Lattices with many cycles are dense

Mårten Trolin

2004

Dans cet article, Mårten Trolin présente un résultat négatif concernant la complexité de SVP (Short Vector Problem) et de CVP (Closest Vector Problem) : ces deux problèmes sur les réseaux ayant $n-1$ cycles non triviaux sont aussi difficiles que sur des réseaux en général. Pour cela, il montre en premier lieu la densité des réseaux à $n-1$ cycles dans les réseaux, puis réduit SVP et CVP sur un réseau quelconque au même problème sur un réseau à $n-1$ cycles en n'effectuant que des transformations polynomiales en temps, ce qui permet d'affirmer que ces deux problèmes sur des réseaux à $n-1$ cycles sont $\mathcal{NP}$-difficiles.

Nous allons présenter ici l'idée de l'algorithme de Trolin pour montrer la densité des réseaux à $n-1$ cycles, en expliquant comment le mettre en pratique efficacement. Nous verrons ensuite comment réduire CVP sur un réseau quelconque à CVP sur un réseau à $n-1$ cycles, le procédé étant identique pour SVP. Enfin, nous reviendrons sur des cas particuliers d'utilisation des réseaux cycliques.

1 Densité des réseaux à $n-1$ cycles

1.1 Problème

On montre que l'on peut approcher tout réseau quelconque par un réseau à $n-1$ cycles non triviaux (et de même longueur, dans l'algorithme de Trolin) arbitrairement proche.

Pour cela, on se donne un réseau quelconque $\Lambda \subset \mathbb{Z}^n$ donné par une matrice B des vecteurs de base, et $\epsilon > 0$. On cherche à approcher Λ par un réseau à $n-1$ cycles d'un facteur ϵ en norme : on construit une fonction $\sigma_{\Lambda,\epsilon}$ tel que $\sigma_{\Lambda,\epsilon}(\Lambda)$ est un réseau à $n-1$ cycles et $\sigma_{\Lambda,\epsilon}$ étendu aux vecteurs vérifie :

$$\forall u \in \mathbb{Z}^n,\ \|u - \sigma_{\Lambda,\epsilon}(u)\| \leqslant \epsilon \|u\|$$

1.2 Algorithme de Trolin

La fonction $\sigma_{\Lambda,\epsilon}$ est définie par l'application de l'algorithme suivant, prenant B en entrée et renvoyant B' base d'un réseau à $n-1$ cycles arbitrairement proche de B :

1. Mettre B sous forme normale de Hermite (HNF).
2. Transformer la matrice obtenue pour avoir une HNF plus simple (voir annexe A).
3. Appliquer LLL avec $\delta = \frac{3}{4}$ sur les $n-1$ premiers vecteurs pour obtenir $\rho(B)$.

4. 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.
5. Transformer D en une matrice D' engendrant un réseau à $n-1$ cycles de même longueur $(\det(B))^{n-1}$.
6. Poser $B' = D'E$.

Remarques :

1. L'étape 1 de l'algorithme peut être réalisée en temps et en espace bornés par des polynômes en la longueur des données d'entrée d'après [1].
2. En utilisant le **Theorem 1** de l'article, on en déduit qu'après l'étape 6 de l'algorithme, le réseau engendré par B' a bien $n-1$ cycles, car E est unimodulaire.

1.3 Application à un vecteur

On étend $\sigma_{\Lambda,\epsilon}$ aux vecteurs ainsi : aux étapes 2 et 5, on définit respectivement les transformations τ_2 et τ_5 sur B. On note $(b_i)_{i\in[\![1,n]\!]}$ les lignes de B.

Si $u = \sum_{i=1}^{n} t_i b_i$, pour $\alpha \in \{2,5\}$, on définit $\tau_\alpha(u) = \sum_{i=1}^{n} t_i b'_i$ où b'_i est la $i^{\text{ème}}$ ligne de $\tau_\alpha(B)$.

On note $\sigma_{\Lambda,\epsilon}(u) = \tau_5 \circ \tau_2(u)$ pour tout vecteur $u \in \mathbb{Z}^n$. On a alors :

$$\forall u \in \mathbb{Z}^n,\ \|u - \sigma_{\Lambda,\epsilon}(u)\| \leqslant \epsilon\|u\|$$

2 Application à SVP et CVP

D'après le **Lemma 10**, on peut effectuer une réduction d'une instance de CVP sur un réseau quelconque à une instance de CVP sur un réseau à $n-1$ cycles, réduction qui est polynomiale en la taille des données. Comme CVP est $\mathcal{NP}$-difficile [2], on en déduit que CVP sur les matrices à $n-1$ cycles est également $\mathcal{NP}$-difficile.

On a le même résultat pour SVP, qui est $\mathcal{NP}$-difficile pour une matrice à $n-1$ cycles, comme pour une matrice quelconque [3].

3 Remarques et perspectives

D'après l'article, de nombreuses applications se ramènent à des problèmes sur des réseaux possédant certaines propriétés, comme des réseaux cycliques (malheureusement, aucune n'est citée). C'est une des raisons pour lesquelles l'étude de réseaux particuliers peut être intéressante. Cependant, comme présenté dans l'article, même ces propriétés sur les réseaux ne suffisent pas à rendre les problèmes classiques plus simples.

D'après [4], et en particulier :

> *every matrix reduces to a unique matrix in reduced row echelon form by elementary row operations*

et le **Theorem 1**, tout réseau de dimension n sur $\mathbb{Z}^n$ est cyclique, ce qui apporte un intérêt nouvel à l'étude de ces réseaux.

Avec les précédents résultats, on sait maintenant que pour les réseaux ayant 1, $n-1$ ou $n/c \in \mathbb{N}$ cycles, SVP et CVP sont aussi difficiles que pour les réseaux en général. De plus, pour les réseaux ayant au moins n cycles, on peut se ramener à k cycles, pour $k \in [\![1, n-1]\!]$. Il manque donc encore un résultat similaire sur les réseaux à $k \in [\![2, n-2]\!],\ k \neq n/c$ cycles pour pouvoir conclure sur tous les réseaux cycliques. Cependant, d'après Trolin, ce résultat n'est pas directement généralisable à partir des preuves déjà existantes pour les autres réseaux cycliques, et reste donc un problème ouvert.

Conclusion Les réseaux à $n-1$ cycles sont denses dans l'ensemble des réseaux. Trolin en tire un résultat qui peut être négatif : la $\mathcal{NP}$-difficulté de SVP et de CVP pour ces réseaux.

Cependant, on peut remarquer que ce résultat a également un côté positif : la difficulté du problème le rend intéressant pour des applications en cryptographie, par exemple pour définir des systèmes de cryptage qui seront *a priori* plus difficiles à casser que ceux décrits dans [5]. De plus, on se restreint ici à SVP et CVP, mais la densité de certaines matrices cycliques et le fait que l'algorithme de Trolin présenté ici soit constructif pourraient servir à résoudre d'autres problèmes.

Références

[1] R. Kannan et A. Bachem : Polynomial algorithms for computing of the Smith and Hermite normal forms of an integer matrix. *SIAM Journal of computing*, pages 499–507, 1979.

[2] I. Dinur, G. Kindler, S. Safra et R. Raz : Approximating CVP to within almost-polynomial factors is NP-hard. *Combinatorica*, pages 205–243, 2003.

[3] D. Micciancio : The shortest vector in a lattice is hard to approximate within some constant. *SIAM Journal of computing*, pages 2008–2035, 2001.

[4] Reduced row echelon form. `http://en.wikipedia.org/wiki/Hermite_normal_form`.

[5] J. Hoffstein, J. Pipher et J.H. Silverman : NTRU : a ring based public key cryptosystem. *Proc. of ANTS III*, pages 267–288, 1998.

[6] M. Trolin : Lattices with many cycles are dense. *Electronic Colloquium on Computational Complexity*, 2004.

Note : Pour les références [1, 5], n'ayant pu trouver l'article, seul l'abstract a été lu.

A Forme normale de Hermite

Revenons sur l'étape 2 de l'algorithme de Trolin pour montrer la densité des matrices à $n-1$ cycles. On veut, à partir d'une matrice H sous HNF, c'est-à-dire :

$$H = \begin{pmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \dots & h_{1,n-1} & h_{1,n} \\ 0 & h_{2,2} & h_{2,3} & \dots & h_{2,n-1} & h_{2,n} \\ 0 & 0 & h_{3,3} & \dots & h_{3,n-1} & h_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & h_{n-1,n-1} & h_{n-1,n} \\ 0 & 0 & 0 & \dots & 0 & h_{n,n} \end{pmatrix} \text{ où } \forall j < i,\ h_{i,i} > h_{j,i} \geqslant 0$$

obtenir une matrice B sous une HNF plus simple :

$$B = \begin{pmatrix} 1 & 0 & \dots & 0 & a_1 \\ 0 & 1 & \dots & 0 & a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & a_{n-1} \\ 0 & 0 & \dots & 0 & d \end{pmatrix} \text{ où } d = \det(\Lambda)$$

Pour passer sous cette forme, Trolin donne un procédé qui n'utilise que des opérations sur les lignes de H. Il précise également que $0 \leqslant a_i < d\ \forall i \in [\![1, n]\!]$.

Or, en appliquant les opérations décrites sur :

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

on obtient :

$$B = \begin{pmatrix} 1 & 0 & \mathbf{-15} \\ 0 & 1 & 7 \\ 0 & 0 & 47 \end{pmatrix}$$

Cela contredit $a_1 \geqslant 0 \dots$ Cependant, la propriété $0 \leqslant a_i\ \forall i \in [\![1, n]\!]$ n'intervient pas dans la preuve de la densité, ce qui ne pose donc aucun problème.