Réduction de réseaux
Adaptations d'idées provenant du cas polynomial au cas entier.

HAI001I : Stage académique

Lucas Noirot
(lucas.noirot@etu.umontpellier.fr)

Encadrant
Romain Lebreton
(romain.lebreton@lirmm.fr)

17 février - 26 juin 2025

Remerciements

Table des matières

Liste des Algorithmes

Notations, complexité et acronymes

Cette section fait office de lexique et recense l'ensemble des notations et acronymes utilisées tout au long de ce mémoire.

Notations

$\mathbb{Z}$	L'ensemble des entiers relatifs.
$\mathbb{Q}$	L'ensemble des rationnels.
$\mathbb{R}$	L'ensemble des réels.
$\mathbb{K}$	Un corps quelconque.
$\mathbb{F}_p$	Un corps fini de caractéristique p.
$\mathbb{F}_p[x]$	L'anneau des polynômes univariés à coefficients dans le corps fini $\mathbb{F}_p$.
$\mathbb{F}_p^{\leq d}[x]$	Les polynômes à coefficients dans $\mathbb{F}_p$ de degré inférieur ou égal d.
$\mathbb{F}_p[x]^{m\times n}$	Les matrices $m \times n$ à coefficients dans $\mathbb{F}_p[x]$.
$\mathscr{L}$	Un réseau, désigné par une lettre majuscule calligraphiée.
$\mathscr{L}(B)$	Le réseau engendré par la matrice B.
B^t	La transposée de la matrice B.
B^*	La base de Gram-Schmidt associée à B.

Complexité

Multiplication dans $\mathbb{F}_p^{\leq d}[x]$	$\mathrm{M}(d) = \mathcal{O}(d \times \log d \times \log\log d)$.
Multiplication dans $\mathbb{F}_p^{n\times n}$	$\mathrm{MM}(n) = \mathcal{O}(n^\omega)$.
Multiplication dans $\mathbb{F}_p^{\leq d}[x]^{m\times n}$	$\mathrm{MM}(n,d) = \mathcal{O}(\mathrm{MM}(n)\,\mathrm{M}(d)) = \tilde{\mathcal{O}}(n^\omega d)$.

1

2

Acronymes

LLL	Lenstra–Lenstra–Lovász
SVP	Shortest Vector Problem
CVP	Closest Vector Problem

1. $\mathrm{MM}(n,d)$ peut également être obtenu via une approche d'évaluation-interpolation sur une suite géométrique, ce qui permet d'améliorer certaines bornes de complexité.
2. ω est l'exposant de la multiplication matricielle.

Guide de Lecture

DANS ce mémoire, chaque définition est suivie d'un exemple concret illustrant la notion en question, ainsi que d'un contre-exemple visant à en exposer les subtilités et exceptions éventuelles. Cette approche permet de mieux comprendre les conditions et les limitations associées à chaque concept. L'objectif est de clarifier les différences entre les situations avec lesquelles une définition est applicable et celles où elle ne l'est pas, afin de renforcer la compréhension approfondie des théorèmes et constructions présentés.

Le chapitre 1 = Introduction aux réseaux euclidiens Le chapitre 2 = Introduction aux réseaux polynomaiux . Si jamais par ma,nque de temps le lecteur peut juste se contenter de lire la définition, la suite étant juste une motivation Le chapitre 3 = Réduction de réseau polynomiaux Le chapitre 4 = Réduction de réseaux euclidiens Le chapitre 5 = On essaye d'adapter les techniques du chapitre 3 pour faire de la réduction

Introduction

Un réseau euclidien peut être intuitivement vu comme un ensemble discret et régulier de points dans l'espace $\mathbb{R}^n$, formant un sous-groupe discret additif. À titre d'exemple, dans le plan $\mathbb{R}^2$, un réseau correspond aux intersections d'un quadrillage régulier. Malgré leur ressemblance avec les espaces vectoriels classiques, les réseaux euclidiens possèdent des propriétés spécifiques et complexes, rendant invalides de nombreux résultats habituellement vérifiés dans les $\mathbb{K}$-espaces vectoriels. Cette complexité fait des réseaux euclidiens un objet d'étude particulièrement riche à la frontière de plusieurs domaines de mathématiques et d'informatiques, en particulier la cryptographie.

Un des problèmes fondamentaux liés aux réseaux euclidiens est la réduction de réseaux, qui consiste à déterminer une "bonne" base, c'est-à-dire une base qui facilite la résolution efficace de divers problèmes algorithmiques comme celui du vecteur le plus court ou du vecteur le plus proche d'une cible donnée. Ces deux problèmes sont connus pour être NP-complets et constituent précisément la difficulté à la base des cryptosystèmes reposant sur les réseaux euclidiens. Tandis que ces questions apparaissent simples et intuitives en basse dimension, elles deviennent rapidement très complexes et coûteuses à résoudre en grande dimension.

Pour mieux comprendre ces difficultés intrinsèques, il est pertinent d'étudier les réseaux polynomiaux, une classe analogue aux réseaux euclidiens mais algorithmiquement plus abordable. Alors que les réseaux euclidiens posent des problèmes NP-difficiles, les réseaux polynomiaux peuvent être réduits exactement et efficacement en temps polynomial. Cette différence fondamentale ouvre une piste prometteuse : serait-il possible d'adapter certaines méthodes exactes de réduction, initialement développées pour les réseaux polynomiaux, au cas des réseaux entiers ?

L'objectif précis de ce stage est donc d'explorer et de comparer les réseaux euclidiens et polynomiaux, et plus particulièrement d'analyser la possibilité d'adapter les techniques exactes issues des réseaux polynomiaux au contexte des réseaux entiers. Cette étude vise à évaluer l'efficacité potentielle de telles adaptations, à identifier les obstacles théoriques ou pratiques à surmonter, et à mieux comprendre les implications sur la sécurité cryptographique.

La pertinence cryptographique de cette démarche s'inscrit dans un contexte où les systèmes de sécurité actuels, reposant sur la difficulté de la factorisation des grands nombres premiers et du calcul de logarithmes discrets, risquent d'être mis à mal par les progrès en informatique quantique. À l'opposé, les réseaux euclidiens apparaissent comme une alternative prometteuse en cryptographie post-quantique, plusieurs de leurs problèmes fondamentaux semblant résister efficacement aux attaques quantiques. Un exemple emblématique de l'ambivalence algorithmique des réseaux euclidiens est l'algorithme LLL (Lenstra-Lenstra-Lovász), initialement célèbre pour avoir permis la cryptanalyse de systèmes basés sur le problème du sac à dos, mais qui joue aujourd'hui paradoxalement un rôle clé dans la conception de nouveaux schémas cryptographiques robustes. Ainsi, approfondir la compréhension et améliorer les techniques de réduction de réseaux euclidiens représente un enjeu crucial pour le développement futur de la cryptographie sécurisée face aux défis quantiques.

CHAPITRE 1

Réseaux euclidiens

L'ÉTUDE des réseaux euclidiens puise ses racines dans les travaux mathématiques du XVIII^{e} siècle, notamment ceux de Leonhard Euler sur l'organisation géométrique des points dans l'espace. C'est cependant en 1891 qu'Hermann Minkowski établit véritablement les fondements modernes avec l'introduction de la théorie géométrique des nombres, reliant explicitement les réseaux à divers problèmes d'optimisation et de minimisation. Ses résultats joueront ultérieurement un rôle déterminant dans le développement de la cryptographie moderne. Néanmoins, c'est au cours du XX^{e} siècle, et plus particulièrement à partir des années 1990, que les réseaux euclidiens connaissent une véritable intégration dans la cryptographie. Les travaux de chercheurs tels que Ajtai, Dwork ou Regev marquent alors un tournant décisif, en démontrant que les difficultés algorithmiques intrinsèques aux réseaux peuvent constituer une base solide pour la conception de nouveaux systèmes cryptographiques résistants aux attaques conventionnelles et quantiques. Ce chapitre vise précisément à introduire de manière approfondie les concepts fondamentaux liés aux réseaux euclidiens.

1.1 *Définitions et exemples*

Nous considérons un espace euclidien, c'est-à-dire un espace vectoriel réel de dimension finie muni d'un produit scalaire, noté $\langle f, g\rangle$. Dans ce chapitre, nous utiliserons le produit scalaire usuel défini par $\langle f, g\rangle := f \cdot g^t = \sum_{i=1}^{n} f_i g_i$, lequel induit la norme-2 donnée par $\|f\|_2 = \sqrt{\sum_{i=1}^{n} f_i^2}$. Pour alléger les notations, nous omettrons l'indice 2. Ce chapitre se contentera d'exposer les résultats classiques, sans chercher à en fournir de démonstration. Rappelons qu'au sein de $\mathbb{R}^n$, toutes les normes sont équivalentes, ce qui signifie qu'aucune ne change la nature intrinsèque des problèmes que nous aborderons. Nous nous concentrerons néanmoins sur la norme-2 pour sa simplicité géométrique, car elle offre une mesure intuitive des longueurs et des angles, point essentiel pour étudier la structure des réseaux euclidiens. Afin de simplifier l'écriture, nous désignerons par $\mathcal{B} = (b_1, \ldots, b_n)$ une base de $\mathbb{R}^n$ et par $\mathcal{B}^* = (b_1^*, \ldots, b_n^*)$ sa base orthogonalisée par le procédé de Gram-Schmidt. Les matrices B et B^* (sans calligraphie) seront respectivement les matrices dont les lignes sont $b_1, \ldots, b_n$ et $b_1^*, \ldots, b_n^*$. Ces différentes écritures faciliteront la manipulation des vecteurs dans la suite de ce manuscrit.

On commence par introduire la notion de réseau euclidien. Pour un rappel sur les groupes ou les modules le lecteur est invité à lire les annexes correspondantes. Le point de vue de module nous sera utile dans la suite de ce manuscrit.

Définition 1.1. Soit $n \in \mathbb{N}^*$, $b_1, \ldots, b_n \in \mathbb{R}^n$. Les définitions suivantes sont équivalentes. [1]

- Un **réseau euclidien** $\mathscr{L}$ est un sous-groupe discret additif de $\mathbb{R}^n$.
  - — Sous-groupe additif : $0 \in \mathscr{L}$, et pour tout $x, y \in \mathscr{L}$, $x + y, -x \in \mathscr{L}$
  - — Discret : $\forall x \in \mathscr{L}, \exists \epsilon > 0$ tel que $B(x, \epsilon) \cap \mathscr{L} = x$ (où $B(x, \epsilon)$ désigne la boule ouverte de rayon ϵ centrée en x)
- Un **réseau euclidien** $\mathscr{L}$ est un $\mathbb{Z}$-module libre de type fini de $\mathbb{R}^n$.

1. Plusieurs définitions équivalentes d'un réseau euclidien coexistent dans la littérature, selon que l'on se place ou non dans un espace euclidien $\mathbb{R}^n$, ou dans un espace $\mathbb{R}^n$ équipé explicitement d'une forme quadratique définie positive. Dans tous les cas, l'idée générale reste la même : un réseau euclidien est un sous-ensemble discret de $\mathbb{R}^n$ formé par toutes les combinaisons linéaires entières d'un ensemble de vecteurs générateurs.

Définition 1.2. Un **sous-réseau** $\mathscr{L}'$ de $\mathscr{L}$ est un sous groupe de $\mathscr{L}$, on notera $\mathscr{L}' \subseteq \mathscr{L}$.

Exemple. Les entiers de Gauss, définit par $\mathbb{Z}[i] := \mathbb{Z} \oplus i\mathbb{Z}$ forment un réseau de rang 2 dans $\mathbb{C}$, c'est même un anneau.

Exemple. Un exemple plus exotique, $\mathbb{Z} \oplus \sqrt{2} \cdot \mathbb{Z}$ est un réseau de rang 2 dans $\mathbb{R}$.

Contre exemple. $\mathbb{Q}$ n'est pas un réseau euclidien , car $\mathbb{Q}$ est dense dans $\mathbb{R}$, ce qui brise la discrétude, bien que ce soit un sous-groupe de $\mathbb{R}$.

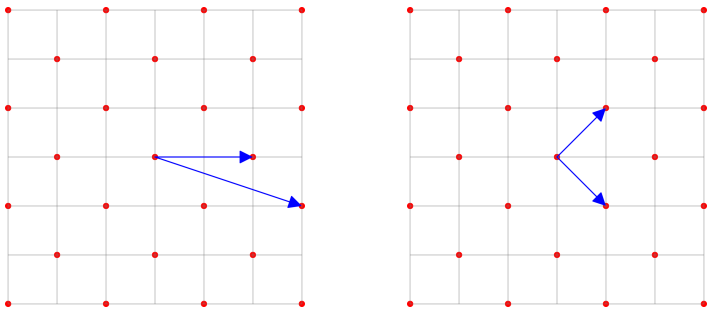
En particulier on peut montrer qu'il existe une famille $\mathbb{Z}$-libre maximale $(b_i)_{1\leq i\leq m}$ dans $\mathscr{L}$ telle que

$$\mathscr{L} = \bigoplus_{1\leq i\leq m} \mathbb{Z}b_i := \{a_1b_1 + \cdots a_mb_m : a_i \in \mathbb{Z}\}$$

Cette famille est appelée **base** de $\mathscr{L}$, si on note B la matrice de la famille $(b_i)_{1\leq i\leq m}$ on notera $\mathscr{L}(B)$ le réseau de base B, donc **engendré** par la famille $(b_i)_{1\leq i\leq m}$. L'entier m est commun à toutes les bases de $\mathscr{L}$ et on l'appelle **rang** de $\mathscr{L}$. Lorsque $n = m$, on dit que le réseau est de **rang plein**.

Exemple. [2] On a la suite d'inclusions $2\mathbb{Z} \subset \mathbb{Z} \subset \frac{1}{2}\mathbb{Z}$. Bien que $\mathrm{rang}(2\mathbb{Z}) = \mathrm{rang}(\mathbb{Z}) = \mathrm{rang}(\frac{1}{2}\mathbb{Z})$, ces ensembles sont distincts. Cela montre que, contrairement aux espaces vectoriels, avoir une relation d'inclusion et avoir le même rang ne suffit pas à garantir l'égalité des réseaux.

Existe-t-il une notion de "bonne" base ? Nous verrons qu'une base idéale est celle qui est la plus orthogonale possible. Il n'existe pas toujours de base strictement orthogonale, ce qui justifie la notion de quasi-orthogonalité. Nous allons rajouter des façons de mesurer la qualité d'une base dans le chapitre suivant et introduire la notion de réduction, qui consistera à trouver une bonne base.



(a) $B = \begin{pmatrix} 3 & -1 \\ 2 & 0 \end{pmatrix}$. (b) $B' = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

FIGURE 1.1 – Deux bases pour le même réseau de $\mathbb{R}^2$ tel que $B = \begin{pmatrix} 1 & 2 \\ 1 & 1 \end{pmatrix} B'$.

Définition 1.3. La **taille** [3] d'un réseau $\mathscr{L}(B)$ est $|\det(B)|$ et est noté $|\mathscr{L}|$. La taille d'un réseau est indépendante de la base choisie.

Remarque. On peut définir de façon équivalente $|\mathscr{L}| := \sqrt{\det(\mathrm{Gram}(B))}$ où $\mathrm{Gram}(B) = B^t \cdot B$. On peut voir $|\mathscr{L}|$ comme le **volume du domaine fondamental** de $\left\{ \sum_{i=0}^{n} \lambda_i b_i \mid 0 \leq \lambda_i < 1 \right\}$.

2. On rappelle que pour $a \in \mathbb{R}$, on a $a\mathbb{Z} = \{an \in \mathbb{R} | n \in \mathbb{Z}\}$.

3. **taille** et **volume** sont synonymes, mais l'usage le plus courant dans la littérature anglaise est "déterminant du réseau" ou "volume".

Théorème 1.1 (Inégalité d'Hadamard)**.** Soit $B \in M_n(\mathbb{K})$ et $\mathscr{L}(B)$ un réseau. Alors

$$|\mathscr{L}| = |\det(B)| \leq \prod_{i=1}^{n} \|b_i\|$$

La borne est atteinte si, et seulement si $(b_i)_{1\leq i\leq n}$ est une famille orthogonale.

Toutes les bases d'un réseau euclidien diffèrent d'une transformation de déterminant ± 1. L'ensemble de ces transformations est connu sous le nom de groupe unimodulaire. On remarque également qu'un ensemble de points non alignés dans un réseau ne constitue pas nécessairement une base si son déterminant est différent de $\pm|\mathscr{L}|$.

Proposition 1.1. Soit $\mathscr{L}$ et $\mathscr{L}'$ deux réseaux de rang n de base B et B'. Alors $\mathscr{L} = \mathscr{L}'$ si et seulement si il existe $U \in \mathrm{GL}_n(\mathbb{Z})$ tel que $B' = BU$. Où $\mathrm{GL}_n(\mathbb{Z}) = \{M \in \mathbb{Z}^{n\times n} | \det(M) = \pm 1\}$

Remarque. On a l'action de groupe

$$\begin{aligned} \mathrm{GL}_n(\mathbb{Z}) \times \mathrm{GL}_n(\mathbb{R}) &\longrightarrow \mathrm{GL}_n(\mathbb{R}) \\ (U, \mathbf{B}) &\longmapsto \mathbf{B}U \end{aligned}$$

Un réseau est exactement une orbite de cette action. La réduction de réseaux consiste à trouver un bon représentant pour chaque orbite.

Nous présentons maintenant deux **invariants** fondamentaux d'un réseau :

— La **taille du vecteur minimal** du réseau, notée $\lambda_1(\mathscr{L})$,
— Le **volume du réseau**, aussi appelé la **taille du réseau** souvent désigné par $|\mathscr{L}|$.

Proposition 1.2. Soit $\mathscr{L}$, $\mathscr{L}'$ deux réseaux de $\mathbb{R}^n$ tel que $\mathscr{L}' \subseteq \mathscr{L}$ alors $\frac{|\mathscr{L}'|}{|\mathscr{L}|} \in \mathbb{N}$.

Remarque. Ce résultat est une conséquence directe du théorème de Lagrange.

Définition 1.4. On appelle **minimum d'un réseau**[4] $\mathscr{L}$ la quantité

$$\lambda_1(\mathscr{L}) = \min_{\substack{v\in\mathscr{L} \\ v\neq 0}} \|v\|$$

Plus généralement, pour $k \in \{1, \ldots, n\}$, on pose $\lambda_k(\mathscr{L})$ le plus petit réel r tel qu'il existe k vecteurs $\mathbb{R}$-linéairement indépendants dans $\mathscr{L}$ de norme au plus r.

Remarque. $\lambda_1(\mathscr{L})$ correspond à la distance minimale entre deux points quelconques de $\mathscr{L}$.

Exemple. Soit $\mathscr{L}$ le réseau de la figure 1.1. On a $|\mathscr{L}| = 2$ et $\lambda_1(\mathscr{L}) = \lambda_2(\mathscr{L}) = \sqrt{2}$.

Théorème 1.2 (Premier théorème de Minkowski)**.** Pour tout $n \in \mathbb{N}^*$, il existe une constante $C_n > 0$ telle que pour tout réseau $\mathscr{L}$ de $\mathbb{R}^n$, on a :

$$\lambda_1(\mathscr{L}) \leq C_n |\mathscr{L}|^{1/n}$$

4. on peut aussi le définir comme $\lambda_1(\mathscr{L}) = \min\{r > 0 : |\mathcal{B}(r) \cap \mathscr{L}| > 1\} \in \mathbb{R}_+$

On peut prendre cette constante égale à $C_n = (2/\sqrt{\pi})\Gamma(n/2+1)^{1/n}$.[5] On appelle **constante de Hermite-Minkowski** le carré de la constante optimale possible pour cette inégalité, noté γ_n, en particulier, on a $\gamma_n \leq C_n$. En développant, on a

$$\gamma_n = \sup_{\dim(\mathscr{L})=n} \gamma(\mathscr{L}), \quad \text{où } \gamma(\mathscr{L}) := \frac{\lambda_1(\mathscr{L})^2}{\det(\mathscr{L})^{2/n}}$$

Proposition 1.3.

$$\gamma_n = 4\left(\frac{\Delta_n}{V_n}\right)^{2/n} \quad \forall n \in \mathbb{N}^*$$

où $V_n = \frac{\pi^{n/2}}{\Gamma(\frac{n}{2}+1)}$ et Δ_n est la densité d'un empilement compact de densité maximum d'hypersphères.

On ne connaît pas la valeur exacte de γ_n pour tout n. Seuls les valeurs dans le tableau suivant sont connues de manières exactes.[6]

n	1	2	3	4	5	6	7	8	24
γ_n^n	1	$\frac{4}{3}$	2	4	8	$\frac{64}{3}$	64	256	4^{24}

FIGURE 1.2 – Valeurs connues de γ_n

On sait que $(\gamma_n)_n$ est une suite d'ordre de croissance linéaire mais on ne sait pas si elle est croissante.

Exemple. Le **réseau d'Eisenstein**, ou **réseau en nid d'abeille**, est un réseau de $\mathbb{R}^2$ de rang 2, engendré par la base $\begin{pmatrix} 1 & \frac{1+\sqrt{3}}{2} \\ 1 & \frac{1-\sqrt{3}}{2} \end{pmatrix}$. On a $|\mathscr{L}| = \sqrt{3}, \lambda_1 = \sqrt{2}, \gamma(\mathscr{L}) = \frac{2}{\sqrt{3}}$. En traçant des sphères de centre les points du réseaux et de rayon $\sqrt{2}$, on obtient l'empilement compact le plus dense en dimension 2.

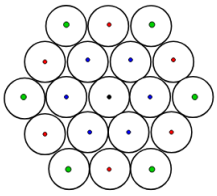


FIGURE 1.3 – Empilement hexagonal dans $\mathbb{R}^2$

Proposition 1.4. Soit $\mathscr{L} \subset \mathbb{R}^n$ un réseau de base $(b_i)_{1\leq i\leq n}$ et sa base de Gram-Schmidt associée $(b_i^*)_{1\leq i\leq n}$. Alors pour tout $v \in \mathscr{L}\backslash\{0\}$, on a

$$\lambda_1(\mathscr{L}) \geq \|v\| \geq \min_{1\leq i\leq n} \|b_i^*\|$$

Tous ces résultats et inégalités vont nous donner des critères de mesure de la qualité de réduction d'une base dans le chapitre sur la réduction de réseaux euclidiens.

5. La fonction Γ, appelée fonction gamma, généralise la notion de factorielle aux nombres réels (et complexes). Elle est définie, pour tout $z \in \mathbb{C}$, par la formule suivante :

$$\Gamma(z) = \int_0^{+\infty} t^{z-1}e^{-t}\,dt.$$

6. **En dimension** $n = 1$, n'importe quel réseau de dimension 1 atteint cette borne. **En dimension** $n = 2$, le réseau optimal est celui des entiers d'Eisenstein, également appelé réseau en nid d'abeille. **En dimension** $n = 3$, le réseau optimal est présenté dans (HALES et al. 2015), mais sa démonstration nécessite environ 130 pages de minimisation de fonctions analytiques. Des avancées majeures ont été obtenues pour les **dimensions** $n = 8$ et $n = 24$ grâce à la mathématicienne ukrainienne Maryna Viazovska, qui a démontré l'optimalité du réseau E_8 (VIAZOVSKA 2016) et, en collaboration, celle du réseau de Leech (COHN et al. 2016).

1.2 *Quelques problèmes algorithmiques liés aux réseaux euclidiens*

Cette section s'inspire largement des travaux de Boudgoust (2023), auxquels le lecteur intéressé pourra se référer pour un traitement plus approfondi. On définira deux problèmes algorithmiques important sur les réseaux euclidiens. Il y en a beaucoup plus, (Stephens-Davidowitz 2015) donne un aperçu des réductions entre ces problèmes.

1.2.1 *Des problèmes faciles*

Certains problèmes liés aux réseaux euclidiens sont relativement simples à résoudre, notamment la vérification de l'appartenance d'un vecteur à un réseau donné, ou encore la décision de l'égalité de deux bases de réseaux. Le lecteur intéressé pourra s'exercer en tentant de résoudre ces questions.

> ***Adhésion***
> Étant donné une base B d'un réseau $\mathscr{L}$, et $v \in \mathbb{R}^n$, décider si $v \in \mathscr{L}$.

> ***Équivalence***
> Étant donné deux bases B et B', décider si $\mathscr{L}(B) = \mathscr{L}(B')$,

1.2.2 *Le problème du vecteur le plus court*

Considérons le problème suivant, paramétré par la dimension n du réseau :

> ***Shortest Vector Problem (SVP), NP-complet, (Ajtai 1996).***
> Étant donné une base B d'un réseau $\mathscr{L}$, trouver un vecteur $v \neq 0$ tel que $\|v\|_2 = \lambda_1(\mathscr{L})$.

On ne connaît que des algorithmes demandant au moins un nombre exponentiel d'opérations pour résoudre ce problème, même en utilisant des algorithmes quantiques. Les algorithmes de type **énumération**[7] et les algorithmes de type **crible**[8] se démarquent pour ce problème. Le calcul d'un plus court vecteur dans un réseau euclidien de $\mathbb{R}^n$ est en général un problème difficile qui sert de fondation à de nombreuses primitives cryptographiques. On s'intéresse souvent à la version approximative :

> ***SVP$_\gamma$, où $\gamma > 0$***
> Étant donné une base B du réseau $\mathscr{L}$, trouver un vecteur $v \neq 0$ tel que $\|v\|_2 \leq \gamma \cdot \lambda_1(\mathscr{L})$.

L'état des connaissances actuelles est le suivant :

- Pour $\gamma = \mathcal{O}(1)$, le problème est prouvé **NP-complet**, (Ajtai 1996).
- Pour $\gamma = \text{poly}(n)$, il existe des algorithmes en **temps exponentiel**.
- Pour $\gamma = 2^{\mathcal{O}(n)}$, l'algorithme **LLL** (Lenstra 1982) permet de le résoudre en **temps polynomial**.

> ***GapSVP$_\gamma$, où $\gamma > 0$***
> Étant donné une base B du réseau $\mathscr{L}$, et $r \in \mathbb{R}_+^*$. Décider si $\lambda_1(\mathscr{L}) \leq r$ (instance positive) ou $\lambda_1(\mathscr{L}) > \gamma \cdot r$ (instance négative).

7. ils énumèrent tous les vecteurs du réseau qui sont dans une certaine boule bien choisie, en pratique ils sont utilisés jusqu'aux dimensions $n \approx 80$. On peut leur ajouter des optimisations et des heuristiques.

8. on génère deux listes d'éléments du réseau, puis on construit la liste de toutes les différences entre les éléments des deux listes. On espère obtenir des vecteurs plus court. On recommence le procédé. Le temps d'exécution est en $2^{\mathcal{O}(n)}$.

Théorème 1.3 ((Banaszczyk 1993),(Stephens-Davidowitz 2015))**.** .
SVP_γ n'est pas plus simple que GapSVP_γ

Conjecture 1.1. Il n'existe aucun algorithme classique ou quantique en temps polynomial qui approxime les problèmes de réseaux SVP_γ, GapSVP_γ ou à un facteur polynomial près γ (pour tous les réseaux d'entrée possibles).

1.2.3 *Le problème du vecteur le plus proche*

Un autre problème important concerne la recherche de vecteurs proches d'une cible dans un réseau.

> ***Closest Vector Problem (CVP), NP-complet**, (Ajtai 1996).*
> Étant donnés $t \in \mathbb{R}^n$, un réseau $\mathscr{L}(B)$, trouver $v \in \mathscr{L}$ tel que $\|t-v\|_2 = d(t,\mathscr{L}) := \min_{v\in\mathscr{L}}\{\|t-v\|_2\}$.

Le problème **CVP** est en général difficile pour un réseau arbitraire. Cependant, pour certaines familles spécifiques de réseaux, comme $\mathbb{Z}^n$, des algorithmes en temps polynomial sont connus. La qualité de la base choisie joue un rôle crucial dans la résolution du problème. De même, on peut considérer une version approximative :

> ***CVP**$_\gamma$, $\gamma > 0$*
> Étant donnés $t \in \mathbb{R}^n$, un réseau $\mathscr{L}(B)$, trouver $v \in \mathscr{L}$ tel que $\|t - v\|_2 \leq \gamma \cdot d(t,\mathscr{L})$.

> ***GapCVP**$_\gamma$, $\gamma > 0$*
> Étant donné $r \in \mathbb{R}_+^*$, $t \in \mathbb{R}^n$, un réseau $\mathscr{L}(B)$. Décider si il existe $v \in \mathscr{L}$ tel que $\|t - v\|_2 \leq r$ (instance positive) ou $\|t - v\|_2 > \gamma \cdot r$ (instance négative)

Théorème 1.4 ((Goldreich et al. 1999))**.** .
GapSVP_γ se réduit à GapCVP_γ en temps polynomial.

Théorème 1.5. Il existe un algorithme qui résout $\mathbf{CVP}_{\exp(n)}$ en temps polynomial via l'algorithme **LLL**.

L'efficacité des algorithmes dépend grandement de la qualité de la base du réseau euclidien choisie. Le chapitre sur la réduction abordera des techniques pour améliorer la base, via l'algorithme **LLL**.[9]

9. En dimension fixée, résoudre **exactement** le problème **SVP** pour la norme $\|\cdot\|_\infty$ fournit en fait une γ-approximation (avec γ dépendant de la dimension) pour le problème **SVP** dans la norme $\|\cdot\|_2$. Il existe notamment une constante C telle que $\|v\|_2 \leq C\|v\|_\infty$ pour tout $v \in \mathbb{R}^n$. Ainsi, un vecteur minimisant $\|v\|_\infty$ donne un vecteur $\sqrt{n}$-proche du vecteur réellement le plus court en norme euclidienne.

CHAPITRE 2

Réseaux polynomiaux

LORSQUE les coefficients des matrices appartiennent à un corps $\mathbb{K}$, les opérations classiques telles que la multiplication, l'inversion, le calcul du déterminant ou la résolution de systèmes linéaires possèdent des complexités comparables. En revanche, lorsqu'on considère des matrices à coefficients dans l'anneau $\mathbb{K}[x]$, des différences apparaissent : si le calcul du déterminant conserve la même complexité que celle du produit matriciel, l'inversion est plus coûteuse. Cette complexité découle de la structure de l'anneau $\mathbb{K}[x]$: bien qu'il s'agisse d'un anneau principal (à la différence de $\mathbb{K}[x, y]$), il ne s'agit pas d'un corps. Ainsi, certaines opérations, comme l'inversion, ne sont plus systématiquement réalisables. Notamment, dans $\mathbb{K}[x]$, seuls les polynômes constants non nuls sont inversibles. Cette restriction impose de repenser et redéfinir rigoureusement plusieurs notions de l'algèbre linéaire. Les matrices à coefficients dans $\mathbb{K}[x]$ sont essentielles dans de nombreuses applications. [1]. Ce chapitre vise à explorer les réseaux polynomiaux et leurs propriétés spécifiques.

2.1 *Définitions et exemples*

On commence par introduire la notion de réseau polynomial. Des rappels détaillés sur les anneaux et sur les modules sont dans l'annexe correspondante.

Définition 2.1. Un **réseau polynomial** $\mathscr{L}$ est un $\mathbb{K}[x]$-module libre de type fini.

On peut montrer qu'il existe une famille $\mathbb{K}[x]$-libre maximale $(b_i)_{1\leq i\leq m}$ dans $\mathscr{L}$ telle que

$$\mathscr{L} = \bigoplus_{1\leq i\leq m} \mathbb{K}[x]b_i := \{a_1b_1 + \cdots a_mb_m : a_i \in \mathbb{K}[x]\}$$

Cette famille est appelée **base** de $\mathscr{L}$, si on note $B \in \mathbb{K}[x]^{m\times n}$ la matrice de la famille $(b_i)_{1\leq i\leq m}$ on notera $\mathscr{L}(B)$ le réseau de base B, donc **engendré** par la famille $(b_i)_{1\leq i\leq m}$. L'entier m est commun à toutes les bases de $\mathscr{L}$ et on l'appelle **rang** de $\mathscr{L}$. Lorsque $n = m$, on dit que le réseau est de **rang plein**. Un élément de $\mathbb{K}[x]^{m\times n}$ est appelé matrice polynomiale.

Exemple.

$$B = \begin{pmatrix} 3x+4 & x^9 \\ 5 & x^2+1 \end{pmatrix} \in \mathbb{R}[x]^{2\times 2}$$

est une matrice polynomiale qui représente le réseau $\mathscr{L}(B)$.

Proposition 2.1. Soient P et Q deux bases de lignes d'un même $\mathbb{F}[x]$-module libre . Alors, il existe une matrice unimodulaire U telle que

$$P \;=\; U\,Q.$$

On observe une analogie structurelle entre les matrices et les modules : de même que les matrices à coefficients dans $\mathbb{K}$ sont naturellement liées aux $\mathbb{K}$-espaces vectoriels, les matrices à coefficients dans $\mathbb{K}[x]$ interviennent dans l'étude des $\mathbb{K}[x]$-modules libres.

1. Par exemple dans l'interpolation bivariée, une étape centrale du décodage des codes de Reed-Solomon

2.2 *Deux points de vue sur les matrices polynomiales*

Théorème 2.1. On dispose d'un **isomorphisme structurel** au sens des modules :

$$\mathbb{K}[x]^{m\times n} \;\cong\; \mathbb{K}^{m\times n}\big[x\big]$$

Exemple.

$$\begin{pmatrix} 3x+4 & x^9 \\ 5 & x^2+1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x^9 + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} x^2 + \begin{pmatrix} 3 & 0 \\ 0 & 0 \end{pmatrix} x + \begin{pmatrix} 4 & 0 \\ 5 & 1 \end{pmatrix}$$

On peut donc interpréter une matrice polynomiale soit comme un polynôme à coefficients dans matriciel, soit comme une matrice à coefficients polynomiaux. Le choix de l'approche dépend alors principalement du type de calculs et d'estimations de complexité qu'on souhaite mener. Deux approches principales sont alors envisageables pour effectuer des calculs algébriques (résolution de systèmes, inversion, déterminant, etc.) :

1. **Appliquer les algorithmes d'algèbre linéaire classique sur** $\mathbb{K}[x]^{m\times n}$.
   On traite la matrice comme un objet usuel, en considérant simplement que les coefficients se trouvent dans l'anneau principal $\mathbb{K}[x]$.
   - **Avantage** : cette méthode tire parti de la robustesse théorique et de l'expérience accumulée avec les algorithmes classiques.
   - **Limite** : on peut rapidement générer des calculs complexes, par exemple des fractions polynomiales de grand degré, et obtenir des bornes de complexité peu réaliste.

2. **Appliquer des algorithmes d'arithmétique polynomiale sur** $\mathbb{K}^{m\times n}[x]$
   - **Avantage** : on bénéficie de techniques optimisées pour les polynômes (accélération de la multiplication via FFT, etc.), on a un meilleur contrôle du degré.
   - **Limite** : cela peut parfois s'avérer restrictif[2] ou inefficace[3] , en particulier si les degrés des entrées diffèrent sensiblement d'une ligne ou d'une colonne à l'autre.

2.3 *Complexité des opérations arithmétique*

Ce mémoire avait pour ambition initiale de motiver rigoureusement l'introduction des bases réduites dans le cadre des matrices polynomiales. Toutefois, afin de ne pas aborder un sujet trop éloigné des objectifs du stage, et par souci de concision, nous ne détaillerons pas ici les aspects liés aux mesures de complexité. Notons simplement qu'il est essentiel d'analyser finement le comportement des matrices polynomiales vis-à-vis des opérations algébriques usuelles, en particulier la multiplication. Cela justifie l'introduction de la notion de degré de ligne, qui jouera un rôle central dans le chapitre suivant.

Définition 2.2. Soit $A \in \mathbb{K}[x]^{m\times n}$, la **taille** de A, notée $\mathrm{size}(A)$, est le nombre de coefficients distincts de $\mathbb{K}$ nécessaires pour sa représentation dense.
Et on a la relation

$$\mathrm{size}(A) = \sum_{i,j} \mathrm{size}(a_{i,j}) = \sum_{i,j} (1 + \max(0, \deg(a_{i,j}))).$$

2. Dans la division avec reste, on suppose souvent que B possède un coefficient dominant inversible ($\mathrm{lc}(B) \neq 0$). On pourrait toutefois assouplir cette hypothèse, avec la notion de base "réduite" qui sera définie dans les prochains chapitres

3. Si l'on travaille avec une matrice de degré d dont de nombreuses entrées ont un degré bien inférieur à d, les algorithmes basés uniquement sur le degré maximal risquent de fournir des performances dégradées.

En général, la taille n'est pas compatible avec le produit matriciel, mais cela peut être le cas dans certains cas particuliers.

Notation. On note $[d]$ un polynôme de degré d. Par exemple x^2+1 sera noté $[2]$, x^9 sera noté $[9]$.

Exemple. La multiplication ne se passe pas forcément bien. On voit que cela donne des bornes non pertinentes et il serait utile de rajoute des critères de mesure du degré. Considérons les matrices de degrés :

$$\begin{pmatrix}[100] & [1]\\ [100] & [1]\end{pmatrix}\begin{pmatrix}[1] & [1]\\ [1] & [1]\end{pmatrix} = \begin{pmatrix}[101] & [101]\\ [101] & [101]\end{pmatrix}$$

Définition 2.3. .

- Pour $m = (m_1, \cdots, m_n) \in \mathbb{K}[x]^{1\times n}$, on définit son **degré en ligne** par :

$$\mathrm{rdeg}(m) = \max_{1\le i\le n} \deg(m_i) \in \mathbb{Z}$$

- Pour $M = \begin{pmatrix}\cdots M_1 \cdots\\ \vdots \\ \cdots M_n \cdots\end{pmatrix}$, $M_i \in \mathbb{K}[x]^{1\times n}\ \ \forall 1 \le i \le n$, on définit son **degré en ligne** par :

$$\mathrm{rdeg}(M) = (\mathrm{rdeg}(M_i))_{1\le i\le n} \in \mathbb{Z}^n$$

Exemple. Soit $M = \begin{pmatrix}3x+4 & x^9\\ 5 & x^2+1\end{pmatrix} \in \mathbb{F}_2[x]$. Alors $\mathrm{rdeg}(M) = \begin{pmatrix}\mathrm{rdeg}(\ 3x+4\ ,\ x^9\ )\\ \mathrm{rdeg}(\ 5\ ,\ x^2+1\ )\end{pmatrix} = \begin{pmatrix}9\\ 2\end{pmatrix}$

Cette définition du degré de ligne présente une limite : si $c = bA$, on a bien en général $\mathrm{rdeg}(c) \le \mathrm{rdeg}(b) + \mathrm{rdeg}(A)$, mais cette majoration est souvent trop lâche pour nos besoins. Ce qui nous intéresse est de pouvoir caractériser plus finement le degré de c, voire d'obtenir une égalité. Cela motive l'introduction de la définition de degré décalé.

Définition 2.4. Soit $s = (s_1, \cdots, s_n) \in \mathbb{Z}^n$. On appelle s le **vecteur de décalage**.

- Pour $m = (m_1, \cdots, m_n) \in \mathbb{K}[x]^{1\times n}$, on définit son **degré en ligne décalé** par :

$$\mathrm{rdeg}_s(m) = \max_{1\le i\le n} (\deg(m_i) + s_i)$$

- Pour $M = \begin{pmatrix}\cdots M_1 \cdots\\ \vdots \\ \cdots M_n \cdots\end{pmatrix}$, $M_i \in \mathbb{K}[x]^{1\times n}\ \ \forall 1 \le i \le n$, on définit son **degré en ligne décalé** par :

$$\mathrm{rdeg}_s(M) = \big(\mathrm{rdeg}_s(M_i)\big)_{1\le i\le m} \in \mathbb{Z}^m$$

Notation. Soit $s = (s_1, \cdots, s_n) \in \mathbb{Z}^n$. On note x^s la matrice diagonale $\begin{pmatrix}x^{s_1} & & \\ & \ddots & \\ & & x^{s_n}\end{pmatrix} \in M_n(\mathbb{K}[x])$.

Proposition 2.2. Soit $A \in \mathbb{K}[x]^{m\times n}$, et $s \in \mathbb{Z}^n$. Alors $\mathrm{rdeg}_s(A) = \mathrm{rdeg}(Ax^s)$.

Exemple. Soit $M = \begin{pmatrix}3x+4 & x^9\\ 5 & x^2+1\end{pmatrix} \in \mathbb{R}_2[x]$ et $s = (8, 0)$

Alors

$$\mathrm{rdeg}_s(M) = \mathrm{rdeg}(M \cdot x^s) = \mathrm{rdeg}\begin{pmatrix}3x^9+4x^8 & x^9\\ 5x^8 & x^2+1\end{pmatrix} = (9, 8)$$

Proposition 2.3. Soit $A \in \mathbb{K}[x]^{m\times n}$ et $s \in \mathbb{Z}^n$. Alors, on a les propriétés suivantes :
- $\mathrm{rdeg}_s(A) = v$ si et seulement si $\mathrm{rdeg}(x^{-v}Ax^s) = 0$.
- $\mathrm{rdeg}_s(A) \leq v$ si et seulement si $\mathrm{rdeg}(x^{-v}Ax^s) \leq 0$.

Exemple. Soit

$$F = \begin{pmatrix} 1 & 0 & 1 \\ x & 1 & x+1 \\ 1 & x^3+x^2 & x \end{pmatrix}, \quad u = (1,0,0,1).$$

Alors

$$v = \mathrm{rdeg}_u(F) = (1,2,3,4) \quad \text{et} \quad x^{-v}Ax^u = \begin{pmatrix} 1 & 0 & x^{-1} \\ 1 & x^{-2} & x^{-2}+x^{-1} \\ x^{-2} & x^{-1}+1 & x^{-2} \end{pmatrix}.$$

Proposition 2.4. Soit $A \in \mathbb{K}[x]^{m\times n}$, $b \in \mathbb{K}[x]^{1\times m}$ et $c = bA$. Soit $v = \mathrm{rdeg}_u(A)$ et $w = \mathrm{rdeg}_v(b)$. Alors

$$\mathrm{rdeg}_u(c) \leq w.$$

On va définir un ordre sur les degrés de ligne, bien que non total.

Définition 2.5. Soit $m \in \mathbb{N}^*$ et soient $u = (u_1, \ldots, u_m)$, $v = (v_1, \ldots, v_m) \in \mathbb{Z}^m$ deux vecteurs de degrés de ligne, triés par valeur croissante. On définit une relation d'ordre partiel, notée $\leq_{ob}$ (ordre obtenu par composantes), par :

$$u \leq_{ob} v \quad \text{si et seulement si} \quad u_i \leq v_i \quad \text{pour tout } i \in \{1, \ldots, m\}.$$

Les notions de degrés introduites précédemment, ainsi que les techniques qui en découlent, permettent d'accélérer certains algorithmes dans des situations spécifiques. Cependant, elles présentent des limites structurelles importantes : notamment, les degrés de lignes et de colonnes ne possèdent pas de bonnes propriétés vis-à-vis de la multiplication matricielle. De plus, plusieurs problèmes restent ouverts quant à la possibilité d'obtenir des algorithmes plus efficaces pour le calcul du déterminant ou de l'inverse de matrices polynomiales. Ces obstacles mettent en évidence l'intérêt crucial de la réduction des matrices polynomiales, outil indispensable pour contrôler la croissance des degrés et améliorer ainsi les performances des algorithmes associés.

CHAPITRE 3

Réduction de réseaux polynomiaux

La réduction de réseaux joue un rôle central en cryptographie, car la difficulté à trouver une « bonne base » d'un réseau est précisément ce qui assure la robustesse de nombreux systèmes cryptographiques actuels et post-quantiques. La réduction consiste à transformer une base arbitraire en une base mieux adaptée aux calculs, c'est-à-dire constituée de vecteurs courts et quasi orthogonaux dans le cas des réseaux euclidiens. Cependant, la complexité algorithmique de ce processus varie considérablement selon la nature du réseau étudié :

— La réduction exacte de réseaux définis sur $\mathbb{F}_p[x]$ peut être réalisée efficacement, en temps polynomial.
— En revanche, la réduction exacte des réseaux définis sur $\mathbb{Z}$ est un problème connu pour être NP-difficile.

Ce chapitre est dédié à l'étude approfondie de ces deux types de réduction. Nous commencerons par examiner en détail les méthodes de réduction pour les réseaux polynomiaux, ce qui nous permettra de mieux identifier les obstacles intrinsèques à la réduction efficace des réseaux euclidiens, et d'en comprendre les implications cryptographiques.
La réduction des réseaux polynomiaux est une étape essentielle dans plusieurs applications algorithmiques, en particulier dans le décodage efficace des codes de Reed-Solomon généralisés. Cette opération vise à transformer une base quelconque d'un réseau polynomial en une base simplifiée ou réduite, facilitant ainsi les calculs ultérieurs. Dans cette section, nous détaillerons les principaux concepts, outils et algorithmes permettant de réaliser cette réduction en temps polynomial, en mettant l'accent sur les approches les plus performantes actuellement connues. Nous ferons ainsi un état de l'art des avancées récentes en matière de complexité et d'efficacité algorithmique, tout en discutant des défis encore ouverts dans le domaine. On notera $\mathbb{K}$ un corps quelconque.

3.1 *Généralités et notion de base d'ordre*

Soit $F \in \mathbb{F}[x]^{m\times n}$, et un **degré de précision** $\sigma \in \mathbb{N}$. On définit

$$(F,\sigma) := \{v \in \mathbb{F}[x]^{1\times m} \mid vF = 0 \mod x^\sigma\}$$

Proposition 3.1. (F,σ) est un réseau polynomial de dimension m.

On va s'intérésser à étudier les bases de ce réseau et définir une notion de base réduite.

Définition 3.1. Une (F,σ)-**base d'ordre** est une base [1] de (F,σ) de degré minimale.

Quelle est la définition du degré ? Que signifie "minimale" dans ce contexte ? Le lecteur est invité à relire les définitions de degré de ligne et degré de ligne décalé dans le chapitre précédent si il ne se sent plus familier avec la notation.

1. au sens des $\mathbb{F}[x]$-modules

Définition 3.2. Soit $F \in \mathbb{F}[x]^{m\times n}$. On dit que F est **réduite par ligne** si, pour tout $U \in \mathbb{F}[x]^{m\times m}$ unimodulaire, on a :

$$\mathrm{rdeg}(F) \leq_{ob} \mathrm{rdeg}(UF).$$

Pour parler d'un minimum, il faut un ordre total. On voit enfin la nouvelle définition de base réduite.

Définition 3.3. Une **base d'ordre** est une base de (F, σ) qui est réduite par ligne.

Proposition 3.2. Il existe une base réduite par ligne de (F, σ).

Exemple. $$\begin{pmatrix} 1 & 0 & 1 \\ x & 1+x & 0 \\ x^2+x^3 & x & 0 \end{pmatrix} \cdot \begin{pmatrix} x+x^2+x^3+x^4+x^5+x^6 \\ 1+x+x^5+x^6+x^7 \\ 1+x^2+x^4+x^5+x^6+x^7 \end{pmatrix} = 0^{4\times 1} \mod x^8$$

L'existence d'une base réduite est garantie, mais elle n'est pas nécessairement unique. Pour assurer l'unicité, une condition supplémentaire est requise : la forme de Popov.

Preuve naïve (incorrecte). Considérons le minimum de tous les $\mathrm{rdeg}(PU)$ triés, pour toutes les matrices unimodulaires $U \in \mathbb{F}[x]^{m\times m}$.
Toute base PU ayant un degré minimal est une base d'ordre. Attention : l'ordre $\leq_{ob}$ n'est pas un ordre total. En effet, il est possible d'avoir deux bases dont les degrés en ligne sont respectivement $(1, 2, 3)$ et $(1, 1, 4)$. On ne peut pas encore garantir l'existence d'un minimum ! ■

Définition 3.4. Soit $A \in \mathbb{F}[x]^{m\times n}$ et soit $v = \mathrm{rdeg}_u(A)$. On définit la **matrice des coefficients dominants**[2] de A, notée $\mathrm{lcoeff}(A) \in \mathbb{F}^{m\times n}$, comme étant la matrice obtenue en extrayant la partie constante de $x^{-v}A$, c'est-à-dire :

$$\mathrm{lcoeff}(A) = \lim_{x\to\infty} x^{-v}A.$$

Exemple. Soit

$$F = \begin{pmatrix} 1 & 0 & 1 \\ x & 1 & 1+x \\ 1 & x^2+x^3 & x \end{pmatrix}, \quad \vec{v} := \mathrm{rdeg}(F) = (0, 1, 3)$$

Alors

$$x^{-\vec{v}} \cdot F = \begin{pmatrix} 1 & 0 & 1 \\ 1 & x^{-1} & x^{-1}+1 \\ x^{-3} & x^{-1}+1 & x^{-2} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} + \mathcal{O}_{x\to\infty}(x^{-1})$$

Proposition 3.3 (Transitivité, revisitée)**.** Soient $c := b \cdot A$, $v = \mathrm{rdeg}_u(A)$ et $w = \mathrm{rdeg}_v(b)$.
Si $\mathrm{lcoeff}(A)$ est **injective à gauche**, alors $\mathrm{rdeg}_u(c) = w$.

Proposition 3.4. Soit A une matrice. Si $\mathrm{lcoeff}(A)$ est injective à gauche, alors A est réduite par ligne.

2. La matrice des coefficients dominants est accessible dans `SageMath` via la méthode `leading_matrix()`.

Maintenant que l'on connaît la définition d'une base réduite, comment la calculer efficacement ?

Définition 3.5. On définit le **pivot** d'une ligne comme l'élément non nul de degré maximal le plus à droite dans cette ligne.

Définition 3.6. Soit $W \in \mathbb{F}[x]^{m\times n}$. La matrice W est dite en **forme Popov faible** si chaque ligne de W possède un pivot, et si les indices de colonnes de ces pivots sont distincts deux à deux.

Exemple. Soit $W_1, W_2 \in M_2(\mathbb{Z}_7[x])$ tel que

$$W_1 = \begin{pmatrix} 3x+4 & x^9 \\ 5 & x^2+1 \end{pmatrix},\ W_2 = \begin{pmatrix} 2x^7+5x^5+3x+4 & x^5 \\ 5 & x^2+1 \end{pmatrix}$$

On note les pivots en rouge. W_1 n'est pas en forme de Popov faible car les pivots sont sur la même colonne, et W_2 est en forme Popov faible , car les pivots, ont des indices de colonnes distincts.

Pour transformer une matrice en forme Popov faible, on peut utiliser l'algorithme proposé dans (Mulders et Storjohann 2003), qui fournit une méthode systématique pour y parvenir en appliquant des transformations unimodulaires par lignes.

Définition 3.7. On appelle **transformation simple** de la ligne k sur la ligne l l'opération consistant à soustraire cx^e fois la ligne k à la ligne l, où $c \in \mathbb{F}$ et $e \in \mathbb{N}$. On dit qu'elle est du **premier type** si les pivots de la ligne k et de la ligne l ont les mêmes indices et du **deuxième type** sinon.

[3]

Algorithme 1 : *WeakPopovForm (Mulders et Storjohann 2003)*

Entrée : $\mathcal{M} \in \mathbb{F}[x]^{n\times m}$
Sortie : $\mathcal{N}$ en forme de Popov faible, obtenue par des transformations simples de premier type appliquées à $\mathcal{M}$

$A \leftarrow$ copie($\mathcal{M}$)
Tant que *A n'est pas en forme de Popov faible* **faire**
  Appliquer une transformation simple du premier type à A
$\mathcal{N} \leftarrow$ copie(A)
Retourner $\mathcal{N}$

Théorème 3.1 (Correction et complexité)**.** .
L'algorithme WeakPopovForm est correct. Sa complexité est bornée par $\mathcal{O}(nmrd^2)$ opérations dans le corps de base, où r désigne le rang de la matrice M, et d une borne supérieure sur le degré des coefficients de $\mathcal{M}$.

Exemple.

$$\begin{pmatrix} 3x+4 & x^9 \\ 5 & x^2+1 \end{pmatrix} \xrightarrow{(1)} \begin{pmatrix} 2x^7+3x+4 & 6x^7 \\ 5 & x^2+1 \end{pmatrix} \xrightarrow{(2)} \begin{pmatrix} 2x^7+5x^5+3x+4 & x^5 \\ 5 & x^2+1 \end{pmatrix}$$

1. Ajouter $6x^7$ fois la deuxième ligne à la première ligne.
2. Ajouter x^5 fois la deuxième ligne à la première ligne.

La matrice finale est en forme Popov faible, les pivots ont des indices de colonnes distincts.

Théorème 3.2. Toute matrice en *forme Popov faible* est réduite par ligne.

3. L'algorithme suivant est disponible dans `SageMath` via la méthode `weak_popov_form()`.

3.2 *Algorithmes de réduction de réseaux polynomiaux*

Cette section va montrer qu'il existe des algorithmes plus rapide que (MULDERS et STORJOHANN 2003) pour calculer des bases d'ordres.

3.2.1 *Cas initial quand $\sigma = 1$*

Si $\sigma = 1$, alors

$$\begin{aligned}(F, \sigma) &= \{v \in \mathbb{F}[x]^{1\times m} \mid vF = 0 \mod x^1\} \\ &= \{v \in \mathbb{F}^{1\times m} \mid vF = 0\}\end{aligned}$$

Soit $F \in \mathbb{F}^{m\times n}$, nous cherchons une base de $(F, 1)$.
On remarque que si

$$\begin{pmatrix} S \\ K \end{pmatrix} F = \begin{pmatrix} R \\ 0 \end{pmatrix} \text{ avec } R \text{ de rang maximal}$$

alors

$$\begin{pmatrix} xS \\ K \end{pmatrix} F = \begin{pmatrix} xR \\ 0 \end{pmatrix} = 0 \mod x$$

ce qui implique que

$$\begin{pmatrix} xS \\ K \end{pmatrix} \text{ est une base de } (F, 1)$$

Il existe des candidats naturels pour S et K : K le noyau de F et S le supplémentaire de K. On pourra choisir le noyau K utilisant les lignes de F de plus petit degré, une façon de les calculer consiste alors à obtenir la forme échelonnée par lignes de F.

$$L \cdot \tau \cdot F = E$$

où

$E = \begin{pmatrix} E' \\ 0 \end{pmatrix}$ est ligne échelonnée.

$L = \begin{pmatrix} L_r & 0 \\ G & I_{m-r} \end{pmatrix}$ est triangulaire inférieure.

τ est une permutation.

r est le rang de E

Algorithme 2 : *Basis*

Entrée : $F \in (\mathbb{F}[x]_{\le 0})^{m\times n}$, et un vecteur de décalage s
Sortie : Une $(F, 1, s)$-base d'ordre et son degré de ligne s

1 On suppose que s est croissant.
2 Calculer une forme ligne échelonnée $F = \tau \cdot L \cdot E$ avec :
3 **Retourner** $\begin{pmatrix} xL_r & 0 \\ G & I_{m-r} \end{pmatrix}, \quad \tau^{-1}s + [1_r, 0_{n-r}]$

3.2.2 *Algorithmes pour le cas général*

On va devoir découper le problème pour trouver une base d'ordre pour $\sigma > 1$.

Théorème 3.3. Soit P_1 une base d'ordre de (F, σ_1). Soit $M \in \mathbb{F}[x]^{m\times n}$ tel que $P_1 F = x^{\sigma_1} M$. Soit P_2 une base d'ordre de (M, σ_2).
$P_2 P_1$ est une base d'ordre de $(F, \sigma_1 + \sigma_2)$.

Preuve. $P_2 P_1 F = P_2(x^{\sigma_1} M) = x^{\sigma_1}(P_2 M) = 0 \mod x^{\sigma_1+\sigma_2}$ ■

On présente un algorithme itératif **quadratique**.

$$(F, 1) \to (F, 2) \to (F, 3) \to \cdots (F, \sigma - 1) \to (F, \sigma)$$

Algorithme 3 : *M-Basis*

Entrée : $F \in (\mathbb{F}[x]_{\leq 0})^{m\times n}$
1 $P_0 \leftarrow$ ***Basis***$(F \bmod x)$
2 **Pour** k *de* 1 *à* $\sigma - 1$ **faire**
3 $\quad F' \leftarrow x^{-k} P_{k-1} F$
4 $\quad M_k \leftarrow$ ***Basis***$(F' \bmod x)$
5 $\quad P_k \leftarrow M_k P_{k-1}$
6 **Retourner** $P_{\sigma-1}$

Théorème 3.4. La complexité de l'algorithme *Basis* est $\mathcal{O}(m^\omega \sigma^2)$.

On présente maintenant un algorithme diviser-pour-régner **quasi-linéaire**.

$$(F, 1) \to (F, 2) \to (F, 4) \to \cdots \to \left(F, \frac{\sigma}{2}\right) \to (F, \sigma)$$

Algorithme 4 : *PM-Basis*

Entrée : $F \in (\mathbb{F}[x]_{\leq 0})^{m\times n}$
Sortie :
1 **Si** $\sigma = 1$ **alors**
2 $\quad$ **Retourner** Basis$F \bmod x$
3 **Sinon**
4 $\quad P_{\text{low}} \leftarrow$ PM-Basis$(F, \lfloor \sigma/2 \rfloor)$
5 $\quad$ Soit F' tel que $P_{\text{low}} F = x^k F'$
6 $\quad P_{\text{high}} \leftarrow$ PM-Basis$(F', \lfloor \sigma/2 \rfloor)$
7 $\quad$ **Retourner** $P_{high} \cdot P_{low}$

Théorème 3.5. La complexité de l'algorithme *PM-Basis* est $\mathcal{O}(\mathrm{MM}(m, \sigma) \log(\sigma))$.

CHAPITRE 4

Réduction de réseaux euclidiens

DANS ce chapitre, nous nous concentrerons sur un algorithme de réduction de réseau s'exécutant en temps polynomial : l'algorithme LLL, du nom de ses auteurs A. Lenstra, H. Lenstra et L. Lovász. Nous ne traiterons pas d'autres algorithmes plus avancés comme BKZ, afin de rester dans un cadre plus élémentaire. L'algorithme LLL repose sur une idée simple mais puissante : il produit une approximation entière de la décomposition de Gram-Schmidt et réorganise les vecteurs de la base pour en améliorer la structure. Un rappel sur le procédé d'orthogonalisation de Gram-Schmidt est dans l'annexe dédiée. Celle-ci ne sera pas rappelé ici afin de préserver la concision du texte.

4.1 *La base réduite*

Définition 4.1. Soit $b_1, \ldots, b_n$ une base d'un réseau et U la matrice triangulaire supérieure telle que $B = B^*U$. est dite **propre**[1] si

$$\max_{1 \leq i < j \leq n} |\mu_{i,j}| \leq \frac{1}{2}. \tag{4.1}$$

Définition 4.2. Soit $\mathcal{B}$ une base de $\mathbb{R}^n$ et $\mathcal{B}^*$ sa base de Gram-Schmidt associée.
On dit que la famille $(b_1, \ldots, b_n)$ satisfait la **condition de Lovász** si

$$(\delta - \mu_{i+1,i}^2) \, \|b_i^*\|^2 \leq \|b_{i+1}^*\|^2 \text{ pour } 1 \leq i \leq n, \text{ où } \delta \in \left]\frac{1}{4}, 1\right]$$

Définition 4.3. Une base $\mathcal{B}$ d'un réseau $\mathscr{L}(\mathcal{B})$ d'un réseau est dite **LLL-réduite** si

- $\mathcal{B}$ est propre.
- $\mathcal{B}$ satisfait la condition de Lovàsz.

Remarque. Chaque vecteur de la base réduite a une norme au moins égale à la moitié de celle du précédent, garantissant ainsi une décroissance modérée.

Théorème 4.1. Soit $\mathcal{B}$ une base réduite du réseau $\mathscr{L} \subseteq \mathbb{R}^n$ et soit $v \in \mathscr{L} \setminus \{0\}$. Alors

$$\|b_1\| \leq 2^{(n-1)/2} \cdot \|v\|$$

.

Théorème 4.2. Soit $\mathcal{B}$ une base du réseau $\mathscr{L} \subseteq \mathbb{R}^n$ qui satisfait la condition de Lovász et soit $v \in \mathscr{L} \setminus \{0\}$. Alors $\|b_1\| \leq \frac{1}{(\delta - \mu_{i+1,i}^2)^{\frac{n-1}{2}}} \|v\|$, si le dénominateur ne s'annule pas.

1. Une base propre est aussi connue sous le nom de base size-réduite dans la littérature.

4.2 *Fonctionnement et exemple*

Algorithme 5 : *BasisReduction*

Entrée : Une base $B = (b_1, \ldots, b_n)$
Sortie : Une base réduite $G = (g_1, \ldots, g_n)$ de B

Pour $i = 1$ **to** n **faire**
  $g_i \leftarrow b_i$
$(B^*, U) \leftarrow$ GRAM-SCHMIDT (B)
Tant que $i \leq n$ **faire**
  Pour $j = i-1, i-2, \ldots, 1$ **faire**
    $g_i \leftarrow g_i - \lceil \mu_{i,j} \rfloor \; g_j$
    Mettre à jour B^*, U
  Si $i > 1$ ***et*** $\|g_{i-1}^*\|^2 > 2\|g_i^*\|^2$ **alors**
    Échanger g_{i-1} et g_i
    Mettre à jour B^*, U
    $i \leftarrow i - 1$
  Sinon
    $i \leftarrow i + 1$
Retourner $G = (g_1, \ldots, g_n)$

L'algorithme débute par le calcul de la base orthogonalisée de Gram-Schmidt, qui servira de support pour les opérations de réduction. Le principe consiste à appliquer successivement des étapes de réduction, en réorganisant les vecteurs de la base lorsqu'une certaine condition n'est pas respectée. Une application naïve des réductions seules ne garantit pas la terminaison de l'algorithme : il est nécessaire d'introduire un mécanisme de contrôle, comme la condition de Lovász, afin de déterminer quand effectuer un échange entre deux vecteurs. Cette condition assure la progression de l'algorithme et, in fine, sa terminaison.

Exemple. Soit

$$B = \begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 2 \\ 3 & 5 & 6 \end{pmatrix} \in GL_3(\mathbb{Z}).$$

On a $|\mathscr{L}(B)| = 9, \quad A = \max_{1 \le i \le 3} \|b_i\| = 70$

On va essayer d'estimer un plus court vecteur L'algorithme *BasisReduction* commence par calculer la décomposition de Gram-Schmidt de B, pour plus de détails sur le calcul de cette décomposition, ce calcul est effectué dans l'annexe *Rappels d'algèbres linéaire* .

On obtient la décomposition

$$U = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ \frac{14}{3} & \frac{13}{14} & 1 \end{pmatrix}, \quad B^* = \begin{pmatrix} 1 & 1 & 1 \\ -\frac{4}{3} & -\frac{1}{3} & \frac{5}{3} \\ -\frac{3}{7} & \frac{9}{14} & -\frac{3}{14} \end{pmatrix}.$$

Voici un tableau récapitulant les principales étapes de l'algorithme. Le tableau est volontairement détaillé et fourni, le lecteur pourra revenir sur cet exemple pour comprendre ce que sont d_1, d_2, D ou la signification de $\mathrm{Gram}(G)$.

	G	U	G^*	d_1, d_2 D	$\begin{pmatrix} \|g_1^*\|^2 \\ \|g_2^*\|^2 \\ \|g_3^*\|^2 \end{pmatrix}$	$\mathrm{Gram}(G)$
	$\begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 2 \\ 3 & 5 & 6 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ \frac{14}{3} & \frac{13}{14} & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 & 1 \\ -\frac{4}{3} & -\frac{1}{3} & \frac{5}{3} \\ -\frac{3}{7} & \frac{9}{14} & -\frac{3}{14} \end{pmatrix}$	$3, 14$ 42	$\begin{pmatrix} 3 \\ \frac{14}{3} \\ \frac{9}{14} \end{pmatrix}$	$\begin{pmatrix} 3 & 1 & 14 \\ 1 & 1 & 9 \\ 14 & 9 & 70 \end{pmatrix}$
proprification g_3	$\begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 2 \\ 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ \frac{1}{3} & \frac{-1}{14} & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 & 1 \\ -\frac{4}{3} & -\frac{1}{3} & \frac{5}{3} \\ -\frac{3}{7} & \frac{9}{14} & -\frac{3}{14} \end{pmatrix}$	$3, 14$ 42	$\begin{pmatrix} 3 \\ \frac{14}{3} \\ \frac{9}{14} \end{pmatrix}$	$\begin{pmatrix} 3 & 1 & 1 \\ 1 & 5 & 0 \\ 1 & 0 & 1 \end{pmatrix}$
réduction $g_2 \leftrightarrow g_3$	$\begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ \frac{1}{3} & \frac{-1}{2} & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 & 1 \\ -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} \\ -\frac{3}{2} & 0 & \frac{3}{2} \end{pmatrix}$	$3, 2$ 6	$\begin{pmatrix} 3 \\ \frac{2}{3} \\ \frac{9}{2} \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{1}{3} & \frac{-1}{2} & 1 \end{pmatrix}$
réduction $g_1 \leftrightarrow g_2$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ -1 & 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ \frac{1}{3} & \frac{-1}{2} & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ -\frac{3}{2} & 0 & \frac{3}{2} \end{pmatrix}$	$1, 2$ 2	$\begin{pmatrix} 1 \\ 2 \\ \frac{9}{2} \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{1}{3} & \frac{-1}{2} & 1 \end{pmatrix}$
proprification g_2	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ -1 & 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{1}{3} & \frac{-1}{2} & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ -\frac{3}{2} & 0 & \frac{3}{2} \end{pmatrix}$	$1, 2$ 2	$\begin{pmatrix} 1 \\ 2 \\ \frac{9}{2} \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 5 \end{pmatrix}$

On obtient la base δ - LLL réduite :

$$G_{reduced} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ -1 & 0 & 2 \end{pmatrix} \in GL_3(\mathbb{Z})$$

Remarque. — La valeur d_3 ne nous intéresse pas car il s'agit d'un invariant,
— Gram G se rapproche petit a petit d'une matrice diagonale

État de l'art de la réduction de réseaux euclidiens et l'objectif de mon stage

État de l'art de la réduction de réseaux euclidiens

Voici une rétrospective structurée des avancées majeures dans la réduction de réseaux euclidiens.

1982 LLL (Lenstra, Lenstra, Lovász) (Lenstra 1982) introduisent le premier algorithme de réduction polynomial, basé sur une combinaison de *size reduction* et d'une condition de Lovász. Il garantit :

$$\|b_1\| \leq (4/3)^{(n-1)/2} \cdot \lambda_1(\mathscr{L}), \quad \text{avec complexité binaire } \mathcal{O}(n^5\beta^2)$$

1991 BKZ (Schnorr et Euchner 1994) introduit une approche par blocs. L'algorithme applique un solveur SVP de petite dimension β à des sous-blocs de la base :

$$\|b_1\| \leq \gamma_\beta^{(n-1)/(\beta-1)} \cdot \lambda_1(\mathscr{L})$$

Le coût est exponentiel en β mais reste efficace pour $\beta \leq 40$ en pratique.

2009 L2 (Nguyen et Stehlé 2009) améliore LLL sur le plan de la stabilité numérique, sans gain théorique majeur sur la qualité de la base. Complexité similaire à LLL, mais plus efficace pour des entrées en flottants.

2011 $\tilde{L}_1$ (Novocin, Stehlé et Villard 2011) propose une version rapide de LLL inspirée du GCD rapide de Knuth–Schönhage. Il introduit une stratégie récursive appelée *Lift-Reduction* et atteint une complexité quasi-linéaire :

$$\mathcal{O}\left(d^{5+\varepsilon}\beta + d^{\omega+1+\varepsilon}\beta^{1+\varepsilon}\right)$$

avec une qualité comparable à LLL : $\|b_1\| \leq 2^{\alpha n} \cdot |\mathscr{L}|^{1/n}$.

2011 Terminating BKZ (Hanrot, Pujol et Stehlé 2011) propose une modélisation dynamique affine de BKZ. Ils montrent que même interrompu prématurément, BKZ garantit :

$$\|b_1\| \leq 2^{\frac{\gamma_\beta(n-1)}{2(\beta-1)}+\frac{3}{2}} \cdot |\mathscr{L}|^{1/n}$$

après seulement $\mathcal{O}(n^3/\beta^2 \cdot \log\|B\|)$ appels à un solveur SVP.

2019 KEF (Kirchner, Espitau et Fouque 2021)propose un algorithme heuristique récursif exploitant la *Geometric Series Assumption* (GSA) pour guider la réduction. Il utilise des techniques de FFT et obtient une complexité heuristique :

$$\widetilde{\mathcal{O}}(n^\omega \cdot \log \kappa(B))$$

avec une qualité empirique équivalente à BKZ en grande dimension ($n > 2000$).

2023 Iterated Compression (Ryan et Heninger 2023) présente un algorithme récursif fondé sur des opérations de compression stables, une métrique de *drop*, et des profils dynamiques :

$$\|b_1\| \leq 2^{\alpha n} \cdot |\mathscr{L}|^{1/n}, \quad \|b_n^*\| \geq 2^{-\alpha n} \cdot |\mathscr{L}|^{1/n}$$

avec complexité heuristique :

$$\mathcal{O}(n^\omega(C+n)^{1+\varepsilon}), \quad C = \log(\|B\| \cdot \|B^{-1}\|)$$

Mesures de qualité

La qualité d'une base $B = (b_1, \ldots, b_n)$ se mesure par :

- La norme $\|b_1\|$, en comparaison avec $\lambda_1(L)$,
- Le *facteur d'Hermite* : $\frac{\|b_1\|}{\det(L)^{1/n}}$,
- La décroissance des $\|b_i^*\|$ (orthogonalité), souvent modélisée par la GSA : $\|b_i^*\| \approx a \cdot r^i$.

Applications

La réduction de réseau est centrale en :

- **Cryptanalyse** de RSA (Coppersmith), NTRU, LWE, FHE,
- **Algèbre effective**, systèmes diophantiens,
- **Optimisation discrète**.

Conclusion

L'évolution de la réduction de réseaux est marquée par un passage progressif :

- de méthodes **théoriques lentes mais garanties** (HKZ, LLL),
- vers des approches **rapides, heuristiques et massivement parallélisables** (KEF, Iterated Compression),

en maintenant un objectif constant : approcher au mieux $\lambda_1(L)$ avec un coût algorithmique acceptable, même en très grande dimension.

Objectif de mon stage

CHAPITRE **5**

Adaptation de la réduction de réseaux polynomiaux au cas entier

Dans ce chapitre, nous allons essayer d'adapter l'algorithme PM-Basis, pour les réseaux euclidiens.

CHAPITRE **6**

Réseaux définis par relations plutôt que par générateurs

Dans tout ce chapitre on considère un réseau $\mathscr{L}$ de $\mathbb{R}^n$ de base B. On commence par s'intéresser au dual d'un réseau.

Définition 6.1. Le **dual** de $\mathscr{L}$ est défini par

$$\mathscr{L}^\vee := \{x \in \mathbb{R}^n \mid \forall y \in \mathscr{L}, \langle x, y\rangle \in \mathbb{Z}\}.$$

Proposition 6.1. On a les propriétés suivantes :

- $\mathscr{L}^\vee$ est un réseau de base $(B^t)^{-1}$.
- $\mathrm{rang}(\mathscr{L}) = \mathrm{rang}(\mathscr{L}^\vee)$.
- $|\mathscr{L}^\vee| = |\mathscr{L}|^{-1}$.

Définition 6.2. On dit que $\mathscr{L}$ est **auto-dual** si $\mathscr{L} = \mathscr{L}^\vee$.

Proposition 6.2. On a les propriétés suivantes :

- $(a\mathscr{L})^\vee = \frac{1}{a}\mathscr{L}^\vee$ pour tout $a \in \mathbb{R}^*$.
- $(\mathbb{Z}u)^\vee = \frac{1}{\|u\|^2}\mathbb{Z}u$ pour tout $u \in \mathbb{R}^m$.

Exemple. Le réseau dual de $\mathbb{Z}^n$ est $\mathbb{Z}^n$ et est donc auto-dual.

Exemple. Le réseau dual de $2\mathbb{Z}^n$ est $\frac{1}{2}\mathbb{Z}^n$.

Proposition 6.3. Soit $\mathscr{L}_1$, $\mathscr{L}_2$ des réseaux, alors

$$(\mathscr{L}_1 \oplus \mathscr{L}_2)^\vee = \mathscr{L}_1^\vee \oplus \mathscr{L}_2^\vee$$

Lemme 6.1. Soit $\mathscr{L}$ un réseau de dimension n. On a

- $\lambda_1(\mathscr{L}) \cdot \lambda_1(\mathscr{L}^\vee) \leq n$,
- $\lambda_1(\mathscr{L}) \cdot \lambda_n(\mathscr{L}^\vee) \geq 1$.

Banaszczyk a démontré une relation encore plus forte entre les minima d'un réseau et ceux de son dual, connue sous le nom de théorème de transfert.

Théorème 6.1 ((Banaszczyk 1993))**.** Soit $\mathscr{L}$ un réseau de dimension n. On a

$$1 \leq \lambda_1(\mathscr{L}) \cdot \lambda_n(\mathscr{L}^\vee) \leq n.$$

6.1 *Quelques réseaux usuels*

Définition 6.3. On définit le réseau euclidien $A_n \subset \mathbb{R}^{n+1}$ par :

$$A_n = \left\{ (x_1, \ldots, x_{n+1}) \in \mathbb{Z}^{n+1} \;\middle|\; \sum_{i=1}^{n+1} x_i = 0 \right\}$$

Exemple. On a :

$$\begin{aligned}
&A_0 = \{0\} \subset \mathbb{R},\\
&A_1 = \{(x,-x) \in \mathbb{Z}^2\}, \text{ donc } A_1 \text{ est de rang } 1\\
&A_2 = \{(x,y,z) \in \mathbb{Z}^3 \mid x+y+z=0\},\\
&\quad = \langle a_1, a_2\rangle \quad \text{où } a_1 = (1,-1,0), \quad a_2 = (0,1,-1),\\
&\quad \text{donc } A_2 \text{ est de rang } 2.
\end{aligned}$$

Proposition 6.4. Pour tout $n \in \mathbb{N}$, A_n a pour matrice génératrice :

$$B_n := \begin{pmatrix} 1 & 0 & 0 & \cdots & 0\\ -1 & 1 & 0 & \cdots & 0\\ 0 & -1 & 1 & \cdots & 0\\ \vdots & \vdots & \ddots & \ddots & \vdots\\ 0 & 0 & \cdots & -1 & 1 \end{pmatrix} \in M_n(\mathbb{Z})$$

Proposition 6.5. A_n est un réseau euclidien de rang n, pour tout $n \in \mathbb{N}$.

Proposition 6.6. Soit $\mathbf{B}$ la matrice de la base, soit D la matrice de la base duale associée à B.

— on a $D = B^{-t}$ lorsque le réseau est de rang plein.
— Si L n'est pas de rang plein, on a $D = B(B^tB)^{-1}$.
— Le dual du dual est la base de départ.

Proposition 6.7. On a $\mathscr{L} = \mathscr{L}(b_1, \cdots, b_m)$ si et seulement si $\mathscr{L}^\vee = \mathscr{L}(b_1^\vee, \cdots, b_n^\vee)$

q-réseaux (ou q-ary lattices)

Il existe une classe particulière de réseaux qui joue un rôle important en cryptographie basée sur les réseaux. Étant donnée une matrice $\mathbf{A} \in \mathbb{Z}_q^{m\times n}$ pour certains entiers $n, m, q \in \mathbb{N}$, on peut définir deux réseaux :

$$\Lambda_q(\mathbf{A}) = \{\mathbf{y} \in \mathbb{Z}^m : \mathbf{y} = \mathbf{A}\mathbf{s} \bmod q \text{ pour un certain } \mathbf{s} \in \mathbb{Z}^n\}$$

$$\Lambda_q^\perp(\mathbf{A}^T) = \left\{\mathbf{y} \in \mathbb{Z}^m : \mathbf{A}^T\mathbf{y} = 0 \bmod q\right\}$$

Les deux réseaux sont de dimension m. Le premier est engendré par les lignes de $\mathbf{A}$ et a pour déterminant q^{m-n}, tandis que le second contient tous les vecteurs orthogonaux aux lignes de $\mathbf{A}$ et a pour déterminant q^n.
De plus, ils sont liés par la dualité des réseaux, c'est-à-dire :

$$\Lambda_q^\perp(\mathbf{A}^T) = q \cdot \Lambda_q(\mathbf{A})^\vee \quad \text{et} \quad \Lambda_q(\mathbf{A}) = q \cdot \Lambda_q^\perp(\mathbf{A}^T)^\vee.$$

Définition 6.4 (Réseau q-aire)**.** Un réseau Λ est un réseau q-aire si

$$q\mathbb{Z}^n \subseteq \Lambda \subseteq \mathbb{Z}^n.$$

Conclusion et perspectives

Preuve de la correction, terminaison et complexité de LLL et applications

.1 *Correction*

Théorème .2 (Correction)**.** L'algorithme *BasisReduction* calcule une base réduite de $\mathscr{L}$.

Nous aurons besoin de lemmes intermédiaires afin de prouver la correction. On commence par **étudier la proprification**. L'idée consiste à approcher au mieux, à l'aide d'entiers, les coefficients de la décomposition de Gram-Schmidt, de manière à construire une base réduite.

Notation. On définit l'entier le plus proche de $x \in \mathbb{R}$ par $\lceil x \rfloor = \lfloor x + 1/2 \rfloor$.

Proposition .8. On a $|x - \lceil x \rfloor| \leq \frac{1}{2}$ pour tout $x \in \mathbb{R}$.

On va maintenant regarder des étapes de *BasisReduction* .

Algorithme 6 : *Proprification partielle de* g_i

$g_i \leftarrow g_i - \lceil \mu_{i,j} \rfloor \cdot g_j$
Mettre à jour (B^*, U)

Lemme .2.

1. Soit $G, G^*, M \in \mathbb{Q}^{n\times n}$ et $H, H^*, N \in \mathbb{Q}^{n\times n}$ les matrices des g_k, g_k^*, $\mu_{k,l}$ avant et après *Proprification partielle de* g_i. Soit $E = I_n - \lceil \mu_{i,j} \rfloor E_{i,j} \in \mathbb{Z}^{n\times n}$, où $E_{i,j}$ désigne la matrice élémentaire. Alors
$$H = EG, \quad N = EM, \quad H^* = G^*.$$
2. Avant *Proprification partielle de* g_i, on a :
$$|\mu_{i,l}| \leq \frac{1}{2} \quad \text{pour} \quad j < l < i.$$

Preuve.

1. • *Proprification partielle de* g_i se traduit matriciellement par $H = EG$.
   • La nouvelle famille est
   $$(g_1, \ \ldots, \ g_{i-1}, \ g_i - \lceil \mu_{i,j} \rfloor g_j, \ g_{i+1}, \ \ldots, \ g_n).$$
   On rappelle que i et j sont fixés et que $1 \leq j \leq i-1$. On a par définition du procédé de Gram-Schmidt $h_k^* = g_k^*$ pour $1 \leq k \leq i-1$.
   Pour $k = i$, on a

$$\begin{aligned}
h_i^* &= h_i - \sum_{k=1}^{i-1} \frac{\langle h_i, h_k^* \rangle}{\|h_k^*\|^2} h_k^* \\
&= g_j - \lceil \mu_{i,j} \rfloor \, g_j - \sum_{k=1}^{i-1} \frac{\langle g_i - \lceil \mu_{i,j} \rfloor \, g_j, g_k^* \rangle}{\|g_k^*\|^2} g_k^* \\
&= g_i^* + \lceil \mu_{i,j} \rfloor \sum_{k=1}^{i-1} \frac{\langle g_j, g_k^* \rangle}{\|g_k^*\|^2} g_k^* - \lceil \mu_{i,j} \rfloor \, g_j \\
&= g_i^* + \lceil \mu_{i,j} \rfloor \sum_{k=1}^{j} \frac{\langle g_j, g_k^* \rangle}{\|g_k^*\|^2} g_k^* - \lceil \mu_{i,j} \rfloor \, g_j \\
&= g_i^* + \lceil \mu_{i,j} \rfloor \sum_{k=1}^{j-1} \frac{\langle g_j, g_k^* \rangle}{\|g_k^*\|^2} g_k^* + g_j^* - \lceil \mu_{i,j} \rfloor \, g_j \\
&= g_i^*
\end{aligned}$$

Ainsi $h_i^* = g_i^*$. En poursuivant le procédé de Gram-Schmidt , on en conclut $H^* = G^*$.

- On a $NG^* \overset{(1)}{=} NH^* \overset{(D.1)}{=} H \overset{(1)}{=} EG \overset{(D.1)}{=} EMG^*$. En identifiant on a $N = EM$.

2. Au début de la boucle lorsque $j = i - 1$ on a trivialement l'inégalité.
   La i-ème ligne de M

$$(\mu_{i,1}, \cdots, \mu_{i,j}, \cdots, \mu_{i,i-1})$$

   Devient

$$(\mu_{i,1} - \lceil \mu_{i,j} \rfloor \, \mu_{j,1}, \cdots, \mu_{i,j} - \lceil \mu_{i,j} \rfloor \, \mu_{j,j}, \cdots, \mu_{i,i-1} - \lceil \mu_{i,j} \rfloor \, \mu_{j,i-1})$$

$$(\mu_{i,1} - \lceil \mu_{i,j} \rfloor \, \mu_{j,1}, \cdots, \mu_{i,j} - \lceil \mu_{i,j} \rfloor, \cdots, \mu_{i,i-1})$$

■

Algorithme 7 : *Proprification de g_i*

Pour $j = i-1, i-2, \ldots, 1$ **faire**
  Proprification partielle de g_i

Lemme .3. *Proprification de g_i* ne change pas G^* et à la fin on a

$$|\mu_{i,l}| \leq \frac{1}{2} \text{ pour } 1 \leq l < i.$$

Preuve. D'après le lemme 2.1, après l'exécution de *Proprification de g_i*, G^* n'a pas été modifié et en appliquant le lemme 2.1 pour $j = 1$ on a

$$|\mu_{i,l}| \leq \frac{1}{2} \text{ pour } 1 \leq l < i.$$

■

On étudie maintenant l'étape de réduction, l'idée consiste à réorganiser les vecteurs afin de garantir une progression quantifiable, qui assurera la terminaison de LLL.

Algorithme 8 : *Réduction de* g_{i-1}, g_i

Si $i > 1$ ***et*** $\left\|g_{i-1}^*\right\|^2 > 2\left\|g_i^*\right\|^2$ **alors**
  Échanger g_{i-1} et g_i
  Mettre à jour (B^*, U)
  $i \leftarrow i - 1$
Sinon
  $i \leftarrow i + 1$

Lemme .4. Supposons que g_{i-1} et g_i sont échangés à l'étape *Réduction de* g_{i-1}, g_i. On note h_k les vecteurs après échange et h_k^* leur base orthogonale de Gram-Schmidt. Alors

1. $h_k^* = g_k^* \quad$ pour tout $k \in \{1, \ldots, n\} \setminus \{i-1, i\}$,
2. $\|h_{i-1}^*\|^2 < \frac{3}{4}\|g_{i-1}^*\|^2$,
3. $\|h_i^*\| \leq \|g_{i-1}^*\|$.

Preuve.

1. La famille
$$(g_1,\ \ldots,\ g_{i-2},\ g_{i-1},\ g_i,\ g_{i+1},\ \ldots,\ g_n)$$
devient
$$(h_1 := g_1,\ \ldots,\ h_{i-2} := g_{i-2},\ h_i := g_i,\ h_{i-1} := g_{i-1},\ h_{i+1} := g_{i+1},\ \ldots,\ h_n := g_n)$$
Par construction de Gram-Schmidt on a $h_k^* = g_k^*$ pour $1 \leq k \leq i-2$.
On a
$$\begin{aligned} h_{i+1}^* &= h_{i+1} - \sum_{j=1}^{i} \frac{\langle h_{i+1}, h_j^* \rangle}{\left\|h_j^*\right\|^2} h_j^* \\ &= g_{i+1} - \sum_{j=1}^{i-2} \frac{\langle g_{i+1}, g_j^* \rangle}{\left\|g_j^*\right\|^2} g_j^* + \frac{\langle g_{i+1}, g_i^* \rangle}{\left\|g_i^*\right\|^2} g_i^* + \frac{\langle g_{i+1}, g_{i-1}^* \rangle}{\left\|g_{i-1}^*\right\|^2} g_{i-1}^* \\ &= g_{i+1}^* \end{aligned}$$
On en déduit donc que $h_k^* = g_k^*$ pour $i+1 \leq k \leq n$.
2. Le vecteur h_{i-1}^* est la composante de g_i orthogonale à $\sum_{1 \leq l < i-1} \mathbb{R} g_l$, or $g_i = g_i^* + \sum_{1 \leq l < i-1} \mu_{i,l} g_l^*$.
Alors
$$h_{i-1}^* = g_i^* + \mu_{i,i-1} g_{i-1}^*$$
et donc
$$\begin{aligned} \left\|h_{i-1}^*\right\|^2 &= \left\|g_i^*\right\|^2 + \mu_{i,i-1}^2 \left\|g_{i-1}^*\right\|^2 \\ &\leq \frac{1}{2}\left\|g_{i-1}^*\right\|^2 + \frac{1}{4}\left\|g_{i-1}^*\right\|^2 \\ &= \frac{3}{4}\left\|g_{i-1}^*\right\|^2 \end{aligned}$$
3. Soit $u = \sum_{1 \leq l < i-1} \mu_{i-1,l} g_l^*$ et posons $U = \sum_{1 \leq l < i-1} \mathbb{R} g_l$ pour simplifier l'écriture.
Alors, le vecteur h_i^* est la composante de $g_{i-1} = g_{i-1}^* + u$ orthogonale à $U + \mathbb{R} g_i$.

Or $u \in U \subseteq U + \mathbb{R}g_i$.
Alors, le vecteur h_i^* est la composante de g_{i-1}^* orthogonale à $U + \mathbb{R}g_i$.
Par conséquent,

$$\|h_i^*\| \leq \|g_{i-1}^*\|.$$

■

Algorithme 9 : *LLL*

Tant que $i \leq n$ **faire**
  Proprification de g_i
  Réduction de g_{i-1}, g_i

Lemme .5. Au début de chaque itération de la boucle à l'étape *LLL*, les invariants suivants sont vérifiés :

$$|\mu_{k,l}| \leq \frac{1}{2} \quad \text{pour } 1 \leq l < k < i, \qquad \|g_{k-1}^*\|^2 \leq 2\|g_k^*\|^2 \quad \text{pour } 1 < k < i.$$

Preuve. Au début de l'étape *LLL*, les deux inégalités considérées sont satisfaites.
Supposons qu'elles restent vraies juste avant l'étape de *Proprification de* g_i.
Le lemme 2.1 garantit que cette phase ne les altère pas : les inégalités demeurent donc valides après *Proprification de* g_i.
Passons à l'étape *Réduction de* g_{i-1}, g_i. Un échange effectué durant cette réduction ne modifie aucun coefficient $\mu_{k,l}$ avec $k < i - 1$; par conséquent, la première inégalité reste satisfaite. Par ailleurs, d'après le lemme 2.3, un échange ne modifie pas g_k^* pour $k \notin \{i - 1, i\}$, ce qui préserve la seconde inégalité.
Ainsi, à la fin de l'étape *Réduction de* g_{i-1}, g_i et donc au début de l'itération de *LLL* suivante les deux inégalités sont toujours vérifiées. Par récurrence sur les itérations, elles le sont à chaque instant de l'algorithme, ce qui conclut la démonstration. ■

Preuve de la correction. Le lemme 2.4 implique qu'à la fin de *LLL*, comme $i = n$, on a

$$|\mu_{k,l}| \leq \frac{1}{2} \quad \text{pour } 1 \leq l < k < n, \qquad \|g_{k-1}^*\|^2 \leq 2\|g_k^*\|^2 \quad \text{pour } 1 < k < n.$$

ce qui prouve que la base est réduite. ■

.2 *Terminaison et complexité*

Théorème .3 (Terminaison et complexité)**.** On pose $A = \max_{1 \leq i \leq n} \|g_i\|$. L'algorithme *BasisReduction* termine et utilise $\mathcal{O}(n^4 \log A)$ opérations arithmétiques sur des entiers.

La difficulté est de montrer que la boucle Tant que ne va pas s'exécuter indéfiniment.

Lemme .6.

1. Orthogonalisation de Gram-Schmidt nécessite $\mathcal{O}(n^3)$ opérations dans $\mathbb{Z}$.
2. *Proprification de* g_i nécessite $\mathcal{O}(n^2)$ opérations dans $\mathbb{Z}$
3. *Réduction de* g_{i-1}, g_i nécessite $\mathcal{O}(n)$ opérations dans $\mathbb{Z}$

Preuve.

1. Il faut utiliser le théorème D.2
2. Une exécution de *Proprification partielle de* g_i revient à faire les multiplications $H = EG$ et $N = EM$ se font en $\mathcal{O}(n)$ étapes, ainsi une exécution de *Proprification de* g_i nécessite $\mathcal{O}(n^2)$ opérations dans $\mathbb{Z}$.
3. Si un échange a lieu à *Réduction de* g_{i-1}, g_i, alors seuls g_{i-1}^*, g_i^*, ainsi que les lignes et colonnes $i-1$ et i de la matrice de transition M sont modifiés, et ces éléments peuvent être mis à jour en $O(n)$ opérations.

■

Il reste à borner le nombre d'itérations de la boucle Tant que à l'étape *LLL*.
Pour tout $1 \leq k \leq n$, on pose

$$G_k = \begin{pmatrix} g_1 \\ \vdots \\ g_k \end{pmatrix} \in \mathbb{Z}^{k \times n}, \quad d_0 = 1, \quad d_k = \det(G_k \cdot G_k^T) \in \mathbb{Z}.$$

Lemme .7. Pour tout $1 \leq k \leq n$, on a :

$$d_k = \prod_{1 \leq l \leq k} \|g_l^*\|^2 > 0.$$

Preuve. Soit $1 \leq k \leq n$, on définit $G_k = U_k G_k^*$ la décomposition de Gram-Schmidt de la famille $(g_i)_{1 \leq i \leq k}$
Alors

$$d_k = \det(G_k G_k^t) = \det(U_k G_k^* (G_k^*)^t U_k^t) = \det(G_k^* (G_k^*)^t) = \prod_{1 \leq l \leq k} \|g_l^*\|^2 > 0$$

■

Lemme .8.

1. *Proprification de* g_i ne change pas d_k pour tout $1 \leq k \leq n$.
2. Si g_{i-1} et g_i sont échangés à l'étape *Réduction de* g_{i-1}, g_i, et si d_k^* désigne la nouvelle valeur de d_k, alors :

$$d_k^* = d_k \quad \text{pour tout } k \neq i-1, \quad \text{et} \quad d_{i-1}^* \leq \frac{3}{4} d_{i-1}.$$

Preuve.

1. D'après le lemme 2.2 *Proprification de* g_i ne modifie pas G_k^* et donc ne modifie pas d_k.
2. Pour $k \neq i-1$, une exécution de *Réduction de* g_{i-1}, g_i multiplie G_k par une matrice de permutation, donc $d_k^* = d_k$
   De plus, on a

$$d_{i-1} \overset{(2.6)}{=} \prod_{1 \leq l \leq i-1} \|g_l^*\|^2 \overset{(2.3)}{\leq} \frac{3}{4} \prod_{1 \leq l \leq i-1} \|h_l^*\|^2 \overset{(2.6)}{=} \frac{3}{4} d_{i-1}^*$$

■

On pose

$$D = \prod_{1 \leq k < n} d_k, \quad A = \max_{1 \leq i \leq n} \|f_i\|$$

On désigne D_0 désigne la valeur de D au début de l'algorithme, on a $1 \leq D \in \mathbb{Z}$ et

$$\begin{aligned} D_0 &= \|g_1^*\|^{2(n-1)} \|g_2^*\|^{2(n-2)} \cdots \|g_{n-1}^*\|^2 \\ &\leq \|g_1\|^{2(n-1)} \|g_2\|^{2(n-2)} \cdots \|g_{n-1}\|^2 \\ &\leq A^{n(n-1)} \end{aligned}$$

Puisque f_i^* est une projection de f_i pour tout i.

Lemme .9.

1. *Proprification de* g_i ne modifie pas D.
2. D diminue d'au moins un facteur $3/4$ si un échange a lieu dans *Réduction de* g_{i-1}, g_i

Preuve.

1. D'après le lemme 2.7 *Proprification de* g_i ne modifie pas d_k et donc ne modifie pas D.
2. Si g_{i-1} et g_i sont échangés lors de l'exécution de *Réduction de* g_{i-1}, g_i, en notant D^* la nouvelle valeur de D, alors d'après le lemme 2.7

$$d_k^* = d_k, \quad d_{i-1}^* \leq \frac{3}{4} d_{i-1} \text{ donc } D^* \leq \frac{3}{4} D.$$

■

À tout moment de l'algorithme, soit $e \in \mathbb{N}$ le nombre d'échanges effectués jusqu'à présent, et e^* le nombre de fois où la branche alternative (le *else*) dans *Réduction de* g_{i-1}, g_i a été prise.

Lemme .10. On a

$$e \leq \log_{4/3} D_0 \in \mathcal{O}(n^2 \log A)$$

Preuve. Soit D_e la valeur de D après e échanges.
On doit avoir

$$1 \ \leq \ D_e \ \leq \ \left(\frac{3}{4}\right)^e D_0 \leq \ \left(\frac{3}{4}\right)^e A^{n(n-1)}.$$

En appliquant $log_{3/4}$, aux extrémités de l'inégalité.

$$0 = \log_{3/4}(1) \ \geq \ e + \log_{3/4}(A^{n(n-1)}) = e + n(n-1)\frac{\log A}{\log(3/4)}.$$

On en déduit que $e \ \leq n(n-1)\frac{\log A}{-\log(3/4)}$ et donc $e \in \mathcal{O}(n^2 \log A)$

■

Preuve de la terminaison et la complexité.
Comme i est décrémenté de 1 lors d'un échange et incrémenté de 1 sinon l'entier $i+e-e^*$ est constant tout au long de *LLL*.
Initialement $i+e-e^* = 2$ et à la fin de *LLL* on a $n+1+e-e^* = 2$. On en déduit donc que $e+e^* = 2e+n-1 \in \mathcal{O}(n^2 \log A)$. et donc d'après le lemme 2.5 le coût total de *LLL* est $\mathcal{O}(n^2 \times n^2 \log A)$ opérations dans $\mathbb{Z}$. Ce qui acheve la preuve. ■

Théorème .4. L'algorithme *BasisReduction (LLL)* opère sur des entiers dont la longueur est $\mathcal{O}(n \log A)$.

Il reste à montrer la dernière partie du théorème.

Lemme .11. Soit $g_1, \ldots, g_n \in \mathbb{Z}^n$, et soit G^* et M respectivement la base de Gram-Schmidt et la matrice des coefficients associés. Pour tout $1 \leq l < k \leq n$, on a :

(i) $d_{k-1} g_k^* \in \mathbb{Z}^n$
(ii) $d_l \mu_{k,l} \in \mathbb{Z}$
(iii) $|\mu_{k,l}| \leq \sqrt{d_{l-1}} \|g_k\|$

Preuve.

1. On écrit
$$g_k^* = g_k - \sum_{1 \leq l < k} \lambda_{k,l} g_l, \quad \lambda \in \mathbb{R}.$$
Soit $j < k$. On a
$$0 = \langle g_k^*, g_j \rangle = \left\langle g_k - \sum_{1 \leq l < k} \lambda_{k,l} g_l, g_j \right\rangle.$$
Ce qui implique
$$\langle g_k, g_j \rangle = \sum_{1 \leq l < k} \lambda_{k,l} \langle g_l, g_j \rangle$$
On a donc
$$\begin{pmatrix} \langle g_1, g_1 \rangle & \cdots & \langle g_{k-1}, g_1 \rangle \\ \vdots & & \vdots \\ \langle g_1, g_{k-1} \rangle & \cdots & \langle g_{k-1}, g_{k-1} \rangle \end{pmatrix} \begin{pmatrix} \lambda_{k,1} \\ \vdots \\ \lambda_{k,k-1} \end{pmatrix} = \begin{pmatrix} \langle g_k, g_1 \rangle \\ \vdots \\ \langle g_k, g_{k-1} \rangle \end{pmatrix}$$
D'après la règle de Cramer (à citer) on a
$$d_{k-1}\lambda_{k,1} = \frac{\begin{vmatrix} \langle g_1, g_1 \rangle & \cdots & \langle g_{k-1}, g_1 \rangle \\ \vdots & & \vdots \\ \langle g_1, g_{k-1} \rangle & \cdots & \langle g_{k-1}, g_{k-1} \rangle \end{vmatrix}}{\begin{vmatrix} \langle g_1, g_1 \rangle & \cdots & \langle g_{k-1}, g_1 \rangle \\ \vdots & & \vdots \\ \langle g_1, g_{k-1} \rangle & \cdots & \langle g_{k-1}, g_{k-1} \rangle \end{vmatrix}} det(G_k G_k^t) = \begin{vmatrix} \langle g_1, g_1 \rangle & \cdots & \langle g_{k-1}, g_1 \rangle \\ \vdots & & \vdots \\ \langle g_1, g_{k-1} \rangle & \cdots & \langle g_{k-1}, g_{k-1} \rangle \end{vmatrix} \in \mathbb{Z}$$

2.
$$d_l \mu_{k,l} = d_l \frac{\langle g_k, g_l^* \rangle}{\|g_l^*\|^2} = d_l \frac{\langle g_k, g_l^* \rangle}{d_l / d_{l-1}} = d_{l-1} \langle g_k, g_l^* \rangle = \langle g_k, g_l^* d_{l-1} \rangle \in \mathbb{Z}$$

3.
$$|\mu_{k,l}| = \frac{\langle g_k, g_l^* \rangle}{\|g_l^*\|}^2 \leq \frac{\|g_k\|}{\|g_l^*\|} \leq \sqrt{d_{l-1}/d_l} \|g_k\| \leq \sqrt{d_{l-1}} \|g_k\|$$

■

Nous avons supposé que $\|g_k\| \leq A$ pour tout k. Alors A est également une borne supérieure pour la base orthogonale de Gram-Schmidt initiale : $\|g_k^*\| \leq A$ pour tout k.
On a d'après les lemmes $\max \{\|g_k^*\| : 1 \leq k \leq n\}$ ne croît jamais au cours de l'algorithme. Ainsi, à tout instant et pour tout k, on a :

$$\|g_k^*\| \leq A \quad \text{et} \quad d_k = \prod_{1 \leq l \leq k} \|g_l^*\|^2 \leq A^{2k}.$$

Lemme .12. Soit $1 \leq k \leq n$.

1. À tout moment de l'algorithme, sauf éventuellement à l'étape *Proprification de* g_i lorsque $k = i$, on a :
$$\|g_k\| \leq \sqrt{n}A.$$
2. À chaque exécution de l'étape *Proprification partielle de* g_i, on a :
$$\|g_i\| \leq n(2A)^n.$$

Preuve.

1. Initialement $\|g_k\| \leq A$ pour tout k. L'étape *Réduction de* g_{i-1}, g_i ne modifie pas $\|g_k\|$, il suffit donc d'examiner l'étape *Proprification de* g_i. On a que g_k, pour $k \neq i$, n'est pas affecté par l'étape *Proprification partielle de* g_i.
Soit $m_i = \max\{|\mu_{i,l}| : 1 \leq l \leq i\}$. À partir de
$$g_i = \sum_{1 \leq l \leq i} \mu_{i,l} g_l^*$$
et de l'orthogonalité des vecteurs g_l^*, on obtient :
$$\|g_i\|^2 = \sum_{1 \leq l \leq i} \mu_{i,l}^2 \|g_l^*\|^2 \leq n m_i^2 A^2, \quad \text{donc} \quad \|g_i\| \leq \sqrt{n}\, m_i A. \tag{4}$$
À la fin de *Proprification de* g_i, on a $m_i = 1$ par le lemme 2.1.
2. Le lemme 2.10 et le point 1 impliquent qu'au début de *Proprification de* g_i, on a
$$\begin{aligned} m_i &\leq \max\left\{d_l^{1/2} : 1 \leq l \leq i\right\} \cdot \|g_i\| \\ &\leq A^{n-2} \cdot n^{1/2} A \\ &= n^{1/2} A^{n-1}. \end{aligned}$$
Considérons maintenant le remplacement effectué à l'étape *Proprification partielle de* g_i. Comme $m_i \geq 1$ et que $|\mu_{j,l}| \leq \frac{1}{2}$ pour $1 \leq l < j$, le lemme 2.8 donne :
$$\begin{aligned} |\mu_{i,l} - \lfloor \mu_{i,j} \rceil \mu_{j,l}| &\leq |\mu_{i,l}| + |\lfloor \mu_{i,j} \rceil| \cdot |\mu_{j,l}| \\ &\leq m_i + \left(m_i + \tfrac{1}{2}\right) \cdot \tfrac{1}{2} \\ &= \tfrac{3}{2} m_i + \tfrac{1}{4} \\ &\leq 2 m_i \end{aligned}$$
pour $1 \leq l < j$.
Pour $l = j$, la nouvelle valeur de $\mu_{i,j}$ est par construction au plus $\frac{1}{2}$ en valeur absolue, tout comme les valeurs de $\mu_{i,l}$ pour $l > j$, d'après le lemme 2.1.
On en déduit que pour chaque valeur de j, la valeur de m_i est au plus doublée. Ainsi, pendant *Proprification de* g_i, la valeur de m_i est multipliée au plus par un facteur $2^{i-1} \leq 2^{n-1}$.
On a donc
$$m_i \leq n^{1/2}(2A)^{n-1}$$
Puis
$$\|g_i\| \leq n^{1/2} m_i A \leq n(2A)^n.$$

■

Preuve du theoreme.

— Les dénominateurs d_l des nombres rationnels calculés pendant l'algorithme sont au plus A^{2n}, et leur taille est en $\mathcal{O}(n \log A)$.

— Les numérateurs sont majorés en valeur absolue par :

- $\|g_k\|_\infty \leq \|g_k\| \leq n(2A)^n$ d'après le lemme 2.11
- $\|d_{k-1} g_k^*\|_\infty \leq \|d_{k-1} g_k^*\| \leq A^{2k-2} A \leq A^{2n}$ d'après le lemme 2.10
- $|d_l \mu_{k,l}| \leq d_l d_{l-1}^{1/2} \|g_l\| \leq A^{2l} A^{l-1} n(2A)^n \leq n(2A^4)^n$ d'après les lemmes 2.10 et 2.11

et par conséquent, leur taille est aussi en $\mathcal{O}(n \log A)$.

■

ANNEXE **A**

Rappels sur les groupes

A.1 *Groupes*

ANNEXE **B**

Rappels sur les anneaux

B.1 *Généralités*

Ces résultats sont classiques de la théorie des anneaux commutatifs. Pour plus de détails, on pourra se reporter à un manuel standard d'algèbre.

Définition B.1. Un **anneau** $(A, +, \cdot)$ est un ensemble A muni de deux lois internes :

— $(A, +)$ est un **groupe abélien** (on note 0 son élément neutre et $-a$ l'inverse de a),

— la multiplication $\cdot : A \times A \to A$, notée simplement ab, est **associative** et **distributive** par rapport à $+$, c'est-à-dire :

$$a(b+c) = ab + ac, \quad (b+c)a = ba + ca, \quad \forall a, b, c \in A,$$

et possède un **élément neutre** $1 \in A$ (on dit alors que A est un anneau **unitaire**).

Définition B.2. Un anneau A est **commutatif** si $ab = ba$ pour tout $a, b \in A$.

Dans ce mémoire et dans cette annexe, tous les anneaux seront supposés commutatifs par défaut, conformément aux usages standards, sauf indication explicite du contraire.

Définition B.3. Un anneau A est **intègre** si, pour tout $a, b \in A$, on a

$$ab = 0 \quad \Longrightarrow \quad (a = 0 \text{ ou } b = 0).$$

Remarque. Autrement dit, A est intègre si et seulement si il n'a pas de diviseurs de 0.

Définition B.4. Soit A un anneau commutatif. Un **idéal** $I \subseteq A$ est un sous-groupe additif de $(A, +)$ tel que, pour tout $a \in A$ et tout $x \in I$, on ait $ax \in I$.

Remarque. En commutatif, $ax = xa$, donc une seule condition suffit à le décrire.

Définition B.5. Un idéal I de A est dit **principal** s'il existe un unique $r \in A$ tel que $I = \langle r \rangle = \{ar : a \in A\}$. Un anneau A est **principal** si tous ses idéaux sont principaux.

Définition B.6. Un anneau A est **noethérien** si toutes ses chaînes d'idéaux (i.e. $I_1 \subseteq I_2 \subseteq \ldots$) sont stationnaires, c'est-à-dire qu'il n'existe pas de chaîne infinie strictement croissante d'idéaux.

Remarque. Cette définition est équivalente à dire que tout idéal de A est de type fini, i.e. $I = \langle x_1, \ldots, x_k \rangle$ pour un nombre fini d'éléments.

Définition B.7 (Anneau factoriel)**.** Un anneau intègre A est **factoriel** si tout élément non nul et non inversible de A admet une factorisation en éléments irréductibles unique à l'ordre près (et inversibles près).

Tout **anneau euclidien** est principal, tout **anneau principal** est factoriel et tout **anneau principal** est noethérien.

TABLE B.1 – Quelques exemples d'anneaux et leurs propriétés

Anneau	Commutatif	Intègre	Principal	Factoriel	Noethérien
$\mathbb{Z}$	✓	✓	✓	✓	✓
$\mathbb{Z}/n\mathbb{Z}$ (non premier)	✓	✗	✓	✗	✓
$\mathbb{Z}/p\mathbb{Z}$ (p premier)	✓	✓	✓	✓	✓
$\mathbb{Z}[i]$	✓	✓	✓	✓	✓
$\mathbb{Z}[\sqrt{-5}]$	✓	✓	✗	✗	✓
$\mathbb{Z} \times \mathbb{Z}$	✓	✗	✗	✗	✓
$M_n(\mathbb{K})$, $n \geq 2$	✗	✗	—	—	✓
$C^0([0,1], \mathbb{R})$	✓	✓	✗	✗	✗

B.2 *Anneaux de polynômes*

Les matrices polynomiales que nous étudierons dans ce mémoire sont construites à partir d'anneaux de polynômes, qui en forment la base algébrique.

Théorème B.1. Si A est un anneau factoriel, alors $A[x]$ est aussi factoriel.
Si A est un anneau noethérien, alors $A[x]$ est noethérien.
Si A est intègre, alors $A[x]$ est intègre.

Proposition B.1. Soit $\mathbb{K}$ un corps. Alors l'anneau de polynômes $\mathbb{K}[x]$ est **principal** et l'anneau $\mathbb{K}[x, y]$ **n'est pas** principal.

Remarque. Quand on a deux variables, la structure d'idéaux se complique et ne peut pas être engendrée par un seul polynôme dans la plupart des cas.

TABLE B.2 – Exemples d'anneaux de polynômes et leurs propriétés

Anneau	Commutatif	Intègre	Principal	Factoriel	Noethérien
$\mathbb{K}[x]$	✓	✓	✓	✓	✓
$\mathbb{K}[x, y]$	✓	✓	✗	✓	✓
$\mathbb{Z}[x]$	✓	✓	✗	✓	✓
$\mathbb{F}_p[x]$	✓	✓	✓	✓	✓
$\mathbb{F}_p[x, y]$	✓	✓	✗	✓	✓
$\mathbb{R}[x]/(x^2+1)$	✓	✓	✓	✓	✓
$\mathbb{K}[x]/(x^n)$, $n \geq 2$	✓	✗	✓	✗	✓

ANNEXE C

Rappels sur les modules

C.1 *Généralités*

Définition C.1. Soit A un anneau commutatif unitaire. Un **A-module** $(M, +, \cdot)$ est un ensemble M :

— muni d'une loi interne $+$ faisant de $(M, +)$ un groupe abélien,

— muni d'une loi externe $A \times M \to M,\ (a, m) \mapsto am$, satisfaisant pour tout $a, b \in A$ et $m, m' \in M$:

1. Distributivité : $a(m + m') = am + am'$,
2. Distributivité : $(a + b)m = am + bm$,
3. Associativité : $(ab)m = a(bm)$,
4. Neutre : $1 \cdot m = m$ (où 1 est l'élément neutre de A).

Remarque. Cette définition généralise la notion d'espace vectoriel, en remplaçant le *corps* des scalaires par un *anneau* commutatif. Dans le cas d'un corps, tout élément non nul de A est inversible, tandis qu'ici on n'exige pas cette propriété.

Exemple. Soit $n \in \mathbb{N}^*$. L'ensemble $\mathbb{Z}^n$, muni de l'addition vectorielle et de la multiplication par un scalaire entier, est un $\mathbb{Z}$-module. En effet :

— $(\mathbb{Z}^n, +)$ est un groupe abélien,

— pour tout $a \in \mathbb{Z}$ et $x = (x_1, \ldots, x_n) \in \mathbb{Z}^n$, on a $a \cdot x = (ax_1, \ldots, ax_n) \in \mathbb{Z}^n$,

— toutes les axiomes d'un module sont satisfaits (associativité, distributivité, existence d'un neutre).

C'est un module libre de type fini, de base canonique $(e_1, \ldots, e_n)$.

Contre exemple. Considérons l'ensemble $\mathbb{R}$ et l'anneau $\mathbb{C}$. On cherche à définir une structure de $\mathbb{C}$-module sur $\mathbb{R}$ via la multiplication usuelle :

$$\forall \alpha \in \mathbb{C},\ \forall r \in \mathbb{R}, \quad \alpha \cdot r := \alpha r.$$

Cependant, cette loi externe n'est pas bien définie car elle n'est pas **fermée** :

$$\text{par exemple, } \alpha = i \in \mathbb{C},\ r = 1 \in \mathbb{R} \quad \Rightarrow \quad \alpha \cdot r = i \notin \mathbb{R}.$$

Donc $\mathbb{R}$ n'est pas stable par multiplication par les scalaires complexes. Il ne peut pas être muni d'une structure de $\mathbb{C}$-module.

C.1.1 *Sous-modules, type fini et modules libres*

Définition C.2. Soit M un A-module. Un **sous-module** $N \subseteq M$ est un sous-groupe additif de $(M, +)$ qui est stable par multiplication externe, c'est-à-dire pour tout $a \in A$ et tout $x \in N$, on a $ax \in N$.

TABLE C.1 – Exemples et contre-exemples de modules

Ensemble considéré	Sur quel anneau A	Module ?
$\mathbb{Z}^n$	$\mathbb{Z}$	✓
$\mathbb{Q}^n$	$\mathbb{Q}$	✓
$\mathbb{Z}/n\mathbb{Z}$	$\mathbb{Z}$	✓
$C^0([0,1],\mathbb{R})$	$\mathbb{R}$	✓
$\mathbb{R}[x]$	$\mathbb{R}$	✓
$\mathbb{Q}$	$\mathbb{Z}$	✗
$\mathbb{Z}[\sqrt{2}]$	$\mathbb{Z}[x]$	✗
$\mathbb{R}$	$\mathbb{C}$	✗
$\mathbb{Z}^n$	$\mathbb{Q}$	✗

Définition C.3. Un A-module M est de **type fini** s'il existe un ensemble fini $S \subset M$ tel que tout élément de M s'écrive comme combinaison A-linéaire des éléments de S. On dit alors que S engendre M.

Définition C.4. Un A-module M est **libre** s'il admet une famille $(x_i)_{i\in I}$ telle que tout $x \in M$ s'écrive de manière unique sous la forme

$$x \;=\; \sum_{i\in I} \alpha_i \, x_i,$$

avec $\alpha_i \in A$. Cette famille (x_i) est appelée **base** de M.

Si M est libre et de type fini, il existe donc une base finie de M. La démonstration n'est pas triviale. Dans ce cas, toutes les bases de M ont le même nombre d'éléments, que l'on appelle le **rang** de M.

Exemple. .

— $\mathbb{Z}/n\mathbb{Z}$ est un $\mathbb{Z}$-module de type fini, mais il n'est pas libre pour $n \neq 0$, car aucun élément non nul de $\mathbb{Z}/n\mathbb{Z}$ n'est librement générateur.

— Au contraire, $\mathbb{Z}^n$ est libre de rang n (les vecteurs de la base canonique en constituent une base).

Libre	Type fini	Exemple
Oui	Oui	$\mathbb{Z}^n$
Oui	Non	$\bigoplus_{i\in\mathbb{N}} \mathbb{Z}$
Non	Oui	$\mathbb{Z}/n\mathbb{Z}$ pour $n \geq 2$
Non	Non	$\mathbb{Q}$

TABLE C.2 – Exemples de $\mathbb{Z}$-modules selon leur liberté et leur type fini

C.2 *Modules sur un anneau principal*

Dans la suite, on considère un anneau principal A, c'est-à-dire un anneau (commutatif unitaire) dans lequel *tout idéal* est principal.

Théorème C.1. Soit M un module **libre** sur un anneau principal A. Alors tout sous-module de M est également libre et son rang est inférieur ou égal à celui de M.

Exemple. Dans le $\mathbb{Z}$-module libre $\mathbb{Z}^2$, considérons le sous-ensemble

$$N \;=\; \{(a,b) \in \mathbb{Z}^2 \mid a \equiv b \pmod{10}\}.$$

On constate que N est un **sous-module** de $\mathbb{Z}^2$, et qu'il est engendré par les vecteurs $(1, 1)$ et $(0, 10)$. Ceux-ci sont A-linéairement indépendants, donc N est libre de rang 2, tout comme $\mathbb{Z}^2$ lui-même.

Module M	**Anneau** A	**Libre**	**Type fini**	**Torsion**	**Réf.**
$\mathbb{K}[x]^n$	$\mathbb{K}[x]$	Oui	Oui	Non	(1)
$\mathbb{K}[x]/(x^n)$	$\mathbb{K}[x]$	Non	Oui	Oui	(2)
$\mathbb{K}[x]^\infty$	$\mathbb{K}[x]$	Oui	Non	Non	(3)
$\mathbb{K}(x)$	$\mathbb{K}[x]$	Non	Non	Oui	(4)
$(\mathbb{K}[x])^n/(x \cdot \mathbb{K}[x])^n$	$\mathbb{K}[x]$	Non	Oui	Oui	(5)

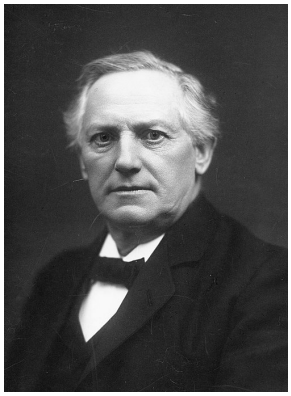
TABLE C.3 – Exemples de modules sur l'anneau $\mathbb{K}[x]$

Commentaires sur les exemples :

1. $\mathbb{K}[x]^n$ est le module libre canonique : c'est un $\mathbb{K}[x]$-module libre de rang n. Il sert de modèle aux réseaux polynomiaux.
2. $\mathbb{K}[x]/(x^n)$ est un module de torsion : tout élément est annulé par une puissance de x. Il n'admet pas de base libre.
3. $\mathbb{K}[x]^\infty$ (somme directe infinie) est un module libre, mais non de type fini. Il possède une base infinie indexée par $\mathbb{N}$.
4. $\mathbb{K}(x)$, le corps des fractions rationnelles, est un module divisible mais non libre. Il contient des éléments sans expression unique comme combinaison de base.
5. $(\mathbb{K}[x])^n/(x \cdot \mathbb{K}[x])^n$ est un module quotient, utilisé dans les algorithmes de bases d'ordre modulo x^σ. C'est un module de torsion.

ANNEXE D

Rappels d'algèbre linéaire



Jørgen Pedersen Gram

JØRGEN Pedersen Gram (1850–1916) était un mathématicien et ingénieur danois. Il est principalement connu pour ses contributions en analyse et en algèbre linéaire, notamment la *matrice de Gram* et le procédé d'orthogonalisation de *Gram-Schmidt* [1], utilisés dans l'étude des produits scalaires, des projections orthogonales et des bases orthonormées. Ses travaux ont également influencé les méthodes numériques et la statistique. Il est mort en 1916, renversé par un cycliste, une fin inattendue.

Dans ce mémoire, on se place dans l'espace euclidien $(\mathbb{R}^n, \langle \cdot, \cdot \rangle)$ muni du produit scalaire canonique usuel qui induit la norme euclidienne $\|\cdot\|_2$. On rappelle que, dans un espace vectoriel, toute application linéaire est représentée, relativement à une base donnée, par une matrice, et que réciproquement, toute matrice définit une application linéaire. Ainsi, dans ce cadre, il est naturel d'identifier une matrice à l'application linéaire qu'elle représente.

Définition D.1. Soient $n, p \in \mathbb{N}^*$, $1 \leq i \leq n$ et $1 \leq j \leq p$. On définit la **matrice élémentaire** $E_{i,j} \in M^{n\times p}(\mathbb{K})$ par :

$$E_{i,j} = (\delta_{k,i}\delta_{l,j})_{1\leq k\leq n,\ 1\leq l\leq p},$$

où δ désigne le symbole de Kronecker. [2]

Exemple. Prenons $n = p = 3$, et considérons $i = 2$, $j = 3$. Alors la matrice élémentaire $E_{2,3} \in M_3(\mathbb{K})$ est donnée par :

$$E_{2,3} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}.$$

Théorème D.1 (Règle de Cramer)**.** Soient $A \in \mathbb{K}^{n\times n}$ une matrice inversible et $b \in \mathbb{K}^n$. Alors les coefficients de l'unique solution $y = (y_1, \ldots, y_n)^t \in \mathbb{K}^n$ du système linéaire $Ay = b$ sont donnés par

$$y_i = \frac{\det(A_i)}{\det(A)},$$

où $A_i \in \mathbb{K}^{n\times n}$ désigne la matrice obtenue à partir de A en remplaçant sa i-ième colonne par le vecteur b.

1. Le procédé d'orthogonalisation porte aussi le nom d'Erhard Schmidt (13 janvier 1876 – 6 décembre 1959), mathématicien allemand né à Dorpat, dans l'ancien Empire russe. Schmidt est reconnu comme l'un des pionniers de l'analyse fonctionnelle abstraite moderne, particulièrement par ses travaux sur la généralisation du procédé d'orthogonalisation aux espaces vectoriels de dimension infinie, aujourd'hui appelés espaces de Hilbert.

2. On appelle **symbole de Kronecker** le nombre noté $\delta_{i,j}$ qui vaut 1 si $i = j$, et 0 sinon.

D.1 *Orthogonalisation de Gram–Schmidt*

Définition D.2 (Base orthogonale)**.** Une base $(b_i)_{1\leq i\leq n}$ de $\mathbb{R}^n$ est dite **orthogonale** si

$$\langle b_i, b_j\rangle = 0 \quad \text{pour tout } i \neq j.$$

Soit $B = (b_1, \ldots, b_n)$ une base de $\mathbb{R}^n$. On construit une base orthogonale associée $B^* = (b_1^*, \ldots, b_n^*)$ de la façon suivante, appelée **procédé d'orthogonalisation de Gram-Schmidt** :

$$b_1^* \coloneqq b_1, \quad b_i^* \coloneqq b_i - \sum_{j=1}^{i-1} \mu_{i,j}\, b_j^*, \quad \mu_{i,j} \coloneqq \frac{\langle b_i, b_j^*\rangle}{\|b_j^*\|^2}.$$

Les coefficients $\mu_{i,j}$ sont appelés **coefficients de Gram-Schmidt**.

Proposition D.1. La famille $(b_i^*)_{1\leq i\leq n}$ obtenue est orthogonale.

Preuve. Pour $n = 1$, la famille est clairement orthogonale.
Supposons que $(b_i^*)_{1\leq i\leq k}$ est orthogonale pour un certain $k < n$.
Alors

$$\begin{aligned}\left\langle b_{k+1}^*, b_i^*\right\rangle &= \left\langle b_{k+1} - \sum_{j=1}^{k} \frac{\langle b_{k+1}, b_j^*\rangle}{\left\|b_j^*\right\|^2} b_j^*, b_i^* \right\rangle \\ &= \langle b_{k+1}, b_i^*\rangle - \left\langle \sum_{j=1}^{k} \frac{\langle b_{k+1}, b_j^*\rangle}{\left\|b_j^*\right\|^2} b_j^*, b_i^* \right\rangle \\ &= \langle b_{k+1}, b_i^*\rangle - \left\langle \frac{\langle b_{k+1}, b_i^*\rangle}{\left\|b_i^*\right\|^2} b_i^*, b_i^* \right\rangle = 0\end{aligned}$$

donc la famille $(b_i^*)_{1\leq i\leq k+1}$ est orthogonale et par le procédé de récurrence la famille $(b_i^*)_{1\leq i\leq n}$ est orthogonale. ■

Définition D.3. Le **complément orthogonal** de U, noté $U^\perp$, est défini par

$$\{x \in \mathbb{R}^n \mid \langle x, d\rangle = 0 \quad \forall\, d \in U\}.$$

Proposition D.2. Pour tout $k \leq n$, g_k^* est la projection de g_k sur $\left(\sum_{1\leq i<k} \mathbb{R}g_i\right)^\perp$.[3]

On en déduit directement que $\mu_{i,i} = 1$ et $\mu_{i,j} = 0$ pour $i < j$. En notant respectivement B et B^* les matrices dont les lignes sont les vecteurs $(b_i)_{1\leq i\leq n}$ et $(b_i^*)_{1\leq i\leq n}$, et en posant

$$U = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ \mu_{2,1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \mu_{n,1} & \cdots & \mu_{n,n-1} & 1 \end{pmatrix},$$

on a la relation suivante :

$$B = U\, B^* \tag{D.1}$$

3. On rappelle que $\mathbb{R}g_i = \{xg_i | x \in \mathbb{R}\}$.

Remarque.

— U est triangulaire inférieure, en particulier $\det(U) = 1$.
— On ne fait pas de normalisation, afin d'éviter l'introduction de racines irrationnelles et de conserver l'information volumétrique du réseau associé.

Exemple. Soit

$$B = \begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 2 \\ 3 & 5 & 6 \end{pmatrix} \in GL_3(\mathbb{Z}).$$

- Calcul de b_1^* :

$$b_1^* = b_1 = (1,1,1)\,, \quad \|b_1^*\|^2 = 3.$$

- Calcul de b_2^* :

$$\begin{aligned} b_2^* &= b_2 - \frac{\langle b_2, b_1^* \rangle}{\|b_1^*\|^2} b_1^* = (-1,0,2) - \frac{1}{3}(1,1,1) \\ &= \left(-\frac{4}{3}, -\frac{1}{3}, \frac{5}{3}\right) \\ \mu_{2,1} &= \frac{1}{3}, \quad \|b_2^*\|^2 = \frac{14}{3}. \end{aligned}$$

- Calcul de b_3^* :

$$\begin{aligned} b_3^* &= b_3 - \frac{\langle b_3, b_1^* \rangle}{\|b_1^*\|^2} b_1^* - \frac{\langle b_3, b_2^* \rangle}{\|b_2^*\|^2} b_2^* \\ &= (3,5,6) - \frac{14}{3}(1,1,1) - \frac{13}{14}\left(-\frac{4}{3}, -\frac{1}{3}, \frac{5}{3}\right) \\ &= \left(-\frac{3}{7}, \frac{9}{14}, -\frac{3}{14}\right) \\ \mu_{3,1} &= \frac{14}{3}, \quad \mu_{3,2} = \frac{13}{14}. \end{aligned}$$

On a donc finalement

$$U = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ \frac{14}{3} & \frac{13}{14} & 1 \end{pmatrix}, \quad B^* = \begin{pmatrix} 1 & 1 & 1 \\ -\frac{4}{3} & -\frac{1}{3} & \frac{5}{3} \\ -\frac{3}{7} & \frac{9}{14} & -\frac{3}{14} \end{pmatrix}.$$

Proposition D.3. On a $\det(B) = \det(B^*) = \prod_{i=1}^{n} \|b_i^*\|$.

Preuve. $\det(B) \overset{(D.1)}{=} \det(UB^*) = \det(U)\det(B^*) = \det(B^*) \overset{(1.3)}{=} \prod_{i=1}^{n} \|b_i^*\|$ ■

Une implémentation de l'algorithme d'orthogonalisation de Gram-Schmidt en SageMath est disponible sur le dépôt GitHub du projet.

Algorithme 10 : *Orthogonalisation de Gram–Schmidt*

Entrée : Une famille libre $B = (b_1, \dots, b_n) \in M_n(\mathbb{Q})$.
Sortie : La famille libre orthogonale $B^* = (b_1^*, \dots, b_n^*)$ et la matrice U des coefficients de Gram–Schmidt.

Pour $k = 1$ **to** n **faire**
  $b_k^* \leftarrow b_k$
  Pour $j = 1$ **to** $k - 1$ **faire**
    $U_{k,j} \leftarrow \dfrac{\langle b_k^*, b_j^* \rangle}{\|b_j^*\|^2}$
    $b_k^* \leftarrow b_k^* - U_{k,j}\, b_j^*$

Théorème D.2. L'algorithme *Orthogonalisation de Gram-Schmidt* effectue au plus $\mathcal{O}(n^3)$ opérations arithmétiques dans $\mathbb{Q}$, où n est la taille de la famille.

D.2 *Matrice de Gram*

Définition D.4. On note $M \in M_n(\mathbb{R})$ la matrice dont les colonnes sont les vecteurs $e_1, \dots, e_n$.
La **matrice de Gram** associée est la matrice symétrique définie par :

$$\mathrm{Gram}(M) \;=\; \big(\langle e_i, e_j \rangle\big)_{1 \le i,j \le n} \;\in\; M_n(\mathbb{R}),$$

Autrement dit,

$$\mathrm{Gram}(M) \;= M^t M$$

Remarque.

— Les éléments diagonaux de G sont les carrés des normes $\|e_i\|^2$.
— G est une matrice symétrique réelle.

Proposition D.4. Soit $M \in M_n(\mathbb{R})$. $\mathrm{Gram}(M)$ est orthogonale si et seulement si ses colonnes forment une famille orthogonale.

Exemple. Soit

$$B^* = \begin{pmatrix} 1 & 1 & 1 \\ -\frac{4}{3} & -\frac{1}{3} & \frac{5}{3} \\ -\frac{3}{7} & \frac{9}{14} & -\frac{3}{14} \end{pmatrix}$$

Alors

$$\mathrm{Gram}(B^*) = (B^*)^t \cdot B^* = \begin{pmatrix} 1 & -\frac{4}{3} & -\frac{3}{7} \\ 1 & -\frac{1}{3} & \frac{9}{14} \\ 1 & \frac{5}{3} & -\frac{3}{14} \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 & 1 \\ -\frac{4}{3} & -\frac{1}{3} & \frac{5}{3} \\ -\frac{3}{7} & \frac{9}{14} & -\frac{3}{14} \end{pmatrix}$$
$$= \begin{pmatrix} 3 & 0 & 0 \\ 0 & \frac{14}{3} & 0 \\ 0 & 0 & \frac{9}{14} \end{pmatrix}$$

On en conclut que la famille est orthogonale, il s'agit en réalité de celle obtenue précédemment par le procédé d'orthogonalisation de Gram-Schmidt.

Rappelons enfin que si u et v sont des vecteurs unitaires, alors $\langle u, v \rangle = \cos(\theta)$, où θ est l'angle entre u et v. Ainsi :

— $\langle u, v \rangle = 1$ signifie que u et v sont alignés et de même sens,
— $\langle u, v \rangle = 0$ signifie que u et v sont orthogonaux,
— $\langle u, v \rangle = -1$ signifie que u et v sont alignés, mais de sens opposé.

Ce point de vue fait de la matrice de Gram un outil privilégié pour évaluer la similarité directionnelle dans un ensemble de vecteurs.

D.3 *Complexité*

Bibliographie

Ajtai, M. (1996). « Generating hard instances of lattice problems (extended abstract) ». In : *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96*. STOC '96. ACM Press, p. 99-108. doi : 10.1145/237814.237838. url : http://dx.doi.org/10.1145/237814.237838.

Banaszczyk, W. (déc. 1993). « New bounds in some transference theorems in the geometry of numbers ». In : *Mathematische Annalen* 296.1, p. 625-635. issn : 1432-1807. doi : 10.1007/bf01445125. url : http://dx.doi.org/10.1007/BF01445125.

Boudgoust, Katharina (fév. 2023). *Hardness Assumptions in Lattice-Based Cryptography*. Crash-Course lecture notes, Aarhus University. Version du 2 février 2023.

Cohn, Henry et al. (2016). « The sphere packing problem in dimension 24 ». In : doi : 10.48550/ARXIV.1603.06518. url : https://arxiv.org/abs/1603.06518.

Goldreich, Oded et al. (1999). « Approximating Shortest Lattice Vectors is not Harder than Approximating Closest Lattice Vectors ». In : *Information Processing Letters* 71.2, p. 55-61. doi : 10.1016/S0020-0190(99)00083-6. url : https://doi.org/10.1016/S0020-0190(99)00083-6.

Hales, Thomas et al. (2015). *A formal proof of the Kepler conjecture*. doi : 10.48550/ARXIV.1501.02155. url : https://arxiv.org/abs/1501.02155.

Hanrot, Guillaume, Xavier Pujol et Damien Stehlé (2011). *Terminating BKZ*. Cryptology ePrint Archive, Paper 2011/198. url : https://eprint.iacr.org/2011/198.

Kirchner, Paul, Thomas Espitau et Pierre-Alain Fouque (2021). « Towards Faster Polynomial-Time Lattice Reduction ». In : *Advances in Cryptology – CRYPTO 2021*. Springer International Publishing, p. 760-790. isbn : 9783030842451. doi : 10.1007/978-3-030-84245-1_26. url : http://dx.doi.org/10.1007/978-3-030-84245-1_26.

Lenstra Lenstra, Lovász (déc. 1982). « Factoring polynomials with rational coefficients ». In : *Mathematische Annalen* 261.4, p. 515-534. issn : 1432-1807. doi : 10.1007/bf01457454. url : http://dx.doi.org/10.1007/BF01457454.

Mulders, T. et A. Storjohann (avr. 2003). « On lattice reduction for polynomial matrices ». In : *Journal of Symbolic Computation* 35.4, p. 377-401. issn : 0747-7171. doi : 10.1016/s0747-7171(02)00139-6. url : http://dx.doi.org/10.1016/S0747-7171(02)00139-6.

Nguyen, Phong Q. et Damien Stehlé (jan. 2009). « An LLL Algorithm with Quadratic Complexity ». In : *SIAM Journal on Computing* 39.3, p. 874-903. issn : 1095-7111. doi : 10.1137/070705702. url : http://dx.doi.org/10.1137/070705702.

Novocin, Andrew, Damien Stehlé et Gilles Villard (juin 2011). « An LLL-reduction algorithm with quasi-linear time complexity : extended abstract ». In : *Proceedings of the forty-third annual*

ACM symposium on Theory of computing. STOC'11. ACM, p. 403-412. DOI : 10.1145/1993636.1993691. URL : http://dx.doi.org/10.1145/1993636.1993691.

RYAN, Keegan et Nadia HENINGER (2023). « Fast Practical Lattice Reduction Through Iterated Compression ». In : *Advances in Cryptology – CRYPTO 2023*. Springer Nature Switzerland, p. 3-36. ISBN : 9783031385483. DOI : 10.1007/978-3-031-38548-3_1. URL : http://dx.doi.org/10.1007/978-3-031-38548-3_1.

SCHNORR, C. P. et M. EUCHNER (août 1994). « Lattice basis reduction : Improved practical algorithms and solving subset sum problems ». In : *Mathematical Programming* 66.1–3, p. 181-199. ISSN : 1436-4646. DOI : 10.1007/bf01581144. URL : http://dx.doi.org/10.1007/BF01581144.

STEPHENS-DAVIDOWITZ, Noah (2015). *Dimension-Preserving Reductions Between Lattice Problems*. http://noahsd.com/latticeproblems.pdf. Accessed : 2025-05-05.

VIAZOVSKA, Maryna (2016). « The sphere packing problem in dimension 8 ». In : DOI : 10.48550/ARXIV.1603.04246. URL : https://arxiv.org/abs/1603.04246.