf{m} à partir de $\mathbf{c} = \mathbf{m} + \mathbf{z}$ revient à trouver le plus proche vecteur de \mathbf{c} dans le réseau $\mathbf{H}\mathbb{Z}^n$. À cause d'une construction particulière du réseau, cette opération peut être faite facilement pour qui a accès aux informations sur sa génération.

Dans ce qui suit, on identifie \mathbb{Z}^n au quotient $\mathbb{Z}[x]/\langle x^n - 1 \rangle$. Les colonnes de la matrice \mathbf{H} correspondent aux polynômes $\mathbf{h}(x)$, $x \cdot \mathbf{h}(x)$, \dots , $x^{n-1} \cdot \mathbf{h}(x) \bmod x^n - 1$ où \mathbf{h} est le polynôme $\mathbf{h}(x) = h_0 + h_1x + h_2x^2 + \dots + h_{n-1}x^{n-1}$.

Génération de clés Alice fixe un entier premier n et entiers p et q premiers entre eux tels que $q > p$. Alice choisit deux polynômes \mathbf{f} et \mathbf{g} de degré inférieur à $n - 1$ et à coefficients dans $\{-1, 0, 1\}$. Alice veille à ce que f soit inversible dans $\mathbb{F}_p[x]/\langle x^n - 1 \rangle$ et dans $\mathbb{F}_q[x]/\langle x^n - 1 \rangle$, c'est-à-dire qu'il existe des polynômes \mathbf{f}_p et $\mathbf{f}_q \in \mathbb{Z}[x]$ tels que $\mathbf{f} \cdot \mathbf{f}_p = 0 \pmod{\langle p, x^n - 1 \rangle}$ et $\mathbf{f} \cdot \mathbf{f}_q = 0 \pmod{\langle q, x^n - 1 \rangle}$. Alice transmet à Bob $\mathbf{h} = p\mathbf{f}_q \cdot \mathbf{g} \pmod{q}$.

Clé privée $(\mathbf{f}, \mathbf{f}_p)$

Clé publique (\mathbf{h})

Chiffrement Bob veut envoyer $\mathbf{m} \in \{-1, 0, 1\}^n$. Il génère un polynôme $\mathbf{r} \in \mathbb{Z}[x]/\langle x^n - 1 \rangle$ à petits coefficients. Il calcule et transmet $\mathbf{e} = \mathbf{r}\mathbf{h} + \mathbf{m} \pmod{\langle x^n - 1, q \rangle}$ à Alice.

Déchiffrement Alice calcule $\mathbf{a} = \mathbf{f}\mathbf{e} \pmod{\langle x^n - 1, q \rangle}$. Elle emploie la représentant des coefficients par des entiers centrés en 0. Puis Alice calcule $\mathbf{b} = \mathbf{a} \pmod{p}$. Enfin, Alice calcule $\mathbf{c} = \mathbf{f}_p \cdot \mathbf{b} \pmod{p}$.

Justification du protocole : Lorsque Alice calcule $\mathbf{a} = \mathbf{f}\mathbf{e} \pmod{\langle x^n - 1, q \rangle}$, elle obtient aussi en vérité

$$\begin{aligned} \mathbf{a} &= \mathbf{f}\mathbf{e} \pmod{\langle x^n - 1, q \rangle} \\ &= \mathbf{f}(\mathbf{r}\mathbf{h} + \mathbf{m}) \pmod{\langle x^n - 1, q \rangle} \\ &= p\mathbf{r}\mathbf{f}\mathbf{f}_q\mathbf{g} + \mathbf{f}\mathbf{m} \pmod{\langle x^n - 1, q \rangle} \\ &= p\mathbf{r}\mathbf{g} + \mathbf{f}\mathbf{m} \pmod{\langle x^n - 1, q \rangle} \end{aligned}$$

À ce stade, il faut supposer que les coefficients des différents éléments ont été choisis petits par rapport à q pour que $p\mathbf{r}\mathbf{g} + \mathbf{f}\mathbf{m}$ soit vraiment le représentant à coefficients centrés dans $\llbracket -q/2, q/2 \rrbracket$ de la même quantité. Alors, \mathbf{b} est simplement égal à $\mathbf{f}\mathbf{m} \pmod{\langle x^n - 1, p \rangle}$ et \mathbf{c} vaut $\mathbf{m} \pmod{p}$.

Exemple 547. Dans cet exemple, $n = 13$, $p = 5$, $q = 81$,

$$\begin{aligned} \mathbf{f} &= -3x^{12} + x^{11} + 2x^{10} - 2x^8 - 2x^7 - 3x^6 + 2x^5 - x^4 + 4x^3 + 2x^2 + x - 2 \\ \mathbf{f}_p &= 3x^{12} + 2x^{11} + 4x^{10} + 2x^9 + 3x^8 + 4x^7 + 4x^6 + 3x^4 + x^3 + x^2 + 2 \\ \mathbf{f}_q &= 8x^{12} + 6x^{11} + 6x^{10} + 26x^9 + 34x^8 + 65x^7 + 40x^6 + 63x^5 + 34x^4 + 6x^3 + 15x^2 + 76x + 25 \\ \mathbf{g} &= -2x^{11} - 2x^{10} + 4x^9 + 2x^7 - x^5 + x^4 + 4x^3 + 2x^2 + 3 \\ \mathbf{h} &= -20x^{12} - 25x^{11} + 30x^{10} - 16x^9 + 32x^8 + 22x^7 + 31x^6 + 16x^5 + 11x^4 + 27x^3 - 5x^2 - 4x + 8 \end{aligned}$$

Bob veut envoyer

$$\mathbf{m} = x^{12} + x^{11} + x^{10} + x^9 + x^8 - x^7 - x^6 - x^5 - x^3 - x^2 - x + 1$$

Il choisit

$$\mathbf{r} = x^{12} + x^{10} + x^7 - 2x^6 - x^3 - x + 1$$

et génère

$$\mathbf{e} = 60x^{12} + 31x^{11} + 13x^{10} + 35x^9 + 41x^8 + 57x^7 + 38x^6 + 6x^5 + 25x^4 + 70x^3 + 44x^2 + 57x + 9$$

qu'il envoie à Alice.

Pour déchiffrer, Alice passe par

$$\mathbf{a} = 11x^{12} + 20x^{11} - 25x^{10} - 19x^9 - 23x^8 + 7x^7 - 20x^6 - 34x^5 - 13x^4 + 34x^3 - 9x^2 + 33x + 38$$

$$\mathbf{b} = x^{12} + x^9 + 2x^8 + 2x^7 + x^5 + 2x^4 - x^3 + x^2 - 2x - 2$$

$$\mathbf{c} = x^{12} + x^{11} + x^{10} + x^9 + x^8 - x^7 - x^6 - x^5 - x^3 - x^2 - x + 1$$

LEARNING WITH ERRORS

Problème 548. LWE On donne des entiers n, m, q et une distribution de probabilité χ sur \mathbb{Z}_q . Une instance du problème de l'apprentissage avec erreurs (learning with errors) est la donnée d'un couple (\mathbf{A}, \mathbf{v}) où \mathbf{A} est une matrice tirée uniformément dans $\mathbb{Z}_q^{m \times n}$ et \mathbf{v} est au choix tiré uniformément dans \mathbb{Z}_q^m ou bien $\mathbf{v} = \mathbf{A}\mathbf{s} + \mathbf{e}$ avec \mathbf{s} tiré uniformément dans \mathbb{Z}_q^n et \mathbf{e} est tiré selon la loi de χ^m . Le but est de distinguer avec une probabilité non négligeable une situation de l'autre.

On peut reformuler le problème comme suit : étant donné (\mathbf{A}, \mathbf{v}) , décider si \mathbf{v} est la réduction mod q d'un vecteur du réseau $\Lambda_q(\mathbf{A}^t)$ légèrement perturbé ou si \mathbf{v} est tiré uniformément dans \mathbb{Z}_q^m .

Il a été montré en 2013 que ce problème est aussi difficile que certains problèmes SVP dans le pire des cas.

Remarque 549. La loi χ est typiquement une Gaussienne arrondie : Ψ_α de moyenne 0 et de déviation standard $\frac{\alpha q}{\sqrt{2\pi}}$.

Soit f la fonction $\mathbb{Z}_t^\ell \rightarrow \mathbb{Z}_q^\ell$ qui multiplie chaque coordonnée par q/t et arrondi le résultat. On note f^{-1} la pseudo inverse.

Paramètres Entiers n, m, ℓ, t, r, q et un réel α .

Clé privée Matrice $\mathbf{S} \in \mathbb{Z}_q^{n \times \ell}$ choisie uniformément

Clé publique On choisit $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ uniformément et $\mathbf{E} \in \mathbb{Z}_q^{m \times \ell}$ dont chaque coefficient selon Ψ_α . La clé publique est $(\mathbf{A}, \mathbf{P} = \mathbf{A}\mathbf{S} + \mathbf{E})$

Chiffrement Étant donné un message à envoyer $\mathbf{v} \in \mathbb{Z}_t^\ell$, choisir aléatoirement $\mathbf{a} \in \{-r, -r+1, \dots, r\}$. Envoyer $\mathbf{u} = \mathbf{A}^t \mathbf{a} \in \mathbb{Z}_q^n, \mathbf{c} = \mathbf{P}^t \mathbf{a} + f(\mathbf{v}) \in \mathbb{Z}_q^\ell$

Déchiffrement Calculer $f^{-1}(\mathbf{c} - \mathbf{S}^t \mathbf{u})$

Esquisse de preuve de sécurité. On peut résumer la preuve comme suit :

Point 1 Il est difficile de distinguer une clé publique valide d'un choix de matrice (\mathbf{A}, \mathbf{P}) , sauf à résoudre LWE.

Point 2 Si on utilise une clé (A, P) qcq, le chiffré ne contient aucune information statistique

Conséquence si on peut casser le système avec une certaine proba, à cause du point 2, on peut distinguer une distribution aléatoire de (A, P) d'une vraie clé. Mais cette tâche est difficile.

□

CHIFFREMENT TOTALEMENT HOMOMORPHE

Le *chiffrement totalement homomorphe* est un type de chiffrement qui permet à ses utilisateurs d'effectuer n'importe quel calcul (formellement, évaluer n'importe quelle fonction booléenne) sur le chiffré et de retrouver le résultat de ce calcul lors du déchiffrement. Le chiffrement totalement homomorphe permet ainsi de déléguer des calculs à un serveur lointain sur lequel sont stockés des données privées chiffrées que l'on ne souhaite pas révéler au moment du calcul. La construction de méthodes compétitives aurait des applications par exemple dans le contexte de l'informatique en nuage (*cloud computing*), du vote électronique, de la statistique sur des données sensibles (données médicales, etc).

Le premier cryptosystème totalement homomorphe a été construit en 2009 par Craig Gentry. Des améliorations et des variantes ont été proposées depuis. Elles s'appuient toutes sur des réseaux et l'idée que l'on cache le message en lui adjoignant du bruit. En général, le bruit augmente au fur et à mesure que l'on effectue des opérations sur le chiffré. À un certain point, il devient nécessaire de débruiter le chiffré pour ne pas perdre l'information qu'il contient (opération de *bootstrapping*). La grande prouesse de Gentry a été de montrer que l'on peut faire cela sans déchiffrer le chiffré.

TP 13 : Courbes elliptiques

Buts : Se familiariser avec les calculs dans une courbe elliptique, découvrir la méthode de factorisation ECM, s'initier aux attaques par canaux cachés.

Travaux préparatoires : Cours et exercice 560 questions 1-3

Évaluation du TP : Exercices 560 question 4 et 5 (addition d'une courbe elliptique), 572 (courbe ANSSI), 575 (comptage de points), 580 (factorisation ECM), 594 (exponentiation rapide et canaux cachés), 595 (courbe d'Edwards).

Introduction

Il existe de nombreuses motivations à l'étude des courbes elliptiques. Historiquement, le terme est apparu car ces courbes étaient associées au calcul de la longueur de l'arc d'une ellipse. Ces courbes apparaissent notamment dans des problèmes de mécanique classique ou dans la preuve du « grand » théorème de Fermat (en théorie des nombres).

En cryptographie, ces courbes ont été promues à partir de 1985 et sont utilisées largement depuis les années 2005. Comme elles possèdent une structure de groupe abélien, beaucoup de protocoles cryptographiques historiques (par exemple El-Gamal ou Diffie-Hellman, voir ex. 598) ont été transposés en remplaçant le groupe $(\mathbb{F}_p^\times, \cdot)$ par des courbes elliptiques. Ceci permet de réduire la taille des clés pour un niveau de sécurité comparable.

Exemple 550. Le cryptosystème de Menezes & Vanstone est un protocole (aujourd'hui déconseillé) de cryptographie asymétrique qui permet à Alice de recevoir un message de Bob

Génération de clés Alice fixe une courbe elliptique $\mathcal{E}(\mathbb{F}_p)$, des points G, H et un entier m tel que $mG = H$.

Clé privée L'entier m est sa clé secrète.

Clé publique Elle publie la courbe et les points G et H .

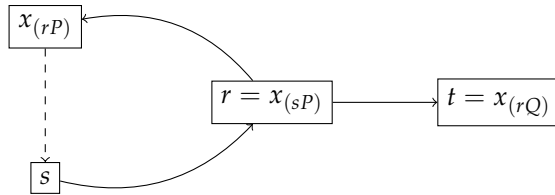
Chiffrement Bob peut envoyer des messages $(x_1, x_2) \in \mathbb{F}_p^2$. Pour cela, il choisit un entier aléatoire k . Il calcule $Y = kG$. Il calcule $(c_1, c_2) = kH$. Il calcule enfin $s_1 = c_1 x_1 \bmod p$ et $s_2 = c_2 x_2 \bmod p$. Il envoie à Alice le triplet $(Y, s_1, s_2) \in \mathcal{E} \times \mathbb{F}_p \times \mathbb{F}_p$.

Déchiffrement Alice calcule $mY = mkG = kH = (c_1, c_2)$. À partir de c_1 et de c_2 , elle retrouve le message avec $m_1 = c_1^{-1} s_1 \bmod p$ et $m_2 = c_2^{-1} s_2 \bmod p$.

Exemple 551 (Dual-EC-DRBG). Le *Dual Elliptic Curve Deterministic Random Bit Generator* est un mécanisme de génération de bits pseudo-aléatoires qui s'appuie sur des courbes elliptiques.

On fixe une courbe elliptique \mathcal{E} (par exemple la courbe P-256 standardisée par le NIST), on fixe deux points $P = (x_P, y_P)$ et $Q = (x_Q, y_Q)$ de cette courbe. On initialise s_0 avec un germe quelconque. On calcule successivement

$$\forall i \in \mathbb{N}, \quad \begin{cases} r_i &= x_{s_i P} \\ t_i &= x_{r_i Q} \\ s_{i+1} &= x_{r_i P} \end{cases}$$



La valeur de s représente l'état interne du générateur. Les bits de t_i (en fait, tous sauf les 16 plus forts) sont utilisés comme bits aléatoires. Du point de vue de la sécurité, il est crucial qu'un observateur ne puisse pas identifier s à partir de t .

Cet algorithme a été présenté initialement comme cryptographiquement sûr. Cependant, on peut conduire l'attaque suivante si l'attaquant connaît un entier e tel que $eQ = P$. On peut d'abord retrouver $x_{r_i Q}$ par une attaque par force brute (seuls 16 bits sont à deviner). On en déduit le point $\pm r_i Q$. On peut alors calculer $\pm e r_i Q = \pm r_i P$ et retrouver s_{i+1} . Aussi, en une seule observation de l'aléa produit, on peut connaître l'état interne s .

Le NIST a promu ce mécanisme à partir de 2006 avec des paramètres P et Q fixés par lui, avant de se rétracter en 2014, quand bien

même l'attaque ci-dessus était déjà connue et publique dès 2007. L'incapacité du NIST à justifier les origines de P et de Q et son insistance à voir ce générateur adopté par l'industrie a été critiquée par la communauté des cryptographes. (Détails dans le document NIST SP 800-90A dans sa version initiale : cf. page 74 de <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-90.pdf>)

On sait aujourd'hui que la cryptographie à base de courbe elliptique actuellement utilisée ne résistera pas à l'arrivée d'ordinateurs quantiques. Il existe toutefois un candidat encore en lice dans la compétition pour la standardisation de la cryptographie post-quantique organisée par le N.I.S.T. qui est basé sur les isogénies (les morphismes) entre courbes elliptiques.

Les courbes elliptiques sont aussi au cœur de certains algorithmes de factorisation d'entiers les plus performants à ce jour.

Structure de groupe abélien

Diviseurs : rappels et cas des courbes elliptiques

Dans ce qui suit, \mathcal{E} désigne une courbe elliptique, c'est-à-dire une courbe algébrique projective lisse de genre 1, définie sur un certain corps fini \mathbb{F}_q et possédant un point, disons $0_{\mathcal{E}}$, sur ce corps. Par ailleurs, $k(\mathcal{E})$ désigne le corps des fonctions rationnelles sur \mathcal{E} à coefficients dans le corps k .

GROUPE DES DIVISEURS Un *diviseur* D de \mathcal{E} est une somme formelle finie de points de cette courbe, c'est-à-dire une expression de la forme

$$D = \sum_{P \in \mathcal{E}(\bar{\mathbb{F}}_q)} n_P(P), \quad (n_P)_{P \in \mathcal{E}(\bar{\mathbb{F}}_q)} \subseteq \mathbb{Z}, \quad (50)$$

telle que $n_P = 0$ pour tout point P sauf un ensemble fini. L'ensemble des diviseurs forme un groupe abélien que nous notons $\text{div}(\mathcal{E})$. On appelle *degré* d'un diviseur l'entier

$$\deg D = \sum_{P \in \mathcal{E}(\bar{\mathbb{F}}_q)} n_P.$$

L'ensemble des diviseurs de degré nul forme aussi un groupe, noté $\text{div}^0(\mathcal{E})$.

DIVISEURS PRINCIPAUX Un diviseur permet opportunément de représenter l'ensemble des multiplicités des zéros et des pôles d'une fonction rationnelle sur \mathcal{E} (déjà vu à la remarque 278). Nous noterons, pour toute fonction rationnelle $f \in k(\mathcal{E})$,

$$\text{div}(f) = \sum_{P \in \mathcal{E}(\bar{\mathbb{F}}_q)} \text{ord}_P(f) \cdot (P)$$

Par ailleurs, on dit que le diviseur D est *principal* lorsqu'il existe une fonction f telle que $D = \text{div}(f)$. Un tel diviseur est toujours de degré 0 (voir 3.7.3 de²⁸). On note $\text{Princ}(\mathcal{E})$ le groupe des diviseurs principaux.

RIEMANN-ROCH On note $\mathcal{L}(D)$ l'espace des *sections globales du faisceau associé au diviseur D* (voir définition 3.8.1, *ibid.*), c'est-à-dire

$$\mathcal{L}(D) = \{f \in \overline{\mathbb{F}_q}(\mathcal{E}); \text{div}(f) + D \geq 0\}.$$

Cet espace est de dimension finie, nulle quand $\deg D < 0$ (proposition 3.8.2 (i.b), *ibid.*) et calculée via le théorème de Riemann-Roch (théorème 3.10.2, *ibid.*) dans le cas contraire :

$$\dim \mathcal{L}(D) - \dim \mathcal{L}(K - D) = \deg D + 1 - g$$

où K est un diviseur canonique et g le genre de la courbe. Mais K est de degré $\deg(K) = 2g - 2 = 0$ (cf. corollaire 3.10.3, *ibid.*) pour une courbe elliptique, si bien que pour une courbe elliptique, le théorème de Riemann-Roch se résume à :

$$\dim \mathcal{L}(D) = \begin{cases} 0 & \text{si } \deg D < 0 \\ 1 & \text{si } \deg D = 0 \text{ et } D \in \text{Princ}(\mathcal{E}) \\ 0 & \text{si } \deg D = 0 \text{ et } D \notin \text{Princ}(\mathcal{E}) \\ \deg D & \text{si } \deg D > 0 \end{cases} \quad (51)$$

Equation de Weierstraß

Par application du théorème de Riemann-Roch aux espaces

$$\mathcal{L}(n(0_{\mathcal{E}})) = \{f \in \mathbb{F}_q(\mathcal{E}); \text{div}(f) + n(0_{\mathcal{E}}) \geq 0\},$$

on peut par ailleurs établir que l'existence de deux fonctions rationnelles x et y de pôle 2 et 3 respectivement en $0_{\mathcal{E}}$ et en déduire que \mathcal{E} possède une équation dans le plan affine de la forme

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6.$$

Le point $0_{\mathcal{E}}$ est alors l'unique point à l'infini de cette courbe. Ses coordonnées projectives sont $(0 : 1 : 0)$. Lorsque la caractéristique p de \mathbb{F}_q est $p > 3$, on dispose même d'une équation simplifiée :

$$\mathcal{E}_{a,b} : y^2 = x^3 + ax + b, \quad (a, b) \in \mathbb{F}_q^2 \quad (52)$$

que l'on obtient par simple transformation linéaire.

Remarque 552. L'équation projective de la courbe \mathcal{E} est

$$\mathcal{E} : Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3$$

et $0_{\mathcal{E}}$ a toujours pour coordonnées $(0 : 1 : 0)$.

28. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

Avec SageMath, on peut utiliser `EllipticCurve([a1,a2,a3,a4,a6])`.

Avec SageMath, on peut définir $\mathcal{E}_{a,b}$ par `EllipticCurve([a,b])`

Exercice 553. On donne la courbe

$$\mathcal{E} : f(x, y) = y^2 - 2xy + 2y - (x^3 - 4x^2 + 2x) = 0$$

1. Vérifier que \mathcal{E} est bien une courbe lisse (voir définition page ??) (soit en calculant directement les dérivées de f avec `derivative` et en utilisant `solve`, soit par la méthode `is_smooth` de \mathcal{E}).
2. Représenter $\mathcal{E}(\mathbb{R})$ graphiquement (avec `plot` sur \mathcal{E} ou `implicit_plot` sur f).

Définition 554. On appelle *discriminant* de la courbe elliptique $\mathcal{E}_{a,b}$ (équation 52)

$$\Delta(\mathcal{E}_{a,b}) = -16(4a^3 + 27b^2) \quad (53)$$

et j -invariant de $\mathcal{E}_{a,b}$:

$$j(\mathcal{E}_{a,b}) = \frac{-1728(4a)^3}{\Delta}. \quad (54)$$

Le discriminant d'une courbe elliptique est toujours non-nul ; ceci est une condition nécessaire et suffisante pour que la courbe d'équation 52 soit lisse. Deux courbes elliptiques sont isomorphes sur la clôture algébrique si et seulement si elles ont le même j -invariant.

- Exercice 555** (Invariants). 1. Implémentez une fonction calculant le discriminant à partir du couple (a, b) . Comparez avec la méthode `idoine` de SageMath.
2. Implémentez une fonction calculant j à partir du couple (a, b) . Comparez avec la méthode `idoine` de SageMath.

Exercice 556. On donne la courbe

$$\mathcal{E} : f(x, y) = y^2 - 2xy + 2y - (x^3 - 4x^2 + 2x) = 0$$

1. Trouver trois constantes α , β et γ telles que par le changement de variable

$$\begin{cases} y &= y' + \alpha x' + \beta \\ x &= x' + \gamma \end{cases}$$

on obtienne une équation réduite de \mathcal{E} dans les coordonnées x' et y' . (On pourra utiliser la méthode `subs`)

2. Calculer le discriminant de \mathcal{E} .
3. Pour quelles valeurs de $q \in \mathbb{N}$, \mathcal{E} est-elle une courbe elliptique sur \mathbb{F}_q ?

Remarque 557. D'autres formes d'équations que celle de Weierstraß sont parfois utilisées. Par exemple, les courbes elliptiques dites de *Montgomery*, prennent la forme

$$By^2 = x^3 + Ax^2 + x \in \mathbb{F}_p[x, y]$$

avec $A \neq \pm 2$ et $B \neq 0$. Lorsque

$$p = 2^{255} - 19, \quad B = 1 \quad \text{et} \quad A = 486662,$$

la courbe s'appelle *Curve25519* et est employée par de nombreux logiciels (la messagerie de WhatsApp ou Facebook par exemple) avec le schéma Diffie-Hellman.

Loi d'addition, structure et critère de primalité d'un diviseur

LOI DE GROUPE On appelle le quotient $\text{div}^0(\mathcal{E})/\text{Princ}(\mathcal{E})$ le *groupe de Picard de degré 0* de \mathcal{E} (voir définition 3.7.4, *op. cit.*). Comme \mathcal{E} est de genre 1, le théorème de Riemann-Roch permet aussi d'établir que

Proposition 558. *L'application*

$$\psi : \begin{cases} \mathcal{E}(\overline{\mathbb{F}}_q) & \rightarrow \text{div}^0(\mathcal{E})/\text{Princ}(\mathcal{E}) \\ P & \mapsto D_P = (P) - (0_{\mathcal{E}}) \end{cases}$$

est une surjection, et donc aussi une bijection.

Démonstration. Soit $D \in \text{div}^0(\mathcal{E})$ un diviseur de degré 0. Par le théorème de Riemann-Roch, l'espace $\mathcal{L}(D + (0_{\mathcal{E}})) = \{f \in \overline{\mathbb{F}}_q(\mathcal{E}); \text{div}(f) + D + (0_{\mathcal{E}}) \geq 0\}$ est de dimension $\deg(D + (0_{\mathcal{E}})) = 1$. Soit f un générateur de $\mathcal{L}(D + (0_{\mathcal{E}}))$. Comme $\text{div}(f) + D + (0_{\mathcal{E}}) \geq 0$ et $\deg(\text{div}(f)) = 0$, il doit exister un point tel que $\text{div}(f) = -D - (0_{\mathcal{E}}) + (P)$, cqfd. \square

Ceci permet de munir les points de la courbe $\mathcal{E}(\overline{\mathbb{F}}_q)$ d'une structure de groupe abélien héritée de $\text{div}^0(\mathcal{E})/\text{Princ}(\mathcal{E})$ et dont le neutre est le point particulier $0_{\mathcal{E}}$.

En pratique, pour calculer la somme $S = P + Q$ de deux points $P, Q \in \mathcal{E}(\overline{\mathbb{F}}_q)$, il faut considérer la troisième intersection $S = (x_S, y_S)$ de la courbe \mathcal{E} avec la droite (PQ) et poser $R = (x_S, -y_S)$. On a dans ce cas

$$(P) - (0_{\mathcal{E}}) + (Q) - (0_{\mathcal{E}}) = (S) - (0_{\mathcal{E}}) + \text{div}(h_{P,Q}) \quad \text{avec} \quad h_{P,Q} = \frac{\ell_{P,Q}}{\ell_{S,S'}}$$

où $\ell_{P,Q} = 0$ est une équation de la droite (PQ) . Par convention, (PQ) désigne la tangente à \mathcal{E} en P si $P = Q$.

Pour résumer,

- le neutre est l'unique point à l'infini $0_{\mathcal{E}} = (0 : 1 : 0)$,
- l'opposé d'un point affine $P(x_P, y_P)$ est son symétrique $(x_P, -y_P)$ par rapport à l'axe (Ox) ,
- trois points alignés sont de somme nulle.

Ainsi, pour calculer la somme $P + Q$, on recherche la troisième intersection de la droite (PQ) , ou de la tangente à \mathcal{E} en P , avec \mathcal{E} et on renvoie sa symétrie par rapport à l'axe (Ox) (voir figure 43).

On appelle désormais somme : $\text{div}(\mathcal{E}) \rightarrow \mathcal{E}(\overline{\mathbb{F}}_q)$ l'opérateur qui calcule la somme décrite par un diviseur.

Exercice 559. Pourquoi l'intersection d'une courbe elliptique avec une droite compte-t-elle toujours 3 points ?

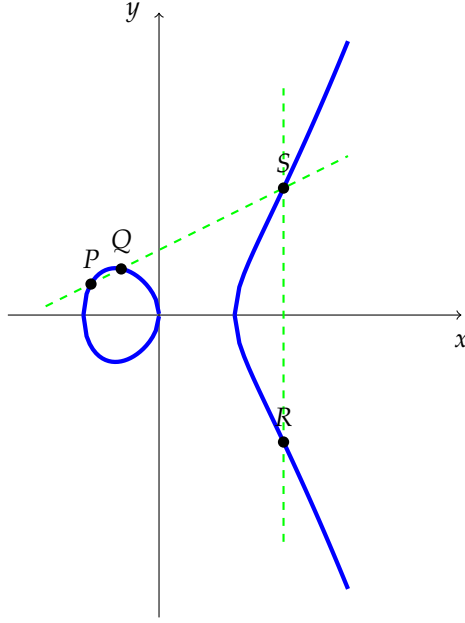


FIGURE 43: Somme de deux points : $P + Q = R$, $S = -R$.

Exercice 560. On désigne par (x_M, y_M) les coordonnées affines du point M .

1. Montrer que l'addition $R = P + Q$ de deux points distincts P et Q de la courbe $\mathcal{E}_{a,b}$ conduit aux formules

$$\lambda = \frac{y_Q - y_P}{x_Q - x_P}, \quad x_R = \lambda^2 - x_P - x_Q, \quad y_R = -y_P + \lambda(x_P - x_R)$$

où λ est la pente de la droite (PQ) .

2. Montrer que la duplication $R = 2P$ d'un point P de $\mathcal{E}_{a,b}$ conduit en général aux formules

$$\lambda = \frac{3x_P^2 + a}{2y_P}, \quad x_R = \lambda^2 - 2x_P, \quad y_R = -y_P + \lambda(x_P - x_R)$$

où λ est la pente de la tangente à \mathcal{E} en P . Que se passe-t-il si $y_P = 0$?

3. Comment faire pour calculer l'ensemble des points de 2-torsion de $\mathcal{E}_{a,b}$? Quelle structure du sous-groupe $\mathcal{E}[2]$ de 2-torsion peut-on obtenir ?

4. Implémenter les formules d'additions ci-dessus.
5. Comparer avec la méthode native de SageMath. On pourra prendre par exemple la courbe $\mathcal{E}_{a,1}(\mathbb{F}_q)$ et différents multiples du point P_0 de coordonnées $(0, 1)$.

Exercice 561. On donne la cubique \mathcal{E} définie sur \mathbb{F}_{13} par

$$y^2 = x^3 + 4x + 7.$$

1. Vérifier qu'il s'agit d'une courbe elliptique et que $G = (4, 3)$ appartient à la courbe.
2. Alice et Bob utilisent le cryptosystème de Menezes et Vanstone pour communiquer, avec le point G et la clé secrète $m = 9$ (notation de l'exemple 550).
 - (a) Calculer le point H de la clé publique d'Alice.
 - (b) Bob veut envoyer le message $(2, 7)$ à Alice. Calculer son chiffré.
 - (c) Alice reçoit le message (Y, s_1, s_2) de la part de Bob avec $Y = (1, 5)$, $s_1 = 7$ et $s_2 = 2$. Retrouver le clair.

Notons par ailleurs que

Théorème 562. *Le groupe des points d'une courbe elliptiques $\mathcal{E}(\mathbb{F}_q)$ est toujours isomorphe au produit d'au plus deux groupes cycliques. Autrement dit,*

$$\mathcal{E}(\mathbb{F}_q) \simeq \mathbb{Z}/r\mathbb{Z} \oplus \mathbb{Z}/s\mathbb{Z}$$

avec r divisant s (éventuellement $r = 1$).

Démonstration. Cf. cours ACCQ 207. □

Exercice 563 (Ordre d'un point). 1. Décrire et implémenter une méthode naïve pour calculer l'ordre d'un point sur une courbe elliptique.

2. Comparez vos implémentations avec les méthodes de SageMath. Commencez par tester le cas

$$a = 1929, b = 1178 \in \mathbb{F}_{2003} \quad \text{et} \quad P = (1259, 1284).$$

3. Comment faire si on connaît le cardinal de la courbe et sa factorisation ?

MULTIPLICATION ET SOUS-GROUPES DE TORSION. Pour tout entier r , la multiplication scalaire par r est notée

$$[r] : \begin{cases} \mathcal{E}(\overline{\mathbb{F}_q}) & \rightarrow & \mathcal{E}(\overline{\mathbb{F}_q}) \\ P & \mapsto & r \cdot P \end{cases} \quad (55)$$

et $\mathcal{E}[r]$ désigne le groupe des points de r -torsion, c'est-à-dire le noyau $[r]$, ou encore le sous-groupe

$$\mathcal{E}[r] = \ker [r] = \{P \in \mathcal{E}(\overline{\mathbb{F}_q}); rP = 0_{\mathcal{E}}\}. \quad (56)$$

Fait 564. Soit \mathcal{E} une courbe elliptique définie sur un corps de caractéristique $p > 0$ et r un entier naturel.

1. Si $p \nmid r$, alors

$$\mathcal{E}[r] \simeq (\mathbb{Z}/r\mathbb{Z})^2$$

2. Si $p \mid r$, alors

$$\mathcal{E}[r] \simeq \begin{cases} \{0\} \\ \text{ou} \\ \mathbb{Z}/r\mathbb{Z} \end{cases}.$$

Dans le premier cas, on dit d'ailleurs que la courbe elliptique \mathcal{E} est supersingulière, dans le second, on dit qu'elle est ordinaire.

Corollaire 565. Soit T un point quelconque de $\mathcal{E}(\overline{\mathbb{F}_q})$, alors il existe toujours un point $T''_0 \in \mathcal{E}(\overline{\mathbb{F}_q})$ tel que $rT''_0 = T$ et

$$\{T'' \in \mathcal{E}(\overline{\mathbb{F}_q}); rT'' = T\} = \{T''_0 + R; R \in \mathcal{E}[r]\}.$$

Démonstration. En effet, $[r]$ est un morphisme non constant, donc $[r]$ est surjectif, ce qui justifie l'existence de T''_0 . Par ailleurs, $rT'' = T$ équivaut à $rR = 0_{\mathcal{E}}$ si on pose $T'' = T''_0 + R$. \square

Proposition 566. Soit $f \in \mathbb{F}_q(\mathcal{E})$ une fonction rationnelle stable par translation par $\mathcal{E}[r]$ (i.e. $\forall P \in \mathcal{E}$ et $\forall R \in \mathcal{E}[r]$, $f(P + R) = f(P)$), alors il existe une fonction $f' \in \mathbb{F}_q(\mathcal{E})$ telle que $f = f' \circ [r]$.

Démonstration. (Esquisse) Notons \mathfrak{T} le groupe des translations par des points de $\mathcal{E}[n]$. Une translation $f(\cdot) \mapsto f(\cdot + R)$ est un automorphisme de corps de l'extension e corps $\overline{\mathbb{F}_q}(\mathcal{E})/[r]^*\overline{\mathbb{F}_q}(\mathcal{E})$. Par un résultat général, on doit avoir $\sharp \text{Aut}(\overline{\mathbb{F}_q}(\mathcal{E})/[r]^*\overline{\mathbb{F}_q}(\mathcal{E})) \leq \deg([r]^*) = r^2$. Les translations étant toutes distinctes, $\mathfrak{T} \simeq \text{Aut}(\overline{\mathbb{F}_q}(\mathcal{E})/[r]^*\overline{\mathbb{F}_q}(\mathcal{E}))$ et l'hypothèse que nous faisons revient à dire que la fonction rationnelle f est stable sous ce groupe. Il s'ensuit que $f \in [r]^*\overline{\mathbb{F}_q}(\mathcal{E})$, autrement dit qu'il existe une fonction rationnelle f' telle que $f = f' \circ [r]$. \square

Enfin, la proposition suivante permet de localiser (dans certains cas) sur quel corps est défini l'ensemble du groupe de r -torsion.

Proposition 567 (Balasubramanian et Koblitz). Soit toujours \mathcal{E} une courbe elliptique définie sur un corps \mathbb{F}_q de caractéristique p . Supposons r premier divisant $\sharp(\mathcal{E}(\mathbb{F}_q))$, $r \neq p$ et $r \nmid q - 1$, alors le groupe de r -torsion $\mathcal{E}[r]$ vérifie $\mathcal{E}[r] \subseteq \mathcal{E}(\mathbb{F}_{q^k})$ si et seulement si $r \mid q^k - 1$.

Voir²⁹, théorème IX.12.

29. I. Blake, G. Seroussi, and N. Smart. *Advances in Elliptic Curve Cryptography*, volume 317 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2005

PRINCIPALITÉ. Restreint à $\text{div}^0(\mathcal{E})/\text{Princ}(\mathcal{E})$, l'opérateur somme est la bijection réciproque de ψ . On déduit immédiatement du point précédent que

Théorème 568 (Abel-Jacobi). *Soit D un diviseur de \mathcal{E} . Pour qu'il existe une fonction $f \in k(\mathcal{E})$ telle que $\text{div}(f) = D$, il faut et il suffit que $\deg D = 0$ et $\text{somme}(D) = 0_{\mathcal{E}}$.*

Exemple 569. Soit $P \in \mathcal{E}(\mathbb{F}_q)$ un point et $n \in \mathbb{N}$ un entier naturel. On appelle *fonction de Miller* toute fonction $f_{n,P}$ telle que

$$\text{div}(f_{n,P}) = n \cdot (P) - ([n]P) + (n-1) \cdot (0_{\mathcal{E}}).$$

Une telle fonction existe en vertu du critère de principalité d'un diviseur (théorème 568) et est unique à une constante multiplicative près.

Lorsque T est un point de r -torsion, on écrit simplement f_T pour une fonction telle que

$$\text{div}(f_T) = r \cdot D_T = r \cdot (T) - r \cdot (0_{\mathcal{E}}).$$

Lemme 570. *Si $T_1, T_2 \in \mathcal{E}[n]$ et $T_3 = T_1 + T_2$, alors il existe une fonction rationnelle h et une constante c telles que*

$$\frac{f_{T_3}}{f_{T_1}f_{T_2}} = c \cdot h^n$$

Démonstration. D'après le critère de principalité, il existe une fonction h_{T_1, T_2} telle que

$$\text{div}(h_{T_1, T_2}) = (T_1) + (T_2) - (T_3) - (0_{\mathcal{E}}).$$

En réalité, nous avons vu il s'agit du quotient $\frac{\ell_{T_1, T_2}}{\ell_{T_3, -T_3}}$ des équations définissant les droites $(T_1 T_2)$ et $(T_3, -T_3)$. Toujours est-il que

$$n \cdot \text{div}(h_{T_1, T_2}) = \text{div}(f_{T_1}) + \text{div}(f_{T_2}) - \text{div}(f_{T_3}) = -\text{div}\left(\frac{f_{T_3}}{f_{T_1}f_{T_2}}\right)$$

ce qui montre que, à une constante c près, $\frac{f_{T_3}}{f_{T_1}f_{T_2}} = ch_{T_1, T_2}^{-n}$ comme voulu. \square

Taille du groupe des points

Aspects cryptographiques sur la taille d'une courbe

En général, le groupe des points $\mathcal{E}(\mathbb{F}_q)$ d'une courbe elliptique se comporte comme un groupe générique. Il peut être utilisé pour des applications cryptographiques telles que le protocole de Diffie-Hellmann (cf. exemple 598) ou d'El Gamal (cf. exemple 600).

Exercice 572. Dans un avis publié le 16 octobre 2011, l'ANSSI a préconisé l'usage d'une certaine courbe elliptique, notée FRP256v1.

1. Que signifie l'acronyme ANSSI ?
2. Retrouver le texte paru au JORF. Donner la taille de p en bits.
3. Quel est l'ordre de FRP256v1 ?
4. Vérifier avec SageMath qu'il s'agit bien d'un nombre premier.
5. Sachant que les algorithmes génériques de calcul du logarithme discret dans un groupe G sont en $O(\sqrt{|G|})$, en déduire que la « sécurité » de la courbe est de 128 bits.

Exercice 573. Simuler un échange de messages par le protocole d'El Gamal (cf. exemple 600) avec la courbe de l'exercice 572.

Estimation sur le comptage de points

Nous pouvons néanmoins citer deux résultats concernant la structure du groupe des points de $\mathcal{E}(\mathbb{F}_q)$ et sur l'estimation de son nombre de points (voir déjà théorème 495).

Théorème 574 (Borne de Hasse). *Soit \mathcal{E} une courbe elliptique définie sur \mathbb{F}_q , alors*

$$q + 1 - 2\sqrt{q} \leq |\mathcal{E}(\mathbb{F}_q)| \leq q + 1 + 2\sqrt{q}$$

Esquisse de justification. Soit a et b les coefficients tels que \mathcal{E} admette pour équation $y^2 = x^3 + ax + b$. En notant $\chi(x) \in \{\pm 1\}$ le symbole de Legendre, pour une valeur $x_0 \in \mathbb{F}_q$ donnée, il y a donc $1 + \chi(x_0^3 + ax_0 + b)$. Au total, en comptant $0_{\mathcal{E}}$,

$$|\mathcal{E}(\mathbb{F}_q)| = 1 + \sum_{x_0 \in \mathbb{F}_q} 1 + \chi(x_0^3 + ax_0 + b) = q + 1 + \sum_{x_0 \in \mathbb{F}_q} \chi(x_0^3 + ax_0 + b).$$

Si $\chi(x_0^3 + ax_0 + b)$ se comportait comme une variable aléatoire uniformément distribuée, la somme se comporterait comme une marche aléatoire sur la droite \mathbb{Z} . Des résultats classiques de probabilité indiquent qu'après q étapes, on se trouve à distance \sqrt{q} de l'origine. \square

Exercice 575. 1. Ecrire un programme « bête » qui compte les points de $\mathcal{E}_{a,b}$ en fonction de a et b .

2. Comparez vos implémentations avec les méthodes de SageMath. Commencez par tester le cas

$$a = 1929, b = 1178 \in \mathbb{F}_{2003}$$

3. Pour différentes valeurs de q , énumérer les courbes elliptiques $\mathcal{E}_{a,b}(\mathbb{F}_q)$ et tracer un histogramme représentant le nombre de courbes en fonction de son nombre de point (cf. figure 44). On peut utiliser `bar_chart`.

4. Comparer à chaque fois l'intervalle obtenu avec les bornes de Hasse.
5. Que remarque-t-on sur la forme du diagramme ?
6. Soit d un élément non carré de \mathbb{F}_q . Calculer $|\mathcal{E}_{a,b}| + |\mathcal{E}_{d^2a,d^3b}|$ pour tout $a, b \in \mathbb{F}_q$. Que remarque-t-on ?

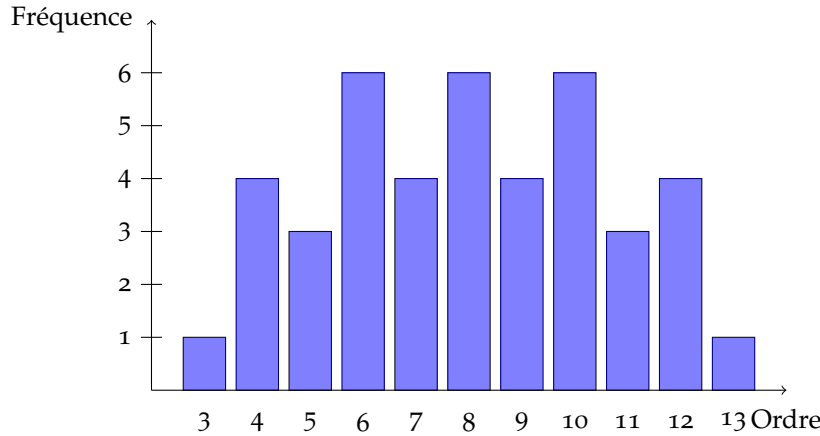


FIGURE 44: Nombre de courbes $\mathcal{E}_{a,b}(\mathbb{F}_7)$ en fonction de leur ordre. Noter que les bornes de Hasse prévoient un ordre entre 3 et 13

Remarque 576. À ce jour, on dispose de l'algorithme SEA, inventé par Schoof et amélioré par Elkies et Atkin pour calculer le cardinal d'une courbe elliptique. Cet algorithme est polynomial en $\log q$ (à comparer avec votre méthode naïve qui est polynomiale en q). C'est grâce à cet algorithme que la cryptographie sur courbe elliptique est envisageable puisque les propriétés cryptographiques d'une courbe sont directement reliées aux caractéristiques de son cardinal (qui doit être proche d'un grand nombre premier).

Méthode de factorisation par courbes elliptiques de Lenstra

Exercice 577. On considère les paramètres

$$a = b = 4, \quad p = 47, \quad q = 59 \quad \text{et} \quad P = (1, 3).$$

On pose $n = pq$.

1. Quel est l'ordre ω_p et ω_q de P dans $\mathcal{E}_{a,b}(\mathbb{F}_p)$ et dans $\mathcal{E}_{a,b}(\mathbb{F}_q)$?
2. Soit ϕ_p la projection modulo p , à savoir $\phi_p : \mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z}) \rightarrow \mathcal{E}_{a,b}(\mathbb{F}_p)$. Quels sont les antécédants du point $(0 : 1 : 0)$ par ϕ_p ?
3. Que se passe-t-il si on cherche à calculer $\omega_p P$ et $\omega_q P$ dans $\mathcal{E}(\mathbb{Z}/n\mathbb{Z})$?

L'algorithme ECM (Elliptic Curve Method) a été inventé en 1985 par Lenstra. Il revisite la méthode $p-1$ de Pollard.

Le succès de l'algorithme de Pollard est conditionné au fait que l'entier n à factoriser possède un facteur premier p tel que $p-1$ soit

ultrafriable. L'algorithme de Lenstra contourne le problème en travaillant dans le groupe $\mathcal{E}(\mathbb{F}_p)$ d'une courbe elliptique (qui est, d'après les bornes de Weil, de taille aléatoire centrée autour de $p + 1$ et que l'on peut espérer friable) à la place du groupe \mathbb{F}_p^\times (qui est de taille fixe égale à $p - 1$).

Soit $\mathcal{E}_{a,b} : y^2 = x^2 + ax + b$ une cubique sur $\mathbb{Z}/n\mathbb{Z}$ (avec $(4a^3 + 27b^2) \wedge n = 1$). Grâce au lemme chinois, on a une application naturelle

$$\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z}) \rightarrow \mathcal{E}_{a,b}(\mathbb{Z}/p\mathbb{Z}) \times \mathcal{E}_{a,b}(\mathbb{Z}/q\mathbb{Z})$$

entre les points affines de ces courbes (que l'on étend au point à l'infini par $0_{\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})} \mapsto (0_{\mathcal{E}_{a,b}(\mathbb{Z}/p\mathbb{Z})}, 0_{\mathcal{E}_{a,b}(\mathbb{Z}/q\mathbb{Z})})$). La méthode ECM consiste à trouver un point $A \in \mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})$ et un entier m tel que

$$\begin{cases} m \cdot A = 0_{\mathcal{E}_{a,b}(\mathbb{Z}/p\mathbb{Z})} & \text{dans } \mathcal{E}_{a,b}(\mathbb{Z}/p\mathbb{Z}) \\ m \cdot A \neq 0_{\mathcal{E}_{a,b}(\mathbb{Z}/p\mathbb{Z})} & \text{dans } \mathcal{E}_{a,b}(\mathbb{Z}/q\mathbb{Z}) \end{cases} . \quad (58)$$

(comparer avec l'équation (17))

Dans cette situation, le calcul de $m \cdot A$ dans $\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})$ échoue. En effet, tous les calculs restent valable modulo p et montrent que $m \cdot A$ est le point $0_{\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})}$. Mais tous les calculs restent valable modulo q aussi et montrent que $m \cdot A$ ne peut pas être le point $0_{\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})}$. Calculer $m \cdot A$ dans $\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})$ revient à faire une suite d'opérations arithmétiques dans $\mathbb{Z}/n\mathbb{Z}$: elles peuvent toutes être réalisées sauf lorsqu'on cherche à diviser par un élément x de $\mathbb{Z}/n\mathbb{Z}$ non inversible. Mais dans ce cas, le pgcd de x et de n est non trivial et fournit un facteur de n .

En faisant le choix $m = \text{ppcm}\{1, 2, 3, \dots, b\}$, on obtient l'algorithme suivant :

Algorithme 37 : Factorisation par courbes elliptiques de Lenstra (ECM)

Entrées : Entier n premier avec 6 et qui n'est pas une puissance d'un nombre premier. Borne B .

Sorties : Diviseur non-trivial de n ou « échec »

```

1 Tirer au hasard  $a, x_0, y_0 \in \llbracket 0, n-1 \rrbracket$ 
2  $b \leftarrow y_0^2 - x_0^3 - ax_0$ 
3  $g \leftarrow \text{pgcd}(4a^3 + 27b^2, n)$ 
4 si  $1 < g < n$  alors
5 |   retourner  $g$ 
6 sinon si  $g = n$  alors
7 |   retourner Echec
8  $\mathcal{E} \leftarrow \mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})$ 
9  $A \leftarrow (x_0, y_0)$ 
10 pour  $p$  premier  $\leq B$  faire
11 |   Trouver  $e$  maximal tel que  $p^e \leq B$ 
12 |   essayer
13 |   |    $A \leftarrow p^e \cdot A$  dans  $\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})$ 
14 |   excepté Erreur d'inversion
15 |   |   retourner Facteur de  $n$  obtenu
16 retourner Echec

```

Choix de la borne de friabilité Il y a de l'ordre de $O(B)$ opérations dans $\mathcal{E}_{a,b}(\mathbb{Z}/n\mathbb{Z})$ pour une tentative d'application de l'algorithme. On peut estimer également la probabilité de trouver un point A dont l'ordre soit B -friable dans $\mathcal{E}_{a,b}(\mathbb{Z}/p\mathbb{Z})$ et ne le soit pas dans $\mathcal{E}_{a,b}(\mathbb{Z}/q\mathbb{Z})$ (en vérité, la deuxième condition est très improbable lorsque la première est satisfaite et peut être oubliée). On obtient une probabilité de friabilité de l'ordre de

$$\text{prob}(B) = c \cdot |\mathcal{S}| \cdot \frac{p^{3/2}}{\log p} \cdot \frac{1}{p^2}$$

où \mathcal{S} est l'ensemble des nombres friables compris entre $p+1-2\sqrt{2}$ et $p+1+2\sqrt{2}$.

On peut ainsi optimiser la complexité de l'algorithme en minimisant $\frac{B}{\text{prob}(B)}$ (en s'appuyant sur des résultats de distribution des nombres friables). On obtient $L\left(1/2, \sqrt{2}/2 + \epsilon; \sqrt{p}\right)$ et une complexité en

$$L\left(1/2, \sqrt{2} + \epsilon; \sqrt{p}\right)$$

où

$$L(\alpha, \beta; N) = \exp\left(\beta(\log N)^\alpha (\log \log N)^{1-\alpha}\right).$$

Remarque 578. La complexité de ECM est liée à la taille du plus petit diviseur de n et non pas à n lui-même. Aussi, la méthode est particulièrement efficace pour éliminer les facteurs de n de taille moyenne. Elle peut être utilisée en complément à d'autres méthodes pour retirer de n les facteurs de petite et moyenne taille, le cas le plus défavorable étant le cas $n = pq$ avec p et q premiers de même taille.

Exemple 579. Cherchons à factoriser $n = 3397$ par la méthode ECM. On tire au hasard la cubique d'équation

$$y^2 = x^3 + 4x + 25$$

et le point $P = (3, -8)$.

— On commence par calculer $2P$ selon les formules habituelles. On calcule la pente de la tangente en P qui vaut

$$\lambda = \frac{3x_P^2 + a}{2y_P} = \frac{31}{-16} = 635 \pmod{n}.$$

On en tire le point $2P$ de coordonnées :

$$x_{2P} = \lambda^2 - 2x_P = 2373 \quad y_{2P} = -y_P + \lambda(x_P - x_{2P}) = 3326.$$

— On calcule ensuite $3P = 2P + P$. La pente de la droite $(P, 2P)$ devrait être

$$\lambda = \frac{y_{2P} - y_P}{x_{2P} - x_P} = \frac{3326 - (-8)}{2373 - 3} = \frac{3334}{2370}.$$

Mais 2370 n'est pas inversible modulo n . En effet, le pgcd $2370 \wedge 3397$ vaut 79. Le calcul de λ échoue et révèle, ce faisant, un facteur de n , à savoir 79.

Exercice 580. Afin d'implémenter l'algorithme ECM avec Sage, nous voulons redéfinir l'algorithme de division afin que le calcul de x/y dans $\mathbb{Z}/n\mathbb{Z}$ quand y n'est pas inversible lève une erreur qui indique le facteur de n que l'on a découvert. On définit

```
class FoundFactor(Exception):
    def __init__(self, value):
        self.value = value
    def __str__(self):
        return repr(self.value)
```

1. Ecrire une fonction division qui prend en entrée deux éléments x et y de $\mathbb{Z}/n\mathbb{Z}$ et renvoie
 - le quotient x/y lorsque celui-ci existe,
 - une erreur `ZeroDivisionError` (avec `raise ZeroDivisionError`) lorsque y est nul dans $\mathbb{Z}/n\mathbb{Z}$

— et une erreur *FoundFactor* (avec **raise** *FoundFactor*(d)) lorsque y est non nul mais n'est pas inversible dans $\mathbb{Z}/n\mathbb{Z}$.

- Écrire une fonction *addition* qui reprend les codes pour l'addition de point de l'exercice 560 en y intégrant la fonction *division*.
- Écrire une fonction *multiplication* qui prend en entrée un point P d'une courbe elliptique et un scalaire λ et renvoie le produit $\lambda \cdot P$. On pourra utiliser une méthode d'exponentiation rapide dont le principe est le suivant :

$$\forall \lambda \in \mathbb{N}, \quad \lambda \cdot P = \begin{cases} 0_{\mathcal{E}_{a,b}} & \text{si } \lambda = 0 \\ P & \text{si } \lambda = 1 \\ \frac{\lambda}{2} \cdot (P + P) & \text{si } \lambda \text{ est pair} \\ P + \frac{\lambda-1}{2} \cdot (P + P) & \text{si } \lambda \text{ est impair} \end{cases}$$

- Écrire une fonction *ECM* qui implante l'algorithme ECM de Lenstra. On pourra s'inspirer du code suivant

```
try:
    [faire quelque chose]
except FoundFactor as FF:
    return FF.value
```

Exercice 581. On pose $n = 42857766101$. Quelle est la plus petite valeur de la borne de friabilité B telle que l'algorithme factorise n avec les paramètres $a = b = 4$ et $P = (1, 3)$? [Indication : factoriser $|\mathcal{E}(\mathbb{F}_p)|$ et $|\mathcal{E}(\mathbb{F}_q)|$ où p et q sont des diviseurs premiers de n .]

Les faiblesses de la multiplication sur une courbe elliptique

L'exponentiation

Nous commençons par nous poser la question suivante : comment calculer la puissance a^n d'un élément a dans un groupe abélien G noté multiplicativement par un entier n avec le moins de produits possibles.

La solution naïve, qui revient à calculer

$$a^2 = a \cdot a, a^3 = a^2 \cdot a, a^4 = a^3 \cdot a, \dots, a = a^{n-1} \cdot a,$$

ce qui requiert $n - 1$ multiplication.

Exemple 582. Calcul de a^{31} en 7 multiplications.

$$a^2 = a \cdot a, \quad a^3 = a \cdot a^2, \quad a^6 = a^3 \cdot a^3, \quad a^{12} = a^6 \cdot a^6,$$

$$a^{24} = a^{12} \cdot a^{12}, \quad a^{30} = a^{24} \cdot a^6, \quad a^{31} = a^{30} \cdot a.$$

Définition 583. On note $\ell(n)$ le nombre minimal de multiplication à effectuer pour calculer a^n .

Proposition 584. L'entier $\ell(n)$ est le plus petit entier ℓ tel qu'il existe une suite finie $\alpha_0, \alpha_1, \alpha_2, \dots, \alpha_\ell = n$ avec $\alpha_0 = 1$ et telle que pour tout $1 \leq \rho \leq n$, il existe $0 \leq i, j < \rho$ tels que $\alpha_\rho = \alpha_i + \alpha_j$.

Exemple 585. On poursuit l'exemple avec le calcul de $\ell(37)$:

$$\alpha_1 = 2 = 1 + 1, \alpha_2 = 3 = 2 + 1, \alpha_3 = 6 = 3 + 3, \alpha_4 = 12 = 6 + 6,$$

$$\alpha_5 = 24 = 12 + 12, \alpha_6 = 30 = 24 + 6, \alpha_7 = 31 = 30 + 1$$

Il est clair que après h étapes, la plus grande valeur que l'on peut espérer avoir calculé est 2^h . Ainsi

Proposition 586. On a

$$\ell(n) \geq \lceil \log_2 n \rceil. \quad (59)$$

Théorème 587. L'algorithme 38 d'exponentiation rapide permet de calculer a^n en $(\lfloor \log_2 n \rfloor - 1 + w)$ multiplication où w est le nombre de bits non-nuls dans l'écriture binaire de n . En particulier

$$\ell(n) \leq \lfloor \log_2 n \rfloor - 1 + w \leq 2 \log_2 n. \quad (60)$$

Algorithme 38 : Exponentiation rapide

Entrées : Élément a d'un anneau A , entier $n \in \mathbb{N}$ de décomposition binaire $n = 2^k + n_{k-1}2^{k-1} + \dots + n_0$

Sorties : $a^n \in A$

```

1  $b \leftarrow a$ 
2 pour  $k-1 \geq i \geq 0$  faire
3   si  $n_i = 1$  alors
4      $b \leftarrow b^2 \cdot a$ 
5   sinon
6      $b \leftarrow b^2$ 
7 retourner  $a$ 
```

Exemple 588. La décomposition de 13 est 1101, donc

$$a^{13} = (((a)^2 \cdot a)^2)^2 \cdot a$$

Ceci correspond à la suite

$$\alpha_0 = 1, \quad \alpha_1 = 1 + 1 = 2, \quad \alpha_2 = 1 + 2 = 3$$

$$\alpha_3 = 3 + 3 = 6, \quad \alpha_4 = 6 + 6 = 12, \quad \alpha_5 = 12 + 1 = 13$$

Exercice 589. Donner une version récursive de l'algorithme d'exponentiation rapide et une version parcourant les bits de n du plus faible au plus fort.

Remarque 590. On connaît de meilleures bornes sur $\ell(n)$, notamment

$$\log_2 n + \log_2 w - 2.13 \leq \ell(n) \leq \log_2 n + \frac{\log_2 n}{\log_2 \log_2 n} + o\left(\frac{\log_2 n}{\log_2 \log_2 n}\right).$$

Exercice 591. Programmer un algorithme de calcul de $\ell(n)$. Comparer vos résultats avec la suite A003313 de l'« On-Line Encyclopedia of Integer Sequences ».

Indication : les suites n'étant pas hachables, on pourra stocker la suite $s = [n_1, \dots, n_k]$ par l'entier Python $s = \text{int}(1\{<\}n_1 + \dots + 1\{<\}n_k)$. Accéder au k -ième bit peut se faire simplement par $(c\{>\}k) \ \& \ \text{int}(1)$.

Exercice 592. Montrer que pour tous entiers $n, m \in \mathbb{N}^*$, $\ell(mn) \leq \ell(m) + \ell(n)$.

Attaque par canaux auxiliaires sur les courbes elliptique

Remarque 593. L'addition de deux points P et Q d'une courbe elliptique procède de formules différentes selon que $P \neq Q$ ou non. Un observateur extérieur peut donc apprendre si un processeur effectue une addition ou une duplication à partir d'une mesure physique (consommation d'énergie, émanations électromagnétiques, rayonnement thermique, bruits émis etc.). Or de nombreux protocoles cryptographiques consistent à calculer $s \cdot P_0$ où P_0 est un point de la courbe et s un entier secret. Ce calcul se fait généralement par l'algorithme d'exponentiation rapide qui lui aussi distingue des phases d'additions et de duplications en fonction des bits de s . Il devient alors enfantin pour un observateur de lire s à partir de sa mesure.

Exercice 594. 1. Coder l'algorithme d'exponentiation rapide pour calculer mP où $m \in \mathbb{Z}$ et P est un point quelconque d'une courbe elliptique \mathcal{E} .

(On rappelle que $//$ et $\%$ donnent le quotient et le reste dans la division entière. On pourra avoir besoin de $\text{int}(n).\text{bit_length}()$ ou de $(n\{>>k\})\%2$ pour déterminer le k -ième bit de n .)

2. Vérifier votre algorithme avec la courbe

$$y^2 = x^3 + ax + 1 \quad P_0 = (0, 1)$$

pour différentes valeurs de a et différents multiples de P_0 .

3. (Attaque par canaux auxiliaires) Un observateur étranger au système observe le signal électromagnétique suivant (figure 45) lors de la multiplication d'un point d'une courbe elliptique par un entier secret s . Que peut-on déduire sur s ?

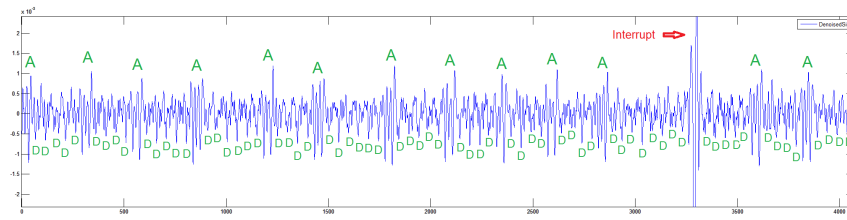


FIGURE 45: Signal électromagnétique enregistré (tiré de <https://www.tau.ac.il/~tromer/mobilesc/>). A signale une addition, D une duplication.

Courbes d'Edwards

On suppose ici que la caractéristique est différente de 2. On appelle *courbe d'Edwards* toute courbe d'équation affine

$$\mathcal{ED}_d : x^2 + y^2 = 1 + dx^2y^2 \quad (61)$$

où $d \in \mathbb{F}_q \setminus \{0, 1\}$.

Les courbes d'Edwards sont des courbes elliptiques. La plupart des courbes elliptiques possède une équation d'Edwards. L'élément neutre de \mathcal{ED}_d a pour coordonnée $(0, 1)$. La somme R de deux points $P = (x_P, y_P)$ et $Q = (x_Q, y_Q)$ de \mathcal{ED}_d a pour coordonnée :

$$x_R = \frac{x_P y_Q + x_Q y_P}{1 + dx_P x_Q y_P y_Q}, \quad y_R = \frac{y_P y_Q - x_P x_Q}{1 - dx_P x_Q y_P y_Q}.$$

Exercice 595. Quel avantage cryptographique y a-t-il à utiliser une courbe d'Edwards ?

Exemple 596. Plus généralement, les *courbes d'Edwards tordues* sont des courbes d'équation affine

$$ax^2 + y^2 = 1 + dx^2y^2 \in \mathbb{F}_p[x, y].$$

La courbe obtenue en prenant

$$p = 2^{255} - 19, \quad a = -1 \quad \text{et} \quad d = -121665/121666$$

s'appelle la courbe *Ed25519* et est employée par de nombreux logiciels dans des schémas de signature digitale.

TP 14 : Logarithme discret et couplages

Buts : Comprendre comment fonctionnent quelques techniques de recherche du logarithme discret

Travaux préparatoires : Cours et exercices 605 (Shanks).

Évaluation du TP : Exercices 604 (Shanks : pas de bébé, pas de géant), 607 (ρ de Pollard), 632 (couplage), 635 (attaque M.O.V.), 614 (calcul d'indice).

Définition 597. Soit G un groupe abélien fini et b, g deux éléments de G . Le problème du *logarithme discret de base b* consiste à trouver un entier k (s'il existe) tel que $g = b^k$.

Avec SageMath, `discrete_log`.

La sécurité de nombreux algorithmes de cryptographie repose sur le fait que le problème du logarithme discret n'admet pas de résolution facile. Selon le groupe utilisé, cette hypothèse de sécurité est de moins en moins correcte de nos jours. De grands progrès ont été accomplis en 2013 rendant la question résoluble en temps quasi-polynomial dans le cas où G est le groupe multiplicatif d'un corps fini de petite caractéristique.

Exemple 598 (Diffie-Hellman). Le protocole de Diffie-Hellman (1976) permet à deux parties, Alice et Bob, de convenir d'un secret commun sur un canal public.

Paramètres Alice et Bob s'accordent sur un groupe abélien et un générateur g .

Clés Secrètement, Alice et Bob choisissent chacun un entier a et b (respectivement).

Échange Alice communique $g_a = g^a$ à Bob, Bob communique $g_b = g^b$ à Alice.

Secret commun Le secret commun est $g^{ab} = (g_a)^b = (g_b)^a$.

Savoir résoudre le problème du logarithme discret permet de casser ce protocole.

Remarque 599. Dans le cas du groupe \mathbb{F}_p^\times , il est conseillé d'utiliser les valeurs de p fixées par le document RFC3526 (cf. <https://tools.ietf.org/html/rfc3526>). D'après l'ANSSI³¹, il ne faut pas utiliser de

31. ANSSI. Guide de sélection d'algorithmes cryptographiques. Technical Report ANSSI-PA-079, Agence nationale de la sécurité des systèmes d'information, 51, boulevard de La Tour-Maubourg, 75700 PARIS 07, 8 mars 2021. Version 1.0

premier p de moins de 3072 bits. Par exemple, p donné en notation hexadécimale :

```

FFFFFFFF FFFFFFFF C90FDAA2 2168C234 C4C6628B 80DC1CD1
29024E08 8A67CC74 020BBEA6 3B139B22 514A0879 8E3404DD
EF9519B3 CD3A431B 302B0A6D F25F1437 4FE1356D 6D51C245
E485B576 625E7EC6 F44C42E9 A637ED6B 0BFF5CB6 F406B7ED
EE386BFB 5A899FA5 AE9F2411 7C4B1FE6 49286651 ECE45B3D
C2007CB8 A163BF05 98DA4836 1C55D39A 69163FA8 FD24CF5F
83655D23 DCA3AD96 1C62F356 208552BB 9ED52907 7096966D
670C354E 4ABC9804 F1746C08 CA18217C 32905E46 2E36CE3B
E39E772C 180E8603 9B2783A2 EC07A28F B5C55DF0 6F4C52C9
DE2BCBF6 95581718 3995497C EA956AE5 15D22618 98FA0510
15728E5A 8AAAC42D AD33170D 04507A33 A85521AB DF1CBA64
ECFB8504 58DBEF0A 8AEA7157 5D060C7D B3970F85 A6E1E4C7
ABF5AE8C DB0933D7 1E8C94E0 4A25619D CEE3D226 1AD2EE6B
F12FFA06 D98A0864 D8760273 3EC86A64 521F2B18 177B200C
BBE11757 7A615D6C 770988C0 BAD946E2 08E24FA0 74E5AB31
43DB5BFC E0FD108E 4B82D120 A93AD2CA FFFFFFFF FFFFFFFF

```

et $g = 2$.

Dans le cas du groupe des points d'une courbe elliptiques, des courbes telles que la courbe de l'exemple 571 sont actuellement recommandées.

Exemple 600 (El Gamal). Le système d'Elgamal (1985) est un système de cryptographie asymétrique basé sur Diffie-Hellman qui permet une partie (Bob) d'envoyer des messages à une seconde partie (Alice).

Génération de clés Alice choisit un groupe cyclique G d'ordre q et un générateur b . Elle choisit également un entier $s \in \llbracket 1, q-1 \rrbracket$ et calcule $g = b^s$.

Clé privée (G, s)

Clé publique (G, q, b, g) .

Chiffrement Pour envoyer un message m à Alice, Bob choisit $t \in \llbracket 1, q-1 \rrbracket$; il calcule $c_1 = b^t$ et $c_2 = m \cdot g^t$. Bob envoie (c_1, c_2) .

Déchiffrement Alice peut retrouver m en calculant $c_2 \cdot c_1^{-s}$.

Exemple 601 (Shamir, Massey-Omura). Un protocole à trois passes pour l'échange de messages est un protocole qui permet de transmettre un message en toute confidentialité au terme de trois communications et sans échange de clés préalable. Nous en présentons un qui s'appuie sur deux primitives E et D de chiffrement et de déchiffrement qui « commutent » entre elles.

Déroulement : Alice souhaite envoyer un message m à Bob.

Etape 1 Alice fixe une clé de chiffrement s et de déchiffrement t . Elle envoie $E(m, s)$ à Bob.

Etape 2 Bob fixe une clé de chiffrement r et de déchiffrement q . Il chiffre $E(m, s)$ avec r et envoie $E(E(m, s), r)$ à Alice.

Etape 3 Alice déchiffre $E(E(m, s), r)$ avec t et renvoie $D(E(E(m, s), r), t)$ à Bob.

À cause de la commutativité, Bob reçoit en réalité $E(m, r)$ qu'il peut déchiffrer. Si l'on travaille dans un groupe cyclique d'ordre n , on peut choisir comme primitives E et D simplement $E(m, a) = m^a$ et $D(c, b) = c^b$ (protocoles de Shamir et de Massey-Omura). Pour obtenir la clé de déchiffrement, on fixe un couple $b = a^{-1} \bmod n$. La sécurité repose alors sur la difficulté du calcul du logarithme.

Les algorithmes actuels de calcul du logarithme discret se classent en deux familles : celle des algorithmes généraux (qui utilisent uniquement les opérations de groupe tels que des produits, des inversion ou des tests d'égalité et éventuellement des informations sur la taille du groupe, voire sa factorisation) et celle des algorithmes dédiés à des groupes particuliers (qui tirent parti de la description spécifique du groupe : corps finis, courbes elliptiques, etc).

Depuis 2013, on connaît des méthodes très performantes pour calculer le logarithme discret dans le groupe multiplicatif \mathbb{F}_q^\times : en particulier lorsque le corps est de petite caractéristique, les méthodes connues fonctionnent en complexité quasi-polynomiale (i.e. $O(n^{\log n})$, où n est la taille en nombre de bits de l'entrée). À ce jour, il est recommandé de ne pas utiliser de protocoles cryptographiques reposant sur le logarithme discret dans \mathbb{F}_q^\times .

Méthodes génériques de recherche du logarithme

Par force brute

On peut bien sûr calculer $b, b^2, b^3 \dots$ jusqu'à obtenir g .

Exercice 602. Quelle est la complexité de la méthode par force brute ? La recommandez-vous en pratique ?

Algorithme « pas de bébé, pas de géant » de Shanks

La méthode Baby-step-giant-step est due à Shanks (1977). Elle repose sur le fait que si G est d'ordre $\leq s^2$, alors k peut s'écrire $k = is + j$ avec $0 \leq i, j < s$. Mais alors

$$b^k = g \Leftrightarrow b^{is+j} = g \Leftrightarrow gb^{-j} = b'^i \text{ avec } b' = b^s.$$

Elle améliore la méthode par force brute en stockant gb^{-j} pour différentes valeurs de j (les pas de bébé) et parcourt b'^i (les pas de géant).

Algorithme 39 : Pas de bébé, pas de géant

Entrées : Générateur b d'un groupe $(G, \cdot, 1_G)$ d'ordre n et élément $g \in G$.

Sorties : Entier $k = \log_b(g)$ tel que $b^k = g$.

```

1  $s \leftarrow \lfloor \sqrt{n} \rfloor + 1$ 
2  $T \leftarrow$  table de hachage vide
3 pour  $j \in \llbracket 0, s-1 \rrbracket$  faire
4    $\beta_j \leftarrow gb^{-j}$ 
5    $T[\beta_j] \leftarrow j$ .
6  $i \leftarrow 0$ 
7  $\gamma \leftarrow 1_G$ 
8  $b' \leftarrow b^s$ 
9 répéter
10   $i \leftarrow i + 1$ 
11   $\gamma \leftarrow \gamma b'$ 
12 jusqu'à  $\gamma \in \text{clés}(T)$ ;
13  $j \leftarrow T[\gamma]$ 
14 retourner  $k = is + j$ 
```

Exemple 603. On travaille dans le groupe des points de la courbe elliptique $\mathcal{E}(\mathbb{F}_{47})$ d'équation

$$y^2 - x^3 - 12x + 5 = 0 \in \mathbb{F}_{47}[x, y].$$

On a fixé $G = (37, 12)$ et $B = (10, 38)$. On cherche à trouver un entier λ tel que $\lambda B = G$. La borne de Hasse est de $p + 1 + 2\sqrt{p} = 61,7$. On commence donc par calculer une table de hachage contenant les points

j	$G - jB$
0	(37, 12)
1	(18, 11)
2	(22, 35)
3	(21, 22)
4	(24, 30)
5	(27, 36)
6	(0, 29)
7	(41, 6)

On a également $B' = 8B = (17, 6)$. On calcule successivement B' , $2B'$, $3B'$ etc. Notons que B' n'est pas dans la table de hachage mais $2B'$ l'est. On en déduit que $2B' = G - 6B$, soit $G = (2 \cdot 8 + 6)B$. Le logarithme discret est donc $\lambda = 22$.

Exercice 604. 1. Programmer l'algorithme de Shanks dans le cas où

On peut créer une table de hachage avec un dictionnaire $T = \{\}$ et utiliser la méthode `has_key`.

$G = \mathbb{F}_q^\times$ (q puissance d'un premier).

2. Résoudre l'équation $3^x = 693 \pmod{1823}$.

3. Calculer $\log_2 15$ dans \mathbb{F}_{239} .

Exercice 605. Quelle avantage a-t-on à utiliser une table de hachage dans l'algorithme 40 par rapport à d'autres structures de données ?

La méthode ρ de Pollard

L'idée de la méthode ρ de Pollard est de considérer une marche à travers G de la forme $w_i = b^{\alpha_i} g^{\beta_i}$ où α_i et β_i sont des entiers connus. Lorsqu'une collision se produit, c'est-à-dire lorsque deux indices i et j donnent

$$w_i = b^{\alpha_i} g^{\beta_i} = b^{\alpha_j} g^{\beta_j} = w_j,$$

si de plus $\beta_j - \beta_i$ est inversible modulo $|G|$, on en tire

$$\log_b g = \frac{\alpha_i - \alpha_j}{\beta_j - \beta_i} \pmod{|G|}.$$

Si la suite $(w_i)_{i \in \mathbb{N}}$ est aléatoire, par le « paradoxe des anniversaires », on peut s'attendre à obtenir une collision en $O(\sqrt{|G|})$.

La méthode ρ de Pollard utilise une suite définie par récurrence :

$$\begin{cases} w_0 &= b \\ w_{i+1} &= \Phi(w_i) \end{cases}$$

où G est partitionné en trois sous-ensemble de tailles proches G_1 , G_2 et G_3 et

$$\Phi(w) = \begin{cases} g \cdot w & \text{si } w \in G_1 \\ w^2 & \text{si } w \in G_2 \\ b \cdot w & \text{si } w \in G_3 \end{cases}$$

Autrement dit :

$$(\alpha_{i+1}, \beta_{i+1}) = \begin{cases} (\alpha_i, \beta_i + 1) \\ (2\alpha_i, 2\beta_i) \\ (\alpha_i + 1, \beta_i) \end{cases} \text{ respectivement.}$$

La partition de G en trois parties peut se faire à l'aide d'une fonction de hachage. La suite $(w_i)_{i \in \mathbb{N}}$ prend la même forme que décrite par la figure 29.

On ne parvient pas toujours à faire apparaître une collision qui révèle $\log_b g$. On peut dans ce cas recommencer l'algorithme en modifiant la partition de G en 3 parties ou en démarrant avec $w_0 = b^{\alpha_0} g^{\beta_0}$ tiré au hasard.

Algorithme 40 : Rho de Pollard pour le logarithme

Entrées : Générateur b d'ordre n d'un groupe G et
élément $g \in G$.

Sorties : Entier $k = \log_b(g)$ tel que $b^k = g$.

```

1 Définir  $\Phi$ 
2  $\alpha_0, \beta_0 \leftarrow$  nombres aléatoires
3  $x, \alpha_x, \beta_x \leftarrow \Phi(b^{\alpha_0} g^{\beta_0}, \alpha_0, \beta_0)$ 
4  $y, \alpha_y, \beta_y \leftarrow \Phi(\Phi(b^{\alpha_0} g^{\beta_0}, \alpha_0, \beta_0))$ 
5 tant que  $x \neq y$  faire
6    $x, \alpha_x, \beta_x \leftarrow \Phi(x, \alpha_x, \beta_x)$ 
7    $y, \alpha_y, \beta_y \leftarrow \Phi(\Phi(y, \alpha_y, \beta_y))$ 
8 essayer
9   retourner  $(\alpha_x - \alpha_y)(\beta_y - \beta_x)^{-1} \bmod n$ 
10 excepté Erreur d'inversion
11   Changer  $\Phi, \alpha_0$  et  $\beta_0$ . Recommencer

```

Exemple 606. On se place dans le groupe multiplicatif \mathbb{F}_{281}^\times . Pour $i \in \llbracket 1, 3 \rrbracket$, on note $G_i = \{a \in \mathbb{F}_{281}^\times; \tilde{a} \equiv i \bmod 3\}$ où \tilde{a} est le relèvement de a par un entier naturel compris entre 1 et 280. On souhaite calculer le logarithme de $g = 124$ dans la base $b = 257$. On calcule la suite

i	w_i	α_i	β_i
0	257	1	0
1	14	2	0
2	196	4	0
3	138	4	1
4	60	5	1
5	246	6	1
6	278	7	1
7	9	14	2
8	65	15	2
9	10	30	4
10	116	30	5
11	249	60	10
12	206	61	10
13	5	122	20
14	25	244	40
15	9	244	41
16	65	245	41

Le première collision survient entre w_7 et w_{15} , ce qui fournit l'équation

$$145 = b^{14} g^2 = b^{244} g^{41}.$$

On en déduit que $\lambda = (14 - 244) \cdot (41 - 2)^{-1} = 30 \bmod 280$ satisfait $b^\lambda = g$.

- Exercice 607.** 1. Vérifier expérimentalement que la fonction `hash(b)%3` partage G en trois parties de tailles semblables lorsque $G = \mathbb{F}_q^\times$.
2. Implémenter l'algorithme ρ de Pollard pour $G = \mathbb{F}_q^\times$.
3. Calculer $\log_{239} 263, \log_{127} 165, \log_{199} 210$ dans \mathbb{F}_{281}^\times .

Algorithme de Pohlig-Hellman

La méthode de Pohlig-Hellman permet de montrer que si l'ordre $|G|$ de G est connu et peut être factorisé, la difficulté du problème du logarithme discret est essentiellement liée à la taille des facteurs de $|G|$. Cette méthode est utile dans le cas où $|G|$ est un nombre friable.

Lemme chinois Notons $n = |G|$ et supposons que n possède la factorisation

$$n = p_1^{e_1} p_2^{e_2} \cdots p_t^{e_t}$$

en produit de nombres premiers. Pour retrouver k tel que $b^k = g$, il suffit de déterminer k modulo chacun des $p_i^{e_i}$ puis d'utiliser le lemme chinois pour reconstruire k . Notons dans ce qui suit k_i un résidu de k modulo $p_i^{e_i}$.

Pour tout $1 \leq i \leq t$, on pose

$$b_i = b^{n/p_i^{e_i}}, \quad G_i = \langle b_i \rangle \quad \text{et} \quad g_i = g^{n/p_i^{e_i}}.$$

Notons que G_i est un groupe d'ordre $p_i^{e_i}$. En passant à la puissance $n/p_i^{e_i}$ de l'équation $b^k = g$, on obtient

$$b^{kn/p_i^{e_i}} = b_i^k = b_i^{k_i} = g_i,$$

ce qui réduit le problème de détermination de k à t problèmes de logarithmes discrets dans les sous-groupes plus petits G_i .

Relèvement p -adique Nous supposons désormais que $|G| = p^e$. Nous cherchons toujours k tel que $b^k = g$. Nous notons $k = k_{e-1}p^{e-1} + \cdots + k_1p + k_0$ la décomposition de k en base p (avec $0 \leq k_i < p$). Nous notons dans ce qui suit

$$b_i = b^{p^i} \quad \text{et} \quad g_i = g^{p^i}.$$

En élevant à la puissance p^{e-1} l'équation $b^k = g$, on obtient

$$b_{e-1}^{k_0} = g_{e-1}.$$

qui est un problème de logarithme discret dans l'unique sous-groupe $G_0 = \langle b_{e-1} \rangle$ d'ordre p de G . En supposant avoir déjà calculé k_0, \dots, k_j , on a de même

$$b_{e-1-j}^{k_j} = g_{e-1-j} b_{e-1-j}^{-(k_0 + k_1p + \cdots + k_{j-1}p^{j-1})}$$

qui est encore une fois le même problème de logarithme discret dans le groupe G_0 d'ordre p .

Remarque 608. Il peut être intéressant de précalculer le logarithme discret de l'ensemble des éléments du sous-groupe G_0 .

Exercice 609. 1. Programmer un algorithme de calcul du logarithme lorsque $|G|$ est une puissance de p

2. Programmer un algorithme de calcul du logarithme lorsque $|G|$ est de la forme $n = p_1^{e_1} p_2^{e_2} \cdots p_t^{e_t}$.

Dans chaque cas, on pourra utiliser un algorithme vu précédemment pour les cas finaux.

Exemple 610. On se place dans \mathbb{F}_{8101} avec $b = 6$ et $g = 7531$. On a $8100 = 2^2 3^4 5^2$. On cherche k tel que $6^k = 7531 \pmod{8101}$. On obtient

$$k \equiv \begin{cases} 1 & \pmod{2^2} \\ \overline{1202}_{(3)} = 47 & \pmod{3^4} \\ \overline{24}_{(5)} = 14 & \pmod{5^2} \end{cases}$$

ce qui conduit à $k = 6689 \pmod{8100}$.

Logarithme par calcul d'indice

La méthode par calcul d'indice consiste en plusieurs phases.

Phase 1 : On commence par fixer un certain sous-ensemble d'éléments du groupe G que l'on appelle base de friabilité ou base de friabilité $\{g_i; i \in I\}$ et que l'on choisit petits selon un certain critère.

Phase 2 : On génère toute sorte de relations de la forme

$$\prod_{i \in I} g_i^{m_i} = \prod_{i \in I} g_i^{n_i}$$

qui impliquent par suite que

$$\sum_{i \in I} m_i \log_b g_i = \sum_{i \in I} n_i \log_b g_i$$

Phase 3 : Lorsque suffisamment de relations ont été trouvées, on résoud le système obtenu pour calculer les valeurs de $\log_b g_i$ pour $i \in I$.

Phase 4 : On factorise l'élément g dans la base de friabilité $g = \prod_{i \in I} g_i^{\alpha_i}$ et on en déduit son logarithme $\log_b g = \sum_{i \in I} \alpha_i \log_b(g_i)$.

Nous présentons une version de cet algorithme dans le cas où $G = \mathbb{F}_p^\times$ (p premier). Cette version a le mérite d'être de complexité sous-exponentielle. Elle est loin cependant des meilleures complexités que l'on sait atteindre aujourd'hui.

Commençons par choisir une borne de friabilité y et calculons l'ensemble des premiers $\{p_1, \dots, p_k\}$ inférieurs à $\leq y$.

Pour α tiré au hasard, nous espérons que $b^\alpha \pmod p$ soit y -friable et $b^\alpha = p_1^{e_1} \cdots p_k^{e_k}$. Si tel est le cas, nous en tirons

$$\alpha = e_1 \log_b p_1 + \cdots + e_k \log_b p_k \pmod{p-1} \quad (62)$$

Dans l'algorithme ci-dessous, nous ferons le pari que $4k$ équations de ce type seront suffisantes à fournir un système inversible et permettront de calculer $(\log_b p_i)_{i \leq k}$ par résolution d'un système linéaire.

Finalement, nous cherchons un entier β tel que $b^\beta g$ soit aussi y -friable et $b^\beta g = p_1^{f_1} \cdots p_k^{f_k}$. On a aussi

$$\beta + \log_b g = f_1 \log_b p_1 + \cdots + f_k \log_b p_k \pmod{p-1} \quad (63)$$

ce qui à ce point doit permettre de retrouver $\log_b g$.

Remarque 611. Il reste à choisir une borne de friabilité y optimale. Le choix idéal repose sur des résultats de théorie analytique des nombres qui estiment la probabilité d'être y -friable. En prenant, $y = e^{(2^{-1/2} + o(1))\sqrt{\ln p \ln \ln p}}$, on minimise le temps d'exécution en moyenne de l'algorithme qui est alors $e^{(3/\sqrt{2} + o(1))\sqrt{\ln p \ln \ln p}}$.

Remarque 612. Toute l'algèbre linéaire se fait modulo $p-1$ qui n'a aucune raison d'être premier. Cela dit, il est possible de travailler modulo chacun des facteurs de $p-1$ puis de recomposer le résultat par le lemme chinois. On peut même se dispenser d'une factorisation préalable de $p-1$, effectuer des éliminations gaussiennes comme si $p-1$ était premier. Soit le calcul passe, soit erreur se produit lors d'une inversion modulo $p-1$, mais alors, on a trouvé un facteur de $p-1$ et on recommence modulo chacun des facteurs.

Algorithme 41 : Logarithme par calcul d'indice**Entrées** : Éléments $b, g \in \mathbb{F}_p^\times$. Borne de friabilité y **Sorties** : Entier ℓ tel que $b^\ell = g$.

```

1 Déterminer la liste  $\{p_1, \dots, p_k\}$  des premiers  $\leq y$ .
2 répéter
3    $i \leftarrow 1$ 
4   tant que  $i \leq 4k$  faire
5     Choisir un entier  $\alpha_i$  aléatoirement dans  $\llbracket 0, p-2 \rrbracket$ 
6      $\gamma_i \leftarrow b^{\alpha_i} \bmod p$ 
7     si  $\gamma_i$  est  $y$ -friable alors
8       Factoriser  $\gamma_i = p_1^{e_{i,1}} \cdots p_k^{e_{i,k}}$ 
9        $\mathbf{v}_i \leftarrow (e_{i,1}, \dots, e_{i,k})$ 
10       $i \leftarrow i + 1$ 
11 jusqu'à  $(\mathbf{v}_i)_{i \leq 4k}$  engendre  $(\mathbb{Z}/(p-1)\mathbb{Z})^k$ ;
12 Calculer  $(\log_b p_i)_{i \leq k}$  dans  $\mathbb{Z}/(p-1)\mathbb{Z}$  par résolution du
    système d'équations (62).
13 répéter
14   Choisir  $\beta$  aléatoirement dans  $\llbracket 0, p-2 \rrbracket$ 
15 jusqu'à  $b^\beta g$  est  $y$ -friable;
16 Calculer la factorisation  $b^\beta g = p_1^{f_1} \cdots p_k^{f_k}$ .
17 retourner  $\ell = -\beta + f_1 \log_b p_1 + \cdots + f_k \log_b p_k$  d'après l'équation
    63

```

Exemple 613. Disons que l'on souhaite trouver le logarithme discret de $g = 237 \in \mathbb{F}_{439}$ en base $b = 136 \in \mathbb{F}_{439}$. On fixe $y = 11$ comme borne de friabilité. On a tiré des exposants au hasard et obtenu les relations :

$$\begin{aligned}
 b^{434} &= 175 = 5^2 \cdot 7 \\
 b^{72} &= 144 = 2^4 \cdot 3^2 \\
 b^{374} &= 396 = 2^2 \cdot 3^2 \cdot 11 \\
 b^{90} &= 9 = 3^2 \\
 b^{225} &= 243 = 3^5 \\
 b^{320} &= 350 = 2 \cdot 5^2 \cdot 7 \\
 b^{244} &= 90 = 2 \cdot 3^2 \cdot 5
 \end{aligned}$$

On résoud le système

$$\left\{ \begin{array}{lll}
 2 \log_b 5 + \log_b 7 & \equiv & 434 \pmod{438} \\
 4 \log_b 2 + 2 \log_b 3 & \equiv & 72 \pmod{438} \\
 2 \log_b 2 + 2 \log_b 3 + \log_b 11 & \equiv & 374 \pmod{438} \\
 3 \log_b 3 & \equiv & 90 \pmod{438} \\
 5 \log_b 3 & \equiv & 225 \pmod{438} \\
 \log_b 2 + 2 \log_b 5 + \log_b 7 & \equiv & 320 \pmod{438} \\
 \log_b 2 + 2 \log_b 3 + \log_b 5 & \equiv & 244 \pmod{438}
 \end{array} \right.$$

On obtient

$$\begin{cases} \log_b 2 &= 324 \pmod{438} \\ \log_b 3 &= 45 \pmod{438} \\ \log_b 5 &= 268 \pmod{438} \\ \log_b 7 &= 336 \pmod{438} \\ \log_b 11 &= 74 \pmod{438} \end{cases}$$

Par ailleurs, nous avons obtenu (par tirage au sort) la relation

$$b^{420}g = 280 = 2^3 \cdot 5 \cdot 7.$$

Il vient alors

$$\log_b g = -420 + 3\log_b 2 + \log_b 5 + \log_b 7 = 280 \pmod{438}.$$

Exercice 614. Programmer l'algorithme 41 du logarithme par calcul d'indice.

Exercice 615. Par un calcul d'indice avec la base de friabilité $\{2, 3, 5, 7, 11\}$, résoudre $7^x = 13 \pmod{2039}$. Quelle est la plus petite base que l'on peut utiliser pour résoudre ce problème ?

Attaques par couplages

Loi de réciprocité de Weil

On désigne ici par \mathcal{C} , \mathcal{C}_1 et \mathcal{C}_2 des courbes algébriques projectives lisses.

ÉVALUATION D'UNE FONCTION RATIONNELLE SUR UN DIVISEUR Si $f \in \mathbb{F}_q(\mathcal{C})$ une fonction rationnelle et $D \in \text{div}(\mathcal{C})$ est un diviseur décrit par l'équation (50), on peut évaluer f en D selon la règle :

$$f(D) = \prod_{P \in \mathcal{C}(\bar{\mathbb{F}}_q)} f(P)^{n_P} \quad (64)$$

pour peu que $\text{div}(f)$ et D soient à support disjoint.

TIRÉS EN AVANT, POUSSÉS EN ARRIÈRE Soit $\phi : \mathcal{C}_1 \rightarrow \mathcal{C}_2$ un morphisme entre les deux courbes \mathcal{C}_1 et \mathcal{C}_2 . On peut définir deux applications, le *tiré en arrière* et le *poussé en avant*,

$$\phi^* : \begin{cases} k(\mathcal{C}_2) & \rightarrow k(\mathcal{C}_1) \\ f & \mapsto f \circ \phi \end{cases}$$

$$\text{et } \phi_* : \begin{cases} k(\mathcal{C}_1) & \rightarrow k(\mathcal{C}_2) \\ f & \mapsto (\phi^*)^{-1} \circ \text{Norme}_{k(\mathcal{C}_1)/\phi^*k(\mathcal{C}_2)}(f) \end{cases}$$

qui passent au diviseur de la façon suivante

$$\phi^* : \begin{cases} \text{div}(\mathcal{C}_2) & \rightarrow & \text{div}(\mathcal{C}_1) \\ (Q) & \mapsto & \sum_{P \in \phi^{-1}(Q)} e_P \cdot (P) \end{cases} \quad (65)$$

$$\text{et } \phi_* : \begin{cases} \text{div}(\mathcal{C}_1) & \rightarrow & \text{div}(\mathcal{C}_2) \\ (Q) & \mapsto & \phi(Q) \end{cases} \quad (66)$$

où e_P est l'indice de ramification de ϕ au point P , i.e. (voir 3.12.3 de ³²) l'entier tel que $\text{ord}_P \phi^*(f) = e_P \text{ord}_{\phi(P)}(f)$ pour tout $f \in k(\mathcal{C}_1)$.

32. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

Lemme 616. Soient, de plus, $f_1 \in k(\mathcal{C}_1)$ et $f_2 \in k(\mathcal{C}_2)$, alors

1. $\phi^*(\text{div}(f_2)) = \text{div}(\phi^* f_2)$,
2. $\phi_*(\text{div}(f_1)) = \text{div}(\phi_* f_1)$,
3. $\forall D_2 \in \text{div}(\mathcal{C}_2), f_1(\phi^* D_2) = (\phi_* f_1)(D_2)$,
4. $\forall D_1 \in \text{div}(\mathcal{C}_1), f_2(\phi_* D_1) = (\phi^* f_2)(D_1)$.

RÉCIPROCITÉ DE WEIL Nous disposons du résultat remarquable suivant (qui reste vrai pour d'autres courbes que les courbes elliptiques)

Théorème 617 (Loi de réciprocité de Weil). Soient f et $g \in \mathcal{E}(\overline{\mathbb{F}}_q)$ deux fonctions rationnelles telles que $\text{div}(f)$ et $\text{div}(g)$ sont à supports disjoints, alors

$$f(\text{div}(g)) = g(\text{div}(f)).$$

Démonstration. 1. Preuve du résultat quand $f, g \in k(\mathbb{P}^1)$. Dans ce cas, il existe des entiers naturels n et m , des scalaires distincts $(a_i)_{i \leq n}, (b_j)_{j \leq m} \subseteq \overline{\mathbb{F}}_q$, des entiers relatifs $(\alpha_i)_{i \leq n}, (\beta_j)_{j \leq m} \subseteq \mathbb{Z}$ et des scalaires non nuls $\omega, \mu \in \mathbb{F}_q$ tels que

$$f = \omega \cdot \prod_{i=1}^n (x - a_i)^{\alpha_i} \quad \text{et} \quad g = \mu \cdot \prod_{j=1}^m (x - b_j)^{\beta_j}$$

Comme les diviseurs sont disjoints, $\text{ord}_\infty(f) = -\sum_{i=1}^n \alpha_i$ ou $\text{ord}_\infty(g) = -\sum_{j=1}^m \beta_j$ est nul. On a alors

$$\begin{aligned} f(\text{div}(g)) &= \prod_{j=1}^m \left(\omega \prod_{i=1}^n (b_j - a_i)^{\alpha_i} \right)^{\beta_j} \cdot \underbrace{\left(\omega^{-\sum_{j=1}^m \beta_j} \right)}_{\substack{\text{n'apparaît que si} \\ \infty \text{ est un zéro ou pôle de } g}} \\ &= \prod_{j=1}^m \prod_{i=1}^n (b_j - a_i)^{\alpha_i \beta_j} \\ &= \underbrace{(-1)^{\sum_{i=1}^n \alpha_i \sum_{j=1}^m \beta_j}}_{=1} \cdot \prod_{i=1}^n \prod_{j=1}^m (a_i - b_j)^{\alpha_i \beta_j} \\ &= g(\text{div}(f)) \end{aligned}$$

2. Preuve du résultat dans le cas général. On a $\text{div}(g) = \text{div}(g^* \text{id}) = g^*(\text{div}(\text{id}))$ (point 1 du lemme), donc

$$f(\text{div}(g)) = f(g^*(\text{div}(\text{id})))$$

Par le point 3 du lemme, $f(g^*(\text{div}(\text{id}))) = (g_* f)(\text{div}(\text{id}))$. Mais $f \circ g^*$ est une fonction rationnelle de \mathbb{P}^1 , droite sur laquelle nous venons de prouver la loi de réciprocité de Weil. Donc $f \circ g^*(\text{div}(\text{id})) = \text{id}(\text{div}(g_* f))$. À présent, par le point 2 du lemme, $\text{id}(\text{div}(f \circ g^*)) = (g^* \text{id})(\text{div}(f))$. Mais, $g^* \text{id} = g$, donc $\text{id}(\text{div}(f \circ g^*)) = g(\text{div}(f))$, cqfd.

□

Définition des couplages

Définition 618. Soient $(\mathcal{G}_1, +)$, $(\mathcal{G}_2, +)$ deux groupes abélien d'exposant r (i.e. pour tout $P \in \mathcal{G}_1$ ou $P \in \mathcal{G}_2$, $r \cdot P = 0_{\mathcal{G}}$) et (\mathcal{G}_T, \cdot) un groupe abélien cyclique d'ordre r . On appelle *couplage* toute application $e : \mathcal{G}_1 \times \mathcal{G}_2 \rightarrow \mathcal{G}_T$ telle que

1. e est bilinéaire, i.e.

$$\forall P, P' \in \mathcal{G}_1, \forall Q \in \mathcal{G}_2, \quad e(P + P', Q) = e(P, Q) \cdot e(P', Q),$$

$$\forall P \in \mathcal{G}_1, \forall Q, Q' \in \mathcal{G}_2, \quad e(P, Q + Q') = e(P, Q) \cdot e(P, Q'),$$

2. e est non-dégénérée, i.e.

$$\forall P \in \mathcal{G}_1 \setminus \{0_{\mathcal{G}_1}\}, \exists Q \in \mathcal{G}_2, \quad e(P, Q) \neq 1_{\mathcal{G}_T},$$

$$\forall Q \in \mathcal{G}_2 \setminus \{0_{\mathcal{G}_2}\}, \exists P \in \mathcal{G}_1, \quad e(P, Q) \neq 1_{\mathcal{G}_T}.$$

Exemple 619. Le produit scalaire usuel entre espaces vectoriels

$$\langle \cdot, \cdot \rangle : \mathbb{F}_q^n \times \mathbb{F}_q^n \rightarrow \mathbb{F}_q$$

est un couplage.

Pour r premier avec q , nous nous intéressons à la constructions de couplages entre le groupe de torsion $\mathcal{E}[r]$ d'une courbe elliptique \mathcal{E} définie sur un corps fini \mathbb{F}_q et le groupe des racines $r^{\text{ième}}$ de l'unité μ_r (sur la clôture algébrique de \mathbb{F}_q) :

$$\mu_r = \{x \in \overline{\mathbb{F}_q}; x^r = 1\}.$$

Nous notons, une fois pour toute, t l'ordre de q dans $\mathbb{Z}/r\mathbb{Z}$, i.e. le plus petit entier tel que $r|q^t - 1$. Il est bien connu que $\mathbb{F}_q[\mu_r] = \mathbb{F}_{q^t}$ et dans ce qui suit, nous pourrons nous arranger pour définir tous les objets sur \mathbb{F}_{q^t} . Le morphisme de groupe

$$\begin{cases} \mathbb{F}_{q^t}^\times & \rightarrow & \mu_r \\ x & \mapsto & x^{(q^t-1)/r} \end{cases}$$

est surjectif, de noyau $(\mathbb{F}_{q^t}^\times)^n$, ce qui permet d'identifier

$$\mu_r \simeq \mathbb{F}_{q^t}^\times / (\mathbb{F}_{q^t}^\times)^r \quad \text{où} \quad (\mathbb{F}_{q^t}^\times)^r = \{x^r; x \in \mathbb{F}_{q^t}^\times\}.$$

Couplage de Tate

Soient $S, T \in \mathcal{E}[r](\mathbb{F}_{q^t})$ deux points de r -torsion de la courbe elliptique \mathcal{E} définis sur \mathbb{F}_{q^t} . Rappelons (voir exemple 569) qu'il existe une fonction rationnelle $f_S \in \mathcal{E}(\mathbb{F}_{q^t})$, appelée fonction de Miller, définie à une constante près par

$$\text{div}(f_S) = r \cdot (S) - r \cdot (0_{\mathcal{E}}).$$

Par ailleurs, soit $\hat{D}_T \in \text{div}(\mathcal{E})$ un diviseur équivalent à

$$D_T = (T) - (0_{\mathcal{E}}) \pmod{\text{Princ}(\mathcal{E})}.$$

Typiquement, on parvient à trouver un diviseur équivalent à D_T de la forme

$$\hat{D}_T = (T + Q) - (Q)$$

pour un certain point Q .

Définition 620. On appelle *couplage de Tate* la quantité

$$\langle S, T \rangle_r = f_S(\hat{D}_T) \pmod{(\mathbb{F}_{q^t}^\times)^r}.$$

Démonstration. Assurons nous de l'indépendance de la définition du choix de \hat{D}_T . Si $\hat{D}'_T = \hat{D}_T + \text{div}(h)$ avec $h \in \overline{\mathbb{F}_q}(\mathcal{E})$, alors, par la loi de réciprocité de Weil (théorème 617)

$$f(\hat{D}'_T) = f(\hat{D}_T) \cdot f(\text{div}(h)) = f(\hat{D}_T) \cdot h(\text{div}(f_T)).$$

Mais

$$h(\text{div}(f_T)) = (h(S)/h(0_{\mathcal{E}}))^r = 1 \pmod{(\mathbb{F}_{q^t}^\times)^r}.$$

Dans ce qui suit, nous écrirons $f_S(D_T)$ par abus de notation. \square

Théorème 621. *Le couplage de Tate est bilinéaire et non dégénéré.*

Démonstration. 1. Soient $S_1, S_2, T \in \mathcal{E}[r]$ et $S_3 = S_1 + S_2$. Par le lemme 570, il existe une fonction $h \in \mathcal{E}(\mathbb{F}_{q^t})$ telle que $f_{S_3} = f_{S_1}f_{S_2}h^n$. Donc

$$\langle S_1 + S_2, T \rangle = f_{S_3}(D_T) = f_{S_1}f_{S_2}h^n(D_T) = \langle S_1, T \rangle \langle S_2, T \rangle \pmod{(\mathbb{F}_{q^t}^\times)^r}.$$

Soient $S, T_1, T_2 \in \mathcal{E}[r]$ et $T_3 = T_1 + T_2$. Alors $D_{T_3} = D_{T_1} + D_{T_2} \pmod{\text{Princ}}$ et

$$\langle S, T_1 + T_2 \rangle = f_S(D_{T_1} + D_{T_2}) = f_S(D_{T_1})f_S(D_{T_2}) = \langle S, T_1 \rangle \langle S, T_2 \rangle.$$

2. Il existe plusieurs preuves pour justifier la non-dégénérescence du couplage de Tate mais elles requièrent toutes des résultats difficiles. Une première s'appuie sur des méthodes cohomologiques hors de portée de cette classe. Une seconde preuve, plus simple, s'appuie sur le théorème de densité de Tchebotariou et la construction explicite d'un point T tel que $f_S(D_T)$ engendre μ_r .

□

Application 622 (Diffie-Hellman triparti). Les couplages sur les courbes elliptiques ont aussi été utilisés pour proposer un mécanisme à la Diffie-Hellman triparti à une seule passe (Joux, 2000). Supposons que Alice, Bob et Charlie souhaitent s'entendre sur un secret commun.

Paramètres Alice, Bob et Charlie disposent d'une courbe elliptique \mathcal{E} et d'un couple de points (P, Q) de la courbe.

Clés Alice fixe un entier secret a , Bob un entier b et Charlie c .

Échange Alice publie $P_A = aP$ et $Q_A = aQ$. De même, Bob et Charlie publient chacun $P_B = bP$, $Q_B = bQ$ et $P_C = cP$, $Q_C = cQ$.

Secret commun On note F_T l'application $(x, D, D') \mapsto \langle D, D' \rangle^x$ (où x est un entier, D et D' sont des diviseurs et \langle, \rangle le couplage de Tate). Alice calcule $F_T(a, (P_B) - (Q_B), (P_C + Q_C) - (0_{\mathcal{E}}))$. Bob et Charlie calculent $F_T(b, (P_A) - (Q_A), (P_C + Q_C) - (0_{\mathcal{E}}))$ et $F_T(c, (P_A) - (Q_A), (P_C + Q_C) - (0_{\mathcal{E}}))$ respectivement. Ces trois quantités sont égales à

$$F_T(1, (P) - (Q), (P + Q) - (0_{\mathcal{E}}))^{abc}$$

et peuvent servir de secret commun.

Exercice 623. On pose $p = 4801$. On considère la courbe elliptique sur \mathbb{F}_p d'équation

$$y^2 = x^3 + 1 \in \mathbb{F}_p[x, y].$$

On fixe deux points $P = (2746, 392)$ et $Q = (2524, 1070)$.

1. Calculer l'ordre r de la courbe et vérifier que \mathbb{F}_p contient le groupe μ_r des racines r -ièmes de l'unité.
2. Alice, Bob et Charlie ont décidé d'établir un secret commun par le protocole de Diffie-Hellman triparti. Alice tire $a = 721$, Bob $b = 324$ et Charlie $c = 2501$.
 - (a) Calculer $F_T(a, (P_B) - (Q_B), (P_C + Q_C) - (0_{\mathcal{E}}))$ (avec la méthode `tate_pairing`).

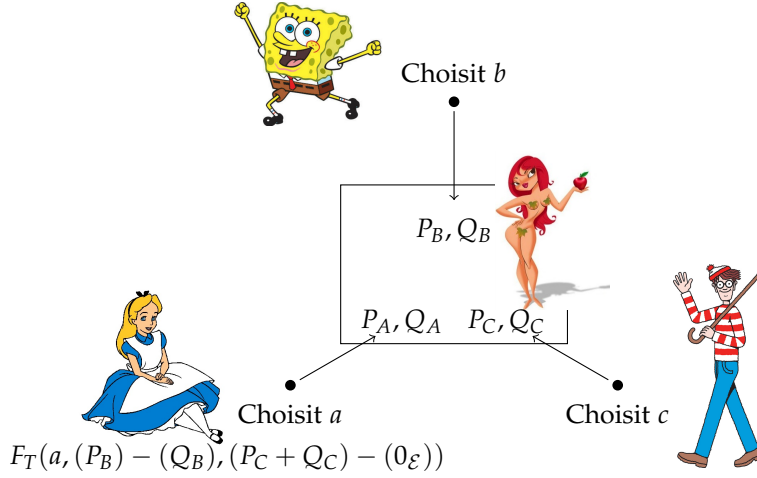


FIGURE 46: Diffie-Hellman triparti

- (b) Calculer de même $F_T(b, (P_A) - (Q_A), (P_C + Q_C) - (0_{\mathcal{E}}))$ et $F_T(c, (P_A) - (Q_A), (P_C + Q_C) - (0_{\mathcal{E}}))$.
- (c) Comparer les résultats avec $F_T(1, (P) - (Q), (P + Q) - (0_{\mathcal{E}}))^{abc}$.

Couplage de Weil

Soient $S, T \in \mathcal{E}[r]$ deux points de r -torsion de la courbe. Il existe, d'après le critère de principalité d'un diviseur (théorème 568), une fonction rationnelle $g_T \in \mathcal{E}(\mathbb{F}_q)$ définie à une constante près par

$$\operatorname{div}(g_T) = [r]^*(D_T) = \sum_{\substack{T'' \text{ tq} \\ rT''=T}} (T'') - \sum_{R \in \mathcal{E}[r]} (R)$$

En effet, le degré de $[r]^*(D_T)$ est clairement nul et le corollaire 565 montre que

$$\sum_{\substack{T'' \text{ tq} \\ rT''=T}} T'' - \sum_{R \in \mathcal{E}[r]} R = \sum_{R \in \mathcal{E}[r]} T_0'' + R - R = [r^2]T_0'' = [r]T = 0_{\mathcal{E}}.$$

Définition 624. On appelle *couplage de Weil* l'application

$$e_r(S, T) : \begin{cases} \mathcal{E}[r] \times \mathcal{E}[r] & \rightarrow \mu_r \\ (S, T) & \mapsto g_T(\hat{D}_S) \end{cases}$$

où \hat{D}_S est le diviseur $(S + P) - (P)$ et $P \in \mathcal{E}(\overline{\mathbb{F}_q})$ est un point quelconque n'annulant par g_T .

Justification. Retrouvons la fonction rationnelle f_T définie dans l'exemple 569 par

$$\operatorname{div}(f_T) = r \cdot D_T = r[(T) - (0_{\mathcal{E}})]$$

et que nous préciserons à constante près plus tard. D'après le point 1 du lemme 616

$$\operatorname{div}(f_T \circ [r]) = \operatorname{div}([r]^* f_T) = [r]^* \operatorname{div}(f_T)$$

Mais

$$[r]^* \operatorname{div}(f_T) = [r]^*(r \cdot D_T) = r \cdot [r]^* D_T = r \cdot \operatorname{div}(g_T) = \operatorname{div}(g_T^r).$$

Donc $f_T \circ [r]$ et g_T^r sont égales à un scalaire près. Quite à renormaliser f_T , on peut donc supposer que

$$f_T \circ [r] = g_T^r. \quad (67)$$

Soit $P \in \mathcal{E}(\overline{\mathbb{F}_q})$ un point tel que g_T ne s'annule pas, alors

$$(g_T(S+P))^r = f_T([r](S+P)) = f_T([r]P) = g_T(P)^r$$

Donc $e_r(S, T) \in \mu_r$. De plus, l'application $P \mapsto \frac{g_T(S+P)}{g_T(P)}$ est un morphisme $\mathcal{E} \rightarrow \mathbb{P}^1$, non-surjective, donc forcément constante (voir 7.3.1 de³³), ce qui montre que le choix de P importe peu pour calculer $e_r(S, T)$. \square

33. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

Proposition 625. *Le couplage de Weil e_r est*

1. bilinéaire,
2. antisymétrique, i.e.

$$\forall (S, T) \in \mathcal{E}[n], \quad e_r(T, T) = 1 \text{ et } e_r(S, T) = e_r(T, S)^{-1},$$

3. non-dégénéré.

Démonstration. 1. Soient S_1, S_2 et $T \in \mathcal{E}[r]$, alors

$$e_r(S_1 + S_2, T) = \frac{g_T(S_1 + S_2 + P)}{g_T(S_1 + P)} \cdot \frac{g_T(S_1 + P)}{g_T(P)} = e_r(S_2, T) e_r(S_1, T).$$

Soient S, T_1 et $T_2 \in \mathcal{E}[r]$. On note $T_3 = T_1 + T_2$.

En utilisant l'équation 67, on en déduit que $f_3 = c' \cdot f_1 \cdot f_2 \cdot (h \circ [r])$, pour une certaine constante c' . Mais alors

$$\begin{aligned} e_r(S, T_1 + T_2) &= \frac{f_{T_3}(S+P)}{f_{T_3}(P)} \\ &= \frac{f_{T_1}(S+P)}{f_{T_1}(P)} \cdot \frac{f_{T_2}(S+P)}{f_{T_2}(P)} \cdot \underbrace{\frac{(h \circ [r])(S+P)}{(h \circ [r])(P)}}_{=h(rS+rP)/h(rP)=1} \\ &= e_r(S, T_1) \cdot e_r(S, T_2) \end{aligned}$$

2. Reprenons le point T_0'' tel que $[r]T_0'' = T$. Comme

$$e_r(S + T, S + T) = e_r(S, S)e_r(S, T)e_r(T, S)e_r(T, T),$$

il est suffisant de montrer que $e_r(T, T) = 1$ pour tout $T \in \mathcal{E}[r]$, c'est-à-dire que pour un certain point P et un point $P' = P + T_0''$

$$\begin{aligned} \frac{g_T(P + T)}{g_T(P)} &= \frac{g_T(P + [r]T_0'')}{g_T(P)} \\ &= \frac{g_T(P + T_0'')g_T(P + [2]T_0'') \cdots g_T(P + [r]T_0'')}{g_T(P)g_T(P + T) \cdots g_T(P + [r-1]T)} \\ &= \frac{g_T(P')g_T(P' + [1]T_0'') \cdots g_T(P' + [r-1]T_0'')}{g_T(P)g_T(P + T) \cdots g_T(P + [r-1]T)} = 1. \end{aligned}$$

Nous allons montrer que l'application $h : P \mapsto g_T(P)g_T(P + T) \cdot g_T(P + [r-1]T)$ est constante, et pour cela vérifier simplement que son diviseur est nul. Or

$$\begin{aligned} \operatorname{div}(h) &= \sum_{j=0}^{r-1} \sum_{R \in \mathcal{E}[r]} (T_0'' + R - jT_0'') - (T_0'' + R - jT_0'') \\ &= \sum_{R \in \mathcal{E}[r]} (R + T_0'') - (R + (1-r)T_0'') \\ &= \sum_{T'', rT''=T} (T'') - \sum_{T'', rT''=T} (T'') \end{aligned}$$

car $r(1-r)T_0'' = T$. Donc $\operatorname{div}(h) = 0$ comme voulu.

3. Supposons que $e_r(S, T) = 1$ pour tout $S \in \mathcal{E}[r]$ et T fixé. Ceci signifie que la fonction rationnelle g_T est stable sous l'action du groupe des translations par des points de $\mathcal{E}[n]$. D'après la proposition 566, il existe une fonction rationnelle h_T telle que $g_T = h_T \circ [r]$. Mais alors

$$h_T^n \circ [r] = (h_T \circ [r])^r = g_T^r = f_T \circ [r]$$

Comme $[r]$ est surjective, il vient que $h_T^r = f_T$ et $r \operatorname{div}(h_T) = f_T$. Donc $\operatorname{div}(h_T) = (T) - (0_{\mathcal{E}})$. Or $\operatorname{somme}(\operatorname{div}(h_T))$ doit être nulle, donc $T = 0_E$ comme attendu. \square

Proposition 626. Les couplages de Weil et de Tate sont reliés par la formule

$$e_r(S, T) = \frac{\langle S, T \rangle_r}{\langle T, S \rangle_r} \pmod{(\mathbb{F}_q(\mathcal{E}[r]))^r}.$$

Démonstration. Soit toujours T_0'' tel que $rT_0'' = T$ et $f_{r, T_0''}$ la fonction de Miller définie par le diviseur

$$\operatorname{div}(f_{r, T_0''}) = rD_{T_0''} - D_T = r \cdot (T_0'') - (T) + (r-1) \cdot (0_{\mathcal{E}})$$

Alors, en utilisant le point 4 du lemme 616

$$\langle S, T \rangle_r = f_S(D_T) = f_S([r]_* D_{T_0''}) = [r]^* f_S(D_{T_0''})$$

Mais, par l'équation 67,

$$[r]^* f_S(D_{T_0''}) = (g_S(D_{T_0''}))^r = g_S(rD_{T_0''})$$

Or

$$g_S(rD_{T_0''}) = g_S(D_T \operatorname{div}(f_{r,T_0''})) = e_r(S, T) \cdot g_S(\operatorname{div}(f_{r,T_0''}))$$

Par la loi de réciprocité de Weil,

$$g_S(\operatorname{div}(f_{r,T_0''})) = f_{r,T_0''}(\operatorname{div}(g_S)) = f_{r,T_0''}([r]^* D_S)$$

Par le point 3 du lemme 616,

$$f_{r,T_0''}([r]^* D_S) = [r]_* f_{r,T_0''}(D_S)$$

Mais $[r]_* f_{r,T_0''}$ et f_T sont égaux à une constante près, car leur diviseur est

$$\operatorname{div}([r]_* f_{r,T_0''}) = [r]_* \operatorname{div}(f_{r,T_0''}) = rD_T = \operatorname{div}(f_T).$$

Donc

$$g_S(\operatorname{div}(f_{r,T_0''})) = f_T(D_S)$$

et le résultat suit comme attendu. \square

En pratique, pour calculer un couplage de Weil, il vaut mieux utiliser cette dernière formule car $\operatorname{div}(g_T)$ était défini sinon par la contribution de n^2 points de torsion.

Application 627 (Schéma cryptographique fondé sur l'identité, Boneh–Franklin). Les schémas fondés sur l'identité sont des cryptosystèmes dans lesquels la clé publique utilisée pour chiffrer un message est simplement dérivée de l'identité du destinataire. Ils permettent de certifier la clé publique, c'est-à-dire de s'assurer que la clé publique est bien celle du destinataire à qui on souhaite envoyer un message.

Dans le schéma suivant, on présume qu'il existe un tiers de confiance, Tom, capable de faire des calculs et de distribuer de l'information. Sa présence permet de distribuer une clé privée qui correspond à une clé publique et compense le caractère non-aléatoire de la clé publique.

Paramètres Tom publie un point de r -torsion P d'une courbe elliptique $\mathcal{E}(\mathbb{F}_q)$ et une fonction de hachage H à valeur dans $\mathcal{E}(\mathbb{F}_q)$ qui ramène une identité à un point de $\mathcal{E}(\mathbb{F}_q)$. On note $P_A = H(\text{Alice})$ l'identité de Alice, calculable par tous.

Clés-maitre Tom fixe un entier secret s et publie la clé-maitre publique $P_T = sP \in \mathcal{E}(\mathbb{F}_q)$.

Clé privée d'Alice Alice reçoit de Tom sa clé privée $Q_A = sP_A$.

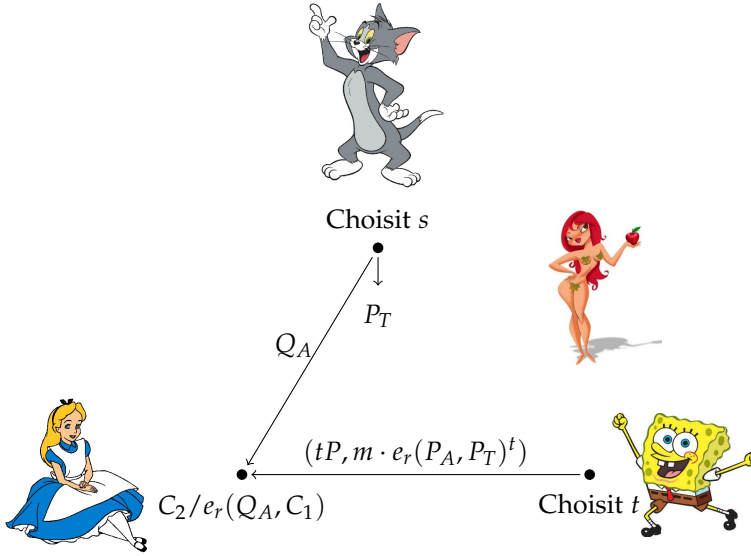
Chiffrement Pour envoyer un message $m \in \mathbb{F}_q^\times$ à Alice, Bob choisit un entier t aléatoire. Il envoie le chiffré

$$(C_1, C_2) = (tP, m \cdot e_r(P_A, P_T)^t).$$

Chiffrement Alice élimine le masque jetable de Bob avec

$$m = C_2 / e_r(Q_A, C_1)$$

FIGURE 47: Protocole de Boneh-Franklin



Exercice 628. On considère $p = 4561$, la courbe elliptique $\mathcal{E}(\mathbb{F}_p)$ d'équation

$$y^2 = x^3 + 2x \in \mathbb{F}_p[x, y]$$

munie du point de 30-torsion $P = (723, 3909)$. On suppose que l'identité d'Alice est $P_A = (1753, 2203)$ qui est aussi un point de 30-torsion. On suppose que la clé privée de Tom est $s = 7$.

1. Quelle est la clé maitre publique P_T de Tom et clé privée Q_A que Tom communique à Alice ?

2. Pour envoyer le message $m = 2000$, Bob tire au sort $r = 20$. Quel est le chiffré ? (utiliser `weil_pairing` de SageMath)
3. Vérifier que les calculs d'Alice permettent bien de retrouver le clair de Bob.

Évaluation d'une fonction de Miller

On note dans ce qui suit h_{P_1, P_2} la fonction $h_{P_1, P_2} = \frac{\ell_{P_1, P_2}}{\ell_{P_1 + P_2, -P_1 - P_2}}$, dont le diviseur est

$$\operatorname{div}(h_{P_1, P_2}) = (P_1) + (P_2) - (P_1 + P_2) - (0_{\mathcal{E}}).$$

(On rappelle que ℓ_{P_1, P_2} désigne une équation de la droite $(P_1 P_2)$).

Exemple 629. On considère $p = 61$, la courbe \mathcal{E} sur \mathbb{F}_{61} d'équation

$$y^2 = x^3 + 11x \in \mathbb{F}_{61}[x, y],$$

et le point $R = (24, 34)$. La tangente passant par R a pour équation $13x + y + 20 = 0$; la droite passant par $2R = (-1, 7)$ et $-2R$ a pour équation $x + 1 = 0$. Donc $h_{R, R}$ peut prendre la forme $(x, y) \mapsto \frac{13x + y + 20}{x + 1}$.

Introduisons des fonctions $f_{m, S}$ (unique à une constante près) telles que³⁴

$$\operatorname{div}(f_{m, S}) = m(S) - (m \cdot S) - (m - 1)(0_{\mathcal{E}}).$$

Quand S est de r -torsion, on retrouve au lieu de $f_{r, S}$ la fonction de Miller f_S vue plus haut pour définir le couplage de Tate. Nous commençons par observer, en vérifiant les diviseurs associés, que

Proposition 630. Pour tous entiers m_1, m_2 et tout point S de r -torsion, les fonctions de Miller vérifient

$$f_{m_1 + m_2, S} = f_{m_1, S} \cdot f_{m_2, S} \cdot h_{m_1 S, m_2 S}$$

$$f_{m_1 m_2, S} = (f_{m_1, S})^{m_2} \cdot f_{m_2, m_1 S}$$

Et en particulier

$$f_{m+1, S} = f_{m, S} \cdot h_{mS, S}$$

$$f_{2m, S} = f_{m, S}^2 \cdot h_{mS, mS}$$

On en déduit l'algorithme suivant, qui permet d'évaluer une fonction de Miller en un point de r -torsion, en utilisant la décomposition binaire de r .

34. le support du diviseur a trois points : S, mS et $0_{\mathcal{E}}$, il est de degré 0 car $m - 1 - (m - 1)$ s'annule.

Algorithme 42 : Algorithme de Miller

Entrées : $r = \overline{r_l r_{l-1} \cdots r_0}_{(2)}$,
 $S, P \in \mathcal{E}[r]$

Sorties : $f_S(P)$

```

1  $R \leftarrow S, f \leftarrow 1$ 
2 pour tous les  $i$  de  $l-1$  à  $0$  faire
3    $f \leftarrow f^2 \cdot h_{R,R}(P)$ 
   // Duplication
4    $R \leftarrow 2R$ 
5   si  $r_i = 1$  alors // Addition
6      $f \leftarrow f \cdot h_{R,S}(P)$ 
7      $R \leftarrow R + S$ 
8 retourner  $f$ 

```

Exemple 631. On considère $p = 61$ et la courbe \mathcal{E} sur \mathbb{F}_{61} d'équation

$$y^2 = x^3 + 11x \in \mathbb{F}_{61}[x, y].$$

Le groupe $\mathcal{E}(\mathbb{F}_{61})$ est isomorphe à $\mathbb{Z}/10\mathbb{Z} \oplus \mathbb{Z}/5\mathbb{Z}$, engendré par exemple par les points $S = (24, 34)$ d'ordre 10 et $T = (5, 27)$ d'ordre 5. Ces deux points sont de r -torsion avec $r = 10$. On souhaite calculer leur couplage de Weil par la formule (proposition 626)

$$e_r(S, T) = \frac{f_S(T + M)}{f_S(M)} \frac{f_T(M')}{f_T(S + M')} \pmod{(\mathbb{F}_p^\times)^r}.$$

où M et M' sont des points choisis aléatoirement de sorte que les calculs ne butent pas sur une division par zéro. Nous notons d'ores et déjà, comme r divise $p - 1$, que le couplage est défini sur \mathbb{F}_{61} et non sur une de ses extensions.

Nous calculons avec $M = M' = (50, 45)$. Nous notons $U = T + M = (11, 7)$. La décomposition binaire de $r = 10$ est $\overline{1010}_{(2)}$. Les calculs de $f_S(T + M)$ et $f_S(M)$ par l'algorithme de Miller conduisent aux étapes suivantes

		Cas $P = T + M$	Cas $P = M$
Initialisation	$R \leftarrow (24, 34)$	$f \leftarrow 1$	$f \leftarrow 1$
Etape $i = 2$	$h_{R,R} : (x, y) \mapsto \frac{13x+y+20}{x+1}$ $R \leftarrow (60, 7)$	$f \leftarrow \frac{57}{34}f^2 = 25$	$f \leftarrow \frac{44}{51}f^2 = 20$
Etape $i = 1$	$h_{R,R} : (x, y) \mapsto \frac{60x+y+53}{x+58}$ $R \leftarrow (3, 50)$ $h_{R,S} : (x, y) \mapsto \frac{24x+y}{x}$ $R \leftarrow (0, 0)$	$f \leftarrow \frac{55}{30}f^2 = 58$ $f \leftarrow \frac{34}{33}f = 8$	$f \leftarrow \frac{48}{47}f^2 = 49$ $f \leftarrow \frac{25}{50}f = 55$
Etape $i = 0$	$h_{R,R} : (x, y) \mapsto \frac{x}{25x+y+60}$ $R \leftarrow 0_{\mathcal{E}}$	$f \leftarrow \frac{33}{5}f^2 = 32$	$f \leftarrow \frac{50}{13}f^2 = 54$

Donc

$$f_S(T + M) = 32 \quad \text{et} \quad f_S(M) = 54.$$

De même,

$$f_T(S + M) = 12 \quad \text{et} \quad f_T(M) = 60.$$

On en déduit que

$$e_r(S, T) = 12 \pmod{(\mathbb{F}_p^\times)^r}.$$

Comme les puissances 10-ièmes sont $(\mathbb{F}_p^\times)^r = \{1, 13, 14, 47, 48, 60\}$, d'autres choix de M et de normalisations auraient conduit aux valeurs $\{12, 15, 27, 34, 46, 49\}$. On retrouve une racine r -ième de l'unité en calculant la puissance 6-ième. On obtient alors dans les 6 cas la valeur 34, qui est bien une racine 10-ième de l'unité.

- Exercice 632** (Couplages). 1. Implémenter une fonction qui calcule $\ell_{P_1, P_2}(S)$ pour trois points P_1, P_2 et S , où ℓ_{P_1, P_2} est l'équation d'une droite passant par P_1 et P_2 .
2. Implémenter une fonction qui calcule $h_{P_1, P_2}(S)$ pour trois points P_1, P_2 et S , où h_{P_1, P_2} est le quotient $\frac{\ell_{P_1, P_2}}{\ell_{P_1 + P_2, -P_1 - P_2}}$.
3. Implémenter une fonction qui calcule le couplage de Tate (définition 620). Vous pourrez utiliser un bloc `try - except` pour tenir compte du fait que l'algorithme de Miller peut butter sur un pôle.
4. Implémenter une fonction qui calcule le couplage de Weil (proposition 626).
5. Comparez avec les méthodes *ad hoc* de SageMath (notamment la méthode `weil_pairing`).

L'attaque M.O.V.

L'attaque de Menezes, Okamoto et Vanstone (M.O.V., 1993) consiste à transposer le problème du logarithme discret sur une courbe elliptique au même problème sur un corps fini via un couplage. Comme on dispose d'algorithmes sous-exponentiels pour ce problème dans des corps finis (par exemple, par du calcul d'indice), il se trouve que lorsque le corps fini d'arrivée en question n'est pas trop grand, l'attaque est pertinente. Initialement, l'attaque M.O.V. utilise le couplage de Weil. Frey et Rück l'ont étendue au couplage de Tate (1994).

Problème 633 (Logarithme discret). Étant donnés deux points $P \in \mathcal{E}(\mathbb{F}_q)$ et un multiple Q de P , trouver un entier λ tel que $[\lambda]P = Q$.

Algorithme 43 : Attaque M.O.V. /
Frey-Rück

Entrées : point $P \in \mathcal{E}(\mathbb{F}_q)$ d'ordre
premier r , Q multiple de
 P

Sorties : entier λ tel que $Q = [\lambda]P$

- 1 Trouver t tels que $r|q^t - 1$
 - 2 Construire \mathbb{F}_{q^t}
 - 3 Trouver $S \in \mathcal{E}(\mathbb{F}_{q^t})$ tel que
 $e(P, S) \neq 1$
 - 4 $\zeta_1 \leftarrow e_r(P, S)$ (couplage de Weil)
 - 5 $\zeta_1 \leftarrow e_r(Q, S)$
 - 6 Trouver λ tel que $\zeta_2 = \zeta_1^\lambda$ dans \mathbb{F}_{q^t}
 - 7 **retourner** λ
-

Remarque 634. On dit qu'une courbe \mathcal{E} définie sur \mathbb{F}_q est *supersingulière* quand la caractéristique p divise $\#(\mathcal{E}(\mathbb{F}_q)) - 1$ (i.e. quand p divise $q + 1 - \#(\mathcal{E}(\mathbb{F}_q))$, qui est aussi la trace du Frobenius). Menezes, Okamoto et Vanstone ont montré que le degré de plongement t est inférieur à 6 lorsque la courbe est supersingulière, ce qui rend ces courbes particulièrement sensibles à cette attaque.

Exercice 635 (Attaque M.O.V.). On travaille avec

$$p = 2199023255579 \quad \mathcal{E} : y^2 = x^3 + x$$

et les points

$$P = (1435967701832 : 123951463462 : 1),$$

$$Q = (1129476910351 : 1383670460733 : 1).$$

1. Vérifier que p est premier.
2. Calculer le j -invariant de \mathcal{E} . Sur un corps fini, on sait qu'une courbe de j -invariant 1728 est supersingulière si et seulement si $p = 2$ ou $p \equiv 3 \pmod{4}$. Que peut-on dire ?
3. Quel est le cardinal de \mathcal{E} ? Quelle est sa factorisation ?
4. Quel est l'ordre r de P ?
5. Quel est le plus petit entier t tel que $r|p^t - 1$. Que représente cet entier ?
6. Quel est le cardinal de $\mathcal{E}(\mathbb{F}_{p^t})$?
7. Trouver un point S de $\mathcal{E}(\mathbb{F}_{p^t})$ d'ordre r tel que $e_r(P, S) \neq 1$ (avec `weil_pairing`).
8. Calculer les valeurs ζ_1 et ζ_2 comme indiquées dans l'algorithme. En déduire λ (vous pouvez utiliser la méthode `log` de SageMath).
9. Que se passe-t-il si on cherche S dans $\mathcal{E}(\mathbb{F}_p)$?

Exercices de révisions II

Tous les exercices peuvent être résolus à la main (sans SageMath).

Exercice 636. On pose $a = 8$ et $n = 45$.

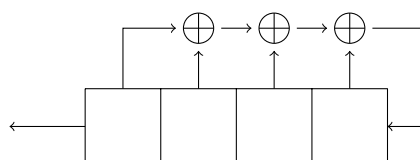
1. Calculer $(a^{11} \bmod n)$, $(a^{22} \bmod n)$ et $(a^{44} \bmod n)$. Que peut-on déduire sur n ?
2. Calculer le symbole de Jacobi $\left(\frac{a}{n}\right)$ et comparer avec $(a^{22} \bmod n)$. Commenter.

Exercice 637. Soit M le \mathbb{Z} -module

$$M = \mathbb{Z}/14\mathbb{Z} \oplus \mathbb{Z}/35\mathbb{Z} \oplus \mathbb{Z}/60\mathbb{Z} \oplus \mathbb{Z}/28\mathbb{Z}.$$

Quelles sont les composantes primaires de M ? Donner le type de chacune.

Exercice 638. On donne le LFSR suivant sur \mathbb{F}_2 .



1. Donner le polynôme de rétroaction $\chi(x)$ et de connexion $c(x)$ du LFSR.
2. Le polynôme χ est-il irréductible sur $\mathbb{F}_2[x]$?
3. Quelle est la période du polynôme de rétroaction χ ?
4. Quelles sont les périodes des suites engendrées par ce LFSR.
5. Ce LFSR est-il maximal ?

Exercice 639. On donne $p = 67$ et $b = 32 \in \mathbb{F}_p$.

1. On a remarqué que

$$b^{18} = 14, \quad b^{24} = 15, \quad b^{30} = 40, \quad b^{45} = 40, \quad b^{62} = 49.$$

En déduire la valeur de $\log_b q$ pour $q \in \{2, 3, 5, 7\}$ par la résolution d'un système linéaire approprié.

2. Alice et Bob ont décidé d'utiliser le protocole de Diffie-Hellman pour s'accorder sur un secret commun. Ils sont convenus publiquement de travailler dans \mathbb{F}_p^\times avec b comme générateur. Ève observe que Alice envoie à Bob le message 42 et Bob envoie à Alice le message 35. Quel est leur secret commun ?

Troisième partie

**Corrigés de certains
exercices**

Solution 2. On peut obtenir la relation de Bézout et une solution du système comme suit.

```
a = 119
b = 435
d,u,v = xgcd(a,b)

alpha = 2
beta = 3
m = 45
n = 14
d,u,v = xgcd(m,n)
x = mod(alpha*v*n + beta*u*m,m*n)
```

Solution 3. À titre préliminaire, on fixe

```
p = 157
k = 4
q = p^k
Fp = FiniteField(p)
Fq.<alpha> = FiniteField(q)
z = 130*alpha^3 + 97*alpha^2 + 99*alpha + 18
```

1. Le Frobenius est l'application $z \mapsto z^p$. Il se trouve que c'est aussi une application F_p linéaire. On peut écrire, pour en définir la matrice :

```
Q = matrix(Fp,k,k)
Q[:,0] = vector(alpha^0)
Q[:,1] = vector(alpha^p)
Q[:,2] = vector(alpha^(2*p))
Q[:,3] = vector(alpha^(3*p))
```

Pour calculer l'image sur une centaine d'exemples, on peut faire

```
t = vector([1,alpha,alpha^2,alpha^3])

sigmaz = t*Q*vector(z)

test1 = True
for _ in range(10):
    x0 = Fp.random_element()
    x1 = Fp.random_element()
    x2 = Fp.random_element()
    x3 = Fp.random_element()
    v = vector([x0,x1,x2,x3])
    x = t*v
    test1 &= (vector([1,alpha,alpha^2,alpha^3])*Q*v == x^p)
```

2. La première manière de calculer τ consiste à inverser le Frobenius en tant qu'application linéaire

```
def tau1(x):
    """
    Première façon de calculer la racine p-ième.
    """
    x=Fq(x)
    return t*Q^(-1)*vector(x)
```

Par ailleurs, on peut noter que comme $x^q = x$ pour tout x , on a aussi $(x^{q/p})^p = x$, ce qui fournit une expression de τ . Une autre façon de remarquer la même chose est de se souvenir que les itérés du σ forment un groupe cyclique d'ordre k et que σ^{-1} correspond ainsi à $\sigma \circ \sigma \cdots \circ \sigma$ ($k-1$ fois).

```
def tau2(x):
    """
    Seconde façon de calculer la racine p-ième.
    """
    x=Fq(x)
    return x^(q/p)

tau1z = tau1(z)
tau2z = tau2(z)

test2 = True
for _ in range(100):
    x = Fp.random_element()
    test2 &= ((tau1(x))^p == x)
    test2 &= ((tau2(x))^p == x)
```

Solution 4. Choisir des petites valeurs des paramètres pour que la liste reste de taille raisonnable

```

q=2
n=4
Fq.<omega> = FiniteField(q)
myPols.<x> = PolynomialRing(Fq)
ListIrred = [x^n+p for p in
myPols.polynomials(max_degree=n-1)
if (x^n+p).is_irreducible()]
ListPrim = [p for p in ListIrred
if p.is_primitive()]
len(ListPrim)*n-euler_phi(q^n-1)
prod(ListPrim) - myPols(
cyclotomic_polynomial(q^n-1))
p1 = ListIrred[0]
p2 = ListIrred[1]
Fq1.<alpha1> = FiniteField(q^n, modulus=p1)
Fq2.<alpha2> = FiniteField(q^n, modulus=p2)
racines = p1.roots(Fq2, multiplicities=false)
phi = Fq1.hom(racines[0],Fq2)
phi(alpha1^2)

```

Solution 5. Instructions avec SageMath

```

A = matrix(QQ, [[-2,1,1], [8,1,-5], [4,3,-3]])
C = matrix(QQ, [[1,2,-1], [2,-1,-1], [-5,0,3]])
for v in W.basis():
    A*v
X0 = C.solve_left(A)
A-X0*C
C.left_kernel().basis_matrix()

```

Les solutions de l'équation forment un espace affine de dimension 3

$$\begin{pmatrix} 0 & -1 & 0 \\ 2 & 3 & 0 \\ 2 & 1 & 0 \end{pmatrix} + \begin{pmatrix} \lambda & 2\lambda & \lambda \\ \mu & 2\mu & \mu \\ \nu & 2\nu & \nu \end{pmatrix}$$

Solution 7. On constate que les coefficients diagonaux deviennent rationnels et que leur taille explose.

Solution 9. On peut considérer le code suivant

```

def myGaussBareiss(MM):
    M=copy(MM)
    n=M.ncols()
    c=1
    s=1
    for k in range(n-1):
        if M[k,k]==0:
            (b,i) = exists( range(k+1,n),
                lambda i : M[i,k]!=0 )
            if b :
                M.swap_rows(k,i)
                s=-s
            else:
                return 0
        for i in range(k+1,n):
            for j in range(k+1,n):
                M[i,j]=(M[k,k]*M[i,j]
                    -M[i,k]*M[k,j])/c
        c=M[k,k]
    return s*M[n-1,n-1]

```

Pour les tests :

```

for n in range(2,20):
    for t in range(1000):
        A = random_matrix(ZZ,4)
        if myGaussBareiss(A) !=A.determinant():
            print "Alerte"

```

Solution 11. Dans \mathbb{F}_p la taille des coefficients ne peut pas exploser.

Solution 16. On peut faire

```

def myFractionContinue(x, k):
    if k==0:
        return [floor(x)]
    else:
        y0=floor(x)
        z=1/(x-y0)
        return [y0] + myFractionContinue(z,k-1)

```

et vérifier que le nombre d'or a pour développement la suite de 1.

```
P = myFractionContinue((1+sqrt(5))/2,50)
```

Inversement, on peut coder

```
def myConvergent(L):
    if len(L)==1:
        return L[0]
    else:
        return L[0]+1/myConvergent(L[1:])
```

et retrouver ϕ

```
(myConvergent(P) - (1+sqrt(5))/2).N()
```

```
def approximation(x,k):
    return myConvergent(myFractionContinue(x,k))
```

Et enfin

```
k=10
ListePremiers = primes_first_n(k)
P = prod(ListePremiers)
N = prod(ListePremiers[:-4])
r = ZZ.random_element(N)
suite_r = [mod(r,p) for p in ListePremiers]
print "Suite initiale"
print suite_r
suite_a = copy(suite_r)
e = randint(0,k-1)
print "Erreur en position ",e, "(facteur p=", ListePremiers[e],")"
suite_a[e]+=1
suite_a = [ZZ(n) for n in suite_a]
print "Nouvelle suite"
print suite_a
a = crt(suite_a, ListePremiers)
x = a/P
for k in range(1,10):
    f = approximation(x,k)
    if abs(x-f) <= 1/(2*(f.denominator())^2) and f.denominator() in ListePremiers:
        print f.denominator().factor()
```

Solution 19. 1. Notons $\alpha + i\beta$ et $b = \gamma + i\delta$. Soient s et t les réels tels que

$$\frac{a}{b} = \frac{\alpha + i\beta}{\gamma + i\delta} = s + it$$

On pose $\sigma = \lfloor s \rfloor$, $\tau = \lfloor t \rfloor$ et $q = \sigma + i\tau$. On a $r = a - qb$. Par construction $\mathcal{N}\left(\frac{a}{b} - q\right) \leq \frac{1}{2^2} + \frac{1}{2^2}$. Donc $\mathcal{N}(r) \leq \frac{1}{2}\mathcal{N}(b)$. On en déduit que $\mathbb{Z}[i]$ est un anneau euclidien. Il est donc aussi principal et factoriel.

2. On a $a^{-1} = \frac{1}{\mathcal{N}(a)}\bar{a}$, donc a est inversible ssi de norme 1, d'où $\mathbb{Z}[i]^\times = \{\pm 1, \pm i\}$.
3. Nous commençons par vérifier que l'énumération propose bien des nombres premiers de $\mathbb{Z}[i]$. Soit π un entier de Gauß de la forme $1 \pm i$ ou de la forme $\alpha \pm i\beta$ avec $\mathcal{N}(\pi) = p$. Supposons que π se factorise en $\pi = \alpha\beta$. Alors $\mathcal{N}(\pi) = \mathcal{N}(\alpha)\mathcal{N}(\beta)$. Mais $\mathcal{N}(\pi)$ est premier. Donc $\mathcal{N}(\alpha)$ ou $\mathcal{N}(\beta)$ vaut 1, autrement dit α ou β est une unité. Soit π de la forme $\pi = p$ où p est premier dans \mathbb{Z} et $p \equiv 3 \pmod{4}$. Supposons que π se factorise en $\pi = \alpha\beta$ où ni α ni β ne sont des unités. Alors $\mathcal{N}(\pi) = \mathcal{N}(\alpha)\mathcal{N}(\beta) = p^2$ et $p = \mathcal{N}(\alpha)$. Mais alors p est une somme de deux carrés, ce qui n'est pas possible (-1 n'est pas un carré modulo p dans ce cas).

Il reste à voir qu'il n'y a pas d'autre facteur premier possible. Soit π un entier de Gauß premier. On a

$$\pi\bar{\pi} = \mathcal{N}(\pi) = p_1 \cdots p_r$$

d'après la factorisation en facteurs premiers sur \mathbb{Z} . Mais comme π est premier, π doit diviser l'un des facteurs $p = p_i$. Alors $\mathcal{N}(\pi)$ doit être un diviseur de p^2 . Dans le cas où $\mathcal{N}(\pi) = p$, on retrouve les deux premiers cas. Sinon, si $\mathcal{N}(\pi)$ vaut p^2 , mais alors $\mathcal{N}(p/\pi)$ vaut 1 et p et π sont égaux à un inversible près. De plus, les factorisations précédente montrent que $p \equiv 3[4]$ pour que p ne soit pas factorisable.

4. On peut considérer l'ensemble des éléments compris dans le carré $0, a, (1+i)a$ et ia (où on exclut deux bords). Il a autant d'éléments que l'aire du carré, soit $\mathcal{N}(a)$ éléments.

Solution 24. 1. non : car $1 = (n/n) \cdot 1 = (1/n) \cdot (n \cdot 1) = 0$

2. non : car $1 = (f/f) \cdot 1 = (1/f) \cdot (f \cdot 1) = 0$

Solution 25. ANALYSE : Soit $u \in \mathbb{F}_p$ tel que $\sqrt{-1} \cdot 1 = u$. Alors, à cause des lois sur un A -module, on a pour tout $\alpha, \beta \in \mathbb{Z}$ et pour tout $y \in \mathbb{F}_p$, on a

$$(\alpha + \beta\sqrt{-1}) \cdot y = (\alpha + \beta u)y$$

En particulier $\sqrt{-1} \cdot u = u^2$. Mais $\sqrt{-1} \cdot u = \sqrt{-1} \cdot \sqrt{-1} \cdot 1 = -1$. Donc -1 doit être un carré dans \mathbb{F}_p , autrement dit $\left(\frac{-1}{p}\right) = 1$. D'après les lois complémentaires de réciprocité quadratique, $p \equiv 1 \pmod{4}$ ou $p = 2$.

SYNTHÈSE : Si $p \equiv 1 \pmod{4}$ ou $p = 2$, il existe $u \in \mathbb{F}_p$ tel que $u^2 = -1$. On pose tout $\alpha, \beta \in \mathbb{Z}$ et pour tout $y \in \mathbb{F}_p$,

$$(\alpha + \beta\sqrt{-1}) \cdot y = (\alpha + \beta u)y$$

On vérifie point par point la définition 20 pour conclure.

Solution 27. Soit M un $\mathbb{Z}[i]$ -module. Alors M est par définition un groupe abélien. Définissons l'application $\phi : M \rightarrow M$ par $x \mapsto \sqrt{-1} \cdot x$. Les propriétés définissant un A -module montrent que ϕ est un endomorphisme de groupe. De plus $\phi^2(x) = \phi \circ \phi(x) = \sqrt{-1}\sqrt{-1}x = -x = -\text{Id}(x)$. Donc M est aussi un groupe abélien muni d'un endomorphisme de groupe ϕ tel que $\phi^2 = -\text{Id}$.

Réciproquement, soit M un groupe abélien muni d'un endomorphisme de groupe ϕ tel que $\phi^2 = -\text{Id}$. Montrons que M est un $\mathbb{Z}[\sqrt{-1}]$ -module. Nous définissons pour $a + \sqrt{-1}b \in \mathbb{Z}[\sqrt{-1}]$ et $x \in M$ la loi de composition externe par

$$(a + \sqrt{-1}b) \cdot x = \underbrace{x + x + \cdots + x}_a \text{ fois} + \underbrace{\phi(x) + \phi(x) + \cdots + \phi(x)}_b \text{ fois}$$

Nous vérifions que les propriétés de la définition 20 sont satisfaites [à faire], ce qui permet de conclure.

Solution 37. Soit $y = (y_1, \dots, y_n) \in \bigoplus_{i=1}^r (M_i/N_i)$. Alors, pour tout $i \leq r$, il existe un représentant $x_i \in M_i$ tel que $\bar{x}_i = x_i + N_i = y_i$. Posons alors

$$z = \overline{x_1 + \cdots + x_r} = x_1 + \cdots + x_r + \left(\bigoplus_{i=1}^r N_i \right).$$

Nous remarquons que z ne dépend pas de $(x_i)_{i \leq r}$. En effet, supposons qu'il existe d'autres représentants x'_i tels que $\bar{x}_i = y_i$, alors $x_i - x'_i \in N_i$ pour tout i et donc

$$(x_1 + \cdots + x_r) - (x'_1 + \cdots + x'_r) \in \left(\bigoplus_{i=1}^r N_i \right).$$

Appelons ψ l'application $y \mapsto z$. Nous remarquons que ψ est linéaire (écrire les équations). De plus, comme la somme des M_i est directe, $\psi(y) = 0$ signifie que $x_1 + \cdots + x_r \in \left(\bigoplus_{i=1}^r N_i \right)$, soit encore $x_i \in N_i$ pour tout i . Donc $y = (0, \dots, 0)$. On en déduit que ψ est injectif. Pour montrer que ψ est surjectif, on considère $z \in \left(\bigoplus_{i=1}^r M_i \right) / \left(\bigoplus_{i=1}^r N_i \right)$ et x un représentant de z . Mais x se décompose en $x_1 + \cdots + x_r$ et $x_i \in M_i$. Et alors $\psi(\bar{x}_1, \dots, \bar{x}_r) = z$.

Solution 42. Soient v_1, \dots, v_g des générateurs de M et soient f_1, \dots, f_{g+1} une famille de vecteurs appartenant à l'espace engendré par v_1, \dots, v_g . Nous raisonnons par récurrence.

Si $g = 1$, cela signifie que $f_1 = \lambda_1 v_1$ et $f_2 = \lambda_2 v_1$ avec λ_1, λ_2 non nuls (si l'un des vecteurs est nul il n'y a rien à montrer), donc $\lambda_1 f_2 = \lambda_2 f_1$, ce qui montre que la famille est liée.

Si $g > 1$, on suppose que les f_i sont libres et que $f_i = \sum_j \alpha_{i,j} v_j$. On considère les vecteurs $f'_i = \alpha_{g+1,g} f_i - \alpha_{i,g} f_{g+1}$ qui appartiennent à l'espace engendré par v_1, \dots, v_{g-1} et sont libres aussi, ce qui est impossible par hypothèse de récurrence.

Solution 45. Tout nombre $x \in \mathbb{Q}$ se décompose en produit d'un signe et d'un produit fini de facteurs premiers avec des puissances relatives.

$$x = (-1)^u p_1^{\alpha_1} \cdots p_k^{\alpha_k}$$

Le signe fournit le terme $\mathbb{Z}/2\mathbb{Z}$ de la somme tandis que l'on peut choisir $M = \mathbb{Q}^+$ comme terme restant, de base dénombrable $\mathcal{P} = \{p \in \mathbb{N}; p \text{ premier}\}$.

Solution 46. M est un \mathbb{K} -ev de dimension r mais ce n'est pas un module libre. C'est un module de type fini, engendré par 1. C'est encore un module de torsion, de rang 0.

Solution 47. Soit M un A module de type fini. On commence par remarquer que si on se restreint à $\mathbb{C} \subseteq A$, on a simplement un \mathbb{C} -espace vectoriel de type fini, autrement dit un \mathbb{C} -espace vectoriel de dimension finie. De plus, la multiplication dans M par ϵ est une opération \mathbb{C} linéaire de M , donc un endomorphisme de E . Soit u cet endomorphisme. On remarque que comme $\epsilon^2 = 0$, $u^2 = 0$. Donc u est la somme directe (voir théorème 406) de l'endomorphisme nul et de blocs dont la matrice est

$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

Réciproquement, si $M_0 = \mathbb{C}$ est défini comme le A -module muni de la loi

$$\forall (\alpha, \beta) \in \mathbb{C}, \forall x \in M_0, \quad (\alpha + \beta\epsilon) \cdot x = \alpha x$$

les conditions de la définition 20 sont satisfaites. D'autre part, si $M_1 = \mathbb{C}^2$ est défini comme le A -module muni de la loi

$$\forall (\alpha, \beta) \in \mathbb{C}, \forall x \in M_0, \quad (\alpha + \beta\epsilon) \cdot x = \alpha x + \beta n(x)$$

avec n l'endomorphisme nilpotent

$$n = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

les conditions de la définition 20 sont satisfaites.

Donc tout A -module de type fini est une somme directe de copies de M_0 et M_1 .

Solution 55. 1. Comme pour un pivot de Gauß, on a tout intérêt à suivre librement l'algorithme pour l'accélérer plutôt que de s'en tenir à la version bête et méchante que peut dérouler une machine.

$$\mathbf{A} = \begin{pmatrix} -2 & 3 & 3 & 1 \\ 2 & -1 & 1 & -3 \\ -4 & 0 & -1 & -4 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

ÉTAPE 1 : le pivot est le coefficient en position (3,4) :

On commence par ramener 1 en position de pivot par $C_3 \leftrightarrow C_4$, puis $C_4 \leftarrow -C_4$.

$$\begin{pmatrix} -2 & 3 & 1 & -3 \\ 2 & -1 & -3 & -1 \\ -4 & 0 & -4 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

À présent on nettoie la dernière ligne par $C_1 \leftarrow C_1 + 4C_4$ et $C_3 \leftarrow C_3 + 4C_4$.

$$\begin{pmatrix} -14 & 3 & -11 & -3 \\ -2 & -1 & -7 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

ÉTAPE 2 : le pivot est le coefficient en position (2,3) :

Puis $C_2 \leftrightarrow C_3$ et $C_3 \leftarrow -C_3$.

$$\begin{pmatrix} -14 & -11 & -3 & -3 \\ -2 & -7 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

On réduit le début de la ligne 2 par $C_1 \leftarrow C_1 + 2C_3$ et $C_2 \leftarrow C_2 + 7C_3$

$$\begin{pmatrix} -20 & -32 & -3 & -3 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

On peut maintenant réduire la fin de la ligne 2 par $C_4 \leftarrow C_4 + C_3$

$$\begin{pmatrix} -20 & -32 & -3 & -6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

ÉTAPE 3 : le pivot est le coefficient en position (1,2) :

On ramène le coefficient le plus petit en position de pivot $C_1 \leftrightarrow C_2$, puis $C_2 \leftarrow -C_2$.

$$\begin{pmatrix} -32 & 20 & -3 & -6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

On réduit

$$\begin{pmatrix} 8 & 20 & -3 & -6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

On ramène le coefficient le plus petit en position de pivot $C_1 \leftrightarrow C_2$

$$\begin{pmatrix} 20 & 8 & -3 & -6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

On réduit

$$\begin{pmatrix} 4 & 8 & -3 & -6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

On ramène le coefficient le plus petit en position de pivot $C_1 \leftrightarrow C_2$

$$\begin{pmatrix} 8 & 4 & -3 & -6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

On réduit

$$\begin{pmatrix} 0 & 4 & -3 & -6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{Z}^{3 \times 4}.$$

Enfin, le début de la ligne ne compte que des 0. On réduit la fin de la ligne par $C_3 \leftarrow C_3 + C_2$ et $C_4 \leftarrow C_4 + 2C_2$. On obtient

$$\mathbf{H} = \begin{pmatrix} 0 & 4 & 1 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

2. Il suffit d'ajouter une instruction $U \leftarrow \mathbf{I}_n$ en début d'algorithme et de reproduire sur U l'ensemble des transformations faites à X .

3.

4. On obtient

$$\mathbf{H} = \begin{pmatrix} 0 & 4 & 1 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

et

$$\mathbf{U} = \begin{pmatrix} 8 & 3 & 3 & 6 \\ 19 & 8 & 7 & 15 \\ -12 & -4 & -4 & -9 \\ -5 & -2 & -2 & -4 \end{pmatrix}$$

Solution 56. On remplace les réductions par des divisions euclidiennes et le test $x_{i,k} < 0$ par une mise à 1 du coefficient dominant de $x_{i,k}$.

```

def myHermiteNormalFormForPolynomialRings(A):
    m = A.nrows()
    n = A.ncols()
    l = max(0, m-n)
    i=m-1
    k=n-1
    U = identity_matrix(Pol,n)
    while i>=l:
        if any(A[i,t]!=0 for t in range(k+1)):
            while any(A[i,t]!=0 for t in range(k)):
                j0 = [A[i,t]!=0 for t in range(k)].index(True)
                A.swap_columns(j0,k)
                U.swap_columns(j0,k)
                if A[i,k].leading_coefficient()!=1:
                    U.set_col_to_multiple_of_col(k,k,1/A[i,k].leading_coefficient())
                    A.set_col_to_multiple_of_col(k,k,1/A[i,k].leading_coefficient())
                for j in range(k):
                    u,_ = A[i,j].quo_rem(A[i,k])
                    A.add_multiple_of_column(j,k,-u)
                    U.add_multiple_of_column(j,k,-u)
                for j in range(k+1,n):
                    u,_ = A[i,j].quo_rem(A[i,k])
                    A.add_multiple_of_column(j,k,-u)
                    U.add_multiple_of_column(j,k,-u)
            i=i-1
            k=k-1
        else:
            i=i-1
    return A, U

```

Solution 64. Comme on fait les calculs à la main, on a tout intérêt à essayer d'accélérer l'algorithme en remontant en position de pivot un petit coefficient (petit au sens de la divisibilité).

On part de

$$\begin{pmatrix} 40 & 70 & 20 \\ 20 & 50 & 60 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Avec $C_2 \leftarrow C_2 - 2C_1$,

$$\begin{pmatrix} 40 & -10 & 20 \\ 20 & 10 & 60 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & -2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Avec $C_2 \leftarrow -C_2$,

$$\begin{pmatrix} 40 & 10 & 20 \\ 20 & -10 & 60 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Avec $C_2 \leftrightarrow C_1$,

$$\begin{pmatrix} 10 & 40 & 20 \\ -10 & 20 & 60 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 2 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Avec $L_2 \leftarrow L_2 + L_1$,

$$\begin{pmatrix} 10 & 40 & 20 \\ 0 & 60 & 80 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 2 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Avec $C_2 \leftarrow C_2 - 4C_1$ et $C_3 \leftarrow C_3 - 2C_1$

$$\begin{pmatrix} 10 & 0 & 0 \\ 0 & 60 & 80 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 2 & -7 & -4 \\ -1 & 4 & 2 \\ 0 & 0 & 1 \end{pmatrix}$$

Avec $C_3 \leftarrow C_3 - C_2$

$$\begin{pmatrix} 10 & 0 & 0 \\ 0 & 60 & 20 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 2 & -7 & 3 \\ -1 & 4 & -2 \\ 0 & 0 & 1 \end{pmatrix}$$

Avec $C_3 \leftrightarrow C_2$

$$\begin{pmatrix} 10 & 0 & 0 \\ 0 & 20 & 60 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 2 & 3 & -7 \\ -1 & -2 & 4 \\ 0 & 1 & 0 \end{pmatrix}$$

Avec $C_3 \leftarrow C_3 - 3C_2$

$$\begin{pmatrix} 10 & 0 & 0 \\ 0 & 20 & 0 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 2 & 3 & -16 \\ -1 & -2 & 10 \\ 0 & 1 & -3 \end{pmatrix}$$

Solution 75. On choisit les vecteurs $e_1 = (1, 0)$ et $e_2 = (1, 1)$. Ils forment une base adaptée car $\mathbb{Z}^2 = \langle e_1, e_2 \rangle$ et $N = \langle e_1, 2e_2 \rangle$ (en prenant $d_1 = 1$ et $d_2 = 2$). De plus d_1 divise bien d_2 .

Solution 76. 1. On peut montrer qu'une base adaptée de M à N est

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \end{pmatrix}, \quad \dots, \quad \mathbf{e}_n = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ -1 \end{pmatrix} \quad \mathbf{e}_{n+1} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

car

$$M = \mathbb{Z}\mathbf{e}_1 \oplus \mathbb{Z}\mathbf{e}_1 \oplus \cdots \oplus \mathbb{Z}\mathbf{e}_{n+1}$$

et

$$N = \mathbb{Z}\mathbf{e}_1 \oplus \mathbb{Z}\mathbf{e}_1 \oplus \cdots \oplus \mathbb{Z}\mathbf{e}_n.$$

2. On peut montrer qu'une base adaptée de M à N est

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} -1 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad \dots, \quad \mathbf{e}_n = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -1 \\ 1 \end{pmatrix}$$

car

$$M = \mathbb{Z}\mathbf{e}_1 \oplus \mathbb{Z}\mathbf{e}_2 \oplus \cdots \oplus \mathbb{Z}\mathbf{e}_n$$

et

$$N = 2\mathbb{Z}\mathbf{e}_1 \oplus \mathbb{Z}\mathbf{e}_2 \oplus \cdots \oplus \mathbb{Z}\mathbf{e}_n.$$

Solution 80. On commence par rechercher une décomposition de Smith de A . On remarque que tous les coefficients de A sont divisibles par 5 (critère : le dernier chiffre est 0 ou 5). On travaillera donc avec la matrice

$$A' = \frac{1}{5}A = \begin{pmatrix} -126 & 147 & 0 & 147 & -126 \\ 255 & -297 & -3 & -294 & 255 \\ 126 & -126 & 0 & -126 & 126 \end{pmatrix}$$

On remarque que les coefficients de A' sont tous divisibles par 3 (critère : la somme des chiffres est divisible par 3). On travaillera donc avec

$$A'' = \frac{1}{15}A = \begin{pmatrix} -42 & 49 & 0 & 49 & -42 \\ 85 & -99 & -1 & -98 & 85 \\ 42 & -42 & 0 & -42 & 42 \end{pmatrix}$$

Pour compléter la correction, nous donnons L et C : notez cependant que nous n'en aurons pas besoin ! On remarque encore qu'il se trouve un 1 dans la matrice. Pour nous faciliter le travail, nous l'amenons en position de pivot par $C_1 \leftrightarrow -C_3$ et $L_1 \leftrightarrow L_2$.

$$\begin{pmatrix} 1 & -99 & 85 & -98 & 85 \\ 0 & 49 & -42 & 49 & -42 \\ 0 & -42 & 42 & -42 & 42 \end{pmatrix} \quad L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad C = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Dernière réduction a priori : $C_5 \leftarrow C_5 - C_3$.

$$\begin{pmatrix} 1 & -99 & 85 & -98 & 0 \\ 0 & 49 & -42 & 49 & 0 \\ 0 & -42 & 42 & -42 & 0 \end{pmatrix} L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad C = \begin{pmatrix} 0 & 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

On réduit la première ligne par $C_2 \leftarrow C_2 + 99C_1$, $C_3 \leftarrow C_3 - 85C_1$,
 $C_4 \leftarrow C_4 + 98C_1$.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 49 & -42 & 49 & 0 \\ 0 & -42 & 42 & -42 & 0 \end{pmatrix} L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad C = \begin{pmatrix} 0 & 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & -99 & 85 & -98 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

À ce stade, on peut remarquer que tous les coefficients sont des multiples de 7

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 7 \cdot 7 & -7 \cdot 6 & 7 \cdot 7 & 0 \\ 0 & -7 \cdot 6 & 7 \cdot 6 & -7 \cdot 6 & 0 \end{pmatrix} L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad C = \begin{pmatrix} 0 & 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & -99 & 85 & -98 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

et que 7 est premier avec 6. Nous voulons donc ramener le pgcd 7 en position de pivot, ce que nous pouvons faire par $C_2 \leftarrow C_2 + C_3$

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 7 & -42 & 49 & 0 \\ 0 & 0 & 42 & -42 & 0 \end{pmatrix} L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad C = \begin{pmatrix} 0 & 1 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & -14 & 85 & -98 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

On peut maintenant nettoyer la ligne 2 : $C_3 \leftarrow C_3 + 6C_2$, $C_4 \leftarrow C_4 - 7C_2$.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 7 & 0 & 0 & 0 \\ 0 & 0 & 42 & -42 & 0 \end{pmatrix} L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad C = \begin{pmatrix} 0 & 1 & 7 & -7 & -1 \\ 0 & 1 & 6 & -7 & 0 \\ -1 & -14 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Dernière étape : $C_4 \leftarrow C_4 + C_3$.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 7 & 0 & 0 & 0 \\ 0 & 0 & 42 & 0 & 0 \end{pmatrix} L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad C = \begin{pmatrix} 0 & 1 & 7 & 0 & -1 \\ 0 & 1 & 6 & -1 & 0 \\ -1 & -14 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Nous en déduisons que la forme de Smith de A est

$$\Delta = \begin{pmatrix} 15 & 0 & 0 & 0 & 0 \\ 0 & 105 & 0 & 0 & 0 \\ 0 & 0 & 630 & 0 & 0 \end{pmatrix}$$

On en déduit immédiatement que

$$M = \mathbb{Z}^3 / N \simeq \mathbb{Z}/15\mathbb{Z} \oplus \mathbb{Z}/105\mathbb{Z} \oplus \mathbb{Z}/630\mathbb{Z}.$$

On a les factorisations $15 = 3 \cdot 5$, $105 = 3 \cdot 5 \cdot 7$ et $630 = 2 \cdot 3^2 \cdot 5 \cdot 7$. Les composantes primaires de M sont donc (en utilisant le lemme chinois pour séparer les facteurs premiers entre eux)

- $\mathbb{Z}/2\mathbb{Z}$ pour la composante 2 primaire,
- $(\mathbb{Z}/3\mathbb{Z})^2 \oplus \mathbb{Z}/3^2\mathbb{Z}$ pour la composante 3-primaire,
- $(\mathbb{Z}/5\mathbb{Z})^3$ pour la composante 5-primaire
- $(\mathbb{Z}/7\mathbb{Z})^2$ pour la composante 7-primaire.

Comme $L^{-1} = L$, nous remarquons de plus que les vecteurs (e_2, e_1, e_3) forment une base adaptée de N par rapport à \mathbb{Z}^3 et que une base simple de N est $(15e_2, 105e_1, 630e_3)$, ce qui permet de mieux comprendre comment on a calculé la structure.

Solution 81. Notons N l'image de ϕ . Soit $LAC = D$ la décomposition de Smith de la matrice A , avec $L, C \in GL_n(\mathbb{Z})$ et

$$D = \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_n \end{pmatrix}$$

où $a_i \neq 0$ pour tout $1 \leq i \leq r$, $a_i = 0$ pour tout $r < i \leq n$ et $a_i | a_{i+1}$ pour $1 \leq i \leq r-1$. Nous remarquons que le déterminant est simplement $|\det A| = |\det D| = \prod_{i=1}^n |a_i|$.

La structure de $M = \mathbb{Z}^n / N$ est par ailleurs

$$\mathbb{Z}^n / N \simeq \bigoplus_{1 \leq i \leq n} \mathbb{Z} / a_i \mathbb{Z}.$$

Donc M est fini si et seulement si aucun des a_i n'est nul, ie si et seulement si $\det A \neq 0$ et le cardinal de M est

$$|M| = \prod_{i=1}^n |a_i|$$

qui est précisément $|\det A|$.

Solution 82. Aaa

Solution 83. On cherche la forme normale de Smith de A . La solution de facilité revient à passer par SageMath :

```
PQQ.<x>=PolynomialRing(QQ)
p=x^2+1
Qi.<i>=NumberField(p,'i')
ZZi=Qi.ring_of_integers()
A=matrix(ZZi,[[1,2+3*i,2-3*i],[3,-3*i,6+9*i],[6,12-18*i,-18*i]])
A.smith_form()
```

Noter qu'il n'est pas nécessaire de calculer les matrices de changement de base L et C (nous le faisons uniquement pour suivre l'algorithme en entier).

On commence par ramener 1 en position de pivot : $L_1 \leftrightarrow L_3$.

$$\begin{pmatrix} 1 & i & -1 \\ 2i+2 & 6i & 6i-8 \\ -2i+3 & 3i+2 & 2i-1 \end{pmatrix} \quad L = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \quad C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

On effectue $C_2 \leftarrow C_2 - iC_1$, $C_3 \leftarrow C_3 + C_1$. On obtient

$$\begin{pmatrix} 1 & 0 & 0 \\ 2i+2 & 4i+2 & 8i-6 \\ -2i+3 & 0 & 2 \end{pmatrix} \quad L = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \quad C = \begin{pmatrix} 1 & -i & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

On arrange encore $L_2 \leftarrow L_2 - (2i+2)L_1$ et $L_3 \leftarrow L_3 - (-2i+3)L_1$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 4i+2 & 8i-6 \\ 0 & 0 & 2 \end{pmatrix} \quad L = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & -2i-2 \\ 1 & 0 & 2i-3 \end{pmatrix} \quad C = \begin{pmatrix} 1 & -i & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

On remarque que 2 divise tous les coefficients. On le porte en position de pivot par $C_2 \leftrightarrow C_3$ et $L_2 \leftrightarrow L_3$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 8i-6 & 4i+2 \end{pmatrix} \quad L = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 2i-3 \\ 0 & 1 & -2i-2 \end{pmatrix} \quad C = \begin{pmatrix} 1 & 1 & -i \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Enfin, on retranche $L_3 \leftarrow L_3 - (4i-3)L_2$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4i+2 \end{pmatrix} \quad L = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 2i-3 \\ -4i+3 & 1 & 16i-3 \end{pmatrix} \quad C = \begin{pmatrix} 1 & 1 & -i \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

La structure du quotient est

$$\mathcal{G}^3/N \simeq \mathcal{G}/2\mathcal{G} \oplus \mathcal{G}/(4i+2)\mathcal{G}.$$

En utilisant l'exercice 19, il y a $\mathcal{N}(2)\mathcal{N}(4i+2) = (2^2 + 0^2)(4^2 + 2^2) = 80$ éléments dans le quotient.

Solution 86. Même preuve que pour les sous-groupes d'un groupe cyclique.

- Solution 89.** 1. $\text{Hom}_{\mathbb{Z}}(\mathbb{Z}^2, \mathbb{Z}) \simeq \mathbb{Z}^2$ (il suffit de choisir une image parmi \mathbb{Z} à $(1, 0)$ et $(0, 1)$).
2. $\text{Hom}_{\mathbb{Z}}(\mathbb{Z}^2, \mathbb{Z}^2) \simeq \mathbb{Z}^4$ (il suffit de choisir une image à $(1, 0)$ et $(0, 1)$ dans la même base
3. $\text{Hom}_{\mathbb{Z}}(\mathbb{Z}, \mathbb{Z}/n\mathbb{Z})$ (il suffit de choisir une image parmi $\mathbb{Z}/n\mathbb{Z}$ à 1)
4. $\text{Hom}_{\mathbb{Z}}(\mathbb{Z}/n\mathbb{Z}, \mathbb{Z}/m\mathbb{Z})$ Soit $\phi \in \text{Hom}_{\mathbb{Z}}(\mathbb{Z}/n\mathbb{Z}, \mathbb{Z}/m\mathbb{Z})$ et $a = \phi(1)$. Alors $0 = \phi(n \times 1) = n\phi(1) = na$. Soit $d = m \wedge n$ et n' tel que $n = dn'$. En particulier $n' \wedge m = 1$. Donc n' est inversible dans $\mathbb{Z}/m\mathbb{Z}$. Donc $na = 0$ équivaut à $da = 0$ dans $\mathbb{Z}/m\mathbb{Z}$, autrement dit a est un élément de $\mathbb{Z}/m\mathbb{Z}$ d'ordre d , i.e. $a \in m'\mathbb{Z}/n\mathbb{Z}$ avec $m' = m/d$.

Réciproquement, si $a \in m'\mathbb{Z}/m\mathbb{Z}$ et on pose $\phi(1) = a$, alors la valeur de $\phi(t) = ta$ ne dépend pas du représentant de t et ϕ est un morphisme bien défini.

Donc $\text{Hom}_{\mathbb{Z}}(\mathbb{Z}/n\mathbb{Z}, \mathbb{Z}/m\mathbb{Z}) \simeq (\mathbb{Z}/(m \wedge n)\mathbb{Z})$.

Solution 90. Soit $N = \langle 2i \rangle_A$.

1. Si $A = \mathbb{Z}$, une base adaptée à (M, N) est $(i, 1)$, car $M = \langle i, 1 \rangle_{\mathbb{Z}}$ et $N = \langle 2i \rangle_{\mathbb{Z}}$, donc $M/N \simeq \mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$.
2. Si $A = \mathbb{Z}[i]$, on a encore $N = \langle 2 \rangle_{\mathbb{Z}[i]}$ puisque i est une unité dans $\mathbb{Z}[i]$. Donc $M/N \simeq \mathbb{Z}[i]/2\mathbb{Z}[i] = \{\bar{0}, \bar{1}, \bar{i}, \bar{1} + \bar{i}\}$.

Solution 96. Attention piège : U est bien un groupe abélien, mais la loi $+$ est le produit ! C'est un module de torsion car toute racine r de l'unité vérifie $r^n = 1$ pour un certain n .

Solution 97. 1. Clair

- 2.(a) $\text{Ann}(M) = 2^4 \cdot 5^1 \cdot 7^1 \mathbb{Z} = 560\mathbb{Z}$ car $560 = \text{ppcm}(4, 7, 10, 16)$.
- (b) $\text{Ann}(M) = \pi(u)A = \prod_{i=1}^m (x - \lambda_i) \mathbb{K}[x]$ où $\pi(u)$ est le polynôme minimal et les λ_i sont les valeurs propres distinctes de u .
- (c) On a $x^2 + 2 = (x - 1)(x + 1)$ et $x^4 + 2x^3 + 2x^2 + 2x + 2 = (x - 1)(x^3 - x + 1)$. Mais $x^3 - x + 1$ ne possède pas de racine dans \mathbb{F}_3 , donc est irréductible dans $\mathbb{F}_3[x]$ et possède ses trois racines dans une extension de degré 3 de \mathbb{F}_3 (en particulier il se factorise dans tout surcorps de \mathbb{F}_{27} , i.e. dans tout $\mathbb{F}_{3^{3k}}$ avec $k \in \mathbb{N}^*$). Donc $\text{Ann}(M) = (x - 1)(x + 1)(x^3 - x + 1)\mathbb{F}_{81}[x]$.

Solution 102. On peut prendre l'idéal $I = \langle x, y \rangle$ de $\mathbb{K}[x, y]$ ou $I = \langle 2, x \rangle$ de $\mathbb{Z}[x]$. Comme ils ne sont pas principaux, ils ne sont pas libres.

Solution 103. On a $\text{rg}(M \oplus M') = \text{rg}(M) + \text{rg}(M')$ et $(M \oplus M')_{\text{tors}} = M_{\text{tors}} \oplus M'_{\text{tors}}$.

- Solution 104.** 1. Soit $x = (x_n)_{n \in \mathbb{N}} \in S$ une suite et $\lambda \in \mathbb{Z} \setminus \{0\}$ un scalaire tels que $\lambda \cdot x = 0$. Alors pour tout $n \in \mathbb{N}$, $\lambda x_n = 0$, donc $x_n = 0$. Ainsi x est la suite nulle. Donc S ne possède pas d'élément de torsion.
2. Pour tout n , l'ensemble $A_n = \{b \in B; \lambda_{n,b} \neq 0\}$ est fini. Or $A = \bigcup_{n \in \mathbb{N}} A_n$. Donc A est dénombrable en tant que réunion dénombrable d'ensembles finis.
3. Comme B est une base, $M + M' = S$. De plus la somme est directe, car M et M' car A et $B \setminus A$ sont disjoints.
4. Comme $\mathbb{N}^{\mathbb{N}}$ est infini non-dénombrable et que l'application $a \mapsto x_a$ est une bijection, il y a une infinité non-dénombrable de suite x_a .
5. Par différence des cardinaux, une telle suite $x_a \in S \setminus M$ existe.
6. Soit x_n la suite contenant les $n - 1$ premiers termes non-nuls de x et nulle sinon. De même, soit z_n la suite contenant les termes non nuls de x à partir du n -ième et nulle sinon. On a évidemment $x = x_n + z_n$. Comme x_n est à support fini, $x_n \in M$. Donc $p(x) = p(y_n)$. Mais tous les termes de z_n sont divisibles par 2^n , donc $y_n = z_n / 2^n$ est encore une suite de S . On a comme voulu $p(x) = 2^n p(y_n)$.
7. Soit x un élément divisible par 2^n pour tout n . Comme L possède une base B , on peut écrire

$$x = \sum_{b \in B} \lambda_b b.$$

$$x' = 2^n x = \sum_{b \in B} \lambda'_b b.$$

avec $\alpha_b, \alpha'_b \in \mathbb{Z}$. Mais comme $x = 2^n x'$, on a immédiatement $\lambda_b = 2^n \lambda'_b$. Donc les entiers λ_b sont divisibles par 2^n pour tout n . Mais alors $\lambda_b = 0$ pour tout b et $x = 0$.

8. S est un \mathbb{Z} -module sans torsion et non-libre.

Solution 113. On rappelle que la proposition 57 permet de trouver le noyau d'une matrice.

Trouver une base de $\mathcal{L}_1 = \{(x, y, z) \in \mathbb{Z}^3; 10x + 6y + 3z = 0\}$, on peut utiliser une décomposition d'Hermite de la matrice $A = (10, 6, 3)$. On obtient les matrices H, C telle que $AC = H$ comme suit :

$$(10 \quad 6 \quad 3) \text{ et } \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Le pivot recherché doit être le pgcd de 10, 6 et de 3. On peut remarquer opportunément que $C_1 \leftarrow C_1 - 3C_3$ fait apparaître un tel

1.

$$(1 \ 6 \ 3) \text{ et } \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -3 & 0 & 1 \end{pmatrix}$$

On place le 1 obtenu en position de pivot par $C_1 \leftrightarrow C_3$.

$$(3 \ 6 \ 1) \text{ et } \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & -3 \end{pmatrix}$$

Il reste à annuler le reste de la ligne par $C_2 \leftarrow C_2 - 6C_3$ et $C_1 \leftarrow C_1 - 3C_3$, ce qui donne

$$(0 \ 0 \ 1) \text{ et } \begin{pmatrix} -3 & -6 & 1 \\ 0 & 1 & 0 \\ 10 & 18 & -3 \end{pmatrix}$$

Nous en déduisons forme

$$\mathcal{L}_1 = \mathbb{Z} \begin{pmatrix} -3 \\ 0 \\ 10 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -6 \\ 1 \\ 18 \end{pmatrix}$$

Il s'agit bien d'un réseau dont la base est explicite. La matrice de Gram est

$$G = \begin{pmatrix} 109 & 198 \\ 198 & 361 \end{pmatrix}$$

Donc $\det \mathcal{L}_1 = 109 \cdot 361 - 198^2 = 145$ et $\text{disc}(\mathcal{L}_1) = \sqrt{145}$. (Remarque : sur cet exemple, on voit assez bien pourquoi il est plus intéressant d'utiliser le déterminant : le discriminant n'est pas rationnel alors que la base est entière).

2. Pour trouver les points de \mathcal{L}_2 , nous pouvons chercher le noyau de la matrice $(10 \ 6 \ 3 \ 11)$ Nous déduisons de la question précédente que par les mêmes opérations

$$(0 \ 0 \ 1 \ 11) \text{ et } \begin{pmatrix} -3 & -6 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 10 & 18 & -3 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Enfin, par $C_4 \leftarrow C_4 - 11C_3$, on arrive à

$$(0 \ 0 \ 1 \ 0) \text{ et } \begin{pmatrix} -3 & -6 & 1 & -11 \\ 0 & 1 & 0 & 0 \\ 10 & 18 & -3 & 33 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathcal{L}_1 = \mathbb{Z} \begin{pmatrix} -3 \\ 0 \\ 10 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -6 \\ 1 \\ 18 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -11 \\ 0 \\ 33 \end{pmatrix}$$

Nous pouvons directement calculer le discriminant

$$\text{disc } \mathcal{L}_2 = |-11| \cdot \left\| \begin{pmatrix} -3 & -6 & 1 \\ 0 & 1 & 0 \\ 10 & 18 & -3 \end{pmatrix} \right\| = 11$$

(car par construction, la matrice restante est dans $GL_3(\mathbb{Z})$, donc de déterminant de module 1). Par suite $\det \mathcal{L}_2 = 121$.

On pourrait aussi calculer directement le déterminant de la matrice de Gram (mais c'est plus long ici) :

3. En fait, on cherche à résoudre $3x = y + 4\lambda_1$ et $y = x - z + 7\lambda_2$ pour $\lambda_1, \lambda_2 \in \mathbb{Z}$, autrement dit, on cherche la projection sur les trois premières coordonnées du noyau de la matrice

$$A = \begin{pmatrix} 3 & -1 & 0 & -4 & 0 \\ 1 & -1 & 1 & 0 & 7 \end{pmatrix}$$

On calcule une forme normale de Hermite de A :

On commence par arranger le pivot de la seconde ligne par $C_1 \leftrightarrow C_5$

$$\begin{pmatrix} 0 & -1 & 0 & -4 & 3 \\ 7 & -1 & 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Il est alors possible de faire $C_1 \leftarrow C_1 - 7C_5$, $C_2 \leftarrow C_2 + C_5$ et $C_3 \leftarrow C_3 - C_5$

$$\begin{pmatrix} -21 & 2 & -3 & -4 & 3 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} -7 & 1 & -1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

On s'attaque à la ligne 1. On peut remarquer que $C_4 \leftarrow -(C_4 - C_3)$ fait apparaître un bon pivot

$$\begin{pmatrix} -21 & 2 & -3 & 1 & 3 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} -7 & 1 & -1 & -1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

On peut finir la ligne par $C_1 \leftarrow C_1 + 21C_4$, $C_2 \leftarrow C_2 - 2C_4$, $C_3 \leftarrow C_3 + 3C_4$ et $C_5 \leftarrow C_5 - 3C_4$.

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} -28 & 3 & -4 & -1 & 4 \\ 0 & 1 & 0 & 0 & 0 \\ 21 & -2 & 4 & 1 & -3 \\ -21 & 2 & -3 & -1 & 3 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

On en déduit que

$$\begin{aligned} \mathcal{L}_3 &= \mathbb{Z} \begin{pmatrix} 28 \\ 0 \\ 21 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -4 \\ 0 \\ 4 \end{pmatrix} \\ &= \mathbb{Z}7 \begin{pmatrix} 4 \\ 0 \\ 3 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z}4 \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} \end{aligned}$$

Le discriminant vaut $4 \cdot 7 \cdot |1 \cdot (4 \cdot 1 - 3 \cdot 1)| = 28$ et le déterminant 28^2

4. On a $x^3 - 2x = x(x^2 - 2)$. Mais quand $x \in \mathbb{Z}$, la seule racine possible est 0, donc L_4 est le réseau \mathbb{Z}^2 , de déterminant 1.

Solution 114. L'ensemble \mathcal{L} est un réseau, comme nous allons voir en exhibant une base. Notons d'abord que par multiplication par -1 , on peut transformer l'équation $4x + 4y + 53z \equiv 0 \pmod{5}$ en $x + y + 2z \equiv 0 \pmod{5}$. On résout le système suivant dans \mathbb{Z}^4

$$\begin{cases} 2x - 3y + 4z &= 0 \\ x + y + 2z - 5\lambda &= 0 \end{cases}$$

en utilisant la proposition 57.

On recherche une forme normale d'Hermite de la matrice

$$A = \begin{pmatrix} 2 & -3 & 4 & 0 \\ 1 & 1 & 2 & -5 \end{pmatrix}$$

On commence par chercher un bon pivot, en effectuant $C_1 \leftrightarrow C_4$

$$\begin{pmatrix} 0 & -3 & 4 & 2 \\ -5 & 1 & 2 & 1 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

À présent, on peut nettoyer la seconde ligne par $C_1 \leftarrow C_1 + 5C_4$, $C_2 \leftarrow C_2 - C_4$ et $C_3 \leftarrow C_3 - 2C_4$

$$\begin{pmatrix} 10 & -5 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 5 & -1 & -2 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

On continue avec la ligne 1 par $C_3 \leftrightarrow C_2$ et $C_3 \leftarrow -C_3$

$$\begin{pmatrix} 10 & 0 & 5 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 5 & -2 & 1 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

et finalement $C_1 \leftarrow C_1 - 2C_3$

$$\begin{pmatrix} 0 & 0 & 5 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 3 & -2 & 1 & 1 \\ 2 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Aussi

$$\mathcal{L} = \mathbb{Z} \begin{pmatrix} 3 \\ 2 \\ 0 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -2 \\ 2 \\ 1 \end{pmatrix}$$

ce qui confirme que \mathcal{L} est un réseau (car engendré sur \mathbb{Z} par deux vecteurs non colinéaires dans \mathbb{R}). Il est de rang 2 et de dimension 3.

Solution 115. D'après le théorème de la base adaptée, il existe une base (e_1, \dots, e_n) de \mathcal{L} et des entiers non nuls $a_1 | \dots | a_n$ tels que

$$(a_1 e_1, \dots, a_n e_n)$$

est une base de \mathcal{L}' . Mais alors

$$\det(a_1 e_1, \dots, a_n e_n) = a_1 \cdots a_n \cdot \det(e_1, \dots, e_n).$$

Donc

$$\det \mathcal{L}' = (a_1 \cdots a_n)^2 \cdot \det \mathcal{L}$$

ce qui montre la divisibilité voulue.

Solution 118. 1. On peut remarquer que déterminer \mathbb{A}_n revient à déterminer le noyau sur \mathbb{Z}^n

$$A = (1, \dots, 1)$$

Or, une décomposition d'Hermite donne $A = DC^{-1}$ avec

$$D = (0, \dots, 0, 1) \text{ et } \begin{pmatrix} -1 & \dots & -1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Une base est donc (d'après la proposition 57) $(e_0 - e_i)_{1 \leq i \leq n}$. Par ailleurs $\langle e_0 - e_i, e_0 - e_j \rangle = 2$ si $i = j$ et $\langle e_0 - e_i, e_0 - e_j \rangle = 1$ si $i \neq j$.

Donc

$$\det \mathbb{A}_n = \begin{vmatrix} 2 & 1 & \cdots & 1 \\ 1 & 2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 1 & \cdots & 1 & 2 \end{vmatrix} = \begin{vmatrix} 2 & 1 & \cdots & 1 \\ -1 & 1 & 0 & 0 \\ \vdots & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{vmatrix} = n+1$$

(en développant la dernière colonne et en raisonnant par récurrence). On peut aussi calculer le déterminant par diagonalisation de la matrice (il est facile de voir que 1 est valeur propre d'ordre $n-1$ et $n+1$ valeur propre d'ordre $n+1$) ou utiliser le fait que la matrice est circulante.

2. On peut remarquer que déterminer \mathbb{D}_n revient à résoudre l'équation sur \mathbb{Z}

$$x_1 + \cdots + x_n - 2y = 0$$

Or, une mise sous forme normale d'Hermite de

$$A = (1, \dots, 1, -2)$$

donne $A = DC^{-1}$ avec

$$D = (0, \dots, 0, 1) \text{ et } \begin{pmatrix} -1 & \cdots & -1 & 2 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 \\ & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

D'après la proposition 57, une base est donc $(2e_1) \cup (e_i - e_1)_{2 \leq i \leq n}$. Il est facile de voir que le discriminant, qui est la valeur absolue du déterminant de cette famille de vecteur est

$$\text{disc } \mathbb{D}_n = 2 \quad \text{et} \quad \det \mathbb{D}_n = 4$$

3. On remarque que

$$\begin{pmatrix} -1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} + 4 \begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} - 2 \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix} = 0$$

Donc une base de \mathbb{E}_8 est donnée par

$$\begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix}$$

Elle est très clairement de déterminant 1, donc $\text{disc } \mathbb{E}_8 = \det \mathbb{E}_8 = 1$.

4. Pour tout vecteur de la base de \mathbb{A}_n , \mathbb{D}_n ou \mathbb{E}_8 que nous avons calculée, $\langle x, x \rangle = 2$ et $\langle x, y \rangle \in \{-2, 1, 0\}$ ce qui suffit à prouver le résultat voulu.
5. Simple déduction de la construction.
6. Dans tous les cas, aucune coordonnée n'est privilégiée dans la construction de \mathbb{A}_n , \mathbb{D}_n ou \mathbb{E}_8 et toute permutation des coordonnées d'un vecteur reste dans le réseau considéré.

Si un vecteur de \mathbb{A}_n a pour norme carrée 2 et à permutation des coordonnées près, il s'agit forcément soit de $(\pm 1, \pm 1, 0, \dots, 0)$. Après contrôle, seul

$$\pm(1, -1, 0, \dots, 0)$$

convient.

Si un vecteur de \mathbb{A}_n a pour norme carrée 2 et à permutation des coordonnées près, il s'agit forcément soit de $(\pm 1, \pm 1, 0, \dots, 0)$. Après contrôle, tous ces vecteurs conviennent.

Si un vecteur de \mathbb{E}_8 est de norme 2, ses coordonnées appartiennent à $\{0, \pm 1\}$ ou à $\{\pm \frac{1}{2}, \pm \frac{3}{2}\}$. Il n'est pas possible d'utiliser $\frac{3}{2}$ avec $\frac{1}{2}$ pour faire une norme 2. Par contre il est possible d'avoir des vecteurs $(\pm 1/2, \dots, \pm 1/2)$. Comme $\sum x_i \equiv 0 \pmod{2}$, sont exclus les vecteurs où le nombre de signe est impair. Tous les autres cas sont possibles

$$\pm \frac{1}{2}(1, 1, 1, 1, 1, 1, 1, 1), \pm \frac{1}{2}(1, 1, 1, 1, 1, 1, -1, -1), \quad \frac{1}{2}(1, 1, 1, 1, -1, -1, -1, -1)$$

et on vérifie qu'ils appartiennent au réseau. Avec des vecteurs entiers, on a les vecteurs

$$(\pm 1, \pm 1, 0, 0, 0, 0, 0, 0)$$

7. En combinant les questions 5. et 7, les vecteurs minimaux sont les vecteurs de norme carré 2. Il y en a $(n+1)n$ pour \mathbb{A}_n (attention, il y a $n+1$ coordonnées), $2n(n-1)$ pour \mathbb{D}_n et

$$2 + 2 \cdot \binom{8}{2} + \binom{8}{4} + 4 \binom{8}{2} = 240$$

pour le réseau \mathbb{E}_8

Solution 119. Soient $(x_i)_{1 \leq i \leq n}$ les coordonnées d'un vecteur non nul, alors (par le théorème de Pythagore)

$$\|x_1 \mathbf{b}_1 + \cdots + x_n \mathbf{b}_n\|^2 = x_1^2 \|\mathbf{b}_1\|^2 + \cdots + x_n^2 \|\mathbf{b}_n\|^2$$

Comme chaque terme est positif, un vecteur n'est pas minimum si deux de ses coordonnées sont non nulles.

Solution 129. On peut calculer Gram-Schmidt avec

```
def myGS(B):
    m=B.nrows()
    n=B.ncols()
    Betoile = matrix(QQ,m,n)
    mu = matrix(QQ,n)

    for i in range(n):
        Betoile[:,i]=B[:,i]
        mu[i,i]=1
        for j in range(i):
            mu[j,i] = (B.column(i)*Betoile.column(j))/(Betoile.column(j)*Betoile.column(j))
            Betoile[:,i] += - mu[j,i]*Betoile[:,j]

    return Betoile,mu
```

et vérifier sur un exemple

```
n=3
B= matrix(n,[[1,5,-2], [2,3,4], [-3,2,1]])

Betoile ,mu = myGS(B)
B-Betoile*mu
```

Solution 135. 1. Il suffit que qu'aucun échange n'ait lieu (ligne 7) du code pour que la base soit LLL -réduite.

2.

3. En intégrant la mise à jour, on arrive à

Algorithme 44 : Réduction LLL d'un réseau (version bis)

Entrées : Base quelconque $B = \mathbf{b}_1, \dots, \mathbf{b}_n$ de \mathcal{L}
Sorties : Base LLL -réduite $\mathbf{b}_1, \dots, \mathbf{b}_n$ de \mathcal{L}

- 1 Calculer la base de Gram-Schmidt $(\mathbf{b}_i^*)_{1 \leq i \leq n}$ et la matrice de passage $\mu = ((\mu_{j,i}))_{1 \leq j \leq i \leq n}$ triangulaire sup. tels que $[\mathbf{b}_i] = [\mathbf{b}_i^*] \cdot \mu$
- 2 **répéter**
 - 3 **pour tous les** $2 \leq i \leq n$ **faire** // Etape de réduction
 - 4 **pour tous les** $i-1 \geq j \geq 1$ **faire**
 - 5 $\mathbf{b}_i \leftarrow \mathbf{b}_i - \lfloor \mu_{j,i} \rfloor \mathbf{b}_j$
 - 6 $\mu_i \leftarrow \mu_i - \lfloor \mu_{j,i} \rfloor \mu_j$
 - 7 $Bool \leftarrow \top$
 - 8 **si** $\exists i$ tq $\|\mathbf{b}_i^*\|^2 > 2\|\mathbf{b}_{i+1}^*\|^2$ **alors** // Etape d'échange
 - 9 $Bool \leftarrow \perp$
 - 10 $\text{Echanger}(\mathbf{b}_i, \mathbf{b}_{i+1})$
 - 11 $\mathbf{s} \leftarrow \mathbf{b}_i^*$
 - 12 $\mathbf{t} \leftarrow \mathbf{b}_{i+1}^* + \mu_{i,i+1} \mathbf{b}_i^*$
 - 13 $\text{Echanger}(\mu_i, \mu_{i+1})$
 - 14 $\mathbf{b}_i^* \leftarrow \mathbf{t}$
 - 15 $\mu_{i,i+1} \leftarrow \langle \mathbf{s}, \mathbf{b}_i^* \rangle / \|\mathbf{b}_i^*\|^2$
 - 16 $\mathbf{b}_{i+1}^* \leftarrow \mathbf{s} - \mu_{i,i+1} \mathbf{b}_i^*$
 - 17 **pour** $i+2 \leq k \leq n$ **faire**
 - 18 $\mu_{i,k} \leftarrow \langle \mathbf{b}_k, \mathbf{b}_i^* \rangle / \|\mathbf{b}_i^*\|^2$
 - 19 $\mu_{i+1,k} \leftarrow \langle \mathbf{b}_k, \mathbf{b}_{i+1}^* \rangle / \|\mathbf{b}_{i+1}^*\|^2$
- 20 **jusqu'à** $Bool$;
- 21 **retourner** B

Solution 136. 1. \mathcal{L}_1 a pour base (b_1, b_2) :

$$\mathcal{L}_1 = \mathbb{Z} \begin{pmatrix} 3 \\ -5 \\ 0 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -3 \\ 6 \\ -2 \end{pmatrix}$$

Or $u = \langle b_1, b_2 \rangle / \langle b_1, b_1, = \rangle \frac{-39}{34}$ qui s'arrondit à -1 . On remplace $b_2 \leftarrow b_2 + b_1$. On obtient la base

$$\mathcal{L}_1 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ -5 \\ 0 \end{pmatrix}$$

Encore une fois, $u = \langle b_1, b_2 \rangle / \langle b_1, b_1, = \rangle \frac{-5}{5} = -1$. On applique $b_2 \leftarrow b_2 + b_1$

$$\mathcal{L}_1 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ -4 \\ -2 \end{pmatrix}$$

On peut vérifier que l'on obtient une base orthogonal (c'est plutôt exceptionnel) et que le déterminant de la nouvelle base est toujours $5 \cdot 29 = 145$.

2. En utilisant le point précédent \mathcal{L}_2 a pour base (b_1, b_2, b_3) :

$$\mathcal{L}_2 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ -4 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 11 \\ -22 \\ 11 \end{pmatrix}.$$

On commence par calculer une orthogonalisation de Gram-Schmidt. On obtient les vecteurs

$$[b_1^* | b_2^* | b_3^*] = \begin{pmatrix} 0 & 3 & \frac{22}{29} \\ 1 & -4 & \frac{66}{145} \\ -2 & -2 & \frac{33}{145} \end{pmatrix}$$

De plus, $\|b_1^*\|^2 = 5$, $\|b_2^*\|^2 = 29$, $\|b_3^*\|^2 = 121/145$ et les coefficients $(\mu_{i,j})_{j \leq i}$:

$$[b_1 | b_2 | b_3] = [b_1^* | b_2^* | b_3^*] \begin{pmatrix} 1 & 0 & -\frac{44}{5} \\ 0 & 1 & \frac{99}{29} \\ 0 & 0 & 1 \end{pmatrix}$$

Le vecteur b_2 ne peut pas être raccourci par un multiple de b_1 car les deux vecteurs déjà sont orthogonaux. Comme $\lfloor -\frac{44}{5} \rfloor = -9$, on effectue $b_3 \leftarrow b_3 + 9b_1$. D'où la nouvelle base :

$$\mathcal{L}_2 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ -4 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 11 \\ -13 \\ -7 \end{pmatrix}$$

Ceci ne change pas la matrice $[b_1^* | b_2^* | b_3^*]$ mais la nouvelle matrice (μ) est maintenant (faire $1/5 = -44/5 + 9$)

$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & \frac{1}{5} \\ 0 & 1 & \frac{99}{29} \\ 0 & 0 & 1 \end{pmatrix}$$

Comme $\lfloor \frac{99}{29} \rfloor = 3$, on effectue $b_3 \leftarrow b_3 - 3b_1$ et on obtient :

$$\mathcal{L}_2 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ -4 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 2 \\ -1 \\ -1 \end{pmatrix}$$

La nouvelle matrice (μ) est maintenant (faire $12/29 = 99/29 - 3$)

$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & \frac{1}{5} \\ 0 & 1 & \frac{12}{29} \\ 0 & 0 & 1 \end{pmatrix}$$

Nous cherchons à présent, s'il est nécessaire d'échanger des vecteurs, i.e. si $\|b_2^*\|^2 > 2\|b_3^*\|^2$. C'est le cas. Nous échangeons b_2 avec b_3 . On a donc

$$\mathcal{L}_2 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 2 \\ -1 \\ -1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ -4 \\ -2 \end{pmatrix}$$

avec

$$[b_1^* | b_2^* | b_3^*] = \begin{pmatrix} 0 & 2 & -\frac{33}{29} \\ 1 & -\frac{6}{5} & -\frac{44}{29} \\ -2 & -\frac{3}{5} & -\frac{22}{29} \end{pmatrix}$$

avec $\|b_1^*\|^2 = 5$, $\|b_2^*\|^2 = \frac{29}{5}$, $\|b_3^*\|^2 = \frac{121}{29}$ et

$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{5} & 0 \\ 0 & 1 & \frac{60}{29} \\ 0 & 0 & 1 \end{pmatrix}$$

Il n'est pas possible de réduire b_2 . Par contre comme $\lfloor \frac{60}{29} \rfloor = 2$, on peut effectuer $b_3 \leftarrow b_3 - 2b_2$. On obtient

$$\mathcal{L}_2 = \mathbb{Z} \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 2 \\ -1 \\ -1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -1 \\ -2 \\ 0 \end{pmatrix}$$

et

$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{5} & -\frac{2}{5} \\ 0 & 1 & \frac{2}{29} \\ 0 & 0 & 1 \end{pmatrix}$$

La base est désormais LLL réduite.

3. On part de

$$\mathcal{L}_3 = \mathbb{Z} \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ 4 \\ 4 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ 0 \\ 7 \end{pmatrix}.$$

On commence par calculer

$$[b_1^* | b_2^* | b_3^*] = \begin{pmatrix} 1 & -\frac{24}{19} & 0 \\ 3 & \frac{4}{19} & -\frac{7}{2} \\ 3 & \frac{4}{19} & \frac{7}{2} \end{pmatrix} \text{ et } \begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{24}{19} & \frac{21}{19} \\ 0 & 1 & \frac{7}{8} \\ 0 & 0 & 1 \end{pmatrix}$$

avec $\|b_1^*\|^2 = 19$, $\|b_2^*\|^2 = \frac{32}{19}$, $\|b_3^*\|^2 = \frac{42}{2}$.

Comme $\lfloor \frac{24}{19} \rfloor = 1$, on effectue $b_2 \leftarrow b_2 - b_1$. On obtient la base

$$\mathcal{L}_3 = \mathbb{Z} \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ 0 \\ 7 \end{pmatrix}.$$

Et

$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{5}{19} & \frac{21}{19} \\ 0 & 1 & \frac{7}{8} \\ 0 & 0 & 1 \end{pmatrix}$$

Comme $\lfloor \frac{21}{19} \rfloor = 1$, on effectue $b_3 \leftarrow b_3 - b_1$. On obtient la base

$$\mathcal{L}_3 = \mathbb{Z} \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -1 \\ -3 \\ 4 \end{pmatrix}.$$

Et

$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{5}{19} & \frac{2}{19} \\ 0 & 1 & \frac{7}{8} \\ 0 & 0 & 1 \end{pmatrix}$$

Comme $\lfloor \frac{7}{8} \rfloor = 1$, on effectue $b_3 \leftarrow b_3 - b_2$. On obtient la base

$$\mathcal{L}_3 = \mathbb{Z} \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ -4 \\ 3 \end{pmatrix}.$$

Et

$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{5}{19} & \frac{2}{19} \\ 0 & 1 & -\frac{1}{8} \\ 0 & 0 & 1 \end{pmatrix}$$

Or $2\|b_2^*\| < \|b_1^*\|$, donc on échange b_1 et b_2 . On a

$$\mathcal{L}_3 = \mathbb{Z} \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ -4 \\ 3 \end{pmatrix}.$$

avec

$$[b_1^* | b_2^* | b_3^*] = \begin{pmatrix} -1 & \frac{8}{3} & 0 \\ 1 & \frac{4}{3} & -\frac{7}{2} \\ 1 & \frac{4}{3} & \frac{7}{2} \end{pmatrix} \text{ et } \begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{5}{3} & -\frac{1}{3} \\ 0 & 1 & -\frac{1}{8} \\ 0 & 0 & 1 \end{pmatrix}$$

avec $\|b_1^*\|^2 = 3$, $\|b_2^*\|^2 = \frac{3}{2}$, $\|b_3^*\|^2 = \frac{49}{2}$.

Comme $\lfloor \frac{5}{3} \rfloor = 2$, on effectue $b_2 \leftarrow b_2 - 2b_1$. On obtient la base

$$\mathcal{L}_3 = \mathbb{Z} \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 0 \\ -4 \\ 3 \end{pmatrix}.$$

Et

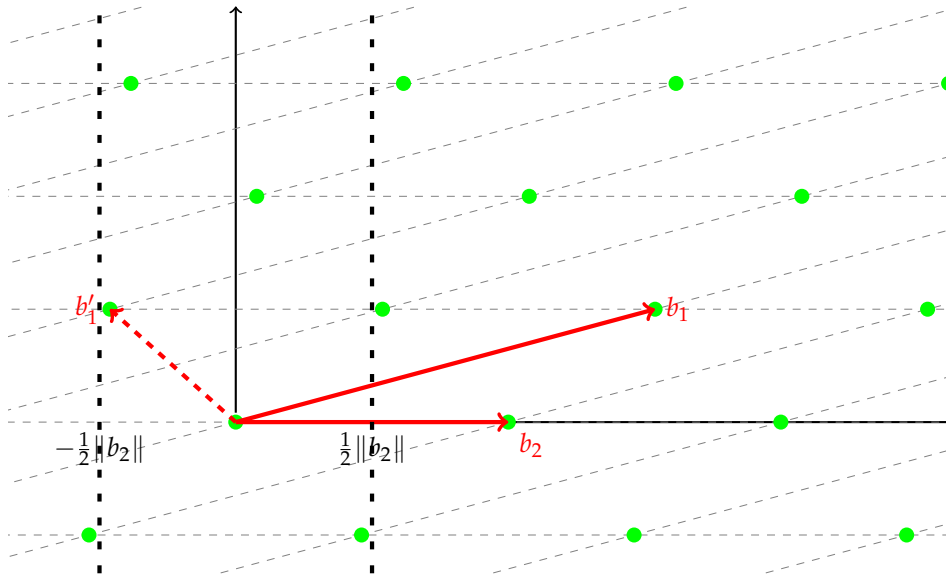
$$\begin{pmatrix} 1 & \mu_{2,1} & \mu_{3,1} \\ 0 & 1 & \mu_{3,2} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \frac{-1}{3} & \frac{-1}{3} \\ 0 & 1 & -\frac{1}{8} \\ 0 & 0 & 1 \end{pmatrix}$$

Cette base est LLL réduite.

Solution 137. Cherchons à présent une base LLL :

Étape 1 : En posant $\mathbf{b}_1 = \begin{pmatrix} 4 \\ 4 \\ 1 \end{pmatrix}$ et $\mathbf{b}_2 = \begin{pmatrix} 3 \\ 2 \\ 0 \end{pmatrix}$, nous avons une première

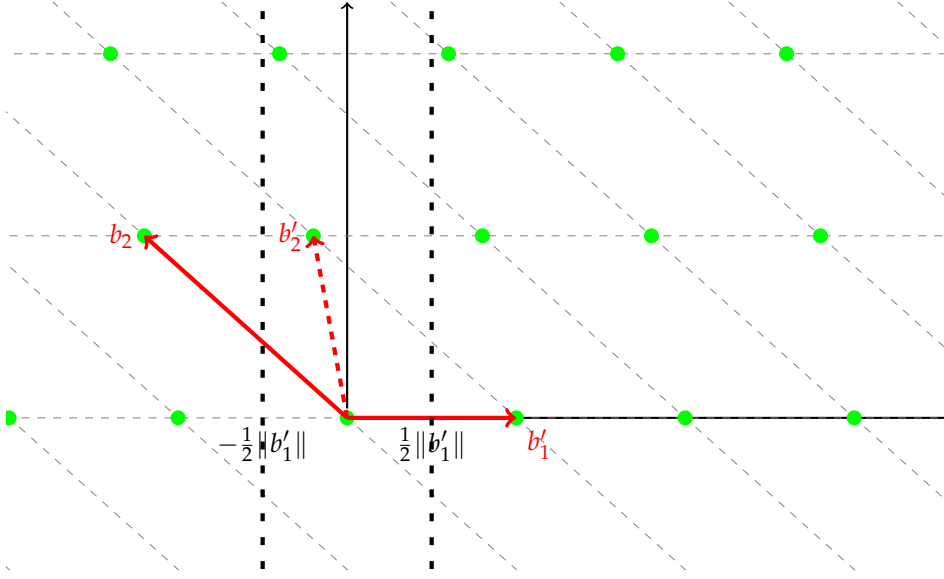
base qui vérifie $\|\mathbf{b}_1\|^2 = 33$ et $\|\mathbf{b}_2\|^2 = 13$. De plus, $\langle \mathbf{b}_1, \mathbf{b}_2 \rangle = 20$. Nous pourrions dessiner le réseau comme suit (en calculant le cosinus de l'angle (b_1, b_2) pour tracer le dessin exact) :



La base $(\mathbf{b}_2, \mathbf{b}_1)$ est-elle LLL réduite? Or $\lfloor \frac{20}{13} \rfloor = 2$, donc on peut rem-

placer \mathbf{b}_1 par $\mathbf{b}'_1 = \mathbf{b}_1 - 2\mathbf{b}_2 = \begin{pmatrix} -2 \\ 0 \\ 1 \end{pmatrix}$.

Étape 2 : La base $(\mathbf{b}'_1, \mathbf{b}_2)$ est-elle LLL réduite? Nous avons à présent $\|\mathbf{b}'_1\|^2 = 5$, $\|\mathbf{b}_2\|^2 = 13$ et $\langle \mathbf{b}'_1, \mathbf{b}_2 \rangle = -6$. Nous pourrions dessiner la situation comme suit (à la même échelle).



Comme $\lfloor \frac{-6}{5} \rfloor = -1$, on peut remplacer \mathbf{b}_2 par $\mathbf{b}'_2 = \mathbf{b}_2 - \mathbf{b}'_1 = \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$

qui est de norme 6.

Étape 3 : la base $(\mathbf{b}'_1, \mathbf{b}'_2)$ est-elle LLL réduite ? Nous calculons $\langle \mathbf{b}'_1, \mathbf{b}'_2 \rangle = 1$ et $\left| \frac{\langle \mathbf{b}'_1, \mathbf{b}'_2 \rangle}{\langle \mathbf{b}'_1, \mathbf{b}'_1 \rangle} \right| \leq \frac{1}{2}$. Donc la base $(\mathbf{b}'_1, \mathbf{b}'_2)$ est LLL -réduite.

$$\mathcal{L} = \mathbb{Z} \begin{pmatrix} -2 \\ 0 \\ 1 \end{pmatrix} \oplus \mathbb{Z} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$$

Les minima successifs sont $\sqrt{5}$ et $\sqrt{6}$ d'après le théorème 124. Le déterminant est

$$\det \mathcal{L} = \begin{vmatrix} 5 & 1 \\ 1 & 6 \end{vmatrix} = 29$$

et le discriminant est $\sqrt{29}$. Remarque : nous aurions aussi pu calculer le déterminant directement à partir de la première base : $\det \mathcal{L} =$

$$\begin{vmatrix} 33 & 20 \\ 20 & 13 \end{vmatrix} = 29.$$

Solution 157. Si -1 est un carré modulo p , on a déjà

$$x^4 + 1 = (x^2 + \sqrt{-1})(x^2 - \sqrt{-1}). \quad (68)$$

Si 2 est un carré modulo p , on a

$$x^4 + 1 = (x^2 + 1)^2 - 2x^2 = (x^2 + \sqrt{2}x + 1)(x^2 - \sqrt{2}x + 1). \quad (69)$$

Si -2 est un carré modulo p , on a

$$x^4 + 1 = (x^2 - 1)^2 - (-2)x^2 = (x^2 + \sqrt{-2}x - 1)(x^2 - \sqrt{-2}x - 1). \quad (70)$$

Si -1 , 2 et -2 sont des carrés (ils ne peuvent l'être que simultanément tous les trois, d'après les règles de calcul du symbole de Legendre), alors (si $p \neq 2$) :

$$x^4 + 1 = \left(x - \frac{\sqrt{2}}{2}(1 + \sqrt{-1})\right) \left(x - \frac{\sqrt{2}}{2}(1 - \sqrt{-1})\right) \\ \left(x - \frac{\sqrt{2}}{2}(-1 + \sqrt{-1})\right) \left(x - \frac{\sqrt{2}}{2}(-1 - \sqrt{-1})\right) \quad (71)$$

Nous utilisons maintenant les lois de réciprocité quadratique pour déterminer chaque cas. D'abord si $p = 2$, on a $x^4 + 1 = (x + 1)^4$. Pour p impair, nous utilisons les lois de réciprocité quadratique pour déterminer chaque cas : nous avons

	$p \equiv 1 \pmod{8}$	$p \equiv 3 \pmod{8}$	$p \equiv 5 \pmod{8}$	$p \equiv 7 \pmod{8}$
1	\square	/	\square	/
2	\square	/	/	\square
-2	\square	\square	/	/
Factorisation	(71)	(70)	(68)	(69)

Solution 163. 1. On définit \mathbb{F}_{1849} comme $\mathbb{F}_{43}[\omega]$ où $\omega^2 = -1$.

- (a) On calcule de symbole de Legendre de -1 avec la première loi complémentaire de la réciprocité quadratique. Comme -1 est un non-carré dans \mathbb{F}_{43} , la factorisation de $x^2 + 1$ en deux termes linéaires ne peut pas se faire.
- (b) On note $\sigma : t \mapsto t^{43}$ l'endomorphisme de Frobenius. Comme l'extension est de degré 2, on a $\sigma \circ \sigma = \text{Id}$. Donc $\sigma(t)$ est toujours une racine 43ième de t . Mais $\sigma(\omega)$ est l'autre racine de $x^2 + 1$, donc $\sigma(\omega) = -\omega$. Par \mathbb{F}_{43} -linéarité, $\sigma(a + \omega b) = a - \omega b$.
- (c) Les monômes sont tous en des puissances multiples de 43. On a

$$\tilde{u}(x) = x^4 + (3 + 2\omega)x^3 + 5\omega x^2 + (2 - 4\omega)x - 1 + \omega.$$

Solution 170. 1. Dans \mathbb{F}_3 , $f'(x) = x^{999}$ est premier avec f , car le seul diviseur irréductible de f' est x , qui ne divise pas $f(x)$. Donc f est bien sans facteur carré.

Dans \mathbb{F}_5 , $f'(x) = 0$, donc f est une puissance de 5 et n'est largement pas sans facteur carré. En fait,

$$f(x) = (x^8 + 2)^{5^3} = (x^8 + 2)^{125}$$

et $x^8 + 2$ est sans facteur carré. En effet, $(x^8 + 2)' = 8x^7$ n'a pas de racine commune avec $x^8 + 2$.

- 2. Faux : prendre $f(x) = g(x) = x$.

Solution 171. 1. On a dans ce cas $f \wedge g$ est la partie sans facteurs carrés de $(f \cdot g)$.

2. Si la décomposition de f et de g en facteurs irréductibles est

$$f = \prod_j p_j^{\alpha_j} \quad g = \prod_j p_j^{\beta_j}$$

alors

$$f \wedge g = \prod_j p_j^{\min \alpha_j, \beta_j}$$

Or $f_i = \prod_{j, \alpha_j=i} p_j$, $g_i = \prod_{j, \beta_j=i} p_j$, donc le facteur p_j est présent dans les α_j premiers produits $f_i \cdots f_s$ et les β_j premiers produits $g_i \cdots g_t$ et seulement $\min(\alpha_j, \beta_j)$ fois dans leur pgcd, comme voulu.

3.

Solution 172. Remarque préliminaire : vu le nombre de calculs que nous ferons, il peut être utile de précalculer la table des inverses dans \mathbb{F}_7 :

$$\begin{array}{ll} 1^{-1} = 1 & (-1)^{-1} = 6^{-1} = -1 = 6 \\ 2^{-1} = 4 & (-2)^{-1} = 5^{-1} = 3 \\ 3^{-1} = 5 & (-3)^{-1} = 4^{-1} = 2 \end{array}$$

Nous notons $f = \prod_i (f_i(x))^i$ la factorisation sans facteurs carrés de f . On a

$$f'(x) = 3x^9 + 5x^8 + 2x^2 + x = 3(x^9 + 4x^8 + 3x^2 + 5x).$$

Nous commençons par calculer le pgcd de f et f' . On applique l'algorithme d'Euclide comme suit :

$$\begin{aligned} f(x) &= \underbrace{(x+2)}_{q_1} \cdot f'(x) - \underbrace{(x^8 + 4x^7 + 3x + 5)}_{r_1} \\ f'(x) &= \underbrace{x}_{q_2} \cdot r_1 + \underbrace{0}_{r_2} \end{aligned}$$

On en déduit que le pgcd est

$$f \wedge f' = x^8 + 4x^7 + 3x + 5$$

Mais on sait que $f \wedge f' = \prod_{7|i} f_i^{i-1} \prod_i f_{7i}^{7i}$. On divise f par $f \wedge f'$. On obtient

$$t(x) = \frac{f(x)}{(f \wedge f')(x)} = \prod_{7 \nmid i} f_i = (x^2 + 2x + 6)$$

Essayons de factoriser $t(x)$. Le discriminant est $\Delta = b^2 - ac = 4 + 4 = 1 = 1^2$, donc t possède deux racines :

$$x = \frac{-b \pm \sqrt{\Delta'}}{2a} = \frac{-2 \pm 1}{2} = 2 \text{ ou } 3.$$

D'où

$$t(x) = (x+4)(x+5)$$

Nous remarquons que $f'(3) = 0$ mais $f'(2) \neq 0$. Donc $x+5$ est un facteur simple de f , ce qui donne déjà $f_1(x) = x+5$. Il reste à analyser le facteur $x+4$. Pour des raisons de degrés, soit $x+4$ est de multiplicité 2 et il manque un facteur de degré 1 et multiplicité 7, soit $x+4$ est de multiplicité 9. Nous factorisons $x+4$ dans $f \wedge f'$. Nous obtenons

$$f \wedge f'(x) = (x+4)(x^7+3) = (x+4)(x+3)^7$$

ce qui montre que nous sommes dans le premier cas et que

$$f = (x+5)(x+4)^2(x+3)^7.$$

Solution 177. On en déduit que f ne possède pas de facteurs irréductibles de degré 1 ou 2. Donc f est lui-même irréductible.

Solution 178. 1. Il faut vérifier que $q^r \equiv q \pmod{r}$, ce qui est simplement le petit théorème de Fermat ou le fait que \mathbb{F}_r^\times est un groupe multiplicatif d'ordre $r-1$.

2. Les polynômes irréductibles unitaires de degré divisant r forment l'ensemble des facteurs de $x^{q^r} - x$. Parmi eux, les seuls degrés possibles sont r ou 1 comme r est premier. Soit t_d le nombre polynômes irréductibles unitaires de degré d . On a $t_1 + rt_r = \deg(x^{q^r} - x) = q^r$. Mais il y a exactement $t_1 = q$ polynômes unitaires de degré 1 : les polynômes $(x - \alpha)$ pour $\alpha \in \mathbb{F}_q$. Donc $t_r = (q^r - q)/r$ comme attendu.

3. En général, on a encore

$$\sum_{d|r} d \cdot t_d = \deg(x^{q^r} - x) = q^r$$

Mais comme r est une puissance d'un nombre premier disons p , on a encore $\sum_{d|r, d \neq r} d \cdot t_d + rd_r = q^{r/p} + rd_r = q^r$, on a donc

$$t_r = \frac{q^r - q^{r/p}}{r}$$

En fait, en général, la formule $\sum_{d|r} d \cdot t_d = q^r$ s'inverse à l'aide de la fonction de Möbius μ . Ici, on en déduit immédiatement que

$$d_r = \frac{1}{r} \sum_{d|n} \mu(n/d) q^d.$$

Solution 179. On vérifie que $(f \wedge x^{q^{10}} - x) = f$ ce qui prouve que f ne possède que des facteurs irréductible de degré divisant 10, $(f \wedge x^{q^2} - x) = 1$ et $(f \wedge x^{q^5} - x) = 1$ ce qui exclut des facteurs irréductibles de degré divisant 2 ou 5.

Solution 180. Notons d'abord que pour être irréductible, un binôme doit être de la forme $x^n - \alpha$.

Supposons que $n \nmid q-1$ et considérons l'application $\tau : \mathbb{F}_q^\times \rightarrow \mathbb{F}_q^\times$ telle que $\tau(x) = x^n$. Alors τ est un morphisme de groupe et son noyau est l'ensemble des éléments d'ordre divisant n . Mais les éléments de \mathbb{F}_q sont d'ordre divisant $q-1$, et $q-1$ est premier avec n . Donc $\ker \tau$ est réduit à $\{1\}$ et τ est une bijection. Ainsi tout binôme $x^n - \alpha$ admet un facteur linéaire.

Réciproquement, si $n|q-1$, alors τ a un noyau formé de toutes les n racines n -ièmes de l'unité et il existe $\alpha \in \mathbb{F}_q^\times \setminus \text{Im}(\tau)$. Pour un tel α , $f(x) = x^n - \alpha$ n'a pas de facteur linéaire, donc $f \wedge (x^q - x) = 1$. Il reste à vérifier que $x^{q^n} - x \equiv 0 \pmod f$ et le critère de Rabin sera satisfait. Or $x^n \equiv \alpha \pmod f$ donc $x^{(q-1)n} \equiv \alpha^{q-1} \equiv 1 \pmod f$. Il suffit donc de prouver que $q^n \equiv 1 \pmod{n(q-1)}$. Mais,

$$\begin{aligned} q^n - 1 &= (1 + (q-1))^n \\ &= 1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{n-1}(q-1)^{n-1} + (q-1)^n - 1 \end{aligned}$$

Or $\binom{n}{1}(q-1) = n(q-1)$, par ailleurs $(q-1)^i$ pour $i \geq 2$ est divisible par $n(q-1)$ car $n|q-1$. Donc $q^n - 1 \equiv 0 \pmod{n(q-1)}$ comme souhaité. D'où le résultat attendu.

Solution 183. Dans ce cas, 1 est toujours racine.

Solution 193. 1. On se contente de calculer

$$r = x^q - x \pmod f$$

puis

$$g = f \wedge r.$$

Ceci fournit le produit g de tous les facteurs de degré 1 de f . Puis on utilise l'algorithme de Cantor-Zassenhaus pour factoriser complètement g sous la forme $(x - \alpha_1) \cdots (x - \alpha_\ell)$ et en déduire les racines $(\alpha_i)_{i \leq \ell}$.

Solution 201. 1. On donne $f = x^3 - x^2 - 1 \in \mathbb{F}_3[x]$.

(a) On a $x^3 \equiv x^2 + 1 \pmod f$ et $x^6 \equiv (x^2 + 1)^2 = x - 1 \pmod f$. On en déduit la matrice

$$\begin{pmatrix} 1 & 1 & -1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

(b) Calculons le noyau de

$$\begin{pmatrix} 0 & 1 & -1 \\ 0 & -1 & 1 \\ 0 & 1 & -1 \end{pmatrix}$$

On a visiblement les vecteurs voulus qui correspondent aux polynômes 1 et $x^2 - x + 1$

- (c) Comme le pgcd de f avec $x^2 - x + 1$ est $x + 1$ ce qui fournit un facteur. L'autre facteur, obtenu par division, est $x^2 + x - 1$.

Solution 202. On construit les polynômes

$$\ell_i(x) = \prod_{j=1, j \neq i}^k f_j(x)$$

Par construction $\ell_i \equiv 0 \pmod{f_j}$ pour $j \neq i$ et $\ell_i \not\equiv 0 \pmod{f_i}$ (pour des raisons de degrés). Le polynôme $\ell_i(x)$ est donc inversible dans $\mathbb{F}_q[x]/\langle f_i \rangle$ et $\ell_i^{-1} \ell_i(x)$ est le polynôme interpolateur de Lagrange, il correspond au vecteur $(0, \dots, 1, 0, \dots, 0)$ dans l'identification avec $\bigoplus_{i=1}^k \mathbb{F}_p[x]/\langle f_i \rangle$.

Solution 207. 1. On a $f^*(x) = x^n f(1/x) = x^{\deg g} g(1/x) x^{\deg h} h(1/x) = g^*(x) h^*(x)$.

2. Avant de factoriser f , on vérifie si $|f_0| < |f_n|$. Si c'est le cas, on factorise f^* et on renverse à nouveau les facteurs à la fin de l'algorithme.

Solution 208. Le polynôme

$$\zeta_8(x) = x^4 + 1$$

est un polynôme irréductible sur \mathbb{Z} . Il s'agit du 8-ième polynôme cyclotomique, dont les racines sont les $\varphi(8) = 4$ racines primitives 8-ièmes de l'unité : $\pm \exp(\pm i\pi/4)$ ou encore $\pm \frac{\sqrt{2}}{2} \pm i \frac{\sqrt{2}}{2}$. Or $\zeta_8(x)$ se factorise en $(x+1)^4$ en caractéristique 2 et se factorise au moins partiellement en caractéristique impaire (d'après l'exercice 157).

Solution 209. On commence par remarquer que comme $h \in \mathbb{Z}[x]$ est unitaire, si $k \in \mathbb{Z}[x]$ vérifie $k \equiv 0 \pmod{m}$, alors le quotient q et le reste r de k par h sont encore congrus à 0 \pmod{m} . Cette affirmation est claire si le quotient est nul. Dans le cas contraire, on remarque que le terme dominant $\text{td}(q)$ doit être $\equiv 0 \pmod{m}$ et on raisonne par récurrence sur $k - \text{td}(q)h$.

Ceci montre que dans notre algorithme, $q, r \equiv 0 \pmod{m}$ car $se \equiv 0 \pmod{m}$.

On note à présent que

$$\begin{aligned} f - g^* h^* &\equiv f - (g + te + qg)(h + se - qh) \\ &= f - gh - (sg + th)e - st \underbrace{e^2}_{\equiv 0} - (sg - th) \underbrace{qe}_{\equiv 0} + gh \underbrace{q^2}_{\equiv 0} \\ &\equiv (1 - sg - th)e \equiv 0 \pmod{m^2} \end{aligned}$$

Solution 211. 1. On peut faire

```

def SymMod(a,m):
    a=mod(a,m).lift()
    if a<=m/2:
        return a
    else:
        return a-m

def mod_polyn(f,m):
    return f.parent()(map(lambda x:SymMod(x,m), f))

```

2. On peut faire

```

def MyHensellift(f,g,h,s,t,m):
    e = mod_polyn(f-g*h,m^2)
    q,r = (s*e).quo_rem(h)
    q = mod_polyn(q,m^2)
    r = mod_polyn(r,m^2)
    gg = mod_polyn( g+t*e+q*g, m^2)
    hh = mod_polyn( h+r, m^2)
    b = mod_polyn( s*gg+t*hh-1, m^2)
    c,d = (s*b).quo_rem(hh)
    c = mod_polyn(c,m^2)
    d = mod_polyn(d,m^2)
    ss = mod_polyn( s-d, m^2)
    tt = mod_polyn(t- t*b-c*gg, m^2)
    return (gg,hh,ss,tt )

```

3.

Solution 231. 1. D'après le lemme, il suffit de vérifier terme à terme que les exposants de chaque monôme formant les polynômes appartiennent au diagramme. Comme $(2,1)$ n'est pas dans le diagramme, $x^6y^3 + x^2y$ n'appartient pas à \mathfrak{I} . Comme $(2,4)$ et $(7,3)$ appartiennent au diagramme, $x^2y^4 + x^7y^3$ fait partie de \mathfrak{I} .

2. On voit que l'idéal est engendré par $\{x^3y, xy^3\}$ qui sont les deux coins de l'escalier.

Solution 241. 1. Le reste de f par (f_1, f_2) est x^2 , les quotients étant nuls.

2. Le ppcm est x^3y . Il faut multiplier f_1 par y et f_2 par x .

3. On peut prendre $q_1 = -y$ et $q_2 = x$.

4. La liste (f_1, f_2) n'est pas une base de Groebner de l'idéal qu'elle engendre.

5. Conclure.

Solution 257. On peut faire

```

MPol.<x,y, lambd> = PolynomialRing(QQ,3, order='Lex')
f = x^2*y - 2*x*y+y+1
g = x^2+y^2-1
gradf = f.gradient()
gradg = g.gradient()
I = Ideal([g, gradf[0]-lambd*gradg[0], gradf[1]-lambd*gradg[1]])
B = I.groebner_basis()
Cand = []
listlambd = B[-1].univariate_polynomial().roots(ring=RR, multiplicities=false)
for l in listlambd:
    listy = (B[1].subs(lambd=l)).univariate_polynomial().roots(ring=RR, multiplicities=false)
    for y0 in listy:
        for x0 in (B[0].subs(lambd=l, y=y0)).univariate_polynomial().roots(ring=RR, multiplicities=false):
            Cand.append((x0,y0))
Cand

```

On obtient trois points candidats : le point $(1,0)$ qui correspond à un plat de la fonction (pas d'optimum), les points $(-0.667, \pm 0.745)$ qui correspondent aux deux extremas.

Solution 258. On considère l'idéal

$$\langle 3x^2 + 2yz - 2x\lambda, 2xz - 2y\lambda, 2xy - 2z - 2z\lambda, x^2 + y^2 + z^2 - 1 \rangle$$

dont une base de Groebner pour l'ordre $\lambda > x > y > z$ se termine par le polynôme

$$z^7 - \frac{1763}{1152}z^5 + \frac{655}{1152}z^3 - \frac{11}{288}z$$

Il s'en suit que z peut prendre les valeurs

$$z = 0, \pm 1, \pm 2/3, \pm \sqrt{22}/16.$$

Solution 263. Manifestement, on cherche à travailler sur des relations entre $u(\theta)$, $v(\theta)$, $\sin(\theta)$ et $\cos(\theta)$. On introduit l'anneau $\mathbb{Q}[u, v, s, c]$ et l'idéal $\mathfrak{R} = \langle u - s - c, v - 2sc - c^2 + s^2, s^2 + c^2 - 1 \rangle$. On note que pour tout $p \in \mathfrak{R}$, $p(u(\theta), v(\theta), \sin(\theta), \cos(\theta))$ s'annule. Aussi, une idée pour trouver la solution serait de regarder $s^6 \bmod \mathfrak{R}$ et de chercher un représentant qui ne s'exprime que avec u et v . On choisit l'ordre $c > s > u > v$ sur les monômes et on calcule $s^6 \bmod \mathfrak{R}$ ce qui fournit un résultat acceptable.

Solution 287. Il s'agit d'un cercle de rayon 2 dont le centre se déplace sur une parabole $y = 4 - (x - 2)^2$.

Solution 293. Un tel graphe s'appelle graphe biparti. On effectue un parcours de graphe et on attribue alternativement une couleur ou une autre en commençant par une couleur quelconque. De la sorte, on peut

soit détecter des cycles de longueur impaire (ce qui empêche le graphe d'être biparti), soit colorier le graphe par deux couleurs.

Solution 300. Méthode standard : On essaye de voir si $M + N = M$ ce qui revient à calculer une base normalisée de chaque module avec la forme normale d'Hermite. Or la forme normale d'Hermite de

$$\begin{pmatrix} 1 & 1 & 13 & -1 & 4 \\ 2 & 0 & 7 & -2 & 3 \\ 0 & 3 & -10 & 2 & -4 \end{pmatrix}$$

est

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

alors que la forme normale d'Hermite de

$$\begin{pmatrix} 13 & -1 & 4 \\ 7 & -2 & 3 \\ -10 & 2 & -4 \end{pmatrix}$$

est

$$\begin{pmatrix} 2 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

Les colonnes non-nuls étant distinctes, on peut conclure que $M \not\subseteq N$.

Méthode accélérée : une fois la base de N

$$\begin{pmatrix} 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}$$

calculée, on peut remarquer que le vecteur \mathbf{m}_1 ne pourra jamais s'écrire dans cette base avec des coefficients entiers. Donc $M \not\subseteq N$.

Méthode accélérée : on regarde la dernière ligne et on note que 3 ne peut être la combinaison de nombres pairs.

Solution 301. On calcule la forme normale de Smith de

$$A = \begin{pmatrix} 2 & -7 & 3 & 5 & 0 & 0 \\ 1 & -5 & 4 & 0 & 3 & 0 \\ 4 & 0 & -6 & 0 & 0 & 8 \end{pmatrix}$$

On obtient (par exemple) les matrices Δ , L et C telles que $LAC = \Delta$ avec

$$\Delta = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \end{pmatrix}$$

$$L = \begin{pmatrix} -5 & 7 & 1 \\ -13 & 18 & 2 \\ -20 & 28 & 3 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 214 & 196 & -9 & -214 & -196 & 60 \\ -128 & -119 & 7 & 128 & 119 & -36 \\ -285 & -263 & 14 & 285 & 263 & -80 \\ 161 & 147 & -7 & -161 & -147 & 45 \end{pmatrix}$$

Comme $Lb = (9, 22, 34)$, on en déduit une solution particulière qui est

$$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = 9 \begin{pmatrix} 0 \\ 0 \\ 214 \end{pmatrix} + 22 \begin{pmatrix} 0 \\ 0 \\ 196 \end{pmatrix} + \frac{22}{2} \begin{pmatrix} 0 \\ 0 \\ -9 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 6085 \end{pmatrix}$$

La solution générale est

$$\begin{pmatrix} \lambda \\ \mu \\ 6085 - 214\lambda - 196\mu + 60\nu \end{pmatrix}$$

avec λ, μ, ν dans \mathbb{Z} .

Remarques. Il est inutile de calculer L en entier, on calcule directement Lb . De même, on peut se contenter de calculer la moitié de U .

Solution 302. On calcule une forme normale de Smith de A . On obtient (par exemple) les matrices Δ , L et C telles que $LAC = \Delta$ avec

$$\Delta = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -2 & 0 \\ -3 & -5 & 1 \end{pmatrix}$$

$$C = \begin{pmatrix} -13 & -12 & 1 & 8 & 1 \\ -12 & -11 & 1 & 6 & 2 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

(pour simplifier les calculs, on peut commencer par factoriser 2) On en déduit que \mathbb{Z}^3/N a pour structure

$$\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/6\mathbb{Z}.$$

Le rang est 1, les facteurs invariants $(2, 6)$. Si on souhaite être plus précis, on peut même dire que

$$\mathbb{Z}/N = \mathbb{Z} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \oplus \mathbb{Z}/2\mathbb{Z} \begin{pmatrix} 2 \\ 1 \\ 11 \end{pmatrix} \oplus \mathbb{Z}/6\mathbb{Z} \begin{pmatrix} 1 \\ 0 \\ 3 \end{pmatrix}$$

en calculant

$$L^{-1} = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 0 & 0 \\ 11 & 3 & 1 \end{pmatrix}$$

Solution 303. On commence par chercher la base de G-S associée qui est

$$B^* = \begin{pmatrix} -3 & -\frac{5}{7} & 0 \\ 1 & -\frac{3}{7} & -\frac{2}{5} \\ 2 & -\frac{6}{7} & \frac{1}{5} \end{pmatrix} \quad \mu^* = \begin{pmatrix} 1 & -\frac{4}{7} & \frac{13}{14} \\ 0 & 1 & -\frac{11}{10} \\ 0 & 0 & 1 \end{pmatrix}$$

avec $B = B^*\mu$. On effectue $C_3 \leftarrow C_3 + C_1$, ce qui donne

$$B = \begin{pmatrix} -3 & 1 & -1 \\ 1 & -1 & 0 \\ 2 & -2 & 1 \end{pmatrix} \quad \mu^* = \begin{pmatrix} 1 & -\frac{4}{7} & \frac{5}{14} \\ 0 & 1 & -\frac{1}{10} \\ 0 & 0 & 1 \end{pmatrix}$$

Puis, $C_2 \leftarrow C_2 + C_1$, ce qui donne

$$B = \begin{pmatrix} -3 & -2 & -1 \\ 1 & 0 & 0 \\ 2 & 0 & 1 \end{pmatrix} \quad \mu^* = \begin{pmatrix} 1 & \frac{3}{7} & \frac{5}{14} \\ 0 & 1 & -\frac{1}{10} \\ 0 & 0 & 1 \end{pmatrix}$$

On peut continuer avec l'algorithme classique. On peut aussi essayer d'accélérer les calculs. On trie les vecteurs par ordre croissant de norme.

$$B = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 0 & -1 \\ 1 & 0 & 2 \end{pmatrix}$$

À l'oeil, on devine que $C_3 \leftarrow C_3 - C_2 - C_1$ puis $C_2 \leftarrow C_1$ donnent de courts vecteurs :

$$B = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & -1 \\ 1 & -1 & 1 \end{pmatrix}$$

On obtient en réalité le réseau $\mathbb{A}_3 = \mathbb{D}_3$, dont les minima successifs valent tous 2.

Solution 304. 1. On peut noter que $f = (x^6 + x + 1)^4$ et que $g = x^6 + x + 1$ est premier avec $g' = 1$.

2. On voit que g n'a ni 0 ni 1 comme racine.

3. On utilise le critère de Rabin : $x^6 + x + 1 = (x^3 + 1) \cdot (x^3 + 1) + x$ et $x^3 = x^2 \cdot x + 1$. Ainsi $f \wedge (x^3 + 1) = 1$ donc f n'admet pas de facteur de degré 2.
4. De même, $g \wedge (x^7 + 1) = 1$ car $x^7 + 1 = x \cdot g + x^2 + x + 1$, $g = (x^4 + x^3 + x + 1)(x^2 + x + 1) + x$ et $x^2 + x + 1 = (x + 1)x + 1$. donc g n'admet pas de facteur de degré 3 et f non plus.
5. On en déduit que g est irréductible sur \mathbb{F}_2 .
6. Par conséquent g se factorise sur toute extension \mathbb{F}_{64} , ou encore sur \mathbb{F}_{2^k} pour k divisible par 6. Comme 5 n'est pas dans ce cas, g reste irréductible sur \mathbb{F}_{32} .

Solution 305. 1. $S(f, g) = y^3 f - x^2 g = -x^2 y - x^2 + y^4 - y^3$ or $S = (-y - 1)f + (-y - 1)g$ Donc f, g est déjà une base de Groebner.

- 2.
3. La dimension est 6.
4. On cherche 6 zéros au total. On peut voir que $g = (y - 1)(y^2 + y + 1)$. Notons j une racine de $y^2 + y + 1$ et $\bar{j} = j^3$ l'autre racine dans \mathbb{F}_9 . On obtient les zéros $(0, 1)$, $(\pm\sqrt{1-j}, j)$ et $(\pm\sqrt{1-\bar{j}}, \bar{j})$. Pour des raisons de symétrie, les 4 derniers points doivent avoir le même ordre. La seule possibilité est donc que $(0, 1)$ soit d'ordre 2 et les autres points d'ordre 1.

Solution 309. On procède comme suit

```
def temoinF(n,a):
    return mod(a,n)^(n-1)==1

def testF(n):
    a=ZZ.random_element(2,n)
    if gcd(a,n)>1:
        return false
    return temoinF(n,a)

for _ in range(5): testF(9)

[n for n in range(2,3000) if all([temoinF(n,a)
for a in range(2,n) if gcd(a,n)==1]) and not is_prime(n)]
```

Solution 310. 1. Supposons que $n = p^e r$ avec p premier, $e \geq 2$ et r quelconque. Alors $a^n = 1$ pour tout $a \in \mathbb{Z}/p^e \mathbb{Z} \setminus \{0\}$. Donc $n - 1$ est un multiple de $\varphi(p^e) = (p - 1)p^{e-1}$. Ainsi p divise à la fois $n - 1$ et n ce qui n'est pas possible.

Soit p un diviseur premier de n . Comme n est de Carmichael, on a encore $a^{n-1} = 1 \pmod p$ pour tout $a \in \mathbb{F}_p^\times$ ce qui implique que $n - 1$ est un multiple de $|\mathbb{F}_p^\times| = p - 1$.

2. Si n est pair, $n - 1$ est impair, donc pour tout diviseur de p , $p - 1$ est impair, si bien que p est pair. Mais alors $n = 2$ qui n'est pas de Carmichael.

Supposons que $n = pq$ où p et q sont premiers. Alors $n - 1 = pq - 1 = p(q - 1) + p - 1$. Donc pour tout $a \in \mathbb{F}_q^\times$, $a^{n-1} = a^{p-1} = 1 \pmod{q}$. Donc $q|p - 1$. Symétriquement, $p|q - 1$. Ces deux conditions sont incompatibles.

Remarque : il suffit que n soit sans facteur carré et vérifie que pour tout premier p divisant n , $p - 1$ divise $n - 1$. En effet, dans ce cas, pour tout a premier avec n , $a^{p-1} = 1 \pmod{p}$ implique que $a^{n-1} = 1 \pmod{p}$ et le lemme chinois montre que $a^{n-1} = 1 \pmod{n}$.

Solution 311. Si $a^{n-1} = 1$, alors a est inversible, d'inverse a^{n-2} . Donc $a \wedge n = 1$.

Solution 314. Notons $a_i = a^{2^i m} \in \mathbb{F}_p$. D'après le théorème de Fermat, $a_p = 1$. Dans ce cas, soit $a_i = 1$ pour tout i , soit il existe d tel que $a_d \neq 1$ et $a_{d+1} = a_d^2 = 1$. Mais alors $a_d = -1$ (Comme \mathbb{F}_p est un corps, 1 n'a que deux racines carrées possibles.).

Solution 326. Quand n est premier, $\phi : x \mapsto x^{(n-1)/2}$ est un morphisme de groupe dont les carrés appartiennent au noyau car $\phi(y^2) = y^{n-1} = 1 \pmod{n}$ et dont le noyau ne peut contenir qu'au plus $(n - 1)/2$ éléments (nombre de racines du polynôme $x^{(n-1)/2} - 1$ dans \mathbb{F}_n). Donc $\phi(x)$ vaut 1 si et seulement si x est un carré. Comme $\phi(x)^2 = 1$ et que \mathbb{F}_n est un corps, $\phi(x)$ vaut -1 si x n'est pas un carré.

Réciproquement, supposons que $\left(\frac{a}{n}\right) = (-1)^{(n-1)/2}$ pour tout $a \in (\mathbb{Z}/n\mathbb{Z})^\times$. On a alors $a^{n-1} = \left(\frac{a}{n}\right)^2 = 1 \pmod{n}$, donc n est soit premier, soit un nombre de Carmichael. Dans ce second cas, n se décompose en un produit $p_1 p_2 \cdots p_s$ de plus de 3 facteurs premiers distincts (d'après l'exercice 310). Par ailleurs,

$$\forall a \in (\mathbb{Z}/n\mathbb{Z})^\times, \quad \left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right) \left(\frac{a}{p_2}\right) \cdots \left(\frac{a}{p_s}\right)$$

Notons α_i la classe de a dans \mathbb{F}_{p_i} . Nous avons notamment

$$\left(\frac{a}{n}\right) = \alpha_1^{(n-1)/2} = \left(\frac{\alpha_1}{p_1}\right) \left(\frac{\alpha_2}{p_2}\right) \cdots \left(\frac{\alpha_s}{p_s}\right) \pmod{p_1}$$

qui doit être vrai pour tout $(\alpha_i)_{i \leq s} \subseteq \prod_{i=1}^s \mathbb{F}_{p_i}^\times$. Mais, puisque $r \geq 2$, on peut falsifier à loisir cette égalité une fois α_1 choisi.

L'ensemble

$$\left\{ a \in (\mathbb{Z}/n\mathbb{Z})^\times; \left(\frac{a}{n}\right) = a^{(n-1)/2} \right\}$$

est le noyau d'un morphisme non constant, donc il contient au plus $\phi(n)/2$ éléments, ce qui prouve le corollaire.

Solution 329. On trouve : (45, 19), (45, 26), (65, 14), (65, 51), (85, 16), (85, 69), (105, 8), (105, 13), (105, 41), (105, 64), (105, 92), (105, 97), (117, 53), (117, 64), (145, 59), (145, 86).

Solution 330. 1. On a (en résolvant un système 2×2 à la main, ou avec SageMath pour les plus incapables)

$$\forall k \in \mathbb{N}, \quad u_k = \frac{\rho^k - \sigma^k}{\rho - \sigma} \quad \text{et} \quad v_k = \rho^k + \sigma^k.$$

2. On exploite les formules ci-dessus :

$$u_k v_k = \frac{(\rho^k - \sigma^k)(\rho^k + \sigma^k)}{\rho - \sigma} = \frac{\rho^{2k} - \sigma^{2k}}{\rho - \sigma} = u_{2k},$$

$$v_k^2 - 2q^2 = \rho^{2k} + 2\rho^k \sigma^k + \sigma^{2k} - 2q^2 = \rho^{2k} + \sigma^{2k} = v_{2k},$$

etc.

Solution 334. 1. Il vaut mieux faire le calcul $d^2 \leq B$ qui est conceptuellement plus simple et évite de recourir à des flottants.

2.

```
def crible(B):
    Premiers = []
    T = [True]*B
    d = 1
    while d^2 <= B:
        d = d+1
        Premiers.append(d)
        if T[d]:
            for i in range(2*d, B, d):
                T[i]=false
    return [i for i in range(2, B) if T[i]]

crible(50)
[i for i in range(2, B) if is_prime(i)]
```

Solution 340. Si F_n est premier, $3^{(F_n-1)/2}$ est le symbole de Legendre de 3. Par réciprocité quadratique et comme $F_n \equiv (-1)^{2^n} + 1 = 2 \pmod{3}$, on a comme voulu

$$3^{(F_n-1)/2} = \left(\frac{3}{F_n}\right) = \left(\frac{F_n}{3}\right) = \left(\frac{2}{3}\right) = -1 \pmod{F_n}$$

Réciproquement, si $3^{(F_n-1)/2} \equiv -1 \pmod{F_n}$, alors on peut appliquer le critère de Lehmer-Pocklington avec $a_2 = 3$ (2 est le seul diviseur premier de $F_n - 1$).

Solution 345. $\mathcal{L}(0, c)$: algorithme polynomial, $\mathcal{L}(0, 1)$ algorithme linéaire, $\mathcal{L}(0, 2)$ algorithme quadratique, $\mathcal{L}(1, c)$ algorithme exponentiel.

Solution 346. C'est une compétition qui vise à la factorisation d'un produit de deux grands nombres premiers organisée par la société RSA Security entre mars 1991 et mai 2007. Le plus grand nombre factorisé (en déc 2015) est

RSA-768 = 12301866845301177551304949583849627207 07263657518745202199786469389956474942 50791702612214291346167042921431160222 6902143413 = 334780716989568987860441698482126908177 878002287614711652531743087737814467999489 x 367460436667995904282446337996279526322 511279233373417143396810270092798736308917	728535695953347921973224521517264005 774063845925192557326303453731548268 124047927473779408066535141959745985 704794983713768568912431388982883793 279158164343087642676032283815739666
--	--

qui comporte 768 bits (écriture en binaire).

Solution 348. 1.

- La ligne $d \leftarrow d + 1$ nous amène à tester des diviseurs pairs de n alors que à ce stage, n est impair. Il serait plus judicieux de ne tester que des diviseurs impairs et faire $d \leftarrow d + 2$.
- On commence par tester la divisibilité par p_1, p_2, \dots, p_k . À ce stade, n n'est plus divisible par p_1, p_2, \dots ou p_k . Il est donc inutile de tester un diviseurs d qui ne soit pas premier avec tous ces nombres. Soient a_1, \dots, a_ϕ les $\varphi(p_1 p_2 \cdots p_k)$ inversibles modulo $p_1 p_2 \cdots p_k$ (triés par ordre croissant et pris entre 0 et $n - 1$). On note $\delta_i = a_{i+1} - a_i$ (pour $i < \phi$) et $\delta_\phi = a_1 - a_\phi + n$. Notre nouvel algorithme initialise la boucle avec $d = a_{i^*}$ (où a_{i^*} est le plus petit des a_i supérieur à p_k), puis il incrémente successivement d de $\delta_{i^*}, \delta_{i^*+1}, \dots, \delta_\phi, \delta_1, \dots, \delta_{i^*-1}, \delta_{i^*}$, etc.

On teste donc une proportion

$$\frac{\varphi(p_1 p_2 \cdots p_k)}{p_1 p_2 \cdots p_k} = \prod_i \left(1 - \frac{1}{p_i}\right)$$

de tous les diviseurs.

- Pour $k = 2$, on commence par éliminer les occurrences du facteur 2 puis les occurrences du facteur 3 de n . On obtient un nouvel entier (désormais premier avec 6). Les seuls diviseurs restants d de n vérifient $d \equiv 1 \pmod{6}$ ou bien $d \equiv 5 \pmod{6}$ (on a $a_1 = 1$ et $a_2 = 5$). On applique le même algorithme de divisions successives en partant de $d = 5$ et en modifiant d alternativement avec $d \leftarrow d + 2$ et $d \leftarrow d + 4$ (puisque ici $\delta_1 = 4$ et $\delta_2 = 2$).

Solution 351. Voici un pseudo-code :

Algorithme 45 : Divisions successives depuis une liste**Entrées** : Entier n , suite P de nombres premiers**Sorties** : Factorisation de n

```

1  $L \leftarrow \emptyset$ 
2 pour  $p \in P$  faire
3   tant que  $p$  divise  $n$  faire
4      $n \leftarrow n/p$ 
5      $L \leftarrow L \cup \{p\}$ 
6 si  $n = 1$  alors
7   retourner  $L$ 
8 sinon
9   retourner  $n$  n'est pas friable

```

Solution 353. Pour tout $k \geq 1$,

$$\begin{aligned}
 \mathbb{P}[s \geq k] &= \prod_{i=0}^{k-1} (1 - i/p) \\
 &\leq \prod_{i=0}^{k-1} e^{-i/p} \\
 &\leq e^{-\frac{(k-1)^2}{2p}}.
 \end{aligned}$$

Or

$$\mathbb{E}[s] = \sum_{k=0}^{\infty} \mathbb{P}[s \geq k].$$

Donc

$$\begin{aligned}
 \mathbb{E}[s] &\leq 1 + \sum_{k=1}^{\infty} e^{-\frac{(k-1)^2}{2p}} \\
 &\leq 2 + \sqrt{2p} \int_0^{\infty} e^{-x^2} dx. \\
 &\leq 2 + \frac{\sqrt{2\pi p}}{2}
 \end{aligned}$$

Solution 360. On peut faire

```

b=30
L = [p for p in primes(b+1,7*b) if max((p-1).prime_divisors()) >= b]
print L
p=L[0]
q=L[1]
print factor(p-1), factor(q-1)
n=p*q
Pollard(n,b)

```

Solution 362. 1.(a) Comme p est premier, $a^{(p-1)/2} = \left(\frac{a}{p}\right) = 1$

- (b) $(p+1)/2$ est pair si $p \equiv 3 \pmod{4}$, ce qui permet de définir $b = a^{(p+1)/4}$. Par la question précédente, $b^2 = a^{(p-1)/2+1} = a$ comme voulu.
- 2.(a) D'une part, $D^{2^s} = d^{p-1} = 1$, donc l'ordre de D est un diviseur de 2^s . Mais $\left(\frac{d}{p}\right) = -1$, donc $D^{2^{s-1}} = d^{(p-1)/2} = -1$. Donc D est d'ordre 2^s exactement. D^{-1} est du même ordre.
- (b) Comme $A^{2^{s-1}} = a^{(p-1)/2} = 1$, A est d'ordre divisant 2^{s-1} . Comme le groupe \mathbb{F}_p^\times est cyclique d'ordre $2^s t$, nous venons de montrer que D est un générateur du sous-groupe de 2-torsion et que A appartient à ce groupe. Il s'en suit que A est une puissance de D^{-1} . De plus, pour des raisons d'ordre, cette puissance doit être paire (sinon A et D auraient le même ordre).
- (c) Soit $b = a^{(t+1)/2} D^\mu$. Comme $b^2 = a A D^{-2\mu} = a$, b est bien une racine carrée de a .
3. Nous observons qu'à chaque entrée dans la boucle for, $(AD^m)^{2^{s-1-i}}$ est de carré 1, donc vaut ± 1 . De plus la boucle Si permet de s'assurer qu'à l'issue de chaque passage dans la boucle For, on a $(AD^m)^{2^{s-1-i}} = 1$. En fin de compte, quand i atteint s , on a bien $AD^m = 1$, donc $m = 2\mu$ comme souhaité.

Solution 368. Toute racine b telle que $b^2 = a$ dans $\mathbb{Z}/p^\alpha\mathbb{Z}$ est aussi une racine de a dans $\mathbb{Z}/p\mathbb{Z}$. Or dans $\mathbb{Z}/p\mathbb{Z}$ il n'y a que 2 racines possibles lorsque a est un carré, 1 si a est nul et 0 si a n'est pas un carré. De plus, toute racine se relève de manière unique dans $\mathbb{Z}/p^i\mathbb{Z}$ pour $i \geq 2$ (par le lemme d'Hensel). Donc a possède 2 racines quand $\left(\frac{a}{p}\right) = 1$.

Solution 377. 1. Soit $n = p_1^{\alpha_1} \cdots p_s^{\alpha_s}$ la factorisation de n (2 n'apparaît pas). Par le lemme chinois, $a^2 = 1 \pmod{n}$ si et seulement si

$$\begin{cases} a^2 \equiv 1 \pmod{p_1^{\alpha_1}} \\ \vdots \\ a^2 \equiv 1 \pmod{p_s^{\alpha_s}} \end{cases} \Leftrightarrow \begin{cases} a \equiv \pm 1 \pmod{p_1^{\alpha_1}} \\ \vdots \\ a \equiv \pm 1 \pmod{p_s^{\alpha_s}} \end{cases}$$

car $(\mathbb{Z}/p_i^{\alpha_i}\mathbb{Z})^\times$ est cyclique (quand p_i est impair). Il y a k signes à choisir, donc 2^k solutions.

2. Soit I l'ensemble des indices i tels que $a \equiv 1 \pmod{p_i^{\alpha_i}}$. Alors

$$g = (a-1) \wedge n = \prod_{i \in I} p_i^{\alpha_i}.$$

L'hypothèse $a \not\equiv \pm 1 \pmod{n}$ revient à dire que I n'est ni vide ni égal à $\llbracket 1, s \rrbracket$, donc g est un diviseur propre de n .

3. La probabilité est alors $1 - 2^{-k+1}$ compte tenu de ce qui précède.

Solution 379. 1. On sait que

$$x^2 - n = p_1^{e_{1,x}} p_2^{e_{2,x}} \cdots p_m^{e_{m,x}}$$

où tous les termes sont explicites. Mais alors

$$\begin{aligned} \prod_{x \in K} x^2 - n &= \prod_{x \in K} p_1^{e_{1,x}} p_2^{e_{2,x}} \cdots p_m^{e_{m,x}} \\ &= p_1^{\sum_{x \in K} e_{1,x}} p_2^{\sum_{x \in K} e_{2,x}} \cdots p_m^{\sum_{x \in K} e_{m,x}} \end{aligned}$$

et

$$\sqrt{\prod_{x \in K} x^2 - n} = p_1^{\frac{1}{2} \sum_{x \in K} e_{1,x}} p_2^{\frac{1}{2} \sum_{x \in K} e_{2,x}} \cdots p_m^{\frac{1}{2} \sum_{x \in K} e_{m,x}}$$

2.

Solution 385. 1. On a $496125 = 3^4 \cdot 5^3 \cdot 7^2$, donc

$$\begin{aligned} M &= \mathbb{Z}/496125\mathbb{Z} \\ &= \mathbb{Z}/3^4\mathbb{Z} \oplus \mathbb{Z}/5^3\mathbb{Z} \oplus \mathbb{Z}/7^2\mathbb{Z} \\ &= \mathbb{Z}/5^3\mathbb{Z} \oplus \mathbb{Z}/3^4\mathbb{Z} \oplus \mathbb{Z}/7^2\mathbb{Z} \\ &= \mathbb{Z}/7^2\mathbb{Z} \oplus \mathbb{Z}/3^4\mathbb{Z} \oplus \mathbb{Z}/5^3\mathbb{Z}. \end{aligned}$$

2. On a $x^3 + x^2 + x + 1 = (x+1)^3$, donc une telle décomposition n'est pas possible.

3. On a $x^4 + 3x^3 + 4x + 1 = (x-2)(x-3)(x^2 + x - 1)$, donc

$$\begin{aligned} M &= A/(x-2)A \oplus A/(x-3)A \oplus A/(x^2 + x - 1)A \\ &= A/(x-3)A \oplus A/(x-2)A \oplus A/(x^2 + x - 1)A \\ &= A/((x-2)(x-3))A \oplus A/(x^2 + x - 1)A \end{aligned}$$

Solution 392. Comme $6436343 = 23^5$, il suffit de passer en revue les types possibles

1. (5) : $\mathbb{Z}/6436343\mathbb{Z}$
2. (1, 4) : $\mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/279841\mathbb{Z}$
3. (2, 3) : $\mathbb{Z}/529\mathbb{Z} \oplus \mathbb{Z}/12167\mathbb{Z}$
4. (1, 1, 3) : $\mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/12167\mathbb{Z}$
5. (1, 2, 2) : $\mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/529\mathbb{Z} \oplus \mathbb{Z}/529\mathbb{Z}$
6. (1, 1, 1, 2) : $\mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/529\mathbb{Z}$
7. (1, 1, 1, 1, 1) : $\mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/23\mathbb{Z} \oplus \mathbb{Z}/23\mathbb{Z}$

Solution 393. On commence par identifier les composantes primaires.

Comme $360 = 2^3 \cdot 3^2 \cdot 5$, il y en a 3.

1. Composante 2-primaire : les types possibles sont (3), (1, 2) ou (1, 1, 1), ce qui donne $\mathbb{Z}/8\mathbb{Z}$, $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/4\mathbb{Z}$ ou $(\mathbb{Z}/2\mathbb{Z})^3$.

2. Composante 3-primaire : les types possibles sont (2) , $(1,1)$, ce qui donne $\mathbb{Z}/9\mathbb{Z}$ ou $\mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z}$.
3. Composante 5-primaire : le seul type possible est (1) , ce qui donne $\mathbb{Z}/5\mathbb{Z}$.

On a donc 6 \mathbb{Z} -modules possibles (en combinant les composantes primaires possibles). De plus, on peut transformer les expressions obtenues à l'aide du lemme chinois pour faire apparaître les facteurs invariants.

1. $\mathbb{Z}/8\mathbb{Z} \oplus \mathbb{Z}/9\mathbb{Z} \oplus \mathbb{Z}/5\mathbb{Z} = \mathbb{Z}/360\mathbb{Z}$, dont les facteurs invariants sont (360) .
2. $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/4\mathbb{Z} \oplus \mathbb{Z}/9\mathbb{Z} \oplus \mathbb{Z}/5\mathbb{Z} = \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/180\mathbb{Z}$, dont les facteurs invariants sont $(2, 180)$.
3. $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/9\mathbb{Z} \oplus \mathbb{Z}/5\mathbb{Z} = \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/90\mathbb{Z}$, dont les facteurs invariants sont $(2, 2, 90)$.
4. $\mathbb{Z}/8\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/5\mathbb{Z} = \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/120\mathbb{Z}$, dont les facteurs invariants sont $(3, 120)$.
5. $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/4\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/5\mathbb{Z} = \mathbb{Z}/6\mathbb{Z} \oplus \mathbb{Z}/60\mathbb{Z}$, dont les facteurs invariants sont $(6, 60)$.
6. $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/3\mathbb{Z} \oplus \mathbb{Z}/5\mathbb{Z} = \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/6\mathbb{Z} \oplus \mathbb{Z}/30\mathbb{Z}$, dont les facteurs invariants sont $(2, 6, 30)$.

Solution 394. On peut compter les modules en fonction du type de chaque décomposition primaire. Pour un nombre premier p_i donné, le type de la composante p_i -primaire a pour somme α_i . Il y a donc $p(\alpha_i)$ types possibles. Au total, on a donc

$$\prod_i p(\alpha_i)$$

modules possibles.

Solution 402. De manière générale, on a un morphisme surjectif de \mathbb{k} -algèbres :

$$\mathbb{k}[x] \rightarrow \mathbb{k}[T]$$

dont le noyau est par définition l'idéal de $\mathbb{k}[x]$ engendré par $\pi_T(x)$. Donc

$$\mathbb{k}[x] / \langle \pi_T(x) \rangle \rightarrow \mathbb{k}[T]$$

est un isomorphisme de \mathbb{k} -algèbres.

— 1. \Rightarrow 2. Soit $f(T) \in \mathbb{k}[T]$ un élément non nul. On considère l'application

$$\psi_f : \begin{cases} \mathbb{k}[T] & \rightarrow & \mathbb{k}[T] \\ g(T) & \mapsto & f(T)g(T) \end{cases}$$

Comme $\mathbb{k}[T]$ est intègre, son noyau est réduit à $\{0\}$. Mais ψ_f est un endomorphisme de \mathbb{k} -espace vectoriel de dimension finie, donc ψ_f est aussi surjectif et 1 admet un antécédant qui est l'inverse de $f(T)$. Donc $\mathbb{k}[T]$ est un corps.

- 2. \Rightarrow 3. D'après la remarque initiale, pour que $\mathbb{k}[T]$ soit un corps, il faut (et il suffit) que $\pi_T(x)$ soit irréductible.
- 3. \Rightarrow 1. Nous raisonnons par la contraposée. Supposons qu'il existe $f(x)$ et $g(x) \in \mathbb{k}[x]$ de degré $< \deg \pi_T(x)$ tels que $f(T)$ et $g(T)$ sont non nuls mais $f(T)g(T)$ soit nul. Alors, $\pi_T(x)$ divise le produit $f(x)g(x)$. Mais par hypothèse, $\pi_T(x)$ ne divise pas $f(x)$ ou $g(x)$, donc $f \wedge \pi_T$ et $g \wedge \pi_T$ forment deux facteurs non triviaux de π_T et π_T n'est pas irréductible.

Solution 405. Nous nous servons de la proposition 404 pour raisonner.

1. Dans le premier cas, soit $z = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \in \mathbb{C}^3$, alors

$$z = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, u(z) = \begin{pmatrix} -1 \\ 1 \\ i \end{pmatrix}, u^2(z) = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$$

forme une famille C-libre de \mathbb{C}^3 , donc u est cyclique. On aurait aussi pu remarquer que u est diagonalisable, de polynôme minimal $\pi(x) = (x-1)(x+1)(x-i)$ et présenter u comme la matrice compagnon de $\pi(x)$.

2. Encore une fois, u est cyclique. En effet, soit $z = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \in \mathbb{C}^3$, alors

$$z = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, u(z) = \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix}, u^2(z) = \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$$

forme une famille C-libre de \mathbb{C}^3 , donc u est cyclique.

3. L'endomorphisme u n'est pas cyclique car il possède un sous-espace propre de dimension supérieure à 1, à savoir celui engendré par

$$\begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},$$

qui ne peut pas être cyclique.

Solution 409. Bien entendu il ne faut pas énumérer les $2^{4 \cdot 4} = 65536$ matrices possibles. Ce problème revient à déterminer l'ensemble des classes d'équivalence des matrices 4×4 sur le corps \mathbb{F}_2 pour la relation de similitude. Il nous suffit de trouver tous les invariants de similitudes possibles, autrement dit toutes les suites $(q_1, q_2, \dots, q_s) \subseteq \mathbb{F}_2[x]$ telles que $\deg q_1 + \dots + \deg q_s = 4$ et $q_1 | q_2, q_2 | q_3, \dots, q_{s-1} | q_s$ à des inversibles près.

PREMIÈRE MÉTHODE : Nous décrivons un algorithme de construction récursif. Soit L une liste d'invariants de similitude vide. On ajoute à L les mono-upplets de polynômes unitaires (q_1) non constants de degré ≤ 4 . Puis, de façon récursive, tant qu'il existe un s -upplet (q_1, \dots, q_s) tel que $\delta = \deg q_1 + \dots + \deg q_s < 4$ dans la liste L , pour tout polynôme unitaire (éventuellement constant) r de degré $4 - \delta - \deg q_s$, on ajoute le $s + 1$ -upplet $(q_1, \dots, q_s, q_s r)$ à L .

À la fin de l'algorithme, on ne conserve que les s -upplets tels que $\delta = \deg q_1 + \dots + \deg q_s = 4$.

On obtient le tableau suivant, dans lequel les solutions ont été passée en fonte grasse et où $a = (x + 1, x + 1, x + 1)$, $b = (x + 1, x + 1, x + 1, x + 1)$ et $c = (x + 1, x + 1, (x + 1)^2)$.

$s = 1$	$s \geq 2$		
	(x, x)	(x, x, x) $\mathbf{f}(x, x, x^2)$	$\mathbf{f}(x, x, x, x)$ \emptyset
(x)	$(x, x \cdot x)$		\emptyset
	$(x, x \cdot (x + 1))$		\emptyset
	$\mathbf{f}(x, x \cdot x^2)$		\emptyset
	$\mathbf{f}(x, x \cdot (x^2 + 1))$		\emptyset
	$\mathbf{f}(x, x \cdot (x^2 + x))$		\emptyset
	$\mathbf{f}(x, x \cdot (x^2 + x + 1))$		\emptyset
$(x + 1)$	$(x + 1, x + 1)$	a $\mathbf{f}c$	$\mathbf{f}b$ \emptyset
	$(x + 1, x^2 + x)$		\emptyset
	$(x + 1, x^2 + 1)$		\emptyset
	$\mathbf{f}(x + 1, (x + 1) \cdot x^2)$		\emptyset
	$\mathbf{f}(x + 1, (x + 1) \cdot (x^2 + 1))$		\emptyset
	$\mathbf{f}(x + 1, (x + 1) \cdot (x^2 + x))$		\emptyset
	$\mathbf{f}(x + 1, (x + 1) \cdot (x^2 + x + 1))$		\emptyset
(x^2)	$\mathbf{f}(x^2, x^2)$		
$(x^2 + 1)$	$\mathbf{f}(x^2 + 1, x^2 + 1)$		
$(x^2 + x)$	$\mathbf{f}(x^2 + x, x^2 + x)$		
$(x^2 + x + 1)$	$\mathbf{f}(x^2 + x + 1, x^2 + x + 1)$		
$(x^3 + p)$ où $\deg p = 2$	\emptyset		
$\mathbf{f}(x^4 + p)$ où $\deg p = 3$	\emptyset		

SECONDE MÉTHODE : Nous nous ramenons à des facteurs pre-

miers. Commençons par trouver la liste des polynômes irréductibles de degré ≤ 4 . On peut les obtenir soit en énumérant tous les polynômes ou bien en factorisant $x^{2^k-1} - 1$ pour $k \leq 4$. Comme $2^1 - 1 = 1$ et $2^3 - 1 = 3$ divisent $2^{15} - 1$, il suffit de les chercher parmi les facteurs de $x^7 - 1$ et de $x^{15} - 1$. On obtient (après étude) et classés par degré les polynômes suivants :

$$p_1 = x, \quad p_2 = x + 1,$$

$$p_3 = x^2 + x + 1,$$

$$p_4 = x^3 + x + 1, \quad p_5 = x^3 + x^2 + 1$$

$$p_6 = x^4 + x + 1, \quad p_7 = x^4 + x^3 + 1, \quad p_8 = x^4 + x^3 + x^2 + x + 1.$$

Les suites possibles d'invariants de similitudes sont donc

1. Si $s = 1$

$$(p_6), (p_7), (p_8),$$

$$(p_4 p_1), (p_4 p_1), (p_5 p_1), (p_5 p_2)$$

$$(p_3^2), (p_3 p_1^2), (p_3 p_1 p_2), (p_3 p_2^2),$$

$$(p_1^4), (p_1^3 p_2), (p_1^2 p_2^2), (p_1 p_2^3), (p_2^4),$$

2. Si $s = 2$, avec $\deg q_1 = 1$ et $\deg q_2 = 3$

$$(p_1, p_1 p_3), (p_1, p_1^3), (p_1, p_1^2 p_2), (p_1, p_1 p_2^2)$$

$$(p_2, p_2 p_3), (p_2, p_2 p_1^2), (p_2, p_2^2 p_1), (p_2, p_2^3)$$

3. Si $s = 2$, avec $\deg q_1 = 2$ et $\deg q_2 = 2$

$$(p_3, p_3), (p_1^2, p_1^2), (p_1 p_2, p_1 p_2), (p_2^2)$$

4. Si $s = 3$, avec $\deg q_1 = 1$, $\deg q_2 = 1$ et $\deg q_3 = 2$

$$(p_1, p_1, p_1^2), (p_1, p_1, p_1 p_2), (p_2, p_2, p_1 p_2), (p_2, p_2, p_2^2),$$

5. Si $s = 4$, avec $\deg q_1 = \deg q_2 = \deg q_3 = \deg q_4 = 1$

$$(p_1, p_1, p_1, p_1), (p_2, p_2, p_2, p_2).$$

Nous pouvons alors donner une matrice dans chaque classe d'équivalence en utilisant la matrice compagnon des polynômes.

1. Si $s = 1$, la forme générale est

$$\begin{pmatrix} 0 & 0 & 0 & * \\ 1 & 0 & 0 & * \\ 0 & 1 & 0 & * \\ 0 & 0 & 1 & * \end{pmatrix}$$

Dans le détail on a :

$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

etc.

2. Si $s = 2$, avec $\deg q_1 = 1$ et $\deg q_2 = 3$, la forme générale est

$$\left(\begin{array}{c|ccc} * & & 0 & \\ \hline & 0 & 0 & * \\ 0 & 1 & 0 & * \\ & 0 & 1 & * \end{array} \right).$$

Dans le détail on a :

$$\left(\begin{array}{c|ccc} 0 & & 0 & \\ \hline & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ & 0 & 1 & 1 \end{array} \right), \left(\begin{array}{c|ccc} 0 & & 0 & \\ \hline & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ & 0 & 1 & 0 \end{array} \right), \left(\begin{array}{c|ccc} 0 & & 0 & \\ \hline & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ & 0 & 1 & 1 \end{array} \right), \left(\begin{array}{c|ccc} 0 & & 0 & \\ \hline & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ & 0 & 1 & 0 \end{array} \right)$$

$$\left(\begin{array}{c|ccc} 1 & & 0 & \\ \hline & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ & 0 & 1 & 0 \end{array} \right), \left(\begin{array}{c|ccc} 1 & & 0 & \\ \hline & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ & 0 & 1 & 1 \end{array} \right), \left(\begin{array}{c|ccc} 1 & & 0 & \\ \hline & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ & 0 & 1 & 0 \end{array} \right), \left(\begin{array}{c|ccc} 1 & & 0 & \\ \hline & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ & 0 & 1 & 1 \end{array} \right)$$

3. Si $s = 2$, avec $\deg q_1 = 2$ et $\deg q_2 = 2$, la forme générale est

$$\left(\begin{array}{cc|c} 0 & * & \\ 1 & * & 0 \\ \hline & 0 & 0 \\ 0 & & 1 \end{array} \right)$$

Dans le détail on a :

$$\left(\begin{array}{cc|c} 0 & 1 & \\ 1 & 1 & 0 \\ \hline & 0 & 1 \\ 0 & & 1 \end{array} \right) \left(\begin{array}{cc|c} 0 & 0 & \\ 1 & 0 & 0 \\ \hline & 0 & 0 \\ 0 & & 1 \end{array} \right) \left(\begin{array}{cc|c} 0 & 0 & \\ 1 & 1 & 0 \\ \hline & 0 & 0 \\ 0 & & 1 \end{array} \right) \left(\begin{array}{cc|c} 0 & 1 & \\ 1 & 0 & 0 \\ \hline & 0 & 1 \\ 0 & & 1 \end{array} \right)$$

4. Si $s = 3$, avec $\deg q_1 = 1$, $\deg q_2 = 1$ et $\deg q_3 = 2$, la forme générale est

$$\left(\begin{array}{c|cc} * & 0 & 0 \\ \hline 0 & * & 0 \\ \hline 0 & 0 & 0 \\ & 1 & * \end{array} \right)$$

Dans le détail on a :

$$\left(\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array} \right), \left(\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{array} \right), \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{array} \right), \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{array} \right).$$

5. Si $s = 4$, avec $\deg q_1 = \deg q_2 = \deg q_3 = \deg q_4 = 1$. la forme générale est

$$\left(\begin{array}{cc|cc} * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ \hline 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{array} \right)$$

Dans le détail on a :

$$\left(\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right), \left(\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array} \right).$$

Solution 410. On reconnait dans chaque cas des blocs diagonaux sous la forme de matrices compagnons.

1. Pour la matrice A , il s'agit des polynômes $x^3 + x^2 - 6x = x(x-2)(x+3)$, $x+1$ et $x^3 - 4x^2 + 4x = x(x-2)^2$. À l'aide du lemme chinois, on peut écrire que \mathbb{Q}^7 en tant que $\mathbb{Q}[x]$ -module a pour structure

$$\begin{aligned} & \mathbb{Q}[x]/x(x-2)(x+3) \oplus \mathbb{Q}[x]/(x+1) \oplus \mathbb{Q}[x]/x(x-2)^2 \\ &= \mathbb{Q}[x]/x(x-2) \oplus \mathbb{Q}[x]/x(x-2)^2(x+1)(x+3) \end{aligned}$$

Les invariants de similitudes de A sont

$$[x(x-2), x(x-2)^2(x+1)(x+3)].$$

2. Pour la matrice B , il s'agit des polynômes $x^2 - 2x = x(x-2)$, $x^2 + x - 6 = (x-2)(x+3)$, $x^3 - x^2 - 2x = x(x-2)(x+1)$. À l'aide du lemme chinois, on peut écrire que \mathbb{Q}^7 en tant que $\mathbb{Q}[x]$ -module a pour structure

$$\begin{aligned} & \mathbb{Q}[x]/x(x-2) \oplus \mathbb{Q}[x]/(x-2)(x+3) \oplus \mathbb{Q}[x]/x(x-2)(x+1) \\ &= \mathbb{Q}[x]/(x-2) \oplus \mathbb{Q}[x]/x(x-2) \oplus \mathbb{Q}[x]/x(x-2)(x+1)(x+3) \end{aligned}$$

Les invariants de similitudes de B sont

$$[(x-2), x(x-2), x(x-2)(x+1)(x+3)].$$

3. Pour la matrice C , il s'agit des polynômes $x^4 - 3x^3 + 4x = x(x - 2)^2(x + 1)$, $x, x - 2, x + 3$. À l'aide du lemme chinois, on peut écrire que \mathbb{Q}^7 en tant que $\mathbb{Q}[x]$ -module a pour structure

$$\begin{aligned} & \mathbb{Q}[x]/x(x-2)^2(x+1) \oplus \mathbb{Q}[x]/x \oplus \mathbb{Q}[x]/x-2 \oplus \mathbb{Q}[x]/x+3 \\ &= \mathbb{Q}[x]/x(x-2) \oplus \mathbb{Q}[x]/x(x-2)^2(x+1)(x+3) \end{aligned}$$

Les invariants de similitudes de C sont

$$\left[x(x-2), x(x-2)^2(x+1)(x+3) \right].$$

Nous en déduisons que A et C sont semblables, mais pas B .

Solution 411. Les matrices sont semblables si et seulement si elles ont les mêmes invariants de similitude. Le polynôme caractéristique $\chi(x)$ est de plus le produit des invariants de similitude. Le dernier polynôme de la suite des invariants de similitude doit contenir au moins tous les facteurs irréductibles de $\chi(x)$.

Si le polynôme caractéristique $\chi(x)$ ne possède que des facteurs irréductibles avec multiplicité 1, alors la seule suite d'invariants de similitude est (χ) .

Par contre, dans le cas contraire, on peut décomposer χ en $\chi = \chi_1\chi_2$ avec $\chi_1 \mid \chi_2$ et alors il y a au moins deux suites d'invariants possibles : à savoir (χ) ou (χ_1, χ_2) .

Solution 412. Il suffit de comparer les invariants de similitudes de nos deux matrices, sachant que le polynôme minimal $\pi(x)$ est le dernier terme de la suite des invariants et que le polynôme caractéristique $\chi(x)$ est le produit des invariants.

1. Pour des matrices 2×2 , il n'y a que deux possibilités comme invariants de similitude :
 Si $\deg \pi = 1$, on doit avoir (π, π) comme invariant de similitude.
 Si $\deg \pi = 2$, on doit avoir (π) comme invariant de similitude.
2. Pour des matrices 3×3 : les possibilités sont les suivantes.
 Si $\deg \pi = 1$, on doit avoir (π, π, π) comme invariant de similitude.
 Si $\deg \pi = 2$, on doit avoir $(\chi/\pi, \pi)$ comme invariant de similitude.
 Si $\deg \pi = 3$, on doit avoir (π) comme invariant de similitude.

Solution 417. Le module est cyclique ssi les invariants de similitudes de u sont réduits à un seul polynôme. Or d'après la remarque 407, dans ce cas, le polynôme caractéristique de u coïncide avec le polynôme minimal de u .

Solution 418. On commence par rechercher les composantes primaires. Il y en a deux : la composante p -primaire et la composante q -primaire avec $p = x - 7$ et $q = x + 11$. Concernant la première, elle peut être de deux formes

1. Si $E(p) = (A/pA)^2 = \mathbb{K}[x]/\langle p \rangle \oplus \mathbb{K}[x]/\langle p \rangle$, alors le bloc de Jordan est

$$B_1 = \begin{pmatrix} 7 & 0 \\ 0 & 7 \end{pmatrix}.$$

Notons que le polynôme minimal est $\pi(B_1) = x - 7$ et que les invariants de similitudes de B_1 sont (p, p) .

2. Si $E(p) = (A/p^2A)^2 = \mathbb{K}[x]/\langle p^2 \rangle$, alors le bloc de Jordan est

$$B_2 = \begin{pmatrix} 7 & 1 \\ 0 & 7 \end{pmatrix}.$$

Notons que le polynôme minimal est $\pi(B_2) = (x - 7)^2$ et que les invariants de similitudes de B_2 sont (p^2) .

Concernant la seconde, elle peut être de trois formes

1. Si $E(q) = (A/qA)^3 = \mathbb{K}[x]/\langle q \rangle \oplus \mathbb{K}[x]/\langle q \rangle \oplus \mathbb{K}[x]/\langle q \rangle$, alors le bloc de Jordan est

$$C_1 = \begin{pmatrix} -11 & 0 & 0 \\ 0 & -11 & 0 \\ 0 & 0 & -11 \end{pmatrix}.$$

Notons que le polynôme minimal est $\pi(C_1) = q = x + 11$ et que les invariants de similitudes de C_1 sont (p, p, p) .

2. Si $E(q) = (A/q^2A) \oplus (A/qA) = \mathbb{K}[x]/\langle q^2 \rangle \oplus \mathbb{K}[x]/\langle q \rangle$, alors le bloc de Jordan est

$$C_2 = \begin{pmatrix} -11 & 1 & 0 \\ 0 & -11 & 0 \\ 0 & 0 & -11 \end{pmatrix}.$$

Notons que le polynôme minimal est $\pi(C_2) = q^2 = (x + 11)^2$ et que les invariants de similitudes de C_2 sont (p, p^2) .

3. Si $E(q) = (A/q^3A) = \mathbb{K}[x]/\langle q^3 \rangle$, alors le bloc de Jordan est

$$C_3 = \begin{pmatrix} -11 & 1 & 0 \\ 0 & -11 & 1 \\ 0 & 0 & -11 \end{pmatrix}.$$

Notons que le polynôme minimal est $\pi(C_3) = q = (x + 11)^3$ et que les invariants de similitudes de C_3 sont (p^3) .

Nous sommes parés pour lister les blocs de Jordan possibles :

1. Si $E = (A/pA)^2 \oplus (A/qA)^3$, alors

$$A_1 = \left(\begin{array}{cc|ccc} 7 & 0 & & & & \\ 0 & 7 & & & & \\ \hline & & 0 & & & \\ 0 & & -11 & 0 & 0 & \\ & & 0 & -11 & 0 & \\ & & 0 & 0 & -11 & \end{array} \right)$$

De plus, en utilisant le lemme chinois (comme p et q sont premiers entre eux), on peut vérifier que $E = (A/qA) \oplus (A/pqA) \oplus (A/pqA)$. Donc les invariants de similitude de A_1 sont (q, pq, pq) et le polynôme minimal est pq .

2. Si $E = (A/pA)^2 \oplus (A/q^2A) \oplus (A/qA)$, alors

$$A_2 = \left(\begin{array}{cc|cc|c} 7 & 0 & & & 0 \\ 0 & 7 & & & \\ \hline & & 0 & & \\ 0 & & -11 & 1 & \\ & & 0 & -11 & 0 \\ \hline 0 & & 0 & & -11 \end{array} \right)$$

De plus, en utilisant le lemme chinois (comme p et q sont premiers entre eux), on peut vérifier que $E = (A/pqA) \oplus (A/pq^2A)$. Donc les invariants de similitude de A_2 sont (pq, pq^2) et le polynôme minimal est pq^2 .

3. Si $E = (A/pA)^2 \oplus (A/q^3A)$, alors

$$A_3 = \left(\begin{array}{cc|cccc} 7 & 0 & & & & \\ 0 & 7 & & & & \\ \hline & & 0 & & & \\ 0 & & -11 & 1 & 0 & \\ & & 0 & -11 & 1 & \\ & & 0 & 0 & -11 & \end{array} \right)$$

De plus, en utilisant le lemme chinois (comme p et q sont premiers entre eux), on peut vérifier que $E = (A/pA) \oplus (A/pq^3A)$. Donc les invariants de similitude de A_3 sont (p, pq^3) et le polynôme minimal est pq^3 .

4. Si $E = (A/p^2A) \oplus (A/qA)^3$, alors

$$A_4 = \left(\begin{array}{cc|ccc} 7 & 1 & & & \\ 0 & 7 & & & \\ \hline & & 0 & & \\ 0 & & -11 & 0 & 0 \\ & & 0 & -11 & 0 \\ & & 0 & 0 & -11 \end{array} \right)$$

De plus, en utilisant le lemme chinois (comme p et q sont premiers entre eux), on peut vérifier que $E = (A/qA) \oplus (A/qA) \oplus (A/p^2qA)$. Donc les invariants de similitude de A_4 sont (q, q, p^2q) et le polynôme minimal est p^2q .

5. Si $E = (A/p^2A) \oplus (A/q^2A) \oplus (A/qA)$, alors

$$A_5 = \left(\begin{array}{cc|cc|c} 7 & 1 & & & \\ 0 & 7 & & & \\ \hline & & 0 & & \\ 0 & & -11 & 1 & \\ & & 0 & -11 & \\ \hline 0 & & 0 & & -11 \end{array} \right)$$

De plus, en utilisant le lemme chinois (comme p et q sont premiers entre eux), on peut vérifier que $E = (A/qA) \oplus (A/p^2q^2A)$. Donc les invariants de similitude de A_5 sont (q, p^2q^2) et le polynôme minimal est p^2q^2 .

6. Si $E = (A/p^2A) \oplus (A/q^3A)$, alors

$$A_3 = \left(\begin{array}{cc|ccc} 7 & 1 & & & \\ 0 & 7 & & & \\ \hline & & 0 & & \\ 0 & & -11 & 1 & 0 \\ & & 0 & -11 & 1 \\ & & 0 & 0 & -11 \end{array} \right)$$

De plus, en utilisant le lemme chinois (comme p et q sont premiers entre eux), on peut vérifier que $E = (A/p^2q^3A)$. Donc les invariants de similitude de A_6 sont (p^2q^3) et le polynôme minimal est p^2q^3 .

Solution 419. Les matrices solutions possèdent nécessairement les blocs

$$\begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

Les trois dimensions restantes peuvent être occupés par des blocs de Jordan élémentaires de valeur propre 0 et de taille ≤ 2 ou de valeur propre 1 et de taille ≤ 3 . On a donc les possibilités suivantes :

$$A_1 = \left(\begin{array}{cc|ccc|ccc} 0 & 1 & & & & 0 & 0 & 0 & 0 \\ 0 & 0 & & & & & & & \\ \hline & & 1 & 1 & 0 & & & & \\ 0 & & 0 & 1 & 1 & 0 & 0 & 0 & \\ & & 0 & 0 & 1 & & & & \\ \hline 0 & & 0 & & & 0 & 0 & 0 & \\ \hline 0 & & 0 & & & 0 & 0 & 0 & \\ \hline 0 & & 0 & & & 0 & 0 & 0 & \end{array} \right)$$

Les invariants de similitude de A_1 sont $(x, x, x, x^2(x-1)^3)$.

$$A_2 = \left(\begin{array}{cc|ccc} 0 & 1 & & 0 & & 0 & 0 & 0 \\ 0 & 0 & & & & & & \\ \hline & & 1 & 1 & 0 & & & \\ 0 & & 0 & 1 & 1 & 0 & 0 & 0 \\ & & 0 & 0 & 1 & & & \\ \hline 0 & & 0 & & & 0 & 0 & 0 \\ \hline 0 & & 0 & & & 0 & 0 & 0 \\ \hline 0 & & 0 & & & 0 & 0 & 1 \end{array} \right)$$

Les invariants de similitude de A_2 sont $(x, x(x-1), x^2(x-1)^3)$.

$$A_3 = \left(\begin{array}{cc|ccc} 0 & 1 & & 0 & & 0 & 0 & 0 \\ 0 & 0 & & & & & & \\ \hline & & 1 & 1 & 0 & & & \\ 0 & & 0 & 1 & 1 & 0 & 0 & 0 \\ & & 0 & 0 & 1 & & & \\ \hline 0 & & 0 & & & 0 & 0 & 0 \\ \hline 0 & & 0 & & & 0 & 1 & 0 \\ \hline 0 & & 0 & & & 0 & 0 & 1 \end{array} \right)$$

Les invariants de similitude de A_3 sont $(x-1, x(x-1), x^2(x-1)^3)$.

$$A_4 = \left(\begin{array}{cc|ccc} 0 & 1 & & 0 & & 0 & 0 & 0 \\ 0 & 0 & & & & & & \\ \hline & & 1 & 1 & 0 & & & \\ 0 & & 0 & 1 & 1 & 0 & 0 & 0 \\ & & 0 & 0 & 1 & & & \\ \hline 0 & & 0 & & & 1 & 0 & 0 \\ \hline 0 & & 0 & & & 0 & 1 & 0 \\ \hline 0 & & 0 & & & 0 & 0 & 1 \end{array} \right)$$

Les invariants de similitude de A_4 sont $(x-1, x-1, x-1, x^2(x-1)^3)$.

$$B_1 = \left(\begin{array}{cc|ccc} 0 & 1 & & 0 & & 0 & 0 & \\ 0 & 0 & & & & & & \\ \hline & & 1 & 1 & 0 & & & \\ 0 & & 0 & 1 & 1 & 0 & 0 & \\ & & 0 & 0 & 1 & & & \\ \hline 0 & & 0 & & & 0 & 1 & 0 \\ & & & & & 0 & 0 & \\ \hline 0 & & 0 & & & 0 & 0 & \end{array} \right)$$

Les invariants de similitude de B_1 sont $(x, x^2, x^2(x-1)^3)$.

$$B_2 = \left(\begin{array}{cc|cc|cc} 0 & 1 & & & 0 & 0 \\ 0 & 0 & & & 0 & 0 \\ \hline & & 1 & 1 & 0 & \\ 0 & & 0 & 1 & 1 & 0 \\ & & 0 & 0 & 1 & \\ \hline 0 & & 0 & & 1 & 1 \\ & & & & 0 & 1 \\ \hline 0 & & 0 & & 0 & 0 \end{array} \right)$$

Les invariants de similitude de B_2 sont $(x(x-1)^2, x^2(x-1)^3)$.

$$B_3 = \left(\begin{array}{cc|cc|cc} 0 & 1 & & & 0 & 0 \\ 0 & 0 & & & 0 & 0 \\ \hline & & 1 & 1 & 0 & \\ 0 & & 0 & 1 & 1 & 0 \\ & & 0 & 0 & 1 & \\ \hline 0 & & 0 & & 0 & 1 \\ & & & & 0 & 0 \\ \hline 0 & & 0 & & 0 & 1 \end{array} \right)$$

Les invariants de similitude de B_3 sont $(x^2(x-1), x^2(x-1)^3)$.

$$B_4 = \left(\begin{array}{cc|cc|cc} 0 & 1 & & & 0 & 0 \\ 0 & 0 & & & 0 & 0 \\ \hline & & 1 & 1 & 0 & \\ 0 & & 0 & 1 & 1 & 0 \\ & & 0 & 0 & 1 & \\ \hline 0 & & 0 & & 1 & 1 \\ & & & & 0 & 1 \\ \hline 0 & & 0 & & 0 & 1 \end{array} \right)$$

Les invariants de similitude de B_4 sont $((x-1)^3, x^2(x-1)^3)$.

$$C_1 = \left(\begin{array}{cc|cc|cc} 0 & 1 & & & & \\ 0 & 0 & & & & \\ \hline & & 1 & 1 & 0 & \\ 0 & & 0 & 1 & 1 & \\ & & 0 & 0 & 1 & \\ \hline 0 & & 0 & & 1 & 1 & 0 \\ & & & & 0 & 1 & 1 \\ & & & & 0 & 0 & 1 \end{array} \right)$$

Les invariants de similitude de C_1 sont $((x-1)^3, x^2(x-1)^3)$.

Solution 420. Notons que α est une racine primitive 7-ième et β est une racine primitive 15-ième de l'unité. On commence par faire la liste

des racines de chaque polynôme pour connaître les valeurs propres de chaque matrice.

polynôme	racines
p_1	0
p_2	1
p_3	β^5, β^{10}
p_4	$\alpha, \alpha^2, \alpha^4$
p_5	$\alpha^3, \alpha^5, \alpha^6$
p_6	$\beta, \beta^2, \beta^4, \beta^8$
p_7	$\beta^7, \beta^{14}, \beta^{13}, \beta^{11}$
p_8	$\beta^3, \beta^6, \beta^{12}, \beta^9$

Les réduites de Jordan possibles sont donc (en précisant chaque fois les invariants de similitude)

$$\begin{array}{lll}
- (p_6) & - (p_7) & - (p_8) \\
\begin{pmatrix} \beta & 0 & 0 & 0 \\ 0 & \beta^2 & 0 & 0 \\ 0 & 0 & \beta^4 & 0 \\ 0 & 0 & 0 & \beta^8 \end{pmatrix} & \begin{pmatrix} \beta & 0 & 0 & 0 \\ 0 & \beta^{14} & 0 & 0 \\ 0 & 0 & \beta^{13} & 0 \\ 0 & 0 & 0 & \beta^{11} \end{pmatrix} & \begin{pmatrix} \beta^3 & 0 & 0 & 0 \\ 0 & \beta^6 & 0 & 0 \\ 0 & 0 & \beta^{12} & 0 \\ 0 & 0 & 0 & \beta^9 \end{pmatrix} \\
- (p_4 p_1) & - (p_4 p_2) & - (p_5 p_1) \\
\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \alpha & 0 & 0 \\ 0 & 0 & \alpha^2 & 0 \\ 0 & 0 & 0 & \alpha^4 \end{pmatrix} & \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \alpha & 0 & 0 \\ 0 & 0 & \alpha^2 & 0 \\ 0 & 0 & 0 & \alpha^4 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \alpha^3 & 0 & 0 \\ 0 & 0 & \alpha^6 & 0 \\ 0 & 0 & 0 & \alpha^5 \end{pmatrix} \\
- (p_5 p_2) & - (p_3^2) & - (p_3 p_1^2) \\
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \alpha^3 & 0 & 0 \\ 0 & 0 & \alpha^6 & 0 \\ 0 & 0 & 0 & \alpha^5 \end{pmatrix} & \left(\begin{array}{cc|cc} \beta^5 & 1 & & 0 \\ 0 & \beta^5 & & \\ \hline & & \beta^{10} & 1 \\ 0 & & 0 & \beta^{10} \end{array} \right) & \left(\begin{array}{cc|cc} \beta^5 & 0 & 0 & 0 \\ 0 & \beta^{10} & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{array} \right) \\
- (p_3 p_2 p_1) & - (p_3 p_2^2) & - (p_1^4) \\
\begin{pmatrix} \beta^5 & 0 & 0 & 0 \\ 0 & \beta^{10} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & \left(\begin{array}{cc|cc} \beta^5 & 0 & 0 & 0 \\ 0 & \beta^{10} & 0 & 0 \\ \hline 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{array} \right) & \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \\
- (p_1^3 p_2) & - (p_1^2 p_2^2) & - (p_1 p_2^3) \\
\left(\begin{array}{c|ccc} 1 & 0 & & \\ \hline 0 & 1 & 0 & \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{array} \right) & \left(\begin{array}{cc|cc} 0 & 1 & & 0 \\ 0 & 0 & & \\ \hline & & 1 & 1 \\ 0 & & 0 & 1 \end{array} \right) & \left(\begin{array}{c|ccc} 0 & 0 & & \\ \hline & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{array} \right)
\end{array}$$

— (p_2^4)

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

etc.

Solution 421. Dans le premier cas, la matrice M_1 est cyclique. La matrice est la matrice compagnon de $x^p - 1$. Son polynôme minimal est donc $x^p - 1$. Comme il est de degré p , les invariants de similitude de M_1 sont $(x^p - 1)$. Dans le second cas, la matrice est un bloc de Jordan, donc les invariants de similitude de M_1 sont $((x - 1)^p)$. Mais sur \mathbb{F}_p , $x^p - 1 = (x - 1)^p$, d'où la similitude.

Solution 422. Pour un bloc de Jordan J , il suffit d'utiliser le changement de base miroir (qui échange les vecteurs e_i et e_{n+1-i} d'une base de n vecteurs) pour montrer que J est semblable à J' . En général, nous savons qu'une matrice M est semblable à sa décomposition de Jordan J . Autrement dit, il existe $P \in GL_n(\mathbb{K})$ tel que $M = PJP^{-1}$. On vient de voir qu'il existe $Q \in GL_n(\mathbb{K})$ tel que $J = QJ'Q^{-1}$. Enfin, comme $J = P^{-1}MP$, on a $J' = P'M'P'^{-1}$. Donc $M = PQP'M'(PQP')^{-1}$ comme voulu.

Solution 437. 1. On peut prendre comme borne $n = q^{\deg \chi - 1}$.

2. On peut coder

```
def MyPeriode(chi):
    m= ceil( chi.base_ring().cardinality()^(chi.degree()/2))
    T = []
    g=1
    for j in range(m):
        T.append(g)
        g = g*x
        _,g = (g).quo_rem(chi)
        if j>0 and g==1:
            return j+1
    h,_ = (-chi/chi(0)+1).quo_rem(x)
    _,h = (h^m).quo_rem(chi)
    gamma = h
    for i in range(1,m):
        for j in range(m):
            if gamma==T[j]:
                return i*m+j
        gamma = gamma*h
```

3. Quelle est la période des polynômes suivant ?

- (a) $\chi_1 = 1 + x + x^2 + x^3 + x^4 \in \mathbb{F}_2[x]$,
 (b) $\chi_2 = 1 + x + x^2 + x^4 \in \mathbb{F}_2[x]$,
 (c) $\chi_3 = 1 + x^3 + x^6 \in \mathbb{F}_2[x]$,
 (d) $\chi_4 = 3 + x^2 \in \mathbb{F}_5[x]$,
 (e) $\chi_5 = 3 + 3x + x^2 \in \mathbb{F}_5[x]$.

4.

Solution 440. On peut écrire :

```
D= DisjointSet( list(cartesian_product_iterator([F2]*4)))
for v in cartesian_product_iterator([F2]*4):
    v=list(v)
    w = lfsr_sequence(chi1.coeffs()[:-1],v,chi1.degree()+1)[1:]
    D.union(tuple(v),tuple(w))
D
```

Solution 441.

```
L = [len(x[1]) for x in D1.root_to_elements_dict().items()]
[(L.count(x),x) for x in set(L)]
```

Solution 449. La période est $q^n - 1 = 2^4 - 1 = 15$. Il faut chercher tous les polynômes primitifs de degré 4 sur \mathbb{F}_2 . Ce sont exactement les diviseurs de $x^{q^n-1} - 1$ qui ne divisent pas $x^{q^{n'}-1}$ pour $n' < n$. Or

$$x^{15} - 1 = \underbrace{(x-1)}_{\text{rac.prim.1}} \underbrace{(x^2+x+1)}_{\text{rac.prim.3ieme}} \underbrace{(x^4+x^3+x^2+x+1)}_{\text{rac.prim.5ieme}} \underbrace{(x^4+x+1)}_{\text{rac.prim.15ieme}} \underbrace{(x^4+x^3+1)}_{\text{rac.prim.15ieme}}$$

Il n'y a que 2 LFSR maximaux, de polynômes caractéristiques $x^4 + x^3 + 1$ et $x^4 + x + 1$.

Solution 482.

Solution 502. 1. $F(X, Y, Z) = Y^2Z + YZ^2 + X^3 + XZ^2 + Z^3$

2. Irréductibilité : si f se factorise, f est de la forme $f(x, y) = (y - t_1(x))(y - t_2(x))$ avec $t_1 + t_2 = 1$ et $t_1 t_2$ de degré 3, ce qui est impossible. Genre 1 par la formule de Plücker
3. $\mathcal{E}(\mathbb{F}_2) = \{(0 : 1 : 0)\}$.
 $\mathcal{E}(\mathbb{F}_4) = \mathcal{E}(\mathbb{F}_2) \cup \{(0 : \alpha : 1), (0 : \alpha^2 : 1), (0 : \alpha^4 : 1), (0 : \alpha^8 : 1)\}$ (les deux derniers points sont les conjugués des premiers).
 Sur \mathbb{F}_8 , on commence par noter que $\{\beta, \beta^2, \beta^4\}$ et $\{\beta^3, \beta^6, \beta^5\}$ sont conjugués. On substitue x par une des orbites et l'on résout $y^2 + y = 0$ dans un cas, $y^2 + y = \beta^2 + \beta$ dans l'autre. $\mathcal{E}(\mathbb{F}_8) = \mathcal{E}(\mathbb{F}_2) \cup \{(\beta : 0 : 1), (0 : \beta : 1)^\sigma, (\beta : 0 : 1)^{\sigma^2}\} \cup \{(\beta : 1 : 1), (\beta : 1 : 1)^\sigma, (\beta : 1 : 1)^{\sigma^2}\} \cup \{(\beta^3 : \beta : 1), (\beta^3 : \beta : 1)^\sigma, (\beta^3 : \beta : 1)^{\sigma^2}\} \cup \{(\beta^3 : \beta + 1 : 1), (\beta^3 : \beta + 1 : 1)^\sigma, (\beta^3 : \beta + 1 : 1)^{\sigma^2}\}$

$$1), (\beta^3 : \beta + 1 : 1)^\sigma, (\beta^3 : \beta + 1 : 1)^{\sigma^2}\}$$

Il se trouve que γ est un élément primitif de \mathbb{F}_{16} (racine $16 - 1 = 15$ de l'unité). Sur \mathbb{F}_{16} , les classes cyclotomiques sont donc $\{0\}, \{1\}, C' = \{\gamma, \gamma^2, \gamma^4, \gamma^8\}$, $C_0 = \{\gamma^3, \gamma^6, \gamma^{12}, \gamma^9\}$ (racines 5-ièmes de l'unité), $\{\alpha = \gamma^5, \alpha^2 = \gamma^{10}\}$ (racines 3-ièmes), $C'' = \{\gamma^7, \gamma^{14}, \gamma^{13}, \gamma^{11}\}$. Pour trouver $\mathcal{E}(\mathbb{F}_{16})$, on résout $y^2 + y = x_0^3 + x_0 + 1$ pour $x_0 = \gamma, \gamma^3, \gamma^5$ et γ^7 . Si $x_0 = \gamma$, il n'y a pas de racines; si $x_0 = \gamma^3$ $y = \gamma^6$ ou γ^{13} ; si $x_0 = \gamma^5$ ou γ^7 $y = \gamma$ ou γ^4 . Par conjugaison, cela permet d'obtenir $8+4+4 = 20$ nouvelles racines. On obtient $|\mathcal{E}(\mathbb{F}_{16})| = 25$.

4. Voir question finale pour une solution simple. Sinon :

```
PlanProj2 = ProjectiveSpace(2, F2, 'x')
PlanProj2.inject_variables()

Equation=[x0*x2^2+x0^2*x2 +x1^3+x1*x0^2+x0^3]
Courbe = PlanProj2.subscheme(Equation)

print "Dimension ", Courbe.dimension()
print "Composantes irréductibles",
Courbe.irreducible_components()
print "Lissité ?", Courbe.is_smooth()
for q in [2,4,8,16] :
    print "Nombre de points sur F",q, ":",
    len(Courbe.rational_points(GF(q,'x')))
```

5. Pour le calcul des diviseurs, il nous faut déterminer les multiplicités des intersections de $\{X = 0\}$, $\{Y = 0\}$ et $\{Z = 0\}$ avec $\{F(X, Y, Z) = 0\}$. Dans le premier cas, on obtient $P_1 = (0 : \alpha : 1)$, $P'_1 = (0 : \alpha^2 : 1)$ et $P_\infty = (0 : 1 : 0)$ avec multiplicité 1 chacun. Dans le deuxième : les trois points $P_2 = (\beta : 0 : 1)$ et leurs conjugués $P'_2 = (\beta^2 : 0 : 1)$ et $P''_2 = (\beta^4 : 0 : 1)$. Enfin pour $\{Z = 0\}$, on obtient (P_∞) avec multiplicité 3. Le théorème de Bézout est vérifié à chaque fois. Ceci donne

$$\operatorname{div} \left(\frac{X}{Z} \right) = P_1 + P'_1 - 2P_\infty \quad \operatorname{div} \left(\frac{X}{Y} \right) = P_2 + P'_2 + P''_2 - 3P_\infty$$

On peut vérifier que le degré est bien 0. On en déduit qu'en général

$$\operatorname{div} \left(\frac{X^\alpha Y^\beta}{Z^{\alpha+\beta}} \right) = \alpha P_1 + \alpha P'_1 + \beta P_2 + \beta P'_2 + \beta P''_2 - (2\alpha + 3\beta) P_\infty$$

6. D'après la question précédente, on voit que $x^\alpha y^\beta \in \mathcal{L}(rP_\infty)$ pour $r \geq 2\alpha + 3\beta$. Les fonctions $x^\alpha y^\beta$ avec $\beta \in \{0, 1\}$ sont linéairement indépendantes car dans la suite

	1	x	y	x ²	xy	x ³	x ² y	x ⁴	x ³ y	...
−ord _{P_∞}	0	2	3	4	5	6	7	8	9	...

les ordres en P_∞ sont étagés. Ainsi aucun des termes de la suite ne peut se trouver dans l'espace engendré par les termes précédents. Une base de $\mathcal{L}(rP_\infty)$ est formé des r premiers termes de la suite

$$1, x, y, x^2, xy, x^3, x^2y, x^4, x^3y, \dots$$

7. On considère le code de Goppa $\mathcal{G}_{(\mathcal{E}, S, D)}$ où S est l'ensemble des points affines sur \mathbb{F}_{16} de \mathcal{E} et $D = (r+1)P_\infty$.

```

x,y = polygens(GF(2), 'x, y')
Pol.<x,y>=PolynomialRing(GF(2), 'x,y')
F2=GF(2)
F4.<a>=GF(4, 'a')
F8.<b>=GF(8, 'b')
F16.<c>=GF(16, 'c')
p = y^2+y+x^3+x+1
ptsaffines4=[ (u,v) for u in F4 for v in F4 if p(u,v)==0]
ptsaffines8=[ (u,v) for u in F8 for v in F8 if p(u,v)==0]
8. ptsaffines16=[ (u,v) for u in F16 for v in F16 if p(u,v)==0]
print len(ptsaffines4), len(ptsaffines8), len(ptsaffines16)
L_Base=[Pol(1)]
for i in range(1,5):
    L_Base.append(x^i)
    L_Base.append(x^(i-1)*y)
L_Code = [[ f(u) for u in ptsaffines4] for f in L_Base[0:3]]
Code = LinearCode(Matrix(L_Code))
print Code.length()
print Code.dimension()
print Code.minimum_distance()
```

Solution 503. 1. On utilise le critère d'Eisenstein avec $p(x) = x$. On en déduit que la courbe est irréductible.

2. Soit $F(X, Y, Z) = Z^4 f(X/Z, Y/Z) = X^3 Y + Y^3 Z + Z^3 X$. On a $\frac{\partial F}{\partial X} = 3X^2 Y + Z^3$ et par permutation circulaire $\frac{\partial F}{\partial Y} = 3Y^2 Z + X^3$ et $\frac{\partial F}{\partial Z} = 3Z^2 X + Y^3$. Si la caractéristique est 3, il est facile de voir que la courbe est lisse. Sinon, soit (X, Y, Z) un point singulier. Alors X, Y, Z sont tous non-nuls car ils ne peuvent être tous nuls. On a alors $(-3Y^2 Z)Y + Y^3 Z + Z^3 X = 0$ soit $-2Y^3 + Z^2 X = 0$. On en tire $-2(-3Z^2 X) + Z^2 X = 7Z^2 X = 0$, d'où $X = Z = 0$ et $Y = 0$ aussi par permutation, si la caractéristique est $\neq 7$. En réinjectant dans F , on peut aussi obtenir $\frac{3}{2}Y^3 + Z^2 X = 0$ et conclure dans le cas restant. Le genre vaut $g = (4-1)(4-2)/2 = 3$.
3. Le polynôme $\alpha^3 + \alpha + 1$ n'a pas de racine dans \mathbb{F}_2 , donc il ne peut se factoriser car de degré 3. Donc α définit une extension de degré 3 et on peut donc utiliser le quotient $\mathbb{F}_2[\alpha]$ pour représenter \mathbb{F}_8 . On

pose aussi $\mathbb{F}_4 = \mathbb{F}_2[\omega]$ où $\omega^2 + \omega + 1 = 0$. Attention \mathbb{F}_4 n'est pas un sous-corps de \mathbb{F}_8 (Rappel : \mathbb{F}_{p^k} est un sous-corps de $\mathbb{F}_{p^{k'}}$ ssi k divise k'). Pour compter les points, on note d'abord qu'il y a deux points à l'infini : $(1 : 0 : 0)$ et $(0 : 1 : 0)$. Puis on recherche les points affines. Comme toujours on utilise le Frobenius pour faire moins de calculs. De plus, on remarque que si $(a : b : c)$ sont des racines, alors $(b : c : a)$ et $(c : a : b)$ aussi. On fixe la valeur de x et on recherche celle possible de y .

x	\mathbb{F}_2	\mathbb{F}_4	\mathbb{F}_8
0	0	0	0
1	\emptyset	\emptyset	$\alpha, \alpha^2, \alpha^4$
ω		ω^2	
α			$1, \alpha, \alpha^6$
α^{-1}			$\alpha^3, \alpha^4, \alpha^6$

On obtient $|\mathcal{K}(\mathbb{F}_2)| = 3$, $\mathcal{K}(\mathbb{F}_4) = 2 + 1 + 2 \cdot 1 = 5$ and $|\mathcal{K}(\mathbb{F}_8)| = 2 + 1 + 3 + 3 \cdot 3 + 3 \cdot 3 = 24$. La borne de Weil ($q + 1 + 6\sqrt{q}$) donne 11,17,25 ; celle de Serre ($q + 1 + 3[2\sqrt{q}]$) 9,17,24. Aussi le cardinal de $\mathcal{K}(\mathbb{F}_8)$ est optimal.

4. On recherche l'intersection de $\{X = 0\} \wedge \{X^3Y + Y^3Z + Z^3X = 0\} = \{X = 0 \wedge (Y = 0 \vee Z = 0)\} = \{P_2, P_3\}$.

Pour le calcul de la multiplicité de P_2 , on se place dans le plan affine $Y = 1$ et on détermine $\dim(\mathbb{F}_q[[x, z]] / \langle x, x^3 + z + z^3 \rangle)$. Mais en tant qu'idéal de $\mathbb{F}_q[[x, z]]$, $\langle x, x^3 + z + z^3 \rangle = \langle x, z + z^3 \rangle = \langle x, z(1 + z^2) \rangle = \langle x, z \rangle$ car $(1 + z^2)$ est inversible (d'inverse $\sum_{i \geq 0} (-z^2)^i$) dans $\mathbb{F}_q[[x, z]]$. La multiplicité de P_2 est 1.

Pour le calcul de la multiplicité de P_3 , on se place dans le plan affine $Z = 1$ et on détermine $\dim(\mathbb{F}_q[[x, y]] / \langle x, x^3y + y^3 + x \rangle)$. Mais en tant qu'idéal de $\mathbb{F}_q[[x, y]]$, $\langle x, x^3y + y^3 + x \rangle = \langle x, x^3y + y^3 \rangle = \langle x, y^3 \rangle$ car un élément $u(x)x + v(x)(x^3y + y^3)$ peut se réécrire $(u(x) + v(x)x^2y)x + v(x)y^3$ et vice versa. On en déduit que la multiplicité est 3. (Le théorème de Bézout est vérifié puisque $1 + 3 = 1 \cdot 4$)

On obtient par permutation circulaire les intersections avec $\{Y = 0\}$ et $\{Z = 0\}$. On en tire

$$\operatorname{div}(x) = \operatorname{div}\left(\frac{X}{Z}\right) = -(P_1) - 2(P_2) + 3(P_3)$$

$$\operatorname{div}(y) = \operatorname{div}\left(\frac{Y}{Z}\right) = 2(P_1) - 3(P_2) + (P_3)$$

(on contrôle que les degrés valent 0 puisque ce sont des diviseurs principaux)

5. On note que pour $i, j \in \mathbb{N}$

$$\operatorname{div} \frac{y^i}{x^j} = (2i + j) \cdot (P_1) + (2j - 3i) \cdot (P_2) + (i - 3j) \cdot (P_3)$$

Ainsi, pour que $\text{div} \frac{y^j}{x^i} \in \mathcal{L}(rP_3)$, il faut que $2i + j \geq 0$ (ce qui est toujours le cas), $2j - 3i \geq 0$ et $i - 3j + r \geq 0$. Considérons la suite

	1	$\frac{1}{x}$	$\frac{y}{x^2}$	$\frac{1}{x^2}$	$\frac{y^2}{x^3}$	$\frac{y}{x^3}$	$\frac{1}{x^3}$	$\frac{y^2}{x^4}$...
ord_{P_1}	0	1	4	2	7	5	3	8	
ord_{P_2}	0	2	1	4	0	3	6	2	
ord_{P_3}	0	-3	-5	-6	-7	-8	-9	-10	

Comme les ordres en P_3 sont échelonnés, à cause du fait que $\text{ord}(f + g) \geq \min(\text{ord}(f), \text{ord}(g))$ (cf. prop 7.2.1 de ³⁵), chaque terme de la suite est indépendant des précédents. Quand $r \geq 5$, on sait d'après le théorème de Riemann Roch, que la dimension de $\mathcal{L}(rP_1)$ est $r - 2$. Les $r - 2$ premiers termes de la suite étant libres et dans $\mathcal{L}(rP_3)$, ils en forment une base.

Toujours à cause de $\text{ord}(f + g) \geq \min(\text{ord}(f), \text{ord}(g))$ et parce que $\mathcal{L}(r \cdot P_3) \subseteq \mathcal{L}(5 \cdot P_3)$ on en déduit que $\mathcal{L}(0) = \mathcal{L}(P_3) = \mathcal{L}(2P_1) = \langle 1 \rangle$ et $\mathcal{L}(3P_3) = \mathcal{L}(4P_1) = \langle 1, \frac{1}{x} \rangle$.

Solution 504. 1. On pose $f(x, y) = x^9 + y^9 + 1$. On remarque que 1 est une racine simple de $x^9 + 1$, ce qui montre qu'en prenant $p(x) = x + 1$, le critère d'Eisenstein s'applique et f est irréductible. Par ailleurs, $\frac{\partial f}{\partial X} = X^8$, $\frac{\partial f}{\partial Y} = Y^8$ et $\frac{\partial f}{\partial Z} = Z^8$ ne s'annulent pas tous simultanément, donc \mathcal{F} est lisse. Par la formule de Plücker, la courbe est de genre $(9 - 1)(9 - 2)/2 = 28$.

2. Comme $3|6$, \mathbb{F}_8 est un sous-corps de \mathbb{F}_{64} ³⁶. L'application $x \mapsto x^9$ est un endomorphisme de groupe. Si $x^9 = 1$, x est une racine 9ième de l'unité. Or \mathbb{F}_8^\times est d'ordre 7 : il ne contient pas de racine non-triviale. Comme le noyau est trivial, l'application est une bijection. Par contre \mathbb{F}_{64}^\times est d'ordre $63 = 7 \times 9$ et contient les 9 racines de l'unité, il est donc 9 à 1. De plus les éléments de l'image satisfont à $y^7 = x^{(9)^7} = x^{63} = 1$: il font donc partie de \mathbb{F}_8 . (En fait il s'agit de la norme de l'extension de corps \mathbb{F}_{8^2} de \mathbb{F}_8 .)
3. Il y a 9 points à l'infini : $P_\infty = (1 : 1 : 0)$ et les 8 nouveaux points $P_\zeta = (1 : \zeta : 0)$ où ζ est une racine 9ième de l'unité.
4. Sur \mathbb{F}_2 , il n'y a que 3 points : $P_1 = (0 : 1 : 1)$, $P_2 = (1 : 0 : 1)$ et $P_\infty = (1 : 1 : 0)$. Sur \mathbb{F}_8 , outre l'unique point à l'infini P_∞ , il y a pour tout choix de x_0 une seule racine de $f(x_0, y) = 0$, ce qui donne 9 points en tout. Enfin, sur \mathbb{F}_{64} , il y a d'abord 9 points à l'infini : P_∞ et les 8 nouveaux points $(1 : \zeta : 0)$ où ζ est une racine 9ième de l'unité. Par ailleurs, si $x_0 \in \mathbb{F}_{64}$, soit x_0 est l'une des 9 racines de l'unité et l'équation $f(x_0, y) = 0$ a 1 racine distincte ou soit x_0 est l'une des 55 non-racines de l'unité et l'équation $f(x_0, y) = 0$ a 9 racines. Cela fait en tout $9 + 9 + 55 \times 9 = 513$ éléments et la borne de Weil est atteinte.

35. David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017

36. Une façon de voir les choses est de dire que \mathbb{F}_8 est formé des racines de $x^8 - x$, autrement dit de 0 et des racines 7ièmes de l'unité. Or \mathbb{F}_{64}^\times est d'ordre $63 = 7 \times 9$: il les contient toutes.

5. La longueur du code est $504 = 513 - 9$. La dimension est $\deg D + 1 - g = 9r - 27$ quand $9r > 2g - 2 = 54$ i.e. $r \geq 8$. Nous verrons ci-dessous qu'elle est $(r+2)(r+1)/2$ sinon. La distance assignée est $504 - 9r$.

Posons $P_a = (0 : 1 : a)$ et $Q_a = (1 : 0 : a)$. On calcule $\operatorname{div}(x) = \operatorname{div}\left(\frac{X}{Z}\right) = \sum_{\zeta^9=1} (P_\zeta) - (R_\zeta)$ et $\operatorname{div}(y) = \operatorname{div}\left(\frac{Y}{Z}\right) = \sum_{\zeta^9=1} (Q_\zeta) - (R_\zeta)$ ce qui montre que $x^i y^j \in \mathcal{L}(rD_0)$ si $i + j \leq r$. Pour montrer que le système de fonctions proposées est libre, on peut utiliser la division euclidienne dans $\mathbb{F}_q(y)[x]$ par $x^9 + (y^9 + 1)$. De la sorte, tout polynôme de $\mathbb{F}_q[x, y]$ possède un unique représentant dans $\mathbb{F}_q(y)[x]$ avec $\deg_x \leq 8$. Comme les fonctions $x^i y^j$ avec $i \leq 8$ sont déjà réduites, les fonctions sont linéairement indépendantes.

Enfin, comme $\mathcal{L}(rD_0) \subseteq \mathcal{L}(8D_0)$, on déduit que $\mathcal{L}(rD_0) = \langle x^i y^j, i + j \leq r \rangle$.

37. En réalité, comme le coefficient dominant est 1, il n'y a pas de division par un élément de $\mathbb{F}_q(y)$ et ce représentant est même dans $\mathbb{F}_q[x, y]$

Solution 520. Il est possible de calculer le pgcd d en temps polynomial. Il reste alors à simplifier les poids a_i et la somme s par d . Le problème reste équivalent.

Solution 521. On remarque que la somme est super-croissante. On peut donc décomposer s comme

$$\begin{aligned} s = 127 &= 95 + 32 \\ &= 95 + 24 + 8 \\ &= 95 + 24 + 5 + 3. \end{aligned}$$

Solution 522.

```
#solution 1
B = [356, 278, 417, 27, 132, 464, 521]
s=1287
C=block_matrix([[identity_matrix(7), Matrix(7,1, B) ],
[matrix(1,7), Matrix( [[- s]] ) ]])
shortvect=C.LLL().row(0)
print C.solve_left(shortvect)
# Verif
356+278+132+521-1287
```

Solution 526. On obtient la méthode de Coppersmith avec :

```

#solution
Pol.<x> = PolynomialRing(ZZ)

def Coppersmith(f,N):
    d=f.degree(x)
    m=ceil(ln(N)/d)
    B=ceil(N^(1/d)/(2*exp(1.)))
    List_g_ij=[]
    A = Matrix(ZZ,d*(m+1),d*(m+1))
    compteur=0
    for i in range(m+1):
        for j in range(d):
            p=expand(x^j*N^i*f^(m-i))
            pp=p(x=B*x)
            List_g_ij.append(p)
            for deg in range(d*(m+1)):
                A[compteur,deg]=pp[deg]
            compteur+=1
    shortvec=A.LLL().row(0);
    shortcoeff=A.solve_left(shortvec)
    h=Pol(0)
    for t in range((m+1)*d):
        h+=shortcoeff[t]*List_g_ij[t]
    R=h.roots(multiplicities=false,ring=ZZ)
    R=[r for r in R if abs(r)<=B and mod(f(x=r),N)==0]
    return R;

p=(x+1)*(x-2)*(x-3)*(x-29)
Coppersmith(p,10000)

```

Solution 527. Ce jour là, le message est ...

```

#solution
bin=BinaryStrings()
N = 42564360034887861127
Pol.<x> = PolynomialRing(ZZ)
PolmodN.<y> = PolynomialRing(Integers(N))
e = 3
c= 12843085802751039909
textp="08/06:"
mtilde = mod( ZZ(str(bin.encoding(textp)), base=2),N)*2^16
print mtilde
f=(mtilde+y)^e-c
f = Pol(f.change_ring(ZZ))
C = Coppersmith(f,N)
C = C[0]
mm=C.str(2)
mm = '0'*(8*ceil(len(mm)/8)-len(mm))+mm #padding
bin(mm).decoding()

```

Solution 542.

```

#solution
def Babai(B,t):
    b=t
    G,M=B.gram_schmidt()
    n=B.nrows()
    for j in range(n-1,-1,-1):
        u=b.dot_product(G.row(j))/(G.row(j)).dot_product(G.row(j))
        b=b- round(u)*B.row(j)
    return t-b

```

Il suffit ensuite d'ajuster l'algorithme de Babai

Algorithme 46 : Réduction au parallélépipède orthogonalisé

Entrées : Vecteur t

Sorties : Vecteur $b \in \mathcal{P}$ tel que $x - b \in \mathcal{L}$

```

1  $b \leftarrow t$ 
2 pour  $j = n$  à 1 faire
3    $u \leftarrow \langle b, b_j^* \rangle / \langle b_j^*, b_j^* \rangle$ 
4    $b \leftarrow b - \lfloor u \rfloor b_j$ 
5 retourner  $b$ 

```

Solution 546.

```
# solution
EE= matrix(ZZ,40, {(32, 32): 1, (22, 13): -2, (23, 8): -3,
(15, 31): -1, (22, 37): -1,
(13, 5): -4, (38, 20): 2, (4, 12): 3, (19, 22): -2, (15, 5): 2, (11,
32): -1, (11, 10): 3, (1, 11): -4, (12, 33): 1, (0, 15): 1, (33, 17): 1,
(7, 19): -1, (11, 1): -2, (7, 27): 3, (19, 32): -4, (22, 10): 2, (31,
39): -4, (34, 9): 2, (36, 17): 2, (18, 17): 1, (14, 6): -2, (23, 14): 3,
(23, 34): 2, (12, 11): -3, (0, 21): -3, (27, 22): -2, (4, 29): -3, (23,
5): 1, (4, 6): -2, (24, 7): 2, (5, 38): -2, (33, 13): -1, (9, 35): 3,
(18, 36): 1, (22, 5): 1, (24, 25): 3, (34, 31): 2, (6, 34): -3, (23,
33): -4, (20, 37): -1, (38, 12): 2, (33, 0): -1, (4, 32): 3})
AA=10*identity_matrix(40)+EE
HH=AA.hermite_form()
cc = vector([-2, 0, 2, 0, 0, 1, -1, -1, -3, 0, 0, 2, -1, 13, 7, 2, 0, 2, 27, 2, 1,
17, -2, 899, 50, 15, 11, 1098, 7, 2, -1, 10, -1, 2, 156, 15, 42, 8,
525748584, 37])
a = Babai(AA,cc)-m*HH
h = Babai(HH,c)-m*HH
hlll = Babai(HH.LLL(),c)-m*HH
a,h,hlll
mm = Babai(AA,cc)*H^(-1)
IntsToString(mm)
```

Solution 553. On peut faire

```
Pol.<x,y> = PolynomialRing(QQ,2)
f = -x^3 + 4*x^2 - 2*x*y + y^2 - 2*x + 2*y
E = EllipticCurve(f)
E.is_smooth()
solve([f,f.derivative(x),f.derivative(y)],[x,y])

E.plot()
implicit_plot(f, (x,-10,10), (y,-10,10))
```

Solution 555. 1. La valeur Δ représente le discriminant du polynôme $x^3 + ax + b$, i.e. le résultant entre ce polynôme et sa dérivée. Il s'annule quand il y a une racine double. On peut montrer que la cubique \mathcal{E} possède un point singulier dans ce cas et seulement dans ce cas. Pour contrôler que \mathcal{E} est une courbe elliptique, il faut donc vérifier que $\Delta \neq 0$.

```
q=29 ; Fq = FiniteField(q); a=Fq(1); b=Fq(1)
E=EllipticCurve([a,b])
```

```
def Delta(a,b):
    return -16*(4*a^3+27*b^2)
```

```
E.discriminant(), Delta(a,b)
```

2.

```
def Jinvariant(a,b):
    return -1728*(4*a)^3/(-16*(4*a^3+27*b^2))
```

```
E.j_invariant(), Jinvariant(a,b)
```

Solution 559. Il s'agit du théorème de Bezout : l'intersection d'une cubique (degré 3) avec une droite (degré 1) possède toujours $3 \times 1 = 3$ points.

Solution 560. 1. L'équation de la droite (PQ) est clairement

$$y - y_P = \lambda(x - x_P)$$

Pour trouver S , on doit donc résoudre le système

$$\begin{cases} y - y_P = \lambda(x - x_P) \\ y^2 = x^3 + ax + b \end{cases}$$

qui conduit à l'équation

$$x^3 - \lambda^2 x^2 + [\dots] = 0.$$

Mais

$$x^3 - \lambda^2 x^2 + [\dots] = (x - x_P)(x - x_Q)(x - x_S).$$

Donc par identification

$$x_S = \lambda^2 - x_P - x_Q$$

La conclusion suit immédiatement.

2. On se souvient que la tangente en x_P a pour équation

$$\frac{\partial f}{\partial x}(P) \cdot (x - x_P) + \frac{\partial f}{\partial y}(P) \cdot (y - y_P) = 0$$

ce qui fournit bien la pente $\lambda = \frac{3x_P^2 + a}{2y_P}$. Le reste du calcul est identique la question précédente. Si $y_P = 0$, on obtient une tangente verticale qui intersecte la courbe en $0_{\mathcal{E}}$. Donc $2P = 0_{\mathcal{E}}$.

3. Le sous-groupe $\mathcal{E}[2]$ de 2-torsion correspond aux points P tels que $2P = 0_{\mathcal{E}}$. On retrouve donc le point $0_{\mathcal{E}}$ et l'ensemble des points $(\sigma, 0)$ tels que $\sigma^3 + a\sigma + b = 0$. Il y en a 0, 1 ou 3, ce qui correspond aux groupes trivial, $\mathbb{Z}/2\mathbb{Z}$ ou $(\mathbb{Z}/2\mathbb{Z})^2$.

Solution 561. 1. Le discriminant est non nul.

2.(a) $H = (11, 11)$.

(b) $k = 5$, donc $Y = kG = (11, 2)$. $kH = (1, 5)$. Donc $s_1 = x_1 = 2$ et $s_2 = 5x_2 = 9$.

(c) Le clair est $(5, 8)$.

Solution 563. 1. En général, on ne peut pas faire mieux que

```
def myOrder(P):
    Q=P; n=1
    while Q<>P.curve()(0):
        Q+=P; n+=1
    return n
```

2. Cas pratique

```
Fq=GF(2003)
E=EllipticCurve([Fq(1929), 1178])
P = E(478,469)
myOrder(P), P.order()
```

3. Notons $N = p_1^{a_1} p_2^{a_2} \cdots p_k^{a_k}$ la factorisation de la taille N de la courbe. Soit P un point de la courbe. Appelons

$$\alpha_i = \max \{s \in [0, a_i]; [N/p_i^s]P = 0_{\mathcal{E}}\}.$$

Alors

$$\text{ord}(P) = \prod_{i=1}^k p_i^{a_i - \alpha_i}.$$

En travaillant par exponentiation rapide, le temps de calcul est polynômial.

Solution 572. Le texte est paru au JORF n° 0241 du 16 octobre 2011 page 17533. La courbe proposée est définie sur \mathbb{F}_p où p possède 256 bits.)

Solution 577. 1. On a $\omega_p = 3$ et $\omega_q = 4$.

2. Il n'y a qu'un seul point : $(0 : 1 : 0)$

3. On doit avoir $\phi_p(\omega_p P) = (0 : 1 : 0)$ et $\phi_q(\omega_q P) \neq (0 : 1 : 0)$, ce qui implique (par la question précédente) que $\omega_p P = (0 : 1 : 0)$ et que $\omega_q P \neq (0 : 1 : 0)$. Contradiction. Calculer $\omega_p P$ déclenche une erreur.

Solution 580. 1. On peut faire

```

def division(x,y):
    n = y.modulus()
    if ZZ(n).divides(ZZ(y)):
        raise ZeroDivisionError
    elif gcd(ZZ(y),ZZ(n))>1:
        raise FoundFactor(gcd(ZZ(y),ZZ(n)))
    else:
        return mod(x,n)/mod(y,n)

```

2. Algorithme d'addition

```

def addition(P,Q):
    if P[2]==0:
        return Q
    elif Q[2]==0:
        return P
    elif P[0]==Q[0] and P[1]==-Q[1]:
        return P.curve()(0)
    else:
        if P==Q:
            lamb = division( (3*P[0]^2+P.curve().a4()) , (2*P[1]))
        else:
            lamb = division(Q[1]-P[1], Q[0]-P[0])
        x = lamb^2-P[0]-Q[0]
        y = -P[1]+lamb*(P[0]-x)
        return P.curve()(x,y)

```

3. Multiplication

```

def multiplication(lamb,P):
    E = P.curve()
    if lamb == 0:
        return E(0)
    elif lamb == 1:
        return P
    elif lamb%2 == 0:
        return multiplication(lamb/2,addition(P,P))
    else:
        Q = multiplication(ZZ((lamb-1)/2),addition(P,P))
        return addition(P,Q)

```

4. Algorithme ECM

```

def ECM(n,B):
    Zn = Zmod(n)
    a = Zn.random_element()
    x0 = Zn.random_element()
    y0 = Zn.random_element()
    b = y0^2-x0^3-a*x0
    g = gcd(ZZ(4*a^3+27*b^2),ZZ(n))
    if g==n:
        return False
    elif g>1:
        return g
    E = EllipticCurve(Zn, [a,b])
    A = E(x0,y0)
    for p in primes(B):
        c=1
        while (c<B):
            c*=p
            try:
                A = multiplication(p,A)
            except FoundFactor as FF:
                print 'Facteur trouvé :'
                return FF.value
    return False

```

Solution 581. On peut faire

```

p=63029
q=679969
Ep = EllipticCurve(GF(p),[4,4])
Eq = EllipticCurve(GF(q),[4,4])
Ep.abelian_group(), Eq.abelian_group()
Ep.cardinality().factor() , Eq.cardinality().factor()

```

On remarque alors que les deux groupes sont des groupes cycliques $\mathcal{E}(\mathbb{F}_p) = \mathbb{Z}/62896\mathbb{Z}$ et $\mathcal{E}(\mathbb{F}_q) = \mathbb{Z}/678501\mathbb{Z}$. Comme le plus petit des plus grands facteurs de ces ordres est 3931, il faut donc aller génériquement jusqu'à la borne $B = 3931$ (incluse). Il se trouve que ici P est d'ordre $2^3 \cdot 3931$ (faire `Ep(Pn).order().factor()` pour s'en apercevoir).

Autre solution : tester

```

n = 42857766101
En = EllipticCurve(Integers(n), [4, 4])
Pn = En(1, 3)

Q = Pn
for r in primes(3932):
    for i in range(log(n, r) + 1):
        Q = r * Q

```

Remarque : on aurait pu prendre comme point $P' = (21091706543, 38501272299)$ qui est d'ordre 8 dans $\mathcal{E}(\mathbb{F}_p)$. Dans ce cas, calculer $8P'$ dans $\mathcal{E}(\mathbb{Z}/n\mathbb{Z})$ provoque déjà une erreur.

Solution 589. On peut faire :

Algorithme 47 : Exponentiation rapide récursive

Entrées : Élément a d'un anneau A , entier $n \in \mathbb{N}$

Sorties : $a^k \in A$

```

1 si  $n = 0$  alors
2   | retourner 1
3 sinon si  $n = 1$  alors
4   | retourner  $a$ 
5 sinon
6   | si  $n$  pair alors
7     | retourner  $\text{ExpRap}(a, n/2)$ 
8   | sinon
9     | retourner  $a \cdot \text{ExpRap}(a, \lfloor n/2 \rfloor)$ 

```

Solution 591. On peut faire

```

def MyAddChain(n):
    V = {1:0, 2:1}
    L = { int(0b11) }
    for i in range(2,n+1):
        LL = []
        for c in L:
            for k in range(c.bit_length()):
                b= (c>>k) & int(1)
                if b ==1 :
                    for kk in range(k, c.bit_length()):
                        bb = (c>>kk) & int(1)
                        if bb ==1 :
                            t = k+kk+1
                            if ((c>>t) & int(1))!=0 :
                                cc=c+ (1<<t)
                                if not V.has_key(t+1):
                                    V[t+1]=i
                                LL.append(int(cc))
        L = Set(LL)
    return V

```

Solution 592. Soit $\alpha_\rho = \alpha_{i(\rho)} + \alpha_{j(\rho)}$ une suite permettant de calculer m en $\ell(m)$ opérations et une suite $\alpha'_{\rho'} = \alpha'_{i'(\rho')} + \alpha'_{j'(\rho')}$ permettant de calculer n en $\ell(n)$ opérations. On peut calculer $m \cdot n$ en calculant d'abord m avec $\beta_\rho = \beta_{i(\rho)} + \beta_{j(\rho)}$ puis $m \cdot n$ en effectuant $\beta_{\ell(m)+\rho} = \beta_{\ell(m)+i'(\rho)} + \beta_{\ell(m)+j'(\rho)}$

Solution 594.

```

def ExponentiationRapide(P,n):
    """
    Calcule n*P par la methode d'exponentiation rapide
    """
    if n==0:
        return parent(P)(0)
    Q=P
    for i in range(int(n).bit_length()-1,0,-1):
        print i
        if ((n>>(i-1))%2)==1:
            Q=Q+Q+P
        else:
            Q=Q+Q
    return Q;

```

Solution 595. Les formules d'addition de deux points distincts et de

duplication sont identiques, ce qui rend indistingables les deux opérations. Une attaque par canaux auxiliaires n'est pas possible.

Solution 605. La recherche d'un élément dans une table de hachage se fait en $O(1)$ (à comparer à un coût linéaire pour une liste), ce qui rend l'algorithme en $O(\sqrt{n})$.

Solution 623. L'ordre de la courbe est $p - 1$. On peut faire

```
p=4801
E = EllipticCurve(GF(p),[0,1])
n = E.order()

P = E(2746,392)
Q = E(2524,1070)
R=P+Q

a=721
b=324
c=2501

Pa=a*P
Qa=a*Q
Pb=b*P
Qb=b*Q
Pc=c*P
Qc=c*Q
Ra=Pa+Qa
Rb=Pb+Qb
Rc=Pc+Qc

O = E(0)

print ( (Pb.tate_pairing(Rc,n,1)/Qb.tate_pairing(Rc,n,1)) / (Pb.tate_pairing(O,n,1)/Qb.tate_pairing(O,n,1)) )
print ( (Pa.tate_pairing(Rc,n,1)/Qa.tate_pairing(Rc,n,1)) / (Pa.tate_pairing(O,n,1)/Qa.tate_pairing(O,n,1)) )
print ( (Pa.tate_pairing(Rb,n,1)/Qa.tate_pairing(Rb,n,1)) / (Pa.tate_pairing(O,n,1)/Qa.tate_pairing(O,n,1)) )
print ( (P.tate_pairing(R,n,1)/Q.tate_pairing(R,n,1)) / (P.tate_pairing(O,n,1)/Q.tate_pairing(O,n,1)) )
```

On obtient 1611 comme secret commun.

Solution 628. 1. La clé maitre est (3386, 3790). La clé privée que Tom communique à Alice est (3962, 2508).

2. Bob envoie

$$C_1 = (3748, 1673)$$

$$C_2 = 2534$$

3. On retrouve 2000.

Solution 632. 1. On écrit :

```
def myLine(P1,P2,S):
    E=P.curve()
    x1=P1[0]; y1=P1[1]; z1=P1[2]
    x2=P2[0]; y2=P2[1]; z2=P2[2]
    xS=S[0]; yS=S[1]; zS=S[2]
    if z1==0 and z2==0:
        return P.curve().base_field().one_element()
    if z1==0:
        return xS-x2*zS
    if z2==0:
        return xS-x1*zS
    if x1==x2 and y1==y2:
        return xS-x1*zS
    if x1==x2:
        a=E.a4(); lambd=(3*x1^2+a)/(2*y1)
    else:
        lambd=(y1-y2)/(x1-x2)
    return yS-y1*zS - (xS-x1*zS)*lambd
```

2. On écrit pour h :

```
def myH(P1,P2,S):
    return myLine(P1,P2,S)/myLine(P1+P2,-P1-P2,S)
```

3. On écrit pour l'algorithme de Miller :

```
def myMiller(r,S,P):
    R=S; f=1
    for x in r.bits()[-2::-1]:
        f=f^2*myH(R,R,P)
        R=2*R
        if x==1:
            f=f*myH(R,S,P)
            R=R+S
    return f
```

4. On écrit pour le couplage de Tate :

```

def myTatePairing(S,T,r):
    try:
        sortie = myMiller(r,S,T)
    except (ZeroDivisionError):
        R = S.curve().random_point()
        sortie = myTatePairing(S,T + R, r)/myTatePairing(S,R, r)
    return sortie

A = E.random_element()
B = E.random_element()
r=lcm(A.order(),B.order())
t = (Integers(r)(q)).multiplicative_order()
myTatePairing(A,B,r)^((q^t-1)/r), A.tate_pairing(B,r,t)

```

5. On écrit pour le couplage de Weil :

```

def myWeilPairing(S,T,r):
    return myTatePairing(S,T,r)/myTatePairing(T,S,r)

```

Solution 635. 1. Primalité de p

```

p = 2199023255579
p.is_prime()

```

2. Avec les calculs suivants,

```

Fp = GF(p)
E = EllipticCurve([Fp(1),Fp(0)])
P = E(1435967701832 , 123951463462)
E.j_invariant(); mod(p,4)==3

```

on note que \mathcal{E} est supersingulière.

3. La courbe possède $2199023255580 = 2^2 \cdot 3 \cdot 5 \cdot 36650387593$ points

```

E.cardinality().factor()

```

4. Le point P est d'ordre $r = 36650387593$.

```

P.order()

```

5. L'entier t est $t = 2$

```

(Integers(r)(p)).multiplicative_order()

```

6. Le cardinal de \mathcal{E}_{p^2} est $4835703278581662001136400 = 2^4 \cdot 3^2 \cdot 5^2 \cdot 36650387593^2$

```

Fpp.<alpha> = GF(p^2)
EE = EllipticCurve([Fpp(1),Fpp(0)])
N = EE.cardinality()

```


7. On fait

```
P = EE(1435967701832 , 123951463462)
while True:
    S = 2^4 * 3^2 * 5^2 * EE.random_element()
    w = P.weil_pairing(S,r)
    if w<>1:
        break
```

8. On a $\lambda = 10000000000$.

```
Q = EE(1129476910351 , 1383670460733)
zeta1 = P.weil_pairing(S,r)
zeta2 = Q.weil_pairing(S,r)
zeta2.log(zeta1)
```

Solution 7. 1. Soit on effectue le calcul par force brute, soit on utilise le lemme chinois et on remonte le calcul. Comme 45 se factorise en $\alpha\beta$ avec $\alpha = 5$ et $\beta = 9$, on a

$$\begin{cases} a^{11} &= (3)^{11} = 3^3 = 2 \pmod{\alpha} \\ a^{11} &= (-1)^{11} = -1 \pmod{\beta} \end{cases}$$

$$\begin{cases} a^{22} &= -1 \pmod{\alpha} \\ a^{22} &= 1 \pmod{\beta} \end{cases}$$

Une relation de Bezout entre α et β est $u\alpha + v\beta = 1$ avec $u = 2$ et $\beta = 1$. De plus,

$$u\alpha \equiv \begin{cases} 0 \pmod{\alpha} \\ 1 \pmod{\beta} \end{cases} \quad v\beta \equiv \begin{cases} 1 \pmod{\alpha} \\ 0 \pmod{\beta} \end{cases}$$

Donc

$$a^{11} = 2(v\beta) + (-1)(u\alpha) = 2 \cdot (-9) + (-1) \cdot 2 \cdot 5 = 17 \pmod{45}$$

$$a^{22} = (-1)(v\beta) + (1)(u\alpha) = (-1) \cdot (-9) + 1 \cdot 2 \cdot 5 = 19 \pmod{45}$$

On obtient $a^{11} = 17$, $a^{22} = 19$ et $a^{44} = 1$. On en déduit que a est un témoin de Rabin Miller de la non primalité de n .

2. On a

$$\left(\frac{a}{n}\right) = \left(\frac{8}{3}\right)^2 \left(\frac{8}{5}\right) = -1$$

qui n'est pas égal à a^{22} . Donc a est aussi un témoin de Solovay-Strassen.

Solution 638. 1. On a $\chi(x) = c(x) = 1 + x + x^2 + x^3 + x^4$.

2. Oui. Différentes preuves sont possibles. On peut montrer qu'il n'y a pas de racine ni de factorisation en 2 polynômes de degré 2. On peut aussi voir qu'il s'agit de $(x^5 - 1)/(x - 1)$. Or la plus petite extension à contenir des racines 5-ièmes de l'unité est de degré 4 : $2^2 - 1$ et $2^3 - 1$ ne sont pas divisibles par 5 alors $2^4 - 1$ l'est.
3. Comme déjà vu, le polynôme est de période 5.
4. Il y a une suite de période 1 et 3 suites de période 5 (résultat du cours). On repère alors la suite de période 1 : $[0]^\infty$ et les suites de période 5 : $[1, 0, 0, 0, 1]^\infty$, $[1, 1, 0, 0, 1]^\infty$
5. Ce LFRS est-il maximal ?

Solution 639. 1. On résoud le système

$$\begin{cases} 18 &= \log 2 + \log 7 \pmod{66} \\ 24 &= \log 3 + \log 5 \pmod{66} \\ 30 &= 3\log 2 + \log 5 \pmod{66} \\ 45 &= 2\log 3 + \log 5 \pmod{66} \\ 62 &= 2\log 7 \pmod{66} \end{cases}$$

qui admet pour solution $\log 2 = 53$, $\log 3 = 21$, $\log 5 = 3$ et $\log 7 = 31$.

2. On a $\alpha = \log 42 = \log 2 + \log 3 + \log 7 = 39 \pmod{66}$ et $\beta = \log 35 = \log 5 + \log 7 = 34 \pmod{66}$. Le secret commun est $b^{\alpha\beta} = b^6 = 25$.

*Exercices évalués***TP 1 :**

Aucune évaluation

TP 2 :

Exercices 55, questions 3 à 6 (mise sous HNF) & 64 questions 3 à 6 (mise sous SNF) puis 59 (système linéaire homogène sur \mathbb{Z}), 60 (image d'une matrice), 67 (système linéaire non-homogène sur \mathbb{Z}), 80 (structure du quotient) & 94 (facteurs invariants).

TP 3 :

Exercices 114 (base d'un réseau), 139 (applications numériques de LLL), 135 (algorithme LLL), 118 (réseau \mathbb{E}_8), 145 (densité de réseaux).

TP 4 :

Exercices 163 (sauf question 1, factorisation des puissances), 172 (factorisation SFC), 176 (factorisation EDD), 190 (Cantor-Zassenhauss), 192 (factorisation complète), 193 (question 2, racines), 191 (Etude de Cantor-Zassenhauss).

TP 5 :

Exercices 201 (Berlekamp, sauf question 1), 211 (Hensel) & 214 (Factorisation finale avec LLL).

TP 6 :

Exercices 218 (fonctions SageMath), 221 (division multivariée), 252 (base de Groebner), 253 (appartenance à un idéal), 256 (résolution d'un système), 257 (optimisation), 263 (manipulations algébriques), 262 (ovales de Descartes).

TP 7 :

Exercices 267 (points singuliers), 277 (valuation), 287 (enveloppe), 295 (coloration de graphe), 288 (preuve de théorèmes géométriques), 299 (programmation entière), 282 (surface de Clebsch).

TP 8 :

Exercices 318 (critère de Rabin-Miller), 319 (performances de Rabin-Miller), 327 (critère de Solovay-Strassen), 329 (comparaison entre Rabin-Miller et Solovay-Strassen), 332 (test de Lucas) & 333 (Test BPSW).

TP 9 :

Exercices 348 (divisions successives), 351 (factorisation d'un nombre B -friable), 355 (ρ de Pollard), 358 ($p - 1$ de Pollard) & 379 (crible quadratique). Pour chaque algorithme de factorisation, faites des tests et comparez les temps d'exécution

TP 10 :

Exercices 401 (invariants de similitude), 408 (forme de Frobenius d'une matrice), 416 (forme de Jordan d'une matrice), 455 (Berlekamp-Massey), 440 (orbites d'un LFSR), 437 (période d'un LFSR), 452 (séries génératrices).

TP 11 :

Exercices 465 (code de Hamming), 477 (décodage de Berlekamp-Massey), 482 (décodage en liste), 502 (code algébrique), 472 (bornes), 510 (codes battant GV).

TP 12 :

Exercices 522 (sac à dos), 531 (attaque de Wiener), 526 (méthode de Coppersmith), 527 (messages stéréotypés), 542 (Babai), 546 (cryptosystème GGH).

TP 13 :

Exercices 560 question 4 et 5 (addition d'une courbe elliptique), 572 (courbe ANSSI), 575 (comptage de points), 580 (factorisation ECM), 594 (exponentiation rapide et canaux cachés), 595 (courbe d'Edwards).

TP 14 :

Exercices 604 (Shanks : pas de bébé, pas de géant), 607 (ρ de Pollard), 632 (couplage), 635 (attaque M.O.V.), 614 (calcul d'indice).

Index

- algorithme
 - ρ de Pollard, 156, 275
 - $p - 1$ de Pollard, 157
 - Baby step, giant step, 185, 272
 - Berlekamp, 78
 - Berlekamp-Massey, 189
 - Buchberger, 102
 - calcul d'indice, 278
 - Cantor Zassenhaus, 72
 - division multivariée, 90
 - ECM de Lenstra, 263
 - fact. degrés distincts, 70
 - fact. sans facteurs carrés, 67
 - forme normale d'Hermite, 28
 - forme normale de Smith, 33
 - Gauß-Bareiss, 15
 - Pohlig-Hellman, 276
 - Rho de Pollard, 274
- Apollonius, cercle, 127
- attaque par canaux auxiliaires, 267
- chiffrement homomorphe, 247
- coefficient dominant, 90
- coloration de graphe, 128
- composante primaire, 170
- corps fini, 11
- courbe elliptique
 - j -invariant, 253
 - discriminant, 253
 - Weierstraß, 252
- cryptographie à clé publique multivariée, 108
- cryptographie basée sur les codes, 222
- cryptographie post-quantique, 108, 222, 251
- cryptosystème $A_5/1$, 194
- cryptosystème E_0 , 194
- cryptosystème RSA, 226
- Diffie-Hellman, 269, 283
- diviseur, 123, 251
 - principal, 252
- El Gamal, 270
- empilement de sphères, 58
- entiers de Gauß, 20
- enveloppe de courbe, 125
- exponentiation rapide, 266
- facteurs invariants, 41
- factorisation
 - dégrés distincts, 70
 - sans facteurs carrés, 67
- fonction de Miller, 258, 282, 289
- fractions continues, 17
- friable, 154, 262
- Frobenius, 12, 64
- groupe de Picard, 254
- Hadamard, borne, 15
- intersection d'idéaux, 111
- invariants de similitude, 176
- Jordan, réduction de, 177
- kissing number, 58
- lemme chinois, 275
- LFSR
 - définition, 179
 - maximal, 187
- lisse, 114
- LLL
 - algorithme, 55, 322
 - base réduite, 53
- logarithme discret, 269
- Massey-Omura, 270
- matrice, 11
 - élémentaire, 24
 - équivalence, 31
 - Bezout, 25
- matrices
 - similitude, 176
- McEliece, 222
- Menezes & Vanstone, 249
- Menezes, Okamoto et Vanstone, 291
- module
 - base adaptée, 38
 - définition, 20
 - de type fini, 23, 37
 - homomorphisme, 22
 - libre, 23, 36
 - torsion, 42
- multidegré, 90
- multiplicateurs de Lagrange, 107
- ordre monomial, 89
- Ovales de Descartes, 111
- point régulier, 114
- point singulier, 114
- polynôme, 11
- réseau euclidien
 - A_n , 46
 - D_n , 46
 - E_8 , 48
 - définition, 45
 - déterminant, 47
 - discriminant, 47
 - minimum, 48
- relèvement de Hensel, 84
- sans facteurs carrés, 62
- signature, 108, 227

Sudoku, 129
 surface de Clebsch, 124
 syzygie, 98
 terme dominant, 90
 théorème
 Abel-Jacobi, 258

Cayley-Salmon, 125
 Dickson, 95
 Hasse, 260
 restes chinois, 12
 Riemann-Roch, 252
 torsion, 42, 257

ultrafriable, 156
 valuation d'une fonction, 116
 variété, 113
 vecteur, 11

Bibliographie

- [ANS21] ANSSI. Guide de sélection d’algorithmes cryptographiques. Technical Report ANSSI-PA-079, Agence nationale de la sécurité des systèmes d’information, 51, boulevard de La Tour-Maubourg, 75700 PARIS 07, 8 mars 2021. Version 1.0.
- [BCS97] Peter Bürgisser, Michael Clausen, and M. Amin Shokrolahi. *Algebraic complexity theory*, volume 315 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 1997. With the collaboration of Thomas Lickteig.
- [Bla03] Richard E. Blahut. *Algebraic Codes for Data Transmission*. Cambridge University Press, 2003.
- [BSS05] I. Blake, G. Seroussi, and N. Smart. *Advances in Elliptic Curve Cryptography*, volume 317 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2005.
- [Buc04] Johannes Buchmann. *Introduction to cryptography*. Undergraduate Texts in Mathematics. Springer-Verlag, New York, second edition, 2004.
- [CCC⁺13] Alexandre Casamayou, Nathann Cohen, Guillaume Connan, Thierry Dumont, Laurent Fousse, François Maltéy, Matthias Meulien, Marc Mezzarobba, Clement Pernet, Nicolas Thiéry, and Paul Zimmermann. *Calcul mathématique avec SAGE*. CreateSpace Independent Publishing Platform, 2013.
- [CCS99] Arjeh M. Cohen, Hans Cuypers, and Hans Sterk, editors. *Some tapas of computer algebra*, volume 4 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, 1999.
- [CFA⁺06] Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Tanja Lange, Kim Nguyen, and Frederik Vercaute-

- ren, editors. *Handbook of elliptic and hyperelliptic curve cryptography*. Discrete Mathematics and its Applications (Boca Raton). Chapman & Hall/CRC, Boca Raton, FL, 2006.
- [CLO15] David A. Cox, John Little, and Donal O'Shea. *Ideals, varieties, and algorithms*. Undergraduate Texts in Mathematics. Springer, Cham, fourth edition, 2015. An introduction to computational algebraic geometry and commutative algebra.
- [Coh93] Henri Cohen. *A course in computational algebraic number theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, 1993.
- [CP05] Richard Crandall and Carl Pomerance. *Prime numbers*. Springer, New York, second edition, 2005. A computational perspective.
- [CS99] J. H. Conway and N. J. A. Sloane. *Sphere packings, lattices and groups*, volume 290 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, New York, third edition, 1999. With additional contributions by E. Bannai, R. E. Borcherds, J. Leech, S. P. Norton, A. M. Odlyzko, R. A. Parker, L. Queen and B. B. Venkov.
- [Dem97] Michel Demazure. *Cours d'algèbre*. Nouvelle Bibliothèque Mathématique [New Mathematics Library], 1. Cassini, Paris, 1997. Primauté. Divisibilité. Codes. [Primality. Divisibility. Codes].
- [Flo13] Jean-Pierre Flori. Notes sommaires et TP. Cours MDI 349, Télécom Paris Tech, 24 juin 2013.
- [Gal12] Steven Galbraith. *Mathematics of public key cryptography*. Cambridge University Press, Cambridge, first edition, 2012.
- [HPS08] Jeffrey Hoffstein, Jill Pipher, and J.H. Silverman. *An Introduction to Mathematical Cryptography*. Springer Publishing Company, Incorporated, 1 edition, 2008.
- [Mad17] David Madore. Courbes algébriques. Notes du cours ACCQ 205, Télécom ParisTech, 6 mars 2017.
- [MG02] Daniele Micciancio and Shafi Goldwasser. *Complexity of lattice problems*. The Kluwer International Series in Engineering and Computer Science, 671. Kluwer Academic Publishers, Boston, MA, 2002. A cryptographic perspective.

- [MR09] Daniele Micciancio and Oded Regev. Lattice-based cryptography. In *Post-quantum cryptography*, pages 147–191. Springer, Berlin, 2009.
- [NV10] Phong Q. Nguyen and Brigitte Vallée, editors. *The LLL algorithm*. Information Security and Cryptography. Springer-Verlag, Berlin, 2010. Survey and applications.
- [Sam67] Pierre Samuel. *Théorie algébrique des nombres*. Hermann, Paris, 1967.
- [Sho09] Victor Shoup. *A computational introduction to number theory and algebra*. Cambridge University Press, Cambridge, second edition, 2009. <http://shoup.net/ntb/>.
- [Sil09] Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer, Dordrecht, second edition, 2009.
- [TWo6] Wade Trappe and Lawrence C. Washington. *Introduction to cryptography with coding theory*. Pearson Prentice Hall, Upper Saddle River, NJ, second edition, 2006.
- [vzGG13] Joachim von zur Gathen and Jürgen Gerhard. *Modern computer algebra*. Cambridge University Press, Cambridge, third edition, 2013.
- [Zémoo] Gilles Zémor. *Cours de cryptographie*. Enseignement des mathématiques. Cassini, Paris, 2000.