Electronic Colloquium on Computational Complexity, Report No. 52 (2006)

Inapproximability Results for the Closest Vector Problem with Preprocessing over ℓ_∞ Norm

Wenbin Chen* Jiangtao Meng †

Abstract

We show that the Closest Vector Problem with Preprocessing over ℓ_∞ norm (CVPP$_\infty$) is NP-hard to approximate to within a factor of $(\log n)^{1/2-\epsilon}$, unless **NP**$\subseteq$ **DTIME** $(2^{polylog(n)})$. The result is the same as that in [19] by Regev and Rosen, but our proof methods are different from theirs. Their reductions are based on norm embeddings. However, our reductions are based on the reduction of [2] and the property of Hadamard matrix.

Keywords: Closest Vector Problem, Computational Complexity, NP-hardness, Label Cover.

1 Introduction

Let $\mathbf{B} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ be a set of linearly independent vectors in $\mathbf{R}^m$. The n-dimensional lattice L generated by B is the set of vectors $\{\sum_{i=1}^n a_i\mathbf{v}_i | a_i \in \mathbf{Z}\}$ where B is called the basis for the lattice L. The lattice L is also an additive group. The same lattice could be generated by many different bases. Given a basis for an n-dimensional lattice L and an arbitrary vector $\mathbf{t}$, the Closest Vector Problem (CVP) is to find a vector in L closest to $\mathbf{t}$ in a certain norm. The Shortest Vector Problem (SVP) is a homogeneous analog of CVP, and is defined to be the problem of finding the shortest non-zero vector in L. These lattice problems have a long history and we present some of the results below. The more comprehensive list of references can be found in [12] and [16].

These lattice problems have been studied since they were introduced in the 19th century. Gauss gave an algorithm that works for 2-dimensional lattices ([10], 1801). In 1842, Dirichlet formulated

[*] Department of Computer Science, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P.R. China. Email: cwbiscas@yahoo.com

[†] Department of Computer Science, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P.R. China. Email: globangrilion@yahoo.com

ISSN 1433-8092

the general problem for arbitrary dimensions. The existence of short non-zero vectors in lattices was dealt with by Minkowski in a field called Geometry of Numbers [17]. Lattices have many applications in various fields of mathematics, such as convex analysis, number theory and computer science. The famous LLL algorithm [13] can be used to construct efficient algorithms for many other problems, such as breaking knapsack cryptosystem, factoring polynomials over rationals.

The first intractability results for lattice problems date back to 1981. In [20], van Emde Boas proved that CVP in any ℓ_p norm and SVP in ℓ_∞ norm are NP-hard. The interest in lattice problems has been renewed due to Ajtai's discovery [1] of worst-case to average-case reduction and subsequent construction of a lattice-based public key cryptosystem by Ajtai and Dwork [3]. Their work implies that if it is hard to approximate SVP within some polynomial factor n^c, then a secure cryptosystem can be constructed.

CVP has received much attention in the recent years. CVP was proved to be NP-hard by van Emde Boas [20]. Arora et al. [4] used the PCP characterization of NP to show that approximating CVP within factor $2^{(\log n)^{1-\epsilon}}$ is NP-hard unless $\mathbf{NP} \subseteq \mathbf{DTIME}$ $(2^{polylog(n)})$ and approximating CVP within any constant factor is NP-hard unless $P = NP$. Assuming $P \neq NP$, Dinur et al. [7] proved that it is hard to approximate CVP within factor $n^{c/\log\log n}$ for some constant $c > 0$. All above results for CVP work in all ℓ_p norms. It was showed by Dinur et al. that both CVP in ℓ_∞ norm and SVP in ℓ_∞ norm are hard to approximate within factor $n^{c/\log\log n}$ for some constant $c > 0$ [6].

In this paper we investigate the inapproximability of the Closest Vector Problem with Preprocessing over ℓ_∞ norm (CVPP$_\infty$). This is a variant of the Closest Vector Problem over ℓ_∞ norm (CVP$_\infty$) in which the basis $\mathbf{B}$ of the lattice depends only on the input length, and hence can be assumed to be specified in advance. This means that the basis can be preprocessed arbitrarily and the computed information can be used to solve CVP$_\infty$ on the input $(\mathbf{B}, \mathbf{t})$.

Bruck and Naor [5] showed that the Nearest Codeword with Preprocessing (NCPP) is NP-hard, which is the analog problem of CVPP in coding theory. In this problem, a binary code $\mathbf{C}$ is specified in advance and the goal is find the closest codeword in $\mathbf{C}$ to a given binary vector $\mathbf{v}$ in the Hamming metric. Later Micciancio proved that CVPP over any ℓ_p $(p \geq 1)$ norm is NP-hard [17]. The first inapproximability result was obtained by Feige and Micciancio [8], who proved that it is NP-hard to approximate CVPP over ℓ_p norm within a $(5/3)^{1/p} - \epsilon$ for any $\epsilon > 0$. In the same paper they also proved that approximating NCPP within a $5/3 - \epsilon$ factor is NP-hard. In [18], Regev improved these inapproxiability factor to $3 - \epsilon$ and $3^{1/p} - \epsilon$ respectively for any $\epsilon > 0$. Recently, Alekhnovich et al.[2] proved that it is NP-hard to approximate CVPP over ℓ_p norm within any constant factor and within a factor of $(\log n)^{1/p-\epsilon}$. They also proved that NCPP is NP-hard to approximate within $(\log n)^{1-\epsilon}$ unless $\mathbf{NP} \subseteq \mathbf{DTIME}$ $(2^{polylog(n)})$.

More recently, Regev and Rosen [19] prove that it is NP-hard to approximate CVPP$_\infty$ within $(\log n)^{1/2-\epsilon}$ for any $\epsilon > 0$. They give a deterministic Karp reduction from CVPP$_2$ to CVPP$_\infty$ and a deterministic Cook reduction from from CVPP$_2$ to CVPP$_\infty$. Their reduction are based on *embeddings of normed spaces.*

Our Result

In this paper, we also show that the Closest Vector Problem with Preprocessing over ℓ_∞ norm (CVPP$_\infty$) is NP-hard to approximate within a factor of $(\log n)^{1/2-\epsilon}$, unless $\mathbf{NP} \subseteq \mathbf{DTIME}$ $(2^{polylog(n)})$. But our proof methods are different from Regev and Rosen's methods which are based on norm embeddings.

Technique

In order to obtain our result, we give a reduction from the Label Cover Problem with Preprocessing (LCPP) to CVPP$_\infty$. LCPP is introduced and it is proved to be NP-hard in [2]. In [2], a polynomial time reduction from LCPP to MISPP over ℓ_2 norm is given. Combine the reduction method with the property of Hadamard matrix, we give a polynomial time reduction from LCPP to MISPP over ℓ_∞ norm. Since there exist a polynomial time reduction from MISPP over ℓ_∞ norm to CVPP$_\infty$, we obtain the hardness result of CVPP$_\infty$.

Structure of the Paper

In section 2, we introduce some definitions. In section 3 we propose a weaker hardness result of $\sqrt{2}-\epsilon$ for CVPP$_\infty$ by reducing Label Cover Problem to it. In section 4 we describes a reduction from the Label Cover Problem with Preprocessing to CVPP$_\infty$, which gives a stronger hardness result of $(\log n)^{1/2-\epsilon}$. Finally, in section 5 we present some conclusions and some open problems.

2 Preliminaries

In this section we present formal description of the problems which are used in our reductions.

Let $\mathbf{B}=\{\mathbf{v}_1,\ldots,\mathbf{v}_n\}$ be a set of linearly independent vectors in $\mathbf{R}^m$. The n-dimensional lattice L generated by B is the set of vectors $\{\sum_{i=1}^n a_i\mathbf{v}_i | a_i \in \mathbf{Z}\}$ where B is called the basis for the lattice L. The lattice L is also an additive group. The closest vector problem over ℓ_∞ norm is defined as follows.

Definition 1 *The Closest Vector Problem over ℓ_∞ norm (CVP_∞) is the problem in which one is given a lattice basis $\mathbf{B}$ and a target vector $\mathbf{y}$ and must find a lattice vector $\mathbf{Bx}$ ($\mathbf{x}\in\mathbf{Z}^n$) such that $\|\mathbf{Bx}-\mathbf{y}\|_\infty$ is minimum. In the decisional version of CVP_∞ one is also given a real number t, and must decide whether there exist an integer vector $\mathbf{x}$ such that $\|\mathbf{Bx}-\mathbf{y}\|_\infty \le t$.*

The decisional and search version of CVP$_\infty$ can be easily proved equivalent [17]. The CVPP$_\infty$ is defined as follows.

Definition 2 *The Closest Vector Problem over ℓ_∞ norm with preprocessing (CVPP$_\infty$) asks for a function P(the preprocessing function) and an algorithm D(the decoding algorithm) with the following properties:*

- *On input a lattice basis $\mathbf{B}$, P($\mathbf{B}$) returns a new description L of the lattice L(B) whose size is polynomially related to the size of $\mathbf{B}$, i.e. there exists a constant c such that $size(L) < size(\mathbf{B})^c$ for all bases $\mathbf{B}$ and $L = P(\mathbf{B})$.*
- *Given L and a target vector $\mathbf{y}$, D(L, $\mathbf{y}$) computes a lattice point $\mathbf{Bx}$ such that $\|\mathbf{Bx}-\mathbf{y}\|_\infty$ is minimum. In the decision version of CVPP$_\infty$, D is also given a distance t, and D(L, $\mathbf{y}$, $\mathbf{t}$) decides whether there exists a lattice vector $\mathbf{Bx}$ such that $\|\mathbf{Bx}-\mathbf{y}\|_\infty \le t$.*

As for CVPP$_\infty$, the search and decision versions are also equivalent which is showed in [17].

Since no complexity assumption is made on the preprocessing function P, one may think of P as a preprocessing algorithm with unlimited computational resources. Only the running of D is used to measure the complexity of the decoding process, i.e. we say that CVPP$_\infty$ is solvable in polynomial time if there exists a function P and a polynomial time algorithm D such that D(P($\mathbf{B}$),$\mathbf{y}$, t) solves the CVP$_\infty$ instance ($\mathbf{B}$, $\mathbf{y}$, t).

In our reductions we need the following NP-hard problems [2].

Definition 3 (MISP$_\infty$). *The Minimum Integral Solution Problem over ℓ_∞ norm is the following problem. For a non-negative function f, an instance of GapMISP$_f$(.) is denoted by $(\mathbf{B}_f, \mathbf{B}_v, \mathbf{t}, d)$, where $\mathbf{B}_f \in \mathbf{Z}^{k_1 \times n}$, $\mathbf{B}_v \in \mathbf{Z}^{k_2 \times n}, \mathbf{t} \in \mathbf{Z}^{k_1}$ and $d \in \mathbf{Z}^+$. It is a YES instance if there exists a $\mathbf{x} \in \mathbf{Z}^n$ such that $\mathbf{B}_f\mathbf{x} = \mathbf{t}$ and $\|\mathbf{B}_v\mathbf{x}\|_\infty \leq d$, and a NO instance if for all $\mathbf{x} \in \mathbf{Z}^n$ satisfying $\mathbf{B}_f\mathbf{x} = \mathbf{t}$, $\|\mathbf{B}_v\mathbf{x}\|_\infty > f(n) \cdot d$. $\mathbf{B}_f$ will be referred to as the fixed linear forms on the variables, while $\mathbf{B}_v$ will be referred to as the variable linear forms.*

Definition 4 (Label Cover Problem) *For a non-negative function f, an instance of GapLCP$_{f(\cdot)}$ is*

$$\mathcal{U}(G(V, W, E), [R], [S], n, m, \{\pi_e\}_{e \in E}, \{\mathcal{P}_w, P_w\}_{w \in W}),$$

where $G = (V, W, E))$ is a bipartite graph, with $|V| = n$, $|W| = m$, E is the set of edges, [S] is the set of labels for vertices in V, for every $w \in W$, $\mathcal{P}_w \in \{R_{w,l}\}$ is a set in the partition that represents all permissible labels from which a label can be assigned to $w \in W$, and for every $e \in E$, $\pi_e : [R] \mapsto [S]$. A labeling is a pair of maps $L_W : W \mapsto [R]$, $L_V : V \mapsto [S]$. An edge $e = (v, w)$ is satisfied by a labeling (L_V, L_W) if $L_W(w) \in P_w$ and $\pi_e(L_W(w)) = L_V(v)$. $\mathcal{U}$ is a YES instance if there is a labeling that satisfies all its edges. It is a NO instance if no labeling satisfies more than $f(n)$ fraction of the edges.

Typically, $m \gg n$ and $R \gg S$. We note that in the standard definition of this problem, the partition is trivial: $\mathcal{P} = [R]$ for all $w \in W$, and thence, the set of permissible labels $P_w = [R]$ for all $w \in W$.

Definition 5 (MISPP$_\infty$) *Minimum Integral Solution Problem with Pre-processing is the following problem. From the input to $MISP_\infty$, which is a tuple $(\mathbf{B}_f, \mathbf{B}_v, \mathbf{t}, d)$, $(\mathbf{B}_f, \mathbf{B}_v)$ is the uniform input to MISPP$_\infty$ (i.e, depends only on the input length).*

Definition 6 (Label Cover Problem with Pre-processing) *From the input to GapLCP$_{f(\cdot)}$, which is a tuple $\mathcal{U}$:*

$$(G(V, W, E), [R], [S], n, m, \{\pi_e\}_{e \in E}, \{\mathcal{P}_w, P_w\}_{w \in W}).$$

The uniform part consist of $G(V, W, E)$, n, m, the set of candidate labels $[R]$ for vertices in W, $[S]$, and the projection maps π_e. Further, for every $w \in W$, a partition of [R], $\mathcal{P}_w = \cup_l R_{w,l}$ is fixed, which also depends just on the length of the input. The input to GapLCPP$_{f(\cdot)}$ now consists of a set $P_w \in \{R_{w,l}\}_l$, for every $w \in W$. Recall that this is the set of permissible labels for w, and w is supposed to be assigned a label only from P_w.

The following proposition states that the MISPP$_\infty$ can be reduced to CVPP$_\infty$.

Proposition 1 There exist a polynomial time reduction from MISPP$_\infty$ to CVPP$_\infty$.

The proof of Proposition 1 is similar to that of Proposition 2.7 in [2], which can be obtained by a slight modification of Lemma 10 in [7]. We omit it here.

3 Hardness of $\sqrt{2}-\epsilon$

In this section we first present a weaker hardness result. We show that CVPP$_\infty$ is NP-hard to approximate within $\sqrt{2}-\epsilon$. Our proof is by the reduction from the PCP theorem to MISPP$_\infty$. This reduction combine Regev's reduction in [18] and the property of Hadamard matrix. In [18], Regev give a reduction from PCP to GapMLD (Maximum Likelihood Decoding). Their instance ($\mathbf{C}$, $\mathbf{v}$, t) of GapMLD can be viewed an instance ($\mathbf{B}_f$, $\mathbf{I}$, $\mathbf{v}$, t) of MISPP over Hamming distance. We find that if we replace the matrix $\mathbf{I}$ with another matrix, we can show that it is NP-hard to approximate MISPP$_\infty$ within $\sqrt{2}-\epsilon$. The reduction details are as follows. The start point of reduction is the following PCP theorem which is the Lemma 3.1 summarized by Regev in [18].

Lemma 1 *For any fixed $\epsilon > 0$ the following problem is NP-hard. Let R_X, R_Y be the two sets and d_X, d_Y two integers depending only on ϵ. For any size parameter n there exist two sets of variables $Y, \hat{X}$ and a set of tests $\hat{\Phi}$, i.e., functions from R_X to R_Y indexed by variables $x \in \hat{X}$ and $y \in Y$. Each variable $x \in \hat{X}$ has d_X tests in $\hat{\Phi}$ associated with it. For $i \in [d_X]$ we denote by $z_i(x)$ the i'th variable in Y with which x have a test for some ordering of the variables . Then, an instance of the problem of size parameter n is specified with by a subset $X \subseteq \hat{X}$. Let $\Phi \subseteq \hat{\Phi}$ be the set of tests associated with X. Each variable $y \in Y$ has d_Y tests in Φ associated to it. An assignment A is a function from X to R_X and from Y to R_Y. A test $\varphi_{x,y} \in \Phi$ is satisfied by A if $\varphi_{x,y}(A(x)) = A(y)$. The problem is to distinguish between the case where there exists an assignment that satisfies all the tests Φ and the case where no assignment satisfies more than ϵ of the tests.*

From an instance of Lemma 1, we construct an instance $(\mathbf{B}_f, \mathbf{B}_v, \mathbf{t}, d)$ of MISPP$_\infty$. In fact $\mathbf{B}_v$ and $\mathbf{t}$ have been constructed by Regev in [18]. For completeness we repeat the construction of $\mathbf{B}_f$ and $\mathbf{t}$ as follows.

For any given size parameter let d_Y, d_X, R_X, R_Y, $Y, \hat{X}$, $\hat{\Phi}$ as described in lemma 1. Let S denote the set $[d_Y]^{d_X}$. For a given an PCP instance $X \subseteq \hat{X}$ we define a function $s^* : X \to S$ such that for every variable $y \in Y$ and for each $j \in [d_Y]$ there exist a unique pair $(x, i) \in X \times [d_X]$ such that $z_i(x) = y$ and $s_i^*(x) = j$. Intuitively, for each $y \in Y$ we can label the d_Y tests that contain y with $1, \ldots, d_Y$ and then $s^*(x)$ as the d_X labels that the tests that contain x got. More formally, such an s^* exists and can be efficiently computed, say by the following process: for every $y \in Y$ let $\alpha(y)$ be a value initially set to 0. For each value $x \in X$ in an arbitrary order we increment $\alpha(z_i(x))$ by one for each $i \in [d_X]$ and define $s^*(x)$ as $(\alpha(z_i(x)), \ldots, \alpha(z_{d_X}(x)))$. The process ends when $\alpha(y)$ is d_Y for all $y \in Y$.

We describe the set of equations $\mathbf{B}_f\mathbf{r} = \mathbf{t}$ given by $MISPP_\infty$ as follows. Let T denote the set $\hat{X} \times [d_X] \times S \times R_X$. The vector $\mathbf{r}$ is $|T|$-dimensional and we index its coordinates by $r_{(x,i,s,a)}$ for $(x, i, s, a) \in T$.

$$\forall x \in \hat{X}, i_1, i_2 \in [d_X], s \in S, a \in R_X \;\; r_{(x,i_1,s,a)} = r_{(x,i_2,s,a)} \tag{1}$$

$$\forall y \in Y, j_1, j_2 \in [d_Y], b \in R_Y, \qquad \sum_{\substack{(x,i,s,a)\in T| \\ (z_i(x),s_i)=(y,j_1) \\ \varphi_{x,y}(a)=b}} r_{(x,i,s,a)} = \sum_{\substack{(x,i,s,a)\in T| \\ (z_i(x),s_i)=(y,j_2) \\ \varphi_{x,y}(a)=b}} r_{(x,i,s,a)} \tag{2}$$

$$\forall x \in X, i \in [d_X] \quad \sum_{a \in R_X} r_{(x,i,s^*(x),a)} = 1 \tag{3}$$

$$\forall x \in X, i \in [d_X], s \neq s^*(x) \quad \sum_{a \in R_X} r_{(x,i,s,a)} = 0 \tag{4}$$

$$\forall x \in \hat{X} \backslash X, i \in [d_X], s \in S \quad \sum_{a \in R_X} r_{(x,i,s,a)} = 0 \tag{5}$$

The variable linear forms are as follows:

$$H\mathbf{r}_{(x,i,s)} \qquad \text{for every } (x,i,s)$$

Where $\mathbf{r}_{(x,i,s)} = (r_{(x,i,s,a_1)}, \dots, r_{(x,i,s,a_{|R_X|})})$ and H is a $|R_X| \times |R_X|$ Hadamard matrix. Thus we get $\mathbf{B}_v = \begin{pmatrix} H & \cdots & 0 \\ \vdots & H & \vdots \\ 0 & \cdots & H \end{pmatrix}$, where H is a $|R_X| \times |R_X|$ Hadamard matrix and for every (x,i,s) we have a Hadamard matrix.

It is easy to know that the set of equations has the property that the matrix $\mathbf{B}_f$ are independent of the instance X; Only the target vector depends on X.

Lemma 2 (Completeness) *If there exists an assignment A that satisfies all the tests in Φ then there exists a solution $\mathbf{r}$ to $\mathbf{B}_f\mathbf{r} = \mathbf{t}$ with $\|\mathbf{B}_v\mathbf{r}\|_\infty = 1$.*

Proof: Give an assignment A, let $\mathbf{r}$ be the vector which is 1 in the coordinates $(x, i, s^*(x), A(x))$ for $x \in X$ and $i \in [d_X]$ and 0 elsewhere. In [18], Regev have show that $\mathbf{B}_f\mathbf{r} = \mathbf{t}$. In the following, we show that $\|\mathbf{B}_v\mathbf{r}\|_\infty = 1$.

For the vector $\mathbf{r}_{(x,i,s)} = (r_{(x,i,s,a_1)}, \dots, r_{(x,i,s,a_{|R_X|})})$, when $x \in X$, $i \in [d_X]$ and $s = s^*(x)$, it is an unit vector; Otherwise, it is an all-0 vector. Thus $H\mathbf{r}_{(x,i,s)}$ is one column of H or all-0 vector. So $\|H\mathbf{r}_{(x,i,s)}\|_\infty \leq 1$. Thus $\|\mathbf{B}_v\mathbf{r}\|_\infty = \max_i \|H\mathbf{r}_{(x,i,s)}\|_\infty = 1$.

Lemma 3 (Soundness) *For any $\epsilon > 0$, if there exists a solution $\mathbf{r}$ to $\mathbf{B}_f\mathbf{r} = \mathbf{t}$ with $\|\mathbf{B}_v\mathbf{r}\|_\infty$ is less than $\sqrt{2-\epsilon}$, then there exists an assignment that satisfies at least a $\frac{1}{2}\epsilon$ fraction of tests.*

Proof: Let $\mathbf{r}$ be a vector such that $\mathbf{B}_f\mathbf{r} = \mathbf{t}$ with $\|\mathbf{B}_v\mathbf{r}\|_\infty \leq \sqrt{2-\epsilon}$. Let $\Delta\mathbf{r}$ denotes the weight of $\mathbf{r}$. Then $\Delta\mathbf{r} \leq \|H\mathbf{r}\|_\infty^2$. Thus for every vector $\mathbf{r}_{(x,i,s)}$, $\Delta\mathbf{r}_{(x,i,s)} \leq 2 - \epsilon$. According to the equations (4) and (5), for every $x \in X$, $i \in [d_X]$, $s \neq s^*(x)$ and every $x \in \hat{X}\backslash X$, $i \in [d_X], s \in S$, if $\Delta\mathbf{r}_{(x,i,s)} \neq 0$, then $\Delta\mathbf{r}_{(x,i,s)} \geq 2$. Thus for all these vector $\Delta\mathbf{r}_{(x,i,s)} = 0$. Thus the weight of $\mathbf{r}$ is at most $(2-\epsilon) \cdot d_X \cdot |X| = (2-\epsilon) \cdot |\Phi|$.

Associate each quadruple $(x, i, s, a) \in T$ for which $r_{(x,i,s,a)}$ is non-zero with the pair $(z_i(x), s_i) \in Y \times [d_Y]$. Since the weight of $\mathbf{r}$ is at most $(2-\epsilon) \cdot |\Phi|$, there exist at most $(1 - \frac{1}{2}\epsilon)|\Phi|$ pairs with which we associate at least 2 quadruples. Therefore, there exist $\frac{1}{2}\epsilon|\Psi|$ pairs with which we associate 1 quadruples. The following process is similar to the proof of lemma 3.4 in [18]. Thus by Regev's proof, there exist an assignment that satisfies at least $\frac{1}{2}\epsilon$ fraction of tests.

By lemma 2, lemma 3 and the PCP of lemma 1, we get the following theorem.

Theorem 1 *For any $\epsilon > 0$, it is NP-hard to approximate MISPP$_\infty$ within a factor $\sqrt{2} - \epsilon$.*

By the proposition 1, we get the hardness of approximating CVPP_∞ as follows.

Theorem 2 *For any $\epsilon > 0$, it is NP-hard to approximate $CVPP_\infty$ within a factor $\sqrt{2} - \epsilon$.*

4 Hardness of $(\log n)^{1/2-\varepsilon}$

In this section, we give a reduction from GapLCPP problem to MISPP_∞. The reduction essentially combines the ideas from [2] and the property of Hadamard matrix. In [2], Aleckhnovich et al give a reduction from GapLCPP to MISPP over ℓ_2 norm. The following lemma has been proved in [2].

Lemma 4 *$GapLCPP_{2-\gamma^k}$ is NP-complete, for some γ and k, where $0 < \gamma < 1$ and k is an integer.*

From an instance of $\mathrm{GapLCPP}_{2-\gamma^k}$, we construct an instance $(\mathbf{B}_f, \mathbf{B}_v, \mathbf{t}, d)$ of MISPP_∞. In fact $\mathbf{B}_v$ and $\mathbf{t}$ have been constructed [2]. We construct a new matrix $\mathbf{B}_v$. For completeness we repeat the construction of $\mathbf{B}_f$ and $\mathbf{t}$ as follows.

Consider following the instance $\mathcal{U}^{T,k}$ of $\mathrm{GapLCPP}_{2-\gamma^k}$ from lemma 4:

$$(G(V, W, E), [R], [S], n, m, \{\pi_e\}_{e \in E}, \{\mathcal{P}_w, P_w\}_{w \in W}),$$

where $n = |V|, m = |W|$ are $n^{O(Tk)}$ and R', S' are $2^{O(Tk)}$. The only part of the input which is not uniform (or does not depend on on) is $\{P_w\}_{w \in W}$. Recall that for every $w \in W$, $\mathcal{P}_w = \cup_l R_{w,l}$ a partition of $|R|$, while the input is the set of permissible labels for each $w, \mathcal{P}_w \in \{R_{w,l}\}_l$. The instance is $\delta = 1/T$ smooth. Recall that this means that for any $w \in W$ and any pair of distinct label $i, i' \in [R]$,

$$\Pr_{v \in_R N(w)} [\pi_{(v,w)}(i) = \pi_{(v,w)}(i')] \leq \delta.$$

Now we define the corresponding GapMISP instance. The variable are:

$$x_{w,i} : \forall w \in W, \forall i \in [R]$$

$$y_{w,i} : \forall v \in V, \forall j \in [S]$$

The fixed linear forms are as follows:

$$\sum_{i \in P_w} x_{w,i} = 1 \qquad \forall w \in W \tag{6}$$

$$\sum_{i \in R_{w,l}} x_{w,i} = 0 \qquad \forall w \in W, R_{w,l} \neq P_w \tag{7}$$

$$\sum_{j \in [S]} y_{v,j} = 1 \qquad \forall v \in V \tag{8}$$

$$(\sum_{i:\pi_{(v,w)}[i]=j} x_{w,i}) - y_{v,j} = 0 \qquad \forall e = (v, w) \in E, \forall j \in [S] \tag{9}$$

The variable linear forms are as follows:

$$H\mathbf{x}_w \qquad \forall w \in W \tag{10}$$

$$H\mathbf{y}_v \qquad \forall v \in V \tag{11}$$

Where H of (10) are $R \times R$ Hadamard matrixes (orthonormal ± 1 matrix), H of (11) are $S \times S$ Hadamard matrixes, $\mathbf{x}_w$ is an vector $(x_{w,1}, \ldots, x_{w,R})$ and $\mathbf{y}_v$ is an vector $(y_{w,1}, \ldots, x_{w,S})$, i.e. $\mathbf{B}_v = \begin{pmatrix} H & \cdots & 0 \\ \vdots & H & \vdots \\ 0 & \cdots & H \end{pmatrix}$.

Since the partition $\mathcal{P}_w = \cup_l R_{w,l}$ depends only on n, the only part that depends on $\{P_w\}_{w\in}$ is the r.h.s of (6) and (7), and hence, this is an instance of GapMISPP. Now we analyze the gap of this reduction and its tradeoff with the size of the instance produced.

Completeness

If $\mathcal{U}^{T,k}$ is a YES instance, then there is an assignment to the variables of the corresponding GapMISP instance such that $\|\mathbf{B}_v\mathbf{x}\|_\infty = 1$. Consider a labeling which satisfies all the edges of $\mathcal{U}^{T,k}$. Now we construct a solution to the GapMISP with $\|\mathbf{B}_v\mathbf{x}\|_\infty = 1$ as follows. If the label i is assigned to the vertex $w \in W$, then assign 1 to $x_{w,i}$, and assign 0 to all $x_{w,i}$, for $i \neq i'$. This makes sure that the constraints (6) and (7) are satisfied. Similarly, if the label j is assigned to the vertex $v \in V$, then assign 1 to $y_{v,j}$, and assign 0 to $y_{v,j'}$, for $j \neq j'$. This makes sure that the constraints (8) are satisfied. Further, if labels i and j are assigned to w and v respectively in this satisfying assignment, then for the (v, w), $\pi_{(v,w)}[i] = j$, and hence, the constraints (9) are also satisfied.

Since every vertex is assigned to one label, $\mathbf{x}_w$ and $\mathbf{y}_v$ are an unit vector. Thus $H\mathbf{x}_w$ and $H\mathbf{y}_v$ are one column of H. So $\|H\mathbf{x}_w\|_\infty = 1$ and $\|H\mathbf{y}_v\|_\infty = 1$. So $\|\mathbf{B}_v\mathbf{x}\|_\infty = 1$.

Soundness

We will establish factor h hardness, where h, as well as other parameters, are fixed below. Assume that there is a solution to the GapMISP with $\|\mathbf{B}_v\mathbf{r}\|_\infty = h$. Then $\|H\mathbf{x}_w\|_\infty \leq h$ and $\|H\mathbf{y}_v\|_\infty \leq h$. Thus $\|\mathbf{x}_w\|_2 \leq \|H\mathbf{x}_w\|_\infty \leq h$ and $\|\mathbf{y}_v\|_2 \leq \|H\mathbf{y}_v\|_\infty \leq h$. Hence $\sum\limits_{i\in[R]} x_{w,i}^2 \leq h^2$ and $\sum\limits_{j\in[S]} y_{v,j}^2 \leq h^2$ for all w and v.

We define blocks of variables by $B_w = \{x_{w,i} : 1 \leq i \leq R\}$ and $A_v = \{y_{v,j} : 1 \leq j \leq S\}$. Thus the number of non-zero variables in the B-blocks, as well as the A blocks are at most h^2. The rest of the proof is similar to the proof of the soundness in [2]. By their argument, we obtain the hardness factor is $(\log n)^{1/2-\epsilon}$. We omit them.

Thus we get the following theorem.

Theorem 3 *For any $\epsilon > 0$, it is NP-hard to approximate MISPP$_\infty$ within a factor* $(\log n)^{1/2-\epsilon}$ *unless* **NP**$\subseteq$ **DTIME** $(2^{polylog(n)})$.

By the proposition 1, we get the hardness of approximating CVPP$_\infty$ as follows.

Theorem 4 *For any $\epsilon > 0$, it is NP-hard to approximate CVPP$_\infty$ within a factor* $(\log n)^{1/2-\epsilon}$ *unless* **NP**$\subseteq$ **DTIME** $(2^{polylog(n)})$.

5 Conclusion

In this paper, we have proved that there is no polynomial time algorithm solving CVPP_∞ within a factor $(\log n)^{1/2-\epsilon}$, unless $\mathbf{NP} \subseteq \mathbf{DTIME}$ $(2^{polylog(n)})$. We obtain the result by a polynomial time reduction from Label Cover Problem with preprocessing to CVPP_∞ via MISPP_∞. Our reduction method is the combination of the reduction method [2] with property of Hadamard matrix.

In [6], I. Dinur proved that it is NP-hard to approximate CVP_∞ within almost-polynomial factors, could we obtain the same hardness factor ? Furthermore, could we obtain polynomial factor hardness of approximating CVPP_∞ (i.e. n^ϵ for some $\epsilon > 0$).

Acknowledgments

We would like to thank the anonymous referees for their careful readings of the manuscripts and many useful suggestions. We are grateful to anonymous referees for referring us to Regev and Rosen's new results [19]. The research was partially supported by young teachers research funds.

References

[1] M. Ajtai. Generating hard instances of lattice problems. In *Proc. of the* 28^{th} *Annual ACM Symposium on Theory of Computing*, 1996, pp 99-108.

[2] M. Alekhnovich, S. Khot, G. Kindler, N. Vishnoi. Hardness of approximating the closest vector problem with pre-processing. in *Proc. 46th IEEE Symposium on FOCS,* 2005, pp 216-225.

[3] M. Ajtai, C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proc. of the* 29^{th} *Annual ACM Symposium on Theory of Computing*, 1997, pp 284-293.

[4] S. Arora, L. Babai, J. Stern, E.Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *Journal of Computer and System Sciences (54),* 1997, pp 317–331.

[5] J. Bruck and M. Naor. The hardness of decoding linear codes with preprocessing. *IEEE Transactions on Information Theory,* 36(2):381-385, 1990.

[6] I. Dinur. Approximating SVP_∞ to within almost-polynomial factors is NP-hard. *Proc. of the 4th Italian Conference on algorithms and Complexity*, LNCS, vol 1767, Springer, 2000.

[7] I. Dinur, G. Kindler, S. Safra. Approximating CVP to within almost-polynomial factors is NP-hard. In *Proceedings of the 39th IEEE Symposium on Foundations of Computer Science*, 1998.

[8] U. Feige and D. Micciancio. The inapproximability of lattice and coding problems with preprocessing. *Journal of Computer and System Sciences,* 69(1):45-67, 2004.

[9] M. R. Garey and D. S. Johnson. *Computers and Intractability: a guide to the theory NP-completeness.* W. H. Freeman and Company, San Francisco, 1979.

[10] C.F. Gauss. Disquisitiones arithmeticae. (leipzig 1801), art. 171. Yale Univ. Press, 1966. English translation by A.A. Clarke.

[11] R. M. Karp and R. J. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proc. 12th ACM Symp. Theory of Computing,* 1980, pp. 302-309.

[12] R. Kumar, D. Sivakumar. Complexity of SVP-A reader's digest. SIGACT News, 32(3), *Complexity Theory Column* (ed. L.Hemaspaandra), 2001, pp 40-52.

[13] A.K. Lenstra, H.W. Lenstra, L. Lov*á*sz. Factoring polynomials with rational coefficients. Mathematische Ann., 261, 1982, pp 513-534.

[14] A. Lobstein. The hardness of solving subset sum with preprocessing. *IEEE Transaction on Information Theory,* vol.36, no. 4, pp.943-946, July 1990.

[15] D. Micciancio. The hardness of the closest vector problem with preprocessing. *IEEE Transaction on Information Theory,* 47:1212-1215, 2001.

[16] D. Micciancio, S. Goldwasser. Complexity of lattice problems, A cryptographic perspective. Klumer Academic Publishers, 2002.

[17] H. Minkowski. Geometrie der zahlen. Leizpig, Tuebner, 1910.

[18] O. Regev. Improved inapproximability of lattice and coding problem with preprocessing. In *Proceedings of the Annual IEEE Conference on Computational Complexity,* Number 18, pages 363-370, 2003.

[19] O. Regev and R. Rosen. Lattice problems and norm embeddings. *Proc. of STOC 2006.*

[20] P. van Emde Boas. Another NP-complete problem and the complexity of computing short vectors in a lattice. Tech. Report 81-04, Mathematische Instiut Univ. of Amsterdam, 1981.