Lattices with many cycles are dense

Mårten Trolin

2004

Introduction

Connaissances précédentes :

- SVP et CVP difficiles pour les réseaux en général
- SVP et CVP difficiles pour les réseaux à 1 et à n/c cycles
- Un réseau à au moins n cycles peut se ramener à un réseau à $k \in [\![1, n-1]\!]$ cycles

Article :

- Densité des réseaux à $n-1$ cycles
- SVP et CVP difficiles également pour ces réseaux

Intérêt de l'étude

- Des problèmes pratiques se ramènent parfois à des problèmes sur des réseaux vérifiant certaines propriétés, comme la cyclicité
- Certains systèmes de cryptage peuvent être affaiblis lorsque les réseaux vérifient une certaine propriété

Plan

Première partie I

Présentation

Forme Normale de Hermite

$$H = \begin{pmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \dots & h_{1,n-1} & h_{1,n} \\ 0 & h_{2,2} & h_{2,3} & \dots & h_{2,n-1} & h_{2,n} \\ 0 & 0 & h_{3,3} & \dots & h_{3,n-1} & h_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & h_{n-1,n-1} & h_{n-1,n} \\ 0 & 0 & 0 & \dots & 0 & h_{n,n} \end{pmatrix}$$

$$\text{où } \forall j < i,\ h_{i,i} > h_{j,i} \geqslant 0$$

$\hookrightarrow$ calculable efficacement

Forme Normale de Hermite plus "simple"

$$B = \begin{pmatrix} 1 & 0 & \dots & 0 & a_1 \\ 0 & 1 & \dots & 0 & a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & a_{n-1} \\ 0 & 0 & \dots & 0 & d \end{pmatrix}$$

où $d = \det(\Lambda)$

$\hookrightarrow$ passage de HNF à HNF plus simple en temps polynomial en la taille des données

Forme Normale de Smith

La SNF de B matrice carrée à coefficients entiers est S telle que :

- $S = UBV$
- U et V entières avec $|\det(U)| = |\det(V)| = 1$
- S diagonale et $\forall i,\ \frac{s_{i+1}}{s_i} \in \mathbb{N}$

$\hookrightarrow$ existe toujours

Calcul de la Forme Normale de Smith

$$s_i = \frac{d_i}{d_{i-1}}$$

où d_i est le pgcd des déterminants de toutes les matrices de taille $i \times i$ de lignes et de colonnes de B et $d_0 = 1$

Cyclicité

Λ a la structure cyclique $k_1 \times \cdots \times k_m$ si :

$$\begin{cases} \mathbb{Z}^n/\Lambda \sim \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_m} \\ k_i \text{ divise } k_{i+1} \end{cases}$$

Propriété

Λ un réseau de dimension n, B sa matrice des vecteurs de base. S la HNF de B.
Alors Λ a pour structure cyclique $s_1 \times \cdots \times s_n$.

Deuxième partie II

Densité

Problème

$\Lambda \subset \mathbb{Z}^n$ un réseau, $\epsilon > 0$. On veut construire $\sigma_{\Lambda,\epsilon}$ telle que :

- $\sigma_{\Lambda,\epsilon}(\Lambda)$ est un réseau à $n-1$ cycles
- $\forall u \in \mathbb{Z}^n,\ \|u - \sigma_{\Lambda,\epsilon}(u)\| \leqslant \epsilon \|u\|$

Dans la suite, Λ est donnée par B, une matrice de vecteurs de bases de Λ.

Algorithme de Trolin

1. Mettre B sous HNF simple.
2. Appliquer LLL avec $\delta = \frac{3}{4}$ sur les $n-1$ premiers vecteurs pour obtenir $\rho(B)$.
3. Factoriser $\rho(B)$ en le produit de
$$D = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & 1 & 0 \\ 0 & \dots & 0 & \det(B) \end{pmatrix}$$
et une matrice unimodulaire E.
4. Transformer D en une matrice D' engendrant un réseau à $n-1$ cycles de même longueur $(\det(B))^{n-1}$.
5. Poser $B' = D'E$.

Étape 4

$$D' = \begin{pmatrix} d^{n-2} & d^{n-3} & d^{n-4} & \dots & d^2 & d & 1 & 0 \\ 0 & d^{n-2} & d^{n-3} & \dots & d^3 & d^2 & d & 0 \\ 0 & 0 & d^{n-2} & \dots & d^4 & d^3 & d^2 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & d^{n-2} & d^{n-3} & d^{n-4} & 0 \\ 0 & 0 & 0 & \dots & 0 & d^{n-2} & d^{n-3} & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & d^{n-2} & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & d^{n-1} \end{pmatrix}$$

$\hookrightarrow$ $\gamma'_n(d)$ a $n-1$ cycles de longueur d^{n-1}

Preuve (1/2)

On considère $k_1 = \frac{d_n}{d_{n-1}}$ la longueur du plus long cycle avec :

- $d_n = |det(D')| = d^{(n-1)(n-1)}$
- d_{n-1} est le pgcd de toutes les $D^{i,j}$, matrices de taille $(n-1)^2$ obtenue à partir de D' en ôtant la $i^{\text{ème}}$ ligne et la $j^{\text{ème}}$ colonne.

Étude de cas $\rightsquigarrow d_{n-1} = d^{(n-2)(n-1)}$, donc $k_1 = d^{n-1}$.

Preuve (2/2)

$d_1 = 1$ (pgcd des éléments de D'), donc :

- $n-1$ cycles de produit $d^{(n-1)(n-1)}$
- le plus grand a pour longueur $d^{(n-1)}$

$\Rightarrow$ chacun a pour longueur $d^{(n-1)}$ par divisibilité.

Complexité et correction de l'algorithme

Complexité : polynomiale.

Correction :

- $\sigma_{\Lambda,\epsilon}(\Lambda)$ est un réseau à $n-1$ cycles
- $\forall u \in \mathbb{Z}^n,\ \|u - \sigma_{\Lambda,\epsilon}(u)\| \leqslant \epsilon\|u\|$

$\hookrightarrow$ on montre que chaque étape ne change pas le vecteur u d'un trop grand facteur

Troisième partie III

CVP et SVP

Présentation

Réduction de CVP sur un réseau quelconque à CVP sur un réseau cyclique.
Problème :

- Entrées : $\Lambda \subset \mathbb{Z}^n$, $y \in \mathbb{Z}^n$.
- Sorties : $x \in \Lambda$ tel que $\|x - y\|_p$ minimal.

$\rightsquigarrow$ x et y légèrement modifiés pour se ramener à un réseau cyclique

Preuve (1/5)

On suppose $p \in \mathbb{N}$, et on note $\|.\| = \|.\|_p$.

Théorème :
Si $\|x - y\| < \|z - y\|$ et toutes les coordonnées sont dans $[\![0, \det(\Lambda) - 1]\!]$ alors
$$\left\| \frac{1}{k}\sigma_{\Lambda,\epsilon}(x) - \frac{1}{k}\sigma_{\Lambda,\epsilon}(y) \right\| < \left\| \frac{1}{k}\sigma_{\Lambda,\epsilon}(z) - \frac{1}{k}\sigma_{\Lambda,\epsilon}(y) \right\|$$
où $0 < \epsilon < \dfrac{1}{2pn^{1+\frac{1}{p}}\det(\Lambda)^{p+1}}$ et k est un polynôme de $1/\epsilon$.

Preuve (2/5)

Démonstration

$$\begin{aligned}\left|\|x-y\| - \left\|\frac{1}{k}\sigma_{\Lambda,\epsilon}(x) - \frac{1}{k}\sigma_{\Lambda,\epsilon}(y)\right\|\right| &= \left|\|x-y\| - \left\|\frac{1}{k}\sigma_{\Lambda,\epsilon}(x-y)\right\|\right| \\ &\leqslant \left\|(x-y) - \frac{1}{k}\sigma_{\Lambda,\epsilon}(x-y)\right\| \\ &\leqslant \epsilon\|x-y\|\end{aligned}$$

$\hookrightarrow$ idem pour z et y

Preuve (3/5)

Donc :

$$\left\| \frac{1}{k}\sigma_{\Lambda,\epsilon}(z) - \frac{1}{k}\sigma_{\Lambda,\epsilon}(y) \right\| - \left\| \frac{1}{k}\sigma_{\Lambda,\epsilon}(x) - \frac{1}{k}\sigma_{\Lambda,\epsilon}(y) \right\|$$

$$\geqslant (\|z-y\| - \|x-y\|) - \epsilon(\|z-y\| + \|x-y\|) \geqslant 0$$

pour ϵ choisi

Preuve (4/5)

Lemme :

$$\forall u \in \mathbb{Z}^n,\ x \text{ est solution de } \mathrm{CVP}(\Lambda, y)$$

$$\text{ssi } x - \det(\Lambda)u \text{ est solution de } \mathrm{CVP}(\Lambda, y - \det(\Lambda)u)$$

Preuve :
Il suffit de remarquer que
$\|x - y\| = \|(x - \det(\Lambda)u) - (y - \det(\Lambda)u)\|$.

Preuve (5/5)

On déduit de tout cela que :
Si $\forall i \in [\![1, n]\!],\ y_i \in [\![0, \det(\Lambda) - 1]\!]$, alors :

$$x \text{ est solution de CVP}(\Lambda, y) \text{ ssi } \frac{1}{k}\sigma_{\Lambda,\epsilon}(x) \text{ est solution de}$$

$$\text{CVP}(\frac{1}{k}\sigma_{\Lambda,\epsilon}(\Lambda), \frac{1}{k}\sigma_{\Lambda,\epsilon}(y))$$

pour k et $1/\epsilon$ polynomiaux en n et $\det(\Lambda)$.

Conclusion

CVP est aussi difficile pour des réseaux à $n-1$ cycles que pour des réseaux quelconques.

SVP

x est solution de SVP(Λ) ssi $\frac{1}{k}\sigma_{\Lambda,\epsilon}(x)$ est solution de SVP($\frac{1}{k}$)$\sigma_{\Lambda,\epsilon}(\Lambda)$.

Donc : SVP est aussi difficile pour des réseaux à $n-1$ cycles que pour des réseaux quelconques.

Conclusion

- On a un algorithme constructif pour se ramener d'un réseau quelconque à un réseau à $n-1$ cycles aussi proche que l'on veut.
- On connaît la difficulté de SVP et CVP pour les réseaux à $n-1$ cycles.
- On ne sait pas encore conclure pour les réseaux à k cycles, $k \in [\![2, n-2]\!]$.

Des questions ?

Cinquième partie V

Preuves

Correction (1/4)

Lemme :
Après la première étape de l'algorithme :

$$\forall u \in \mathbb{Z}^n,\ \left\| u - \frac{1}{k}\tau_{\Lambda,k}(u) \right\| \leqslant \frac{1}{k} 2^n \|u\|$$

Preuve : fonctionne grâce aux propriétés de la HNF

Correction (2/4)

Lemme :
Après la deuxième étape de l'algorithme :

$$|t_i| \leqslant 2^{\frac{3}{2}n-u}\|u\| \text{ pour } u = \sum_{i=1}^{n} t_i b_i$$

Preuve : fonctionne car B est LLL-réduite

Correction (3/4)

Lemme :
Après la deuxième étape de l'algorithme :

$$\|b_i\| \leqslant n2^{\frac{n^2}{8}}(d^2 n)^{\frac{1}{4}}$$

Preuve : LLL donne de petits vecteurs

Correction (4/4)

Lemme :
Après la quatrième étape de l'algorithme :

$$\left\| u - \frac{1}{\det(\Lambda)^{n-2}} \gamma(u) \right\| \leqslant \frac{n^{\frac{9}{4}} 2^{\frac{3}{2}n + \frac{n^2}{8}}}{\sqrt{\det(\Lambda)}} \|u\|$$