Solving closest vector instances using an approximate shortest independent vectors oracle

Chengliang Tian[*] Wei Wei [†] Dongdai Lin [‡]

July 14, 2014

Abstract

Given a lattice $L \subset \mathbb{R}^n$ and some target vector, this paper studies the algorithms for approximate closest vector problem (CVP$_\gamma$) by using an approximate shortest independent vectors problem oracle (SIVP$_\gamma$). More precisely, if the distance between the target vector and the lattice is no larger than $\frac{c}{\gamma n}\lambda_1(L)$ for any constant $c > 0$, we give randomized and deterministic polynomial time algorithms to find a closest vector, which improves the known result by a factor of $2c$. Moreover, if the distance between the target vector and the lattice is larger than some quantity with respect to $\lambda_n(L)$, using SIVP_γ oracle and Babai's nearest plane algorithm, we can solve $\mathrm{CVP}_{\gamma\sqrt{n}}$ in deterministic polynomial time. Specially, if the approximate factor $\gamma \in (1, 2)$ in the SIVP$_\gamma$ oracle, we obtain a better reduction factor for CVP.

Key words: Lattices, Closest vector problem, shortest independent vectors problem, Reductions

1 Introduction

Lattices are discrete subgroups of $\mathbb{R}^n$. They are powerful mathematical objects that have been used to efficiently solve many important problems in computer science, most notably in the areas of cryptography and combinatorial optimization. In lattice theory, the most important and widely studied computational problems are Shortest Vector Problem (SVP) and Closest Vector Problem (CVP). Given a lattice $L \subseteq \mathbb{R}^n$, the SVP_γ is the problem of finding a non-zero lattice vector of length at most $\gamma\lambda_1(L)$, where $\lambda_1(L)$ denotes the length of shortest non-zero lattice vector. Given a lattice $L \subseteq \mathbb{R}^n$ and a target vector $\mathbf{t} \in \mathbb{R}^n$, the CVP_γ is the problem of finding a $\mathbf{v} \in L$ such that $\|\mathbf{v} - \mathbf{t}\| \leq \gamma \operatorname{dist}(\mathbf{t}, L)$, where $\operatorname{dist}(\mathbf{t}, L) = \min\{\|\mathbf{u} - \mathbf{t}\| : \forall \mathbf{u} \in L\}$ denotes the distance between $\mathbf{t}$ and L. In 1999, Goldreich, Micciancio, Safra and Seifert [1] first studied the relationship between these two problems and gave a deterministic polynomial-time rank-preserving reduction from SVP$_\gamma$ to CVP$_\gamma$ for any approximate factor $\gamma \geq 1$, which implies that SVP$_\gamma$ is not harder than CVP$_\gamma$.

It is natural to ask whether CVP$_\gamma$ is strictly harder than SVP$_\gamma$. In terms of known computational complexity results, the answer may be Yes. Since for any constant c and approximate factor $\gamma = n^{c/\log\log n}$, CVP$_\gamma$ is NP-hard under deterministic reductions [2], while the proof that SVP$_\gamma$ is NP-hard with the same approximate factor is randomized and under a strong complexity assumption [3]. A possible way to derandomized is giving a deterministic reduction from CVP$_\gamma$ to SVP$_\gamma$. Using an exact SVP oracle, Kannan [4] presented a deterministic polynomial time algorithm for solving approximate closest vector problem CVP$_{\sqrt{n}}$. Ajtai et al. [5] generalized Kannan's reduction technique and proposed a $2^{O(1+1/\epsilon)n}$ time algorithm for solving $\mathrm{CVP}_{1+\epsilon}$ by sampling short vectors. In another survey paper [6], using dual lattice and transference theorem in the geometry of numbers [7], Kannan proved that $\mathrm{CVP}_{\gamma^2 n^{3/2}}$ can be reduced to SVP$_\gamma$ in deterministic polynomial time. Recently, combining Kannan's lattice-embedding technique [4] with the reduction from

[*] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100093, P.R. China. Email: tianchengliang@iie.ac.cn.

[†] Institute for Advanced Study, Tsinghua University, Beijing, 100084, P.R. China. Email: wei-wei08@mails.tsinghua.edu.cn

[‡] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100093, P.R. China. Email: ddlin@iie.ac.cn

$\mathrm{BDD}_{1/2\gamma}$ to uSVP_γ given by Lyubashevsky and Micciancio [8], Dubey and Holenstein [9] improved Kannan's result [6] and obtained a deterministic polynomial-time rank-preserving reduction from $\mathrm{CVP}_{\gamma^2\sqrt{n}}$ to SVP_γ.

Ajtai's groundbreaking work [10] which connected the worst-case and average-case complexity of certain computational problems on lattices has opened the door to cryptography based on worst-case hardness. Regev's results [11] further broaden the foundation of lattice-based cryptography. Their works show that the security of all the cryptographic protocols based on SIS (Small Integer Solution) and LWE (Learning With Errors) depends on the worst-case hardness of SIVP_γ (the definition will be given in Section 2). Therefore it is essential to compare the harness among SIVP_γ, SVP_γ and CVP_γ. In order to study the hardness of SIVP_γ, Blömer and Seifert [12] first gave a deterministic polynomial time reduction from exact CVP to exact SIVP, but the reduction didn't preserve the rank of lattices. Combining the lattice-embedding technique with the relationship of primal-dual lattices, Micciancio [13] improved their result and obtained a deterministic polynomial-time rank-preserving reduction. Furthermore, through constructing sublattice skillfully, the reference [13] also gave a deterministic polynomial-time rank-preserving reduction from SIVP_γ to CVP_γ for any approximate factor $\gamma \geq 1$, which implies that the exact CVP and the exact SIVP are equivalent and SIVP_γ is not harder than CVP_γ. Naturally, we also want to know whether CVP_γ is strictly harder than SIVP_γ. In SODA 2008, Micciancio [13] proposed the following open problem:

Open Problem: Is there a deterministic polynomial time reduction from CVP_γ to SIVP_γ that preserves the rank of the lattice and approximation factor?

OUR RESULTS. Stemmed from the efforts to solve the open problem, we give a helpful exploration about the relationships between SIVP_γ and some special CVP_γ instances. More precisely, if the distance between the target vector and the lattice is less than some quantity with respect to $\lambda_1(L)$, we give randomized and deterministic polynomial time reductions from $\mathrm{BDD}_{\frac{c}{\gamma n}}$ to SIVP_γ for any constant $c > 0$, which improves the known result by a factor of $2c$. Moreover, if the distance between the target vector and the lattice is lager than some quantity with respect to $\lambda_n(L)$, using SIVP_γ oracle and Babai's nearest plane algorithm [14], we can solve $\mathrm{CVP}_{\gamma\sqrt{n}}$ in deterministic polynomial time, and for a uniformly chosen target vector, its distance from the lattice satisfies this constraint with probability not less than $1/2$. Specially, if the approximate factor $\gamma \in (1, 2)$ in the SIVP_γ oracle, we obtain a better result.

ROAD MAP. In Section 2, we review necessary concepts and notations, and then gives some useful lemmas for our proof. Our main results are stated and proved in Section 3 and Section 4. Using the SIVP_γ oracle, two algorithms for finding a closest vector when the target is close to the lattice are presented in Section 3. Section 4 gives polynomial time algorithms to approximate a closest vector when the target is far from the lattice. Finally, we conclude the paper in Section 5.

2 Preliminaries

In this section, we will give some necessary concepts on lattices and some useful lemmas for our proofs. First, we give some notations. For any real x, $\lfloor x \rfloor$ denotes the largest integer not larger than x and $\lceil x \rceil$ denotes the smallest integer not smaller than x. The n-dimensional Euclidean space is represented by $\mathbb{R}^n$. $\|\cdot\|$ denotes the Euclidean norm. We use bold lower letters (e.g., $\mathbf{x}$) to denote vectors, and bold upper case letters (e.g., $\boldsymbol{M}$) to denote matrices. The i-th coordinate of $\mathbf{x}$ is denoted x_i. For a set $S \subseteq \mathbb{R}^n$, $r \in \mathbb{R}$, $rS = \{r\mathbf{y} : \mathbf{y} \in S\}$ denotes the scaling of S by r.

2.1 Lattices and lattice problems

LATTICES. A lattice consists of all linear combinations with integer coefficients of some set of linearly independent vectors in the Euclidean space. If $\mathbf{b}_1, \cdots, \mathbf{b}_n \in \mathbb{R}^m$ are linearly independent, then the lattice spanned by these vectors is given by

$$L = L(\mathbf{B}) = \left\{ \sum_{i=1}^{n} z_i \mathbf{b}_i : z_i \in \mathbb{Z} \right\}.$$

where the matrix $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_n] \subset \mathbb{R}^{m \times n}$ is called a basis of the lattice. Usually, the basis of a lattice L is not unique. The number m is called the dimension of the lattice L and n is called the rank of the lattice L. If $m = n$, the lattice is called full rank. In the Euclid space, every non-full rank lattice is isomorphic to a full rank lattice. Hence without loss of generality, in the rest of our paper, we assume that all the lattices are full rank. The fundamental parallelepiped of $\mathbf{B}$ is defined as

$$\mathcal{P}(\mathbf{B}) = \left\{ \sum_{i=1}^{n} x_i \mathbf{b}_i : x_i \in [0,1) \right\}.$$

We denote the volume of the fundamental parallelepiped as $\det(L)$, which is independent of the choice of the basis.

MINKOWSKI'S MINIMA. For any $1 \leq i \leq n$, the i-th *successive minimum* with respect to a lattice L is defined as

$$\lambda_i(L) = \inf\{r > 0 : \dim(\text{span}(L \cap r\mathcal{B}(0,1))) \geq i\},$$

where $\mathcal{B}(0,1)$ denotes the open unit ball in the Euclidean norm. Specially, $\lambda_1(L) = \min\{\|\mathbf{v}\| : \mathbf{v} \in L, \mathbf{v} \neq \mathbf{0}\}$ denotes the length of the shortest non-zero lattice vector.

COVERING RADIUS. The *covering radius* associated to a lattice L is defined to be $\rho(L) = \max_{\mathbf{t} \in \mathbb{R}^n} \min_{\mathbf{v} \in L} \|\mathbf{v} - \mathbf{t}\|$.

GRAM-SCHMIDT ORTHOGONALIZATION. Let $\mathbf{b}_1, \cdots, \mathbf{b}_n \in \mathbb{R}^n$ be linearly independent vectors. Let π_i denote the projection over the orthogonal supplement of the linear span of $\mathbf{b}_1, \cdots, \mathbf{b}_{i-1}$. The *Gram-Schmidt orthogonalization* (GSO) is the family $(\tilde{\mathbf{b}}_1, \ldots, \tilde{\mathbf{b}}_n)$ defined as: $\tilde{\mathbf{b}}_1 = \mathbf{b}_1$ and for $i \geq 2$, $\tilde{\mathbf{b}}_i = \pi_i(\mathbf{b}_i)$. Then $\tilde{\mathbf{b}}_i = \mathbf{b}_i - \sum_{j=1}^{i-1} \mu_{i,j} \tilde{\mathbf{b}}_j$, where $\mu_{i,j} = \langle \mathbf{b}_i, \tilde{\mathbf{b}}_j \rangle / \|\tilde{\mathbf{b}}_j\|^2$ for $1 \leq j < i \leq n$.

DUALITY. Given a lattice $L = L(\mathbf{B})$, the *dual lattice* of L is the lattice

$$L^\star = \{\mathbf{w} \in \text{span}(L) : \langle \mathbf{w}, \mathbf{v} \rangle \in \mathbb{Z}, \forall \mathbf{v} \in L\}.$$

It is easy to verify that $(\mathbf{B}^{\mathrm{T}})^{-1}$ is a basis of $L^\star$, which is called the dual basis of $\mathbf{B}$.

LATTICE PROBLEMS. For computational purpose, it is usually assumed that all lattices vectors have integer entries, namely, the lattice basis is given by an integer matrix $\mathbf{B} \in \mathbb{Z}^{n \times n}$. There are several important computational problems in lattice theory. Here we give their strict definitions as follows.

Definition 2.1 (Shortest Vector Problem (SVP$_\gamma$)). *Given a basis $\mathbf{B} \in \mathbb{Z}^{n \times n}$ for a lattice $L = L(\mathbf{B})$, find a lattice vector $\mathbf{v} \in L$ such that $\|\mathbf{v}\| \leq \gamma \lambda_1(L)$.*

Definition 2.2 (Closest Vector Problem (CVP$_\gamma$)). *Given a basis $\mathbf{B} \in \mathbb{Z}^{n \times n}$ for a lattice $L = L(\mathbf{B})$ and some vector $\mathbf{t} \in \mathbb{R}^n$ (generally not in L), find a lattice vector $\mathbf{v} \in L$ such that $\|\mathbf{v} - \mathbf{t}\| \leq \gamma \,\text{dist}\,(\mathbf{t}, L)$, where $\text{dist}\,(\mathbf{t}, L) = \min_{\mathbf{u} \in L} \|\mathbf{u} - \mathbf{t}\|$ denotes the distance between $\mathbf{t}$ and L.*

Definition 2.3 (Bounded Distance Decoding (BDD$_\gamma$)). *Given a basis $\mathbf{B} \in \mathbb{Z}^{n \times n}$ for a lattice $L = L(\mathbf{B})$ and a target point $\mathbf{t} \in \mathbb{R}^n$ such that $\text{dist}\,(\mathbf{t}, L) \leq \gamma \lambda_1(L)$, output a lattice vector $\mathbf{v} \in L(\mathbf{b})$ such that $\|\mathbf{v} - \mathbf{t}\| \leq \gamma \lambda_1(L)$.*

Definition 2.4 (Shortest Independent Vectors Problem (SIVP$_\gamma$)). *Given a basis $\mathbf{B} \in \mathbb{Z}^{n \times n}$ for a lattice $L = L(\mathbf{B})$ and our goal is to find n linearly independent vectors $\mathbf{v}_1, \cdots, \mathbf{v}_n \in L$ such that $\max_i \|\mathbf{v}_i\| \leq \gamma \lambda_n(L)$.*

2.2 Useful lemmas

In this subsection, we will give some useful lemmas for our reductions.

Since we study lattices from a computational point of view, without loss of generality, we assume that lattices are represented by a basis with integer coordinates. By the definition of Gram-Schmidt orthogonalization, the following lemma bounds the bit size of the representation of any Gram-Schmidt orthogonalization vector.

Lemma 2.5 ([15]). *For a sequence of n linearly independent vectors $\mathbf{b}_1, \cdots, \mathbf{b}_n$, their Gram-Schmidt orthogonalization is the sequence of vectors $\tilde{\mathbf{b}}_1, \cdots, \tilde{\mathbf{b}}_n$. Then the representation of any vector $\tilde{\mathbf{b}}_i$ as a vector of quotients of natural numbers takes at most $\text{poly}(M)$ bits for $M = \max\{n, \log(\max_i \|\mathbf{b}_i\|)\}$.*

Clearly, any set of n linearly independent lattice vectors is not necessary a lattice basis. The following useful lemma says that any full-rank set of vectors in a lattice can be efficiently converted into a basis of the lattice, without increasing the length of the Gram-Schmidt vectors.

Lemma 2.6 ([15]). *There is a deterministic polynomial time algorithm* $\mathrm{ConverttoBasis}(\mathbf{B}, \boldsymbol{S})$ *that on input a lattice basis* $\mathbf{B}$ *and linearly independent lattice vectors* $\boldsymbol{S} = \{\mathbf{s}_1, \cdots, \mathbf{s}_n\} \subset L(\mathbf{B})$ *such that* $\|\mathbf{s}_1\| \leq \|\mathbf{s}_2\| \leq \cdots \leq \|\mathbf{s}_n\|$, *outputs a basis* $\mathbf{R}$ *equivalent to* $\mathbf{B}$ *such that* $\|\mathbf{r}_k\| \leq \max\{(\sqrt{k}/2)\|\mathbf{s}_k\|, \|\mathbf{s}_k\|\}$ *for all* $k = 1, \cdots, n$. *Moreover, the new basis satisfies* $\mathrm{span}(\mathbf{r}_1, \cdots, \mathbf{r}_k) = \mathrm{span}(\mathbf{s}_1, \cdots, \mathbf{s}_k)$ *and the length of their Gram-Schmidt orthogonalization vectors satisfying* $\|\tilde{\mathbf{r}}_k\| \leq \|\tilde{\mathbf{s}}_k\|$ *for all* $k = 1, \cdots, n$.

About the relationships between primal lattice and its dual, we have the following two important results. Lemma 2.7 shows that, in appropriate order, the Gram-Schmidt orthogonalization vectors of the dual basis are in the same direction as that of the primal basis and their lengths are the inverses of each other. Lemma 2.8 is well known as transference theorem. It reflects the properties of the successive minima between a lattice and its dual.

Lemma 2.7 ([16]). *Let* $\mathbf{b}_1, \cdots, \mathbf{b}_n$ *be some basis of* L *and* $\tilde{\mathbf{b}}_1, \cdots, \tilde{\mathbf{b}}_n$ *be its Gram-Schmidt orthogonalization. Let* $\mathbf{d}_1, \cdots, \mathbf{d}_n$ *be the dual basis of* $\mathbf{b}_1, \cdots, \mathbf{b}_n$ *and let* $\tilde{\mathbf{d}}_n, \cdots, \tilde{\mathbf{d}}_1$ *be its Gram-Schmidt orthogonalization in reverse order. In other words,* $\tilde{\mathbf{d}}_n = \mathbf{d}_n$, $\tilde{\mathbf{d}}_i = \mathbf{d}_i - \sum_{j>i} \nu_{i,j}\tilde{\mathbf{d}}_j$, *where* $\nu_{i,j} = \frac{\langle \mathbf{d}_i, \tilde{\mathbf{d}}_j\rangle}{\langle \tilde{\mathbf{d}}_j, \tilde{\mathbf{d}}_j\rangle}$ *for* $1 \leq i < j \leq n$. *Then*

$$\forall 1 \leq i \leq n, \quad \tilde{\mathbf{d}}_i = \frac{\tilde{\mathbf{b}}_i}{\|\tilde{\mathbf{b}}_i\|^2}.$$

Lemma 2.8 ([7]). *For any n-dimensional lattice* L, $\lambda_1(L)\lambda_n(L^\star) \leq n$.

In SODA'00, Klein [17] proposed a randomized algorithm to find the closest vector when the target vector is unusually close to the lattice. Actually, it is a randomized version of Babai's algorithm [14]. The algorithm randomly samples lattice points from a Gaussian-like distribution and chooses the closest points among all the samples.

Lemma 2.9 ([17]). *There is a randomized algorithm* $\mathrm{Klein}(\mathbf{B}, \mathbf{t})$ *that, when given an n-dimensional lattice L generated by basis vectors* $\mathbf{b}_1, \cdots, \mathbf{b}_n$ *and a target* $\mathbf{t} \in \mathbb{R}^n$ *that's at distance D away from L, will find the closest lattice vector to* $\mathbf{t}$, *in time* $n^{D^2/\min_i \|\tilde{\mathbf{b}}_i\|^2}$, *where* $\tilde{\mathbf{b}}_1, \cdots, \tilde{\mathbf{b}}_n$ *are Gram-Schmidt orthogonalization vectors of* $\mathbf{b}_1, \cdots, \mathbf{b}_n$.

3 Find a closest lattice vector when it's close to the lattice

In this section, we shall study the algorithms for special CVP instance-BDD_γ problem with an SIVP_γ oracle. We improve the presented result in two different algorithms, randomized and deterministic. First, we review some previous work as following.

Lemma 3.1 ([8]). *For any* $\gamma \geq 1$, *there is a polynomial time Cook-reduction from* $\mathrm{BDD}_{1/(2\gamma)}$ *to* uSVP_γ.

Lemma 3.2 ([18]). *For any* $\gamma \geq 1$, *there is a probabilistic polynomial time reduction from* $\mathrm{uSVP}_{\gamma n}$ *to* SIVP_γ.

Combining the above two lemmas, we have the following result which is also shown in reference [19].

Lemma 3.3. *For any* $\gamma \geq 1$, *there is a probabilistic polynomial time reduction from* $\mathrm{BDD}_{1/(2\gamma n)}$ *to* SIVP_γ.

Combining Klein's algorithm [17] and the relationship between primal and dual lattices, we first improve Lemma 3.3 using a randomized reduction algorithm. Namely, we prove the following result.

Theorem 3.4. *For any* $\gamma \geq 1$ *and any constant* $c > 0$, *there exists a randomized polynomial time reduction from* $\mathrm{BDD}_{c/\gamma n}$ *to* SIVP_γ.

Algorithm 1 BDD Algorithm: `BDD` $(\mathbf{B}, \mathbf{t})$

```
Input: A lattice basis $\mathbf{B} \in \mathbb{Z}^{n\times n}$, a target vector $\mathbf{t}$ such that $\operatorname{dist}(\mathbf{t}, L) < \frac{c}{\gamma n}\lambda_1(L)$ and a $\mathrm{SIVP}_\gamma$ oracle $\mathcal{O}$, where
    $1 < \gamma \le \mathrm{poly}(n)$, $c > 0$ be any constant.
Output: A lattice vector $\mathbf{v} \in L$ such that $\operatorname{dist}(\mathbf{t}, L) = \|\mathbf{v} - \mathbf{t}\|$.
1: Compute the dual basis of $\mathbf{B}$: $\boldsymbol{W} = [\mathbf{w}_1, \cdots, \mathbf{w}_n] = (\mathbf{B}^{\mathrm{T}})^{-1}$, which is a basis of $L^\star$.
2: Invoking $\mathrm{SIVP}_\gamma$ oracle on the lattice $L^\star$, output $\boldsymbol{S} = \{\mathbf{s}_1, \cdots, \mathbf{s}_n\} \leftarrow \mathrm{SIVP}_\gamma(L^\star)$.
3: Compute a basis of $L^\star$: $\mathbf{D} = [\mathbf{d}_1, \cdots, \mathbf{d}_n] \leftarrow \mathrm{ConverttoBasis}(\boldsymbol{W}, \boldsymbol{S})$.
4: Compute a basis of the original lattice $L$: $\mathbf{R} = [\mathbf{r}_1, \cdots, \mathbf{r}_n] = (\mathbf{D}^{\mathrm{T}})^{-1}$.
5: Return $\mathbf{v} \leftarrow \mathrm{Klein}(\mathbf{R}, \mathbf{t})$.
```

Proof. Given an SIVP_γ oracle and any constant $c > 0$, we only need to show that Algorithm 1 will output a lattice vector $\mathbf{v} \in L$ such that $\|\mathbf{v} - \mathbf{t}\| = \operatorname{dist}(\mathbf{t}, L)$ in $\mathrm{poly}(n)$ time. In fact, in step 2, for any $1 \le i \le n$, $\|\tilde{\mathbf{s}}_i\| \le \|\mathbf{s}_i\| \le \gamma\lambda_n(L^\star)$. In step 3, by Lemma 2.6, the n linearly independent vectors $\mathbf{s}_1, \cdots, \mathbf{s}_n$ can be converted into a basis of dual lattice $L^\star$: $\mathbf{d}_1, \cdots, \mathbf{d}_n$ satisfying

$$\|\mathbf{d}_i\| \le \max\left\{\frac{\sqrt{i}}{2}\|\mathbf{s}_i\|, \|\mathbf{s}_i\|\right\},\ \|\tilde{\mathbf{d}}_i\| \le \|\tilde{\mathbf{s}}_i\|,$$

where $1 \le i \le n$, $\tilde{\mathbf{d}}_1, \cdots, \tilde{\mathbf{d}}_n$ and $\tilde{\mathbf{s}}_1, \cdots, \tilde{\mathbf{s}}_n$ are Gram-Schmidt orthogonalization vectors of $\mathbf{d}_1, \cdots, \mathbf{d}_n$ and $\mathbf{s}_1, \cdots, \mathbf{s}_n$, respectively.

Assume that $\tilde{\mathbf{r}}_n, \tilde{\mathbf{r}}_{n-1}, \cdots, \tilde{\mathbf{r}}_1$ are the Gram-Schmidt orthogonalization of $\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_n$ in reverse order. Then, by Lemma 2.7 and Lemma 2.8, for all $1 \le i \le n$, $\tilde{\mathbf{r}}_i = \frac{\tilde{\mathbf{d}}_i}{\|\tilde{\mathbf{d}}_i\|^2}$ and

$$\|\tilde{\mathbf{r}}_i\| = \frac{1}{\|\tilde{\mathbf{d}}_i\|} \ge \frac{1}{\|\tilde{\mathbf{s}}_i\|} \ge \frac{1}{\gamma\lambda_n(L^\star)} \ge \frac{\lambda_1(L)}{\gamma n}.$$

Combining with Lemma 2.9, we can find the closest lattice vector to $\mathbf{t}$ in time $n^{D^2/\min_i \|\tilde{\mathbf{r}_i}\|^2} = O(n^{c^2})$. □

Furthermore, we can improved the above algorithm in a deterministic way.

Theorem 3.5. *For any $\gamma \ge 1$ and any constant $c > 0$, there exists a deterministic polynomial time reduction from* $\mathrm{BDD}_{c/\gamma n}$ *to* SIVP_γ.

Proof. We give our algorithm in two steps. Firstly, we show how to reduce $\mathrm{BDD}_{1/(2\gamma n)}$ to SIVP_γ, which, in fact, is a derandomization of Lemma 3.3. Secondly, for arbitrary but finite constant $c > \frac{1}{2}$, we give a self-reduction from $\mathrm{BDD}_{c/\gamma n}$ to $\mathrm{BDD}_{\sqrt{c^2-1/4}/\gamma n}$ with a SIVP_γ oracle.

Step 1: Reducing $\mathrm{BDD}_{1/(2\gamma n)}$ to SIVP_γ.

Our reduction is shown in Algorithm 2. Clearly, using Gaussian elimination, Algorithm 2 will output a lattice vector efficiently. We only need to prove the correctness of Algorithm 2. Let $(L(\mathbf{B}), \mathbf{t})$ be an instance of $\mathrm{BDD}_{1/(2\gamma n)}$ with $\operatorname{dist}(\mathbf{t}, L) < \lambda_1(L)/(2\gamma n)$. Let $\mathbf{v}$ be a lattice vector in L such that $\|\mathbf{t} - \mathbf{v}\| = \operatorname{dist}(\mathbf{t}, L)$. For $1 \le i \le n$, since $\|\mathbf{s}_i\| \le \gamma\lambda_n(L^\star)$ and $\langle \mathbf{v}, \mathbf{s}_i\rangle \in \mathbb{Z}$. Then, by Lemma 2.8,

$$\begin{aligned} &|\langle \mathbf{v}, \mathbf{s}_i\rangle - \langle \mathbf{t}, \mathbf{s}_i\rangle| = |\langle \mathbf{v} - \mathbf{t}, \mathbf{s}_i\rangle| \\ \le\ & \|\mathbf{v} - \mathbf{t}\| \times \|\mathbf{s}_i\| < \frac{\lambda_1(L)}{2\gamma n} \times \gamma\lambda_n(L^\star) \le \frac{1}{2}. \end{aligned}$$

It implies that $\langle \mathbf{v}, \mathbf{s}_i\rangle \in (\langle \mathbf{t}, \mathbf{s}_i\rangle - 1/2, \langle \mathbf{t}, \mathbf{s}_i\rangle + 1/2)$. Since there exists at most one integer in this interval, the lattice vector $\mathbf{v}$ satisfying the system of linear equations $\langle \mathbf{v}, \mathbf{s}_i\rangle = \lceil \langle \mathbf{t}, \mathbf{s}_i\rangle \rfloor$, $1 \le i \le n$.

Algorithm 2 $\texttt{BDD}_{1/(2\gamma n)}(\mathbf{B},\mathbf{t})$

```
Input: A lattice basis $\mathbf{B} \in \mathbb{Z}^{n\times n}$, a target vector $\mathbf{t}$ such that $\mathrm{dist}(\mathbf{t},L) < \frac{1}{2\gamma n}\lambda_1(L)$ and a SIVP$_\gamma$ oracle, where
    $1 < \gamma \leq \mathrm{poly}(n)$.
Output: A lattice vector $\mathbf{v} \in L$ such that $\mathrm{dist}(\mathbf{t},L) = \|\mathbf{v}-\mathbf{t}\|$.
Invoking the SIVP$_\gamma$ oracle on the lattice $L^\star$, output $\boldsymbol{S} = \{\mathbf{s}_1,\cdots,\mathbf{s}_n\} \leftarrow \mathrm{SIVP}_\gamma(L^\star)$.
Solve the linear equations $\langle \mathbf{v},\mathbf{s}_i\rangle = \lceil\langle \mathbf{t},\mathbf{s}_i\rangle\rfloor$ for $1 \leq i \leq n$ and output $\mathbf{v}$.
```

Step 2: Solving $\mathrm{BDD}_{c/(2\gamma n)}$ instances using $\mathrm{BDD}_{\sqrt{c^2-1/4}/\gamma n}$ and SIVP$_\gamma$ oracles.

The algorithm is described in Algorithm 3.

Firstly, we shall prove the correctness of Algorithm 3. Let $(L(\mathbf{B}),\mathbf{t})$ be an instance of $\mathrm{BDD}_{c/(2\gamma n)}$ with $\mathrm{dist}(\mathbf{t},L) < c\lambda_1(L)/(2\gamma n)$. Let $\mathbf{v}$ be a lattice vector in L such that $\|\mathbf{t}-\mathbf{v}\| = \mathrm{dist}(\mathbf{t},L)$. Invoke the SIVP$_\gamma$ oracle on the dual lattice $L^\star$ and return a set of n independent lattice vectors $\{\mathbf{s}_1,\ldots,\mathbf{s}_n\} \subset L^\star$ such that $\|\mathbf{s}_i\| \leq \gamma\lambda_n(L^\star)$ and $\langle\mathbf{v},\mathbf{s}_i\rangle \in \mathbb{Z}$ for $1 \leq i \leq n$. Then, for any $1 \leq i \leq n$,

$$\begin{aligned} &|\langle \mathbf{v},\mathbf{s}_i\rangle - \langle \mathbf{t},\mathbf{s}_i\rangle| = |\langle \mathbf{v}-\mathbf{t},\mathbf{s}_i\rangle| \\ \leq\ & \|\mathbf{v}-\mathbf{t}\| \times \|\mathbf{s}_i\| < \frac{c\lambda_1(L)}{\gamma n} \times \gamma\lambda_n(L^\star) \leq c. \end{aligned}$$

It implies that $\langle\mathbf{v},\mathbf{s}_i\rangle \in (\langle\mathbf{t},\mathbf{s}_i\rangle - c, \langle\mathbf{t},\mathbf{s}_i\rangle + c)$. Since there are at most $\lceil 2c\rceil$ integers in this interval, the integer $\langle\mathbf{v},\mathbf{s}_i\rangle$ could be one of these adjacent integers. Each vector $\mathbf{s}_i \in L^\star$ $(1 \leq i \leq n)$ can partition L into subsets $L \cap H_{i,j}$ $(j \in \mathbb{Z})$ where $H_{i,j}$ denotes an $(n-1)$-dimensional hyperplane $H_{i,j} = \{\mathbf{x} \in \mathbb{R}^n : \langle\mathbf{x},\mathbf{s}_i\rangle = j\}$. Clearly, the distance between any two adjacent hyperplanes $H_{i,j}$ and $H_{i,j+1}$ is $1/\|\mathbf{s}_i\|$. The above analysis shows that the closest vector $\mathbf{v}$ must be located on one of the $\lceil 2c\rceil$ adjacent hyperplanes of $\mathbf{t}$ for each partition induced by $\mathbf{s}_i$. We discuss in the following cases:

Case 1. Suppose that $\mathbf{v}$ is located on all $H_{i,\lceil\langle\mathbf{t},\mathbf{s}_i\rangle\rfloor}$ for $1 \leq i \leq n$. Solving the linear equations $\langle\mathbf{v},\mathbf{s}_i\rangle = \lceil\langle\mathbf{t},\mathbf{s}_i\rangle\rfloor$ for $1 \leq i \leq n$ can immediately recover $\mathbf{v}$.

Case 2. Suppose that $\mathbf{v}$ lies on $H_{i,j}$ for some $1 \leq i \leq n$ and $j \neq \lceil\langle\mathbf{t},\mathbf{s}_i\rangle\rfloor$. Then, by Lemma 2.8, we obtain the following two results

$$\|\mathbf{t}-\mathbf{t}'_{i,j}\| \ \geq\ \frac{1}{2\|\mathbf{s}_i\|} \geq \frac{1}{2\gamma\lambda_n(L^\star)} \geq \frac{\lambda_1(L)}{2\gamma n}.$$

$$\begin{aligned} &\mathrm{dist}(\mathbf{t}_{i,j},L_{i,j}) = \mathrm{dist}(\mathbf{t}'_{i,j},L\cap H_{i,j}) \\ &= (\mathrm{dist}^2(\mathbf{t},L) - \|\mathbf{t}-\mathbf{t}'_{i,j}\|^2)^{1/2} \\ &< \left(\frac{c^2\lambda_1^2(L)}{\gamma^2 n^2} - \frac{\lambda_1^2(L)}{4\gamma^2n^2}\right)^{1/2} \\ &\leq \frac{\sqrt{c^2-1/4}}{\gamma n}\lambda_1(L) \leq \frac{\sqrt{c^2-1/4}}{\gamma n}\lambda_1(L_{i,j}). \end{aligned}$$

It's easy to verify that $L_{i,j}$ is an $(n-1)$-dimensional sublattice of L. Therefore, the recovery of $\mathbf{v}$ is converted to a $\mathrm{BDD}_{\sqrt{c^2-1/4}/(\gamma n)}$ instance $(L_{i,j},\mathbf{t}_{i,j})$.

Now, we analyze the efficiency of Algorithm 3. In step 2 of Algorithm 3, the vector $\mathbf{v}_0$ can be found efficiently by Gaussian elimination. Using Euclidean algorithm, we can find $\mathbf{w}_{i,j}$ efficiently in step 5 of Algorithm 3, and, in step 7, Micciancio [13] presents an efficient and deterministic algorithm to find a basis of $L_{i,j}$. Therefore, invoking $\mathrm{BDD}_{\sqrt{c^2-1/4}/(\gamma n)}$ oracle at most $2cn$ times, we can find a closest vector $\mathbf{v} \in L$ to $\mathbf{t}$ in deterministic polynomial time in n.

For arbitrary but finite constant $c > 0$, given an SIVP$_\gamma$ oracle, the $\mathrm{BDD}_{c/\gamma n}$ can be solved by invoking $O(2cn)$ times $\mathrm{BDD}_{\sqrt{c^2-1/4}/\gamma n}$ oracle. Recursively, the $\mathrm{BDD}_{c/\gamma n}$ problem can be reduced to $\mathrm{BDD}_{\sqrt{c^2-m/4}/\gamma n}$ after $(2cn)^m$ recursions.

Let $\sqrt{c^2 - m/4}/\gamma n \leq 1/(2\gamma n)$, we have $m \geq 4c^2 - 1$. This implies that, combining Algorithm 2 and Algorithm 3, invoking SIVP$_\gamma$ oracle at most $(2cn)^{4c^2-1}$ times, we can solve a BDD$_{c/\gamma n}$ instance in deterministic polynomial time.

Algorithm 3 $\texttt{BDD}_{c/(\gamma n)}(\mathbf{B}, \mathbf{t})$

```
Input: A lattice basis $\mathbf{B} \in \mathbb{Z}^{n\times n}$ and some constant $c > \frac{1}{2}$, a target vector $\mathbf{t}$ such that $\mathrm{dist}(\mathbf{t}, L) < \frac{c}{\gamma n}\lambda_1(L)$, BDD$_{\sqrt{c^2-1/4}/\gamma n}$ and SIVP$_\gamma$ oracles, where $1 < \gamma \leq \mathrm{poly}(n)$.
Output: A lattice vector $\mathbf{v} \in L$ such that $\mathrm{dist}(\mathbf{t}, L) = \|\mathbf{v} - \mathbf{t}\|$.
Invoking SIVP$_\gamma$ oracle on the lattice $L^\star$, output $\boldsymbol{S} = \{\mathbf{s}_1, \cdots, \mathbf{s}_n\} \leftarrow \mathrm{SIVP}_\gamma(L^\star)$.
Solve the linear equations $\langle \mathbf{v}_0, \mathbf{s}_i\rangle = \lceil\langle \mathbf{t}, \mathbf{s}_i\rangle\rfloor$ for $1 \leq i \leq n$ and output $\mathbf{v}_0$.
for $i = 1, \cdots, n$ do
    for $j = \lceil\langle \mathbf{t}, \mathbf{s}_i\rangle - c\rceil, \cdots, \lfloor\langle \mathbf{t}, \mathbf{s}_i\rangle + c\rfloor$ do
        Compute an vector $\mathbf{w}_{i,j} \in L \cap H_{i,j}$.
        Compute the projection of $\mathbf{t}$ on $H_{i,j}$: $\mathbf{t}'_{i,j}$.
        $L_{i,j} \leftarrow L \cap H_{i,j} - \mathbf{w}_{i,j}$, $\mathbf{t}_{i,j} \leftarrow \mathbf{t}'_{i,j} - \mathbf{w}_{i,j}$.
        $\mathbf{v}'_{i,j} \leftarrow \mathrm{BDD}_{\sqrt{c^2-1/4}/\gamma n}(L_{i,j}, \mathbf{t}_{i,j})$
        $\mathbf{v}_{i,j} \leftarrow \mathbf{v}'_{i,j} + \mathbf{w}_{i,j}$.
    end for
end for
Output the closest point to $\mathbf{t}$ among all the points $\mathbf{v}_{i,j}$ and $\mathbf{v}_0$.
```

□

4 Approximate a closer lattice vector when it's far from the lattice

First, we review some previous known results about the distance between a uniformly random chosen target and a lattice,

Lemma 4.1 ([20]). *Given an n-dimensional lattice $L(\mathbf{B})$ and a vector $\mathbf{t}$ chosen uniformly from $\mathcal{P}(L)$, then*

$$\Pr_{\mathbf{t}}\left(\mathrm{dist}(\mathbf{t}, L(\mathbf{B})) \geq \frac{\rho(L)}{2}\right) \geq \frac{1}{2},$$

where $\rho(L)$ denotes the covering radius of L.

Lemma 4.2 ([15]). *For any n-dimensional lattice $L(\mathbf{B})$,*

$$\frac{\lambda_n(L)}{2} \leq \rho(L) \leq \frac{\sqrt{n}}{2}\lambda_n(L)$$

By Lemma 4.1 and Lemma 4.2, we have that for any uniformly chosen target vector $\mathbf{t}$,

$$\begin{aligned}&\Pr_{\mathbf{t}}\left(\mathrm{dist}(\mathbf{t}, L(\mathbf{B})) \geq \frac{\lambda_n(L)}{4}\right)\\ \geq\ &\Pr_{\mathbf{t}}\left(\mathrm{dist}(\mathbf{t}, L(\mathbf{B})) \geq \frac{\rho(L)}{2}\right) \geq \frac{1}{2}.\end{aligned}$$

Given a lattice $L = L(\mathbf{B})$ and a target vector $\mathbf{t} \in \mathbb{R}^n$, If we have n linearly independent vectors $\mathbf{s}_1, \cdots, \mathbf{s}_n$ satisfying that for any $1 \leq i \leq n$, $\|\mathbf{s}_i\| \leq \gamma\lambda_n(L)$ in hand. Then compute their Gram-Schmidt orthogonalization vectors $\tilde{\mathbf{s}}_1, \cdots, \tilde{\mathbf{s}}_n$, using Babai's nearest plane algorithm [14], we can find a vector $\mathbf{v} \in L$ such that

$$\begin{aligned}\mathrm{dist}(\mathbf{v}, \mathbf{t}) &\leq \sqrt{\sum_{i=1}^{n}\left(\frac{\|\tilde{\mathbf{s}}_i\|}{2}\right)^2} \leq \frac{1}{2}\sqrt{\sum_{i=1}^{n}\|\mathbf{s}_i\|^2}\\ &\leq \frac{1}{2}\sqrt{n}\max_i\|\mathbf{s}_i\| \leq \frac{1}{2}\gamma\sqrt{n}\lambda_n(L).\end{aligned}$$

If $\operatorname{dist}(\mathbf{t}, L(\mathbf{B})) \geq \frac{\lambda_n(L)}{4}$, then using SIVP_γ oracle we can find a vector $\mathbf{v} \in L$ such that

$$\operatorname{dist}(\mathbf{v}, \mathbf{t}) \leq \frac{1}{2}\gamma\sqrt{n}\lambda_n(L) \leq 2\gamma\sqrt{n}\operatorname{dist}(\mathbf{t}, L).$$

In summary, the above analysis contains the following result.

Corollary 4.3. *Given an n-dimensional lattice $L = L(\mathbf{B})$ and a target vector $\mathbf{t} \in \mathbb{R}^n$, if $\operatorname{dist}(\mathbf{t}, L) \geq \lambda_n(L)/4$, then $\mathrm{CVP}_{2\gamma\sqrt{n}}$ can be reduced to SIVP_γ in deterministic polynomial time. Specially, for uniformly chosen target vector, the reduction algorithm is correct with probability not less than $1/2$.*

Furthermore, if $1 < \gamma < 2$, using lattice-embedding technique we can get a better result.

Theorem 4.4. *Given an n-dimensional lattice $L = L(\mathbf{B})$ and a target vector $\mathbf{t} \in \mathbb{R}^n$, for any real $k > \frac{\sqrt{3}}{3}$, $1 < \gamma < \frac{2k}{\sqrt{1+k^2}}$, if $\operatorname{dist}(L, \mathbf{t}) = \min_{\mathbf{v}\in L} \|\mathbf{v} - \mathbf{t}\| > \frac{\gamma}{2k}\lambda_n(L)$, then there exists a Cook reduction from $\mathrm{CVP}_{\sqrt{3}k(1+1/n)}$ to SIVP_γ.*

Proof. Let $\mu = \operatorname{dist}(\mathbf{t}, L)$, using Babai's nearest plane algorithm, we can get a real d satisfying $\mu \leq d < 2^n\mu$, namely, $\mu \in (d/2^n, d]$. Divide the interval $(d/2^n, d]$ into $\operatorname{poly}(n)$ small intervals $\left(\frac{d}{2^n}\left(1+\frac{1}{n}\right)^i, \frac{d}{2^n}\left(1+\frac{1}{n}\right)^{i+1}\right]$. For each $i_0 = 0, \cdots, \lceil n \log_{(1+1/n)} 2 \rceil$, guess

$$\mu \in \left(\frac{d}{2^n}\left(1+\frac{1}{n}\right)^{i_0}, \frac{d}{2^n}\left(1+\frac{1}{n}\right)^{i_0+1}\right].$$

Let $\mu_0 = \frac{d}{2^n}\left(1+\frac{1}{n}\right)^{i_0+1}$, then $\mu \leq \mu_0 < \mu\left(1+\frac{1}{n}\right)$.

Let

$$\begin{aligned}\tilde{\mathbf{B}} &= \begin{pmatrix} \mathbf{B} & \mathbf{t} \\ \mathbf{0} & k\mu_0 \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{b}_1 & \cdots & \mathbf{b}_n & \mathbf{t} \\ 0 & \cdots & 0 & k\mu_0 \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{d}_1 & \cdots & \mathbf{d}_n & \mathbf{d}_{n+1} \end{pmatrix}.\end{aligned}$$

The reduction algorithm goes as Algorithm 4.

Algorithm 4 `Lattice-embedding`$(\mathbf{B}, \mathbf{t})$

```
Input: A lattice basis $\mathbf{B} \in \mathbb{Z}^{n\times n}$ parameters $k > \frac{\sqrt{3}}{3}, \mu_0, 1 < \gamma < \frac{2k}{\sqrt{1+k^2}}$ and a $\mathrm{SIVP}_\gamma$ oracle
Output: A lattice vector $\mathbf{v} \in L$ such that $\operatorname{dist}(\mathbf{t}, L) = \|\mathbf{v} - \mathbf{t}\|$.
Construct a new lattice $\tilde{L} = L(\tilde{\mathbf{B}})$.
Invoking $\mathrm{SIVP}_\gamma$ oracle on $\tilde{L}$, $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_{n+1} \leftarrow \mathrm{SIVP}_\gamma(\tilde{L})$.
Express each $\mathbf{v}_i = \sum_{j=1}^{n+1} z_{ij}\mathbf{d}_j$.
Return $\mathbf{v} = \sum_{j=1}^{n} z_{i_0,j}\mathbf{b}_j$ where $(z_{i_0,1}, \cdots, z_{i_0,n}, z_{i_0,n+1})$ satisfies $|z_{i_0,n+1}| \neq 0$.
```

Now we prove the correctness of our algorithm in two cases:

Case 1: $\lambda_{n+1}(\tilde{L}) \leq \sqrt{\mu^2 + (k\mu_0)^2}$. For every vector $\mathbf{v}_i$ can be represented as an integer linear combination of $\mathbf{d}_1, \cdots, \mathbf{d}_{n+1}$, there must be some vector with a non-zero coefficient in $\mathbf{d}_{n+1}$. Without loss of generality, assume that

$$\begin{aligned}\mathbf{v}_{n+1} &= \sum_{i=1}^{n} z_i\mathbf{d}_i + z_{n+1}\mathbf{d}_{n+1} \\ &= \left(\sum_{i=1}^{n} z_i\mathbf{b}_i + z_{n+1}\mathbf{t}, z_{n+1}k\mu_0\right),\ z_{n+1} \neq 0,\end{aligned}$$

Now we will show that $|z_{n+1}| = 1$. In fact, if $|z_{n+1}| \geq 2$, then $\|\mathbf{v}_{n+1}\|^2 \geq 4(k\mu_0)^2$. While, in step 2, we know $\|\mathbf{v}_{n+1}\|^2 = \|\sum_{i=1}^n z_i\mathbf{b}_i + z_{n+1}\mathbf{t}\|^2 + z_{n+1}^2(k\mu_0)^2 \leq \gamma^2\lambda_{n+1}^2(\tilde{L}) \leq \gamma^2\left(\mu^2 + (k\mu_0)^2\right)$, which implies that

$$\begin{aligned} & 4(k\mu_0)^2 \leq \gamma^2\left(\mu_0^2 + (k\mu_0)^2\right) \\ \Rightarrow \quad & 4k^2 \leq \gamma^2(1+k^2) \Rightarrow \gamma \geq \frac{2k}{\sqrt{1+k^2}}. \end{aligned}$$

This contradicts with the condition in our theorem. Therefore $|z_{n+1}| = 1$. Let $\mathbf{v} = \sum_{i=1}^n z_i\mathbf{b}_i$. Then

$$\begin{aligned} & \|\mathbf{v}+\mathbf{t}\|^2 = \mathbf{v}_{n+1}^2 - z_{n+1}^2(k\mu_0)^2 \\ \leq \quad & \gamma^2\mu^2 + (\gamma^2-1)(k\mu_0)^2 \\ \leq \quad & \gamma^2(1+k^2)\mu_0^2 - k^2\mu_0^2 \leq 3k^2\mu_0^2. \\ \Rightarrow \quad & \|\mathbf{v}+\mathbf{t}\| \leq \sqrt{3}k\mu_0 \leq \sqrt{3}k\left(1+\frac{1}{n}\right)\mu. \end{aligned}$$

Case 2: $\lambda_{n+1}(\tilde{L}) > \sqrt{\mu^2 + (k\mu_0)^2}$. In this case, by the definition of $\lambda_{n+1}(\tilde{L})$ and $\lambda_n(L)$, we have $\sqrt{\mu^2 + (k\mu_0)^2} < \lambda_{n+1}(\tilde{L}) \leq \lambda_n(L)$.

Similarly, we also show that $|z_{n+1}| = 1$. In fact, if $|z_{n+1}| \geq 2$, then $\|\mathbf{v}_{n+1}\|^2 \geq 4(k\mu_0)^2$. While, in step 2, we know

$$\begin{aligned} \|\mathbf{v}_{n+1}\|^2 &= \left\|\sum_{i=1}^n z_i\mathbf{b}_i + z_{n+1}\mathbf{t}\right\|^2 + z_{n+1}^2(k\mu_0)^2 \\ &\leq \gamma^2\lambda_{n+1}^2(\tilde{L}) \leq \gamma^2\lambda_n(L)^2. \end{aligned}$$

Hence, $4(k\mu_0)^2 \leq \gamma^2\lambda_n(L)^2 \Rightarrow \mu_0 \leq \frac{\gamma}{2k}\lambda_n(L) \Rightarrow \mu \leq \frac{\gamma}{2k}\lambda_n(L)$. This contradicts with the condition in our theorem. Therefore $|z_{n+1}| = 1$. Let $\mathbf{v} = \sum_{i=1}^n z_i\mathbf{b}_i$. Then

$$\begin{aligned} & \|\mathbf{v}+\mathbf{t}\|^2 = \mathbf{v}_{n+1}^2 - z_{n+1}^2(k\mu_0)^2 \\ \leq \quad & \gamma^2\lambda_n^2(L) - (k\mu_0)^2 < 4k^2\mu^2 - k^2\mu_0^2 \\ \leq \quad & 4k^2\mu^2 - k^2\mu^2 = 3k^2\mu^2. \\ \Rightarrow \quad & \|\mathbf{v}+\mathbf{t}\| < \sqrt{3}k\mu. \end{aligned}$$

Combining the above two cases, we complete the proof of Theorem 4.4.

□

Remark 4.5. *In fact, let $\frac{\gamma}{2k} = \frac{1}{4}$ in Theorem 4.4, we immediately obtain that, if* $\mathrm{dist}(\mathbf{t}, L) > \frac{1}{4}\lambda_n(L)$, $\mathrm{CVP}_{2\sqrt{3}\gamma(1+1/n)}$ *can be reduced to* SIVP_γ *for* $1 < \gamma < \frac{\sqrt{15}}{2}$. *The reduction factor for CVP is much better than that in Corollary 4.3 for* $n \geq 5$. *If we fix the reduction factor, let* $2\gamma\sqrt{n} = \sqrt{3}k(1+1/n)$, *then* $\frac{\gamma}{2k} = \frac{\sqrt{3}(1+1/n)}{4\sqrt{n}} < \frac{1}{4}$ *in the conditions that* $n \geq 5$ *and* $1 < \gamma < \sqrt{4 - 3(1+\frac{1}{n})^2/4n}$. *This implies that, for* $n \geq 5$ *and* $1 < \gamma < \sqrt{4 - 3(1+\frac{1}{n})^2/4n}$, *the reduction in Theorem 4 is valid for much more target vectors than that in Corollary 4.3.*

5 Conclusion

Motivated by the open problem presented by Micciancio in SODA 2008, this paper studies the relationships between CVP and SIVP. Given a lattice and some target vector, intuitively, the hardness is different when the distance between the target vector and the lattice varies. Along this way, we gives some preliminary results about the relations between SIVP and some special CVP instances, which may be helpful for the full and final solution of the open problem. Solving this problem has a great impact on the computational complexity theory and security of lattice-based cryptosystems, which is the direction of our future work.

References

[1] O. Goldreich, D. Micciancio, S. Safra, and J.-P. Seifert. Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. *Information Processing Letters*, 71(2):55 – 61, 1999.

[2] I. Dinur, G. Kindler, R. Raz, and S. Safra. Approximating cvp to within almost-polynomial factors is np-hard. *Combinatorica*, 23(2):205–243, April 2003.

[3] Ishay Haviv and Oded Regev. Tensor-based hardness of the shortest vector problem to within almost polynomial factors. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, STOC '07, pages 469–477, New York, NY, USA, 2007. ACM.

[4] Ravi Kannan. Minkowski's convex body theorem and integer programming. *Math. Oper. Res.*, 12(3):415–440, August 1987.

[5] Miklos Ajtai, Ravi Kumar, and D. Sivakumar. Sampling short lattice vectors and the closest lattice vector problem. In *Proceedings of the 17th IEEE Annual Conference on Computational Complexity*, CCC '02, pages 53–57, Washington, DC, USA, 2002. IEEE Computer Society.

[6] R Kannan. Algorithmic geometry of numbers. *Annual Review of Computer Science*, 2(1):231–267, 1987.

[7] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(1):625–635, 1993.

[8] Vadim Lyubashevsky and Daniele Micciancio. On bounded distance decoding, unique shortest vectors, and the minimum distance problem. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009*, volume 5677 of *Lecture Notes in Computer Science*, pages 577–594. Springer Berlin Heidelberg, 2009.

[9] Chandan Dubey and Thomas Holenstein. Approximating the closest vector problem using an approximate shortest vector oracle. In LeslieAnn Goldberg, Klaus Jansen, R. Ravi, and José D.P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 6845 of *Lecture Notes in Computer Science*, pages 184–193. Springer Berlin Heidelberg, 2011.

[10] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, STOC '96, pages 99–108, New York, NY, USA, 1996. ACM.

[11] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, STOC '05, pages 84–93, New York, NY, USA, 2005. ACM.

[12] Johannes Blömer and Jean-Pierre Seifert. On the complexity of computing short linearly independent vectors and short bases in a lattice. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, STOC '99, pages 711–720, New York, NY, USA, 1999. ACM.

[13] Daniele Micciancio. Efficient reductions among lattice problems. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '08, pages 84–93, Philadelphia, PA, USA, 2008. Society for Industrial and Applied Mathematics.

[14] L. Babai. On lovász lattice reduction and the nearest lattice point problem. *Combinatorica*, 6(1):1–13, 1986.

[15] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers, 2002.

[16] Oded Regev. Dual lattices. Lattices in Computer Science, 2009. Lecture notes of a course given at the Tel Aviv University. Availble at the URL http://www.cims.nyu.edu/regev/teaching/lattices_fall_2009/.

[17] Philip Klein. Finding the closest lattice vector when it's unusually close. In *Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms*, SODA '00, pages 937–941, Philadelphia, PA, USA, 2000. Society for Industrial and Applied Mathematics.

[18] Jin-Yi Cai. A new transference theorem in the geometry of numbers and new bounds for ajtai's connection factor. *Discrete Applied Mathematics*, 126(1):9 – 31, 2003. 5th Annual International Computing and Combinatorics Conference.

[19] Daniele Micciancio. The geometry of lattice cryptography. In Alessandro Aldini and Roberto Gorrieri, editors, *Foundations of Security Analysis and Design VI*, volume 6858 of *Lecture Notes in Computer Science*, pages 185–210. Springer Berlin Heidelberg, 2011.

[20] Venkatesan Guruswami, Daniele Micciancio, and Oded Regev. The complexity of the covering radius problem. *Computational Complexity*, 14(2):90–121, 2005. Preliminary version in CCC 2004.