Some Improved Complexity Results for GapSVP and uSVP

Cheng Kuan

February 3, 2013

Abstract

In this paper, first, it is proved that finding the approximate shortest vector with length in $[\lambda_1, \gamma\lambda_1]$ could be Karp-reduced to $\text{GapSVP}_{\tilde{\gamma}}$ where $\gamma = \gamma^{\frac{1}{n(n+\log_2(\gamma n))(n-1)}}$.

Second, it is proved that shortest vector problem itself could be reduced to GapSVP with a quite small gap.

Third, we improve the complexity results of uSVP, proving uSVP could be reduced from SVP (our results are better than any known result). What's more, we prove that the search version of uSVP could be reduced to decisional version of uSVP with almost the same gap.

1 Introduction

Lattice is a wonderful mathematical structure. It is a set of all integer combinations of linearly independent base vector $b_1, b_2, \ldots, b_n$ in $\mathbb{R}^m$. Shortest vector problem is a NP-hard problem under random reduction which is proved by Micciancio[7]. However, the complexity relation among shortest vector problem and its related problems are still not very explicit.

Our contribution. In section 3 of this paper, we proves that finding the approximate shortest vector with length in $[\lambda_1, \gamma\lambda_1]$ could be Karp-reduced to $\text{GapSVP}_{\tilde{\gamma}}$ where $\tilde{\gamma} = \gamma^{\frac{1}{n(n+\log_2(\gamma n))(n-1)}}$. The reduction between the two problems has never been done before.

In section 4, we reduce SVP to GapSVP. Our work makes an improvement in proving the hardness of GapSVP. In this way, the complexity of GapSVP is more explicit than before.

In section 5, we proved SVP could be reduced to uSVP. Our result is slightly better than [3]. We also proved that the search version of uSVP could be reduced to its decision version, with almost the same gap.

2 Preliminaries

Given a base of n-dimensional space, a Lattice could be formulated by linear combination of base vectors using integer coefficients.

Definition 1 (Lattice)**.** *Given n linearly independent vectors $b_1, b_2, \ldots, b_n \in \mathbb{R}^m$, the lattice generated by them is defined as*

$$\mathcal{L}(b_1, b_2, \ldots, b_n) = \{\sum_{i=1}^{n} x_i b_i | x_i \in \mathbb{Z}\}.$$

We refer to $b_1, \ldots, b_n$ as a basis of the lattice. Equivalently, if we define B as the $m \times n$ matrix whose columns are $b_1, b_2, \ldots, b_n$, then the lattice generated by B is

$$\mathcal{L}(B) = \mathcal{L}(b_1, b_2, \ldots, b_n) = \{Bx | x \in \mathbb{Z}^n\}.$$

Definition 2 (Span). *The span of a lattice $\mathcal{L}(B)$ is the linear space spanned by its vectors,*

$$span(\mathcal{L}(B)) = span(B) = \{By | y \in \mathbb{R}^n\}.$$

Definition 3 (Shortest Vector Problem (SVP)). *Given a basis $B \in \mathbb{R}^{m \times n}$, find a nonzero lattice vector Bx (with $x \in \mathbb{Z}^n \setminus \{0\}$) such that*

$$\|Bx\| \leq \|By\|$$

for any other $y \in \mathbb{Z}^n \setminus \{0\}$.

Always, researchers use λ_i (also $\lambda_i(B)$ with respect to basis B) to denote the ith shortest vector in a lattice. Pay attention that λ_0 is 0, and λ_1 is the shortest vector. I also use Λ to denote a lattice.

GapSVP is defined as the following.

Definition 4 (GapSVP_γ). *The input consists of $B \in \mathbb{Z}^{m \times n}$ and $r \in \mathbb{Q}$.*

- *In YES instances, $\lambda_1(\mathcal{L}(B)) \leq r$.*
- *In NO instances, $\lambda_1(\mathcal{L}(B)) > \gamma \cdot r$.*

Now I give the definition of the problem finding the approximate shortest vector problem of a lattice.

Definition 5 (SVP_γ). *Given lattice $\mathcal{L}(B)$ where basis $B \in \mathbb{R}^{m \times n}$, find the vector v such that $\|v\| \in [\lambda_1(B), \gamma\lambda_1(B))$.*

Definition 6 (Unique Shortest Vector Problem(uSVP_γ)). *Given a lattice B such that $\lambda_2(B) > \gamma\lambda_1(B)$, find a nonzero vector $v \in \mathcal{L}(B)$ of length $\lambda_1(B)$.*

In the following passage, without loss of generality, assume that input basis B is full rank and is an integer matrix.

3 The Reduction from SVP_γ to $\text{GapSVP}_{\tilde{\gamma}}$

In this section, the first result of this paper is given.

3.1 The Capability of GapSVP Oracle

Lemma 1. *Given Lattice Basis $B \in \mathbb{Z}^{n \times n}$, using GapSVP_γ oracle, a range $(\alpha, \gamma\alpha]$, $\alpha \in \mathbb{R}$, could be found such that $\lambda_1(B) \in (\alpha, \gamma\alpha]$.*

Proof. We propose an algorithm just like binary search to prove this lemma.

First, pick a real number α_0 such that $\alpha_0 \geq \sqrt{n} \det(B)^{1/n} \geq \lambda_1$. Run GapSVP_γ oracle on input instance $\langle B, \alpha_0 \rangle$. If it returns "yes", then set $\alpha_1 = \alpha_0/2$. And the range of λ_1 reduces to $(0, \gamma\alpha_0]$. It's impossible for the oracle to return "no" at this step.

Each time we got $\alpha_i, i \geq 0$, we do the following operations. Check the result of GapSVP_γ oracle on input instance $\langle B, \alpha_i \rangle$. If it returns "yes", then set $\alpha_{i+1} = \alpha_i/2$. The range of λ_1 reduces to $(0, \gamma\alpha_i]$. We redo this step until the oracle returns "no".

Once the oracle returns "no", the situation needs to be discussed. We enter the second part of our algorithm. Here, we just consider situation of the most difficult input instance, because other input instance will lead us to get a even smaller range of λ_1.

To be more precisely, assume the oracle returns "no" on input $\langle B, \alpha_{k+1} \rangle$. The range of λ_1 is $(\alpha_{k+1}, \gamma\alpha_k]$. Two situations are here.

- λ_1 is in $(\alpha_k, \gamma\alpha_k]$.
- λ_1 is in $(\alpha_{k+1}, \alpha_k]$.

No matter in which situation, next step, we will set $\alpha_{k+2} = (\alpha_{k+1} + \alpha_k)/2$. On input $\langle B, \alpha_{k+2}\rangle$, if the oracle returns "no", then the range reduces to $(\alpha_{k+2}, \gamma\alpha_k]$. If the oracle returns "Yes", then the range reduces to $(\alpha_{k+1}, \gamma\alpha_{k+2}]$.

Then at each of the following step (considering the $\tilde{k}$th step), we set $\alpha_{\tilde{k}} = (a+b/\gamma)/2$, where $(a, b]$ denotes the range we got of λ_1 at hand. Run GapSVP oracle on $\langle B, \alpha_{\tilde{k}}\rangle$. No matter what the result is, the range reduces. Renew the value of a, b. Do the $\tilde{k}+1$th step the same way as we do the $\tilde{k}$th until, $b < \gamma a$. Let $\alpha = a$. Finally, we got that $\lambda_1(B) \in [\alpha, \gamma\alpha]$.

$\lambda_1 \leq 2^{p(m)}$, where m is the length of input. As a result, the proposed algorithm could be done in polynomial time of input length. This algorithm directly proves the lemma. □

3.2 The Reduction

We give the following theorem to reduce the approximation version of SVP to GapSVP. The method is just adapted from the proof reducing uSVP to GapSVP in [1].

Theorem 1. *For any given $\gamma \geq 1$, Approximate SVP$_\gamma \leq_p$ GapSVP$_{\tilde{\gamma}}$, $\tilde{\gamma} = \gamma^{\frac{1}{n(n+\log_2(\gamma n))(n-1)}}$.*

Proof. Given the input instance $B = [b_1, \ldots, b_n]$ which is the basis of a lattice. Now I will show the algorithm to compute a vector v such that $\|v\| \leq \gamma\lambda_1(B)$.

The main idea is to obtain lower rank sublattice of $\mathcal{L}(B)$ such that the approximate shortest vector are still in the sublattice.

As $\mathcal{L}(B)$ is n dimensional lattice, we only need to lower the rank by $n-1$ times.

To be more precisely, for the given lattice basis $\mathcal{L}(B)$, we find a serial of sublattices with rank decreased gradually. Assume that the serial of sublattices are denoted by $B_1, \ldots, B_n$, where $B_1 = B$ and $\text{rank}(B_i) - 1 = \text{rank}(B_{i+1})$.

We use the following method to lower the rank by 1.

Now I describe how to obtain B_{i+1} from B_i.

Given basis B_i, applying the method proposed in lemma 1, we could found the range $r = [\alpha, \tilde{\gamma}\alpha]$ where $\lambda_1(B_i)$ is in.

Generate three sublattices of $\tilde{B}_0 = B_i$. They are $\hat{B}_0 = [2b_1, b_2, \ldots, b_k]$, $\hat{B}_c = [b_1 + cb_2, 2b_2, b_3, \ldots, b_k], c = 1, 2$. It could be found that the shortest vector of $\tilde{B}_0$ is in one of the three generated sublattices. Apply the method proposed in lemma 1, we could find the range containing the length of shortest vector for each of the three sublattice. Assume the three ranges we got are $r_j, j = 0, 1, 2$. At least one sublattices has r_j intersecting r. Set $\tilde{B}_1$ to be $\hat{B}_j$ if r_j intersects r. If more than one sublattice has r_j intersecting r, set $\tilde{B}_1$ to be arbitrarily one of them. We do this for t times to get a serial of sublattices, where

$$\mathcal{L}(\tilde{B}_0) \supset \mathcal{L}(\tilde{B}_1) \supset \cdots \supset \mathcal{L}(\tilde{B}_t)$$

.

Here, $t > n(n + \log_2(\gamma n))$.

It could be concluded that $\lambda_1(\tilde{B}_t) \leq \tilde{\gamma}^t\lambda_1(\tilde{B}_0)$. Also we know that $\det(\tilde{B}_t) \geq 2^t \det(\tilde{B}_0)$, because each time we select a sublattice the value of the determinant at least doubles. Assume D to be the dual of $\tilde{B}_t$. $\det(D) \leq 1/(2^t \det(\tilde{B}_0))$. Applying the LLL algorithm we can find a vector $u \in \mathcal{L}(D)$ such that

$$\|u\| \leq 2^n\sqrt{n}\det(D)^{1/n} \leq \frac{\sqrt{n}2^n}{2^{t/n}\det(\tilde{B}_0)^{1/n}}.$$

Suppose the shortest vector of $\mathcal{L}(\tilde{B}_0)$ is $\tilde{u}_0$. According to Minkowski's bound, we have $\|\tilde{u}_0\| \leq \sqrt{n}\det(\tilde{B}_0)^{1/n}$. Consequently, the shortest vector of $\tilde{B}_t$ (suppose to be $\tilde{u}_t$) meet the following bound.

$$\|\tilde{u}_t\| \leq \tilde{\gamma}^t \sqrt{n}\det(\tilde{B}_0)^{1/n}$$

Using Cauchy-Schwarz inequality,

$$|\langle \tilde{u}_t, u\rangle| \leq \|\tilde{u}_t\| \cdot \|u\| \leq \tilde{\gamma}^t n 2^{n-t/n} < 1.$$

We know $\tilde{u}_t \in \mathcal{L}(\tilde{B}_t)$, $u \in \mathcal{L}(D)$. It means that $\langle \tilde{u}_t, u\rangle$ is an integer. This concludes that $|\langle \tilde{u}_t, u\rangle| = 0$. Taking the sublattice of $\tilde{B}_t$ orthogonal to u, we get a lower rank sublattice $\mathcal{L}(B_{i+1}) \subset \mathcal{L}(B_i)$ such that $\lambda_1(B_{i+1}) \leq \tilde{\gamma}^t \lambda_1(B_i)$.

Finally, after lowering rank for $n-1$ times, we could finally got $\mathcal{L}(B_n)$ such that its rank is 1 which mean its shortest vector could be found trivially. Also we have

$$\lambda_1(B_n) \leq \tilde{\gamma}^{(n-1)t}\lambda_1(B).$$

We know $\tilde{\gamma} = \gamma^{\frac{1}{n(n+\log_2(\gamma n))(n-1)}}$. As a result, the final conclusion is

$$\lambda_1(B_n) \leq \gamma\lambda_1(B).$$

This completes our proof. □

3.3 Improved Result

What's more, the result still has space for improvement. The improvement is dependent on the effectiveness of approximating algorithm for shortest vector problem. In the proof of theorem 1, we use the LLL algorithm, but there are more effective algorithm, so the result could be improved.

Theorem 2. *According to the effective polynomial time approximating algorithm of shortest vector problem, in theorem 1, $\tilde{\gamma}$ could be set to $\gamma^{\frac{1}{n(n\log\log n/\log n+\log(\gamma n))(n-1)}}$.*

Proof. According to the effective polynomial time approximating algorithm of shortest vector problem proposed in [2],it is clear that in polynomial time of input length, a vector of length in $2^{O(n\log\log n/\log n)}$ could be found. In this way, the parameter t in the proof of theorem 1 could be $n(n\log\log n/\log n+\log n)$. As we need $\tilde{\gamma}^{(n-1)t} \leq \gamma$, $\tilde{\gamma}$ could be set to $\gamma^{\frac{1}{(n-1)t}} = \gamma^{\frac{1}{n(n\log\log n/\log n+\log n)(n-1)}}$.

However, the algorithm is probabilistic, so the reduction turns to be a random reduction.

□

Theorem 3. *According to the effective polynomial time deterministic approximating algorithm of shortest vector problem, in theorem 1, $\tilde{\gamma}$ could be set to $\gamma^{\frac{1}{n(\frac{1}{4}(n-1)+\log(\gamma n))(n-1)}}$.*

Proof. According to the most effective polynomial time approximating algorithm of shortest vector problem proposed in [2],it is clear that in polynomial time of input length, a vector of length in $\frac{4}{3}^{(n-1)/2}\lambda_1$ could be found. In this way, the parameter t in the proof of theorem 1 could be $n(\frac{1}{4}(n-1)+\log n)$. As we need $\tilde{\gamma}^{(n-1)t} \leq \gamma$, $\tilde{\gamma}$ could be set to $\gamma^{\frac{1}{n(\frac{1}{4}(n-1)+\log n)(n-1)}}$.

However, the algorithm is probabilistic, so the reduction turns to be a random reduction.

□

4 The Reduction from SVP to GapSVP

This proof use the method also adapted from the method in [1].

Theorem 4. *SVP could be cook-reduced to GapSVP$_\gamma$, where* $\gamma = \sqrt{1 + \frac{1}{\lambda_1(\mathcal{L}(B))^2}}$.

Proof. Given the input instance $B = [b_1, \ldots, b_n]$ which is the basis of a lattice. The following algorithm computes the shortest vector of $\mathcal{L}(B)$, saying v, using GapSVP oracle.

The main idea is to obtain a lower rank sublattice of $\mathcal{L}(B)$ such that the shortest vector are still in the sublattice.

As $\mathcal{L}(B)$ is n dimensional lattice, we need to lower the rank by $n-1$ times.

To be more precisely, for the given lattice basis $\mathcal{L}(B)$, we find a serial of sublattices with rank decreased gradually. Assume that the serial of sublattices are denoted by $B_1, \ldots, B_n$, where $B_1 = B$ and $\text{rank}(B_i) - 1 = \text{rank}(B_{i+1})$.

We use the following method to lower the rank by 1. The method describes how to obtain B_{i+1} from B_i.

Given basis B_i, applying the method proposed in lemma 1, we could found the range $r = (\alpha, \gamma\alpha]$ which $\lambda_1(B_i)$ is in.

Generate three sublattices of $\tilde{B}_0 = B_i$. They are $\hat{B}_0 = [2b_1, b_2, \ldots, b_k]$, $\hat{B}_c = [b_1 + cb_2, 2b_2, b_3, \ldots, b_k], c = 1, 2$. It could be found that the shortest vector of $\tilde{B}_0$ is in at least one of the three generated sublattices. Apply the method proposed in lemma 1, we could find the range containing the length of shortest vector for each of the three sublattice. Assume the three ranges we got are $r_j, j = 0, 1, 2$. At least one of them contain $\lambda_1(B)$. Assume $\lambda_1(B)$ is in r_i, corresponding to $\mathcal{L}(\hat{B}_i)$. If $\mathcal{L}(\hat{B}_j)$ do not have the shortest vector of $\mathcal{L}(B)$, then $\lambda_1 \notin r_j$. This is because, according to lemma 1, if $\lambda_1 \in r_j$, r_j will not contain $\lambda_1(\hat{B}_j)$, as the gap of the oracle is $\sqrt{1 + \frac{1}{\lambda_1(\mathcal{L}(B))^2}}$ and $\lambda_1(\hat{B}_j)^2 \geq \lambda_1(B)^2 + 1$.

$$\sup\{x | x \in r_j\} < \gamma\lambda_1(B) \leq \lambda_1(\hat{B}_j)$$

As a result, we could find the sublattice $\mathcal{L}(\hat{B}_i)$ which contains the shortest vector. Set $\tilde{B}_1 = \hat{B}_i$. We do this for t times to get a serial of sublattices, where

$$\mathcal{L}(\tilde{B}_0) \supset \mathcal{L}(\tilde{B}_1) \supset \cdots \supset \mathcal{L}(\tilde{B}_t)$$

.

Here, $t > n(n + \log_2 n)$.

It could be concluded that $\lambda_1(\tilde{B}_t) = \lambda_1(B_0)$. Also we know that $\det(\tilde{B}_t) \geq 2^t \det(\tilde{B}_0)$, because each time we select a sublattice the value of the determinant at least doubles. Assume D to be the dual of $\tilde{B}_t$. $\det(D) \leq 1/(2^t \det(\tilde{B}_0))$. Applying the LLL algorithm we can find a vector $u \in \mathcal{L}(D)$ such that

$$\|u\| \leq 2^n \sqrt{n} \det(D)^{1/n} \leq \frac{\sqrt{n} 2^n}{2^{t/n} \det(\tilde{B}_0)^{1/n}}.$$

According to Minkowski's bound,

$$\|v\| \leq \sqrt{n} \det(\tilde{B}_0)^{1/n}$$

Using Cauchy-Schwarz inequality,

$$|\langle v, u \rangle| \leq \|v\| \cdot \|u\| < 1.$$

As $|\langle v, u \rangle|$ is an integer, $|\langle v, u \rangle| = 0$. Taking the sublattice of $\tilde{B}_t$ orthogonal to u, we get a lower rank sublattice $\mathcal{L}(B_{i+1}) \subset \mathcal{L}(B_i)$ such that $\lambda_1(B_{i+1}) = \lambda_1(B)$.

Finally, after lowering rank for $n-1$ times, we could finally got $\mathcal{L}(B_n)$ such that its rank is 1 which mean the shortest vector could be found trivially.

□

5 Complexity Results of Unique Shortest Vector Problem

5.1 Reduction from SVP to uSVP

In [3], it is proved that SVP $\leq_p$ uSVP$_\gamma, \gamma = \sqrt{1+\frac{1}{c\cdot 2^{4n^2}\lambda_1^2}}$. This could be improved.

Lemma 2. *For basis $B = [b_1, b_2, \ldots, b_n]$, it could be reduced to B' such that, if $u = \sum_i \alpha_i b'_i$ is a shortest vector, then $|\alpha_i| < cn(\frac{3}{2})^{n-i}\cdot(r_k(1+\epsilon))^{\frac{pk}{k-1}}$,where r_k denotes the hermite constant, k is the block size ($O(\log n/(\log\log n))$) and $p = n/k$.*

Proof. This could be done according to the slide reduction method proposed in [5]. Here I will explain why this basis reduction will meet the proposed property. The reduction method proposed in [5] require the dimension to be $n = pk$, here I made a little adjustment so that n could be any positive integer.

A basis B of an n-dimensional lattice L where $n = kp + q$ is slide reduced with a factor $\varepsilon \geq 0$ if it is size-reduced and satisfies the following two conditions.

- $\forall i \in [0, p-1]$, the block $B_{[ik+1,ik+k]}$ is HKZ-reduced.
- $\forall i \in [0, p-1]$, the block $B_{[ik+2,ik+k+1]}$ is $(1+\epsilon)$-DSVP-reduced. (If $q = 0$, $i \in [0, p-2]$.)

As a result, we have $\|b^*_{ik+1}\| \leq (r_k(1+\epsilon))^{\frac{k}{k-1}}\|b^*_{ik+k+1}\|$.
This induces $\|b^*_1\| \leq (r_k(1+\epsilon))^{\frac{ik}{k-1}}\|b^*_{ik+1}\| \Rightarrow \|b^*_1\| \leq (r_k(1+\epsilon))^{\frac{pk}{k-1}}\|b^*_{pk+1}\|$.
According to [5] B' is LLL-reduced. We have

$$\|b^*_1\| \leq (r_k(1+\epsilon))^{\frac{pk}{k-1}}\|b^*_{pk+1}\| \leq (\sqrt{\frac{4}{3}}(1+\epsilon))^{q-1}(r_k(1+\epsilon))^{\frac{pk}{k-1}}\|b^*_n\|$$

$$\|b_1\| = \|b^*_1\| \geq \|u\| \geq |\alpha_n|\|b^*_n\| \geq ((\sqrt{\frac{4}{3}}(1+\epsilon))^{q-1}(r_k(1+\epsilon))^{\frac{pk}{k-1}})^{-1}|\alpha_n|\|b^*_1\|$$

This implies $|\alpha_n| \leq (\sqrt{\frac{4}{3}}(1+\epsilon))^{q-1}(r_k(1+\epsilon))^{\frac{pk}{k-1}}$. $q < k = O(\log n/\log\log n)$, so $(\sqrt{\frac{4}{3}}(1+\epsilon))^{q-1}$ is a linear polynomial of n.

Suppose the lemma holds for $\alpha_i, \forall i > n-l$. According to Gram-Schmidt orthogonal method, we have the following.

$$\begin{aligned}\|b^*_1\| \geq \|u\| &\geq |\alpha_{n-l} + (\sum_{j=n-l+1}^{n}\mu_{j,n-l}\alpha_j)|\|b^*_{n-l}\| \\ &\geq \{(r_k(1+\epsilon))^{\frac{\lfloor (n-l-1)/k\rfloor}{k-1}}(\sqrt{\frac{4}{3}}(1+\epsilon))^{(n-l-1) \mod k}\}^{-1} \\ &\cdot |\alpha_{n-l} + (\sum_{j=n-l+1}^{n}\mu_{j,n-l}\alpha_j)|\|b^*_1\|\end{aligned} \tag{1}$$

$\sqrt{\frac{4}{3}}(1+\epsilon))^{(n-l-1) \mod k}$ is also a linear polynomial of n. Suppose it is less than $c_2 n$. We also knows that $\forall 1 \leq j < i \leq n, |\mu_{i,j}| \leq 1/2$.

$$
\begin{aligned}
|\alpha_{n-l}| &\le c_2 n (r_k(1+\epsilon))^{\frac{\lfloor (n-l-1)/k \rfloor}{k-1}} + (\sum_{j=n-l+1}^{n} |\mu_{j,n-l}\alpha_j|) \\
&\le c_2 n (r_k(1+\epsilon))^{\frac{\lfloor (n-l-1)/k \rfloor}{k-1}} + 1/2 \sum_{j=n-l+1}^{n} |\alpha_j|
\end{aligned} \tag{2}
$$

$$
c_2 n (r_k(1+\epsilon))^{\frac{\lfloor (n-l-1)/k \rfloor}{k-1}} + 1/2 \sum_{j=n-l+1}^{n} |\alpha_j| \le c_3 n (r_k(1+\epsilon))^{\frac{pk}{k-1}} (\frac{3}{2})^{n-l}
$$

As a result, we could conclude that $\forall i, |\alpha_i| < cn(\frac{3}{2})^{n-i} \cdot (r_k(1+\epsilon))^{\frac{pk}{k-1}}$.

□

For LLL reduction, we could have a similar result. It is $\alpha_i < 2^{n/2}(\frac{3}{2})^{n-i}$.

Theorem 5. *If the shortest vector u of the input lattice $\mathcal{L}(B)$ could be denoted as $u = \sum_i \alpha_i b_i$, knowing $|\alpha_i| < t_i$, then $SVP \le_p uSVP_\gamma$, $\gamma = \sqrt{1 + \frac{1}{c(\prod_{i=1}^{n} t_i)^2 \lambda_1(\mathcal{L}(B))^2}}$.*

Proof. Suppose $m_j = \prod_{i=1}^{j} t_i$. Denote $m_0 = 1$.

Consider the following matrix.

$$
B' = \begin{pmatrix}
m_n b_1 & m_n b_2 & \dots & m_n b_{n-1} & m_n b_n \\
1 & & & & \\
 & m_1 & & & \\
 & & \dots & & \\
 & & & m_{n-2} & \\
 & & & & m_{n-1}
\end{pmatrix}
$$

I will prove that, $\mathcal{L}(B')$ has a unique shortest vector corresponding to one of the shortest vector of $\mathcal{L}(B)$.

$$
\forall x \in \mathbb{Z}^n, \|B'x\|^2 = m_n^2 \|Bx\|^2 + \sum_{i=1}^{n} (m_{i-1}x_i)^2 < m_n^2(\|Bx\|^2 + 1)
$$

If $\|Bx\| = \|By\| = \lambda_1(B)$, then $\exists k, \forall i, k < i \le n, x_i = y_i, |x_k| > |y_k|$. We could see $\|B'x\| > \|B'y\|$. The reason follows.

It is easy to see that $(x_k m_{k-1})^2 - (y_k m_{k-1})^2 \ge m_{k-1}^2 = (t_{k-1} m_{k-2})^2 \ge (|y_{k-1}| m_{k-2})^2 + m_{k-2}^2 \ge \sum_{j=1}^{k-1} (|y_j| m_{j-1})^2 + m_0^2 > \sum_{j=1}^{k-1} (|y_j| m_{j-1})^2$. So we have that $|x_i| = |y_i|$.

According to [6], when two shortest vectors, say Bx and By, have the same parity vector (for Bx the parity vector is $p(x) = [x_1 \mod 2, \dots, x_n \mod 2]$), then $Bx = By$. This implies if $\|Bx\| = \|By\| = \lambda_1, \forall i \in [n], |x_i| = |y_i|$, then $\forall i \in [n], x_i = y_i$.

As a result, we could see that there is only one unique shortest vector for $\mathcal{L}(B')$.

$\lambda_2^2(B') - \lambda_1^2(B') \ge 1$. So the gap is $\sqrt{1 + \frac{1}{c(\prod_{i=1}^{n} t_i)^2 \lambda_1(\mathcal{L}(B))^2}}$ □

Theorem 6. *SVP could be reduced to $uSVP_\gamma$ with*

$$
\gamma = \sqrt{1 + \frac{1}{c_1 (c_2 n)^{2n} (3/2)^{(n-1)n} (r_k(1+\epsilon))^{2n\frac{pk}{k-1}} \lambda_1(\mathcal{L}(B))^2}},
$$

for some constant c_1, c_2. r_k is the kth hermite constant. k is the block size $(O(\log n/(\log\log n)))$ and $p = n/k$.

If using LLL reduction, we could get that $SVP \le_p uSVP_\gamma, \gamma = \sqrt{1 + \frac{1}{c 2^{n^2} (\frac{3}{2})^{(n-1)n} \lambda_1^2}}$.

Proof. The theorem follows immediately from lemma 4 and theorem 5. □

5.2 Search versus Decision

We will show that the search version uSVP could be reduced to decision version uSVP maintaining almost the same gap.

In order to do this reduction, we adapted the methods of Kannan[4] and the methods of Hu and Pan[2].

Both of the two methods aimed to reduce SVP to decisional SVP. However, the parameters in their methods are very large. If just apply their methods to do the reduction we cannot get results better than [3].

Lemma 3. *Given the value of an integer r, knowing $r = m+\sum_{i=1}^{n} \alpha_i p_i$, where $p_n|m, p_i|p_{i+1}$, $\alpha_i < \lfloor \frac{p_1}{2} \rfloor$, α_i $(i \in [n-1])$ could be computed. Here, $\forall i, p_i$ and α_i are integers.*

Proof. First, we compute $r_n = r \mod p_n$. Here, we do not need to know the value of α_n. Once we have r_i, we compute $r_{i-1} = r_i \mod p_{i-1}$, choosing α_{i-1} such that $\alpha_{i-1} \cdot p_{i-1}$ closest to r_i (α_{i-1} is unique. Only one value of α_{i-1} could be choose). In this way, we could compute $\alpha_i, (i \in [n-1])$ one by one.

□

Lemma 4. *Given the value of an integer r, knowing $r = mp_{n+1} + \sum_{i=1}^{n} \alpha_i p_i$, where $p_i|p_{i+1}$, $\alpha_i < \lfloor \frac{p_{i+1}}{p_i} \rfloor$, α_i $(i \in [n])$ and m could be computed. Here, $m \geq 0, \forall i, p_i > 0, \alpha_i \geq 0$.*

Proof. First, we compute $r_n = r \mod p_{n+1}, m = r/p_{n+1}$. Once we have r_i, we compute $r_{i-1} = r_i \mod p_i$, $\alpha_i = r_i/p_i$. In this way, we could compute $\alpha_i, (i \in [n])$ one by one.

□

Lemma 5. *Using duSVP oracle, the exact length of the shortest vector of the given input lattice could be found.*

Proof. This could be done using binary search.

According to Minkowski's bound, we have the following bound for shortest vector u of input lattice $\mathcal{L}(B)$.

$$\|u\| \leq \sqrt{n} \det(B)^{1/n}$$

First we just take $\langle B, d \rangle$, where $d = \sqrt{n} \det(B)^{1/n}$, as the input for duSVP oracle. Set the original range of $\lambda_1(B)$ to be $[a, b] = [0, d]$ (means $a = 0, b = d = \sqrt{n} \det(B)^{1/n}$). We do the following iteration.

For each time run the duSVP oracle on $\langle B, d \rangle, d = (a + b)/2$. If it returns "Yes", then the range of $\lambda_1(B)$ is set to be $[a, b] = [a, d]$, else set the range to be $[a, b] = [d, b]$. Finally, the length of $\lambda_1(B)$ could be settled in polynomial time of the input length.

□

Theorem 7.

$$\textit{search-uSVP}_{\gamma} \leq_p \textit{decision-uSVP}_{\gamma\sqrt{1-\epsilon}}$$

Proof. According to lemma 4, we could assume that we have the oracle O which could output the length of the unique shortest vector given any input lattice basis with gap γ'.

Now, given the input lattice basis B, we construct the following new lattice.

$$B' = LLL(B)$$

$$B'' = \begin{pmatrix} m_n b'_1 & m_n b'_2 & \dots & m_n b'_{n-1} & m_n b'_n \\ 1 & & & & \\ & m_1 & & & \\ & & \dots & & \\ & & & m_{n-2} & \\ & & & & m_{n-1} \end{pmatrix}$$

$t_i = 2^{n/2}(\frac{3}{2})^{n-i}$, $m_i = \prod_{j=1}^{i} t_j$. We already know that if $u = \sum_{i=1}^{n} \alpha_i b'_i$ then $\alpha_i < t_i$. Assume $m_0 = 1$.

If $B''x$ is the shortest vector of $\mathcal{L}(B'')$ then, $B'x$ is the shortest vector of $\mathcal{L}(B')$. If not, assume $B'y$ is the shortest vector of $\mathcal{L}(B')$. That is $\|B'y\| < \|B'x\|$. It means $\|B'x\|^2 - \|B'y\|^2 \geq 1$. Consider the vector $B''y$ in $\mathcal{L}(B'')$. $\|B''y\|^2 = m_n^2\|B'y\|^2 + \sum_{i=1}^{n}(y_i m_{i-1})^2 < m_n^2\|B'x\|^2 < \|B''x\|^2$. This contradicts $B''x$ is the shortest vector of $\mathcal{L}(B'')$.

We could also know that $\mathcal{L}(B'')$ has a unique shortest vector. If not, assume that $B''x$,$B''y$ are two shortest vector $B''x \neq \pm B''y$. $\|B''x\|^2 = m_n^2\|B'x\| + \sum_{i=1}^{n}(x_i m_{i-1})^2$. It should be $\|B''x\| = \|B''y\|$. So we have $\|B'x\| = \|B'y\|$. It means both $B'x$ and $B'y$ are the shortest vector of $\mathcal{L}(B')$. This is impossible, as $\mathcal{L}(B')$ a unique shortest vector.

Suppose $B''x$ is the shortest vector of $\mathcal{L}(B'')$. Using our oracle, we could get $\lambda_1(B'')$. According the above lemma, we could get $|x_i|, i = 1, \dots, n$.

Now we compute the sign for each x_i.

Construct the following basis.

$$\tilde{B} = \begin{pmatrix} m_1 x_1 b'_1 & m_1 x_2 b'_2 & \dots & m_1 x_{n-1} b'_{n-1} & m_1 x_n b'_n \\ 1 & -1 & & & \end{pmatrix}$$

Assume that $x_1 > 0$, $x_2 \neq 0$. Now we compute the sign of x_2. It is easy to see that $\tilde{B}$ has unique shortest vector.

Run O on $\tilde{B}$. We get $\lambda_1(\tilde{B})$. According to lemma 3 and 4, we could get $\lambda_1(\tilde{B})$ mod m_1. If it is 0, we know x_2 is positive, else it is negative. In this way, all the sign of x_i could be got. So we could get the shortest vector of B.

Next we analysis the gap that O need.

Denote the gap between λ_1 and λ_2 of $\mathcal{B}''$ to be γ''.

If $\gamma' < \gamma''$, run O on B''. We could get $\lambda_1(B'')^2 = m_n^2\|B'x\|^2 + \sum_{i=1}^{n} x_i^2 m_{i-1}^2$. According to lemma 3, we could get x and $B'x$ is the unique shortest vector of $\mathcal{L}(B)$.

$$\gamma'' = \sqrt{\frac{(\lambda''_2)^2}{(\lambda''_1)^2}} > \sqrt{\frac{\lambda_2^2}{\lambda_1^2+1}} = \gamma\sqrt{\frac{\lambda_1^2}{\lambda_1^2+1}}$$

Set $\gamma' = \gamma\sqrt{\frac{\lambda_1^2}{\lambda_1^2+1}} = \gamma\sqrt{1-\frac{1}{\lambda_1^2+1}} = \gamma\sqrt{1-\epsilon}$. The proof is complete.

□

References

[1] Vadim Lyubashevsky, Daniele Micciancio, "On Bounded Distance Decoding, Unique Shortest Vectors, and the Minimum Distance Problem", 2009.

[2] Gengran Hu, Yanbin Pan, "A New Reduction from Search SVP to Optimization SVP", 2012.

[3] Divesh Aggarwal, Chandan Dubey, "Improved Hardness Results for Unique Shortest Vector Problem", 2011.

[4] Ravi Kannan, "Minkowski's Convex Body Theorem and Integer Programming", 1987.

[5] Nicolas Gama, Phong Q. Nguyen, "Finding Short Lattice Vectors within Mordell's inequality", 2008.

[6] R Kumar, D Sivakumar, "A note on the shortest lattice vector problem", 1999.

[7] S. Goldwasser and D. Micciancio, "Complexity of lattice problems", Springer, 2002.