Electronic Colloquium on Computational Complexity, Report No. 16 (1999)

Approximating SVP_∞ to within Almost-Polynomial Factors is NP-hard

I. Dinur *

Abstract

This paper shows SVP_∞ and CVP_∞ to be NP-hard to approximate to within any factor up to $n^{1/\log\log n}$. This improves on the best previous result [ABSS93] that showed quasi-NP-hardness for smaller factors, namely $2^{\log^{1-\varepsilon} n}$ for any constant $\varepsilon > 0$. We show a direct reduction from SAT to these problems, that combines ideas from [ABSS93] and from [DKS98, DKRS99], along with some modifications. Our result is obtained without relying on the PCP characterization of NP, although some of our techniques are derived from the proof of the PCP characterization itself [DFK+].

Introduction

Background

A lattice $L = L(v_1, .., v_n)$, for a basis $v_1, .., v_n \in R^n$ is the additive group generated by the basis vectors, i.e. the set $L = \{\sum a_i v_i \mid a_i \in \mathbf{Z}\}$. Given L, the Shortest Vector Problem (SVP_p) is to find the shortest non-zero vector in L. The length is measured in Euclidean l_p norm ($1 \leq p \leq \infty$). The Closest Vector Problem (CVP_p) is the non-homogeneous analog, i.e. given L and a vector y, find a vector in L, closest to y.

These lattice problems have been introduced in the previous century, and have been studied since. Minkowsky and Dirichlet tried, with little success, to come up with approximation algorithms for these problems. It was much later that the lattice reduction algorithm was presented by Lenstra, Lenstra and Lovász [LLL82] , achieving a polynomial-time algorithm approximating the Shortest Lattice Vector to within the exponential factor $2^{n/2}$, where n is the dimension of the lattice. Babai [Bab86] applied LLL's methods to present an algorithm that approximates CVP to within a similar factor. Schnorr [Sch85] improved on LLL's technique, reducing the factor of approximation to $(1+\varepsilon)^n$, for any constant $\varepsilon > 0$, for both CVP and SVP. These positive approximation results hold for l_p norm for any $p \geq 1$ yet are quite weak, achieving only extremely large (exponential) approximation factors. The shortest vector problem is particularly important, quoting [ABSS93], because even the above relatively weak approximation algorithms have been used in a host of applications, including integer programming, solving low-density subset-sum problems and breaking knapsack based codes [LO85], simultaneous diophantine approximation and factoring polynomials over the rationals [LLL82], and strongly polynomial-time algorithms in combinatorial optimization [FT85].

Interest in lattice problems has been recently renewed due to a result of Ajtai [Ajt96], showing a reduction, from a version of SVP, to the *average-case* of the same problem. Finding a problem

*Tel-Aviv University

ISSN 1433-8092

whose average case complexity is as hard as the worst-case of some other problem is interesting from a theoretic perspective. Yet this result also has significant cryptographic applications - [AD97] showed that NP-hardness for that specific restriction of SVP – although unlikely [GG98] – would imply an unbreakable cryptosystem, unless P=NP.

Only recently [Ajt97] showed a randomized reduction from the NP-complete problem Subset-Sum to SVP. This has been improved [CN98], showing approximation hardness for some small factor $(1+\frac{1}{n^\varepsilon})$. Very recently [Mic98] has significantly strengthened Ajtai's result, showing SVP hard to approximate to within some constant factor.

The above results all apply to SVP_p, for finite p. SVP with the maximum norm l_∞, appears to be a harder problem. A g-approximation algorithm for SVP_2 implies a $\sqrt{n}g$-approximation algorithm for SVP_∞, since for every vector v, $\|v\|_\infty \leq \|v\|_2 \leq \sqrt{n} \cdot \|v\|_\infty$. Thus hardness for approximating SVP_∞ to within a factor $\sqrt{n}g$ will imply the hardness for approximating SVP_2 to within factor g. Lagarias showed SVP_∞ to be NP-hard in its exact decision version. Arora et al. [ABSS93] utilized the PCP characterization of NP to show that both CVP (for l_1 norm) and SVP_∞ are quasi-NP-hard to approximate to within $2^{(\log n)^{1-\varepsilon}}$ for any constant $\varepsilon > 0$. Their result implicitly holds for CVP_∞ as well. Recently, the hardness result for approximating CVP has been strengthened [DKS98, DKRS99] showing that it is NP-hard to approximate to within a factor of $n^{1/\log\log n}$ (where n is the lattice dimension). In this paper we similarly strengthen the hardness result for approximating SVP_∞ and CVP_∞.

So far there is still a huge gap between the positive results, showing approximations for SVP and CVP with exponential factors, and the above hardness results. Nevertheless, some other results provide a discouraging indication for improving the hardness result beyond a certain factor. [GG98] showed that approximating both SVP_2 and CVP_2 to within $\sqrt{n}$ and approximating SVP_∞ and CVP_∞ to within $n/O(\log n)$ is in NP $\cap$ co-AM. Hence it is unlikely for any of these problems to be NP-hard.

Our Result

We prove that approximating SVP_∞ to within a factor of $n^{c/\log\log n}$ is NP-hard (where n is the lattice dimension and $c > 0$ is some fixed constant). Our result is also easily adapted to CVP_∞.

Technique

We obtain our result by modifying (and in some ways, simplifying) the framework of [DKS98, DKRS99].

Our first attempt was to adapt the reduction [DKS98] from $\mathcal{SSAT}$ to CVP_1, to SVP_∞. The main obstacle, however, is that the natural lattice corresponding to an $\mathcal{SSAT}$ instance consists of very-short vectors. These vectors did not pose a problem in the case of CVP_1 since they are very far from the target vector. We overcome this problem by modifying the structure of the $\mathcal{SSAT}$ instance itself. We define a variant of the $\mathcal{SSAT}$ problem which we call $\mathcal{SSAT}_\infty$.. We first show that $\mathcal{SSAT}_\infty$ is NP-hard to approximate to within the above factors of $n^{c/\log\log n}$, and then reduce $\mathcal{SSAT}_\infty$ to SVP_∞ and to CVP_∞.

We prove $\mathcal{SSAT}_\infty$ NP-hard to approximate by modifying the recursive tree-like construction from [DKS98, DKRS99], so as to eliminate the aforementioned very-short lattice vectors. This requires some simple observations regarding the embedding-extension and low-degree-functions (see propositions 1 and 4). The reductions from $\mathcal{SSAT}_\infty$ to SVP_∞ and CVP_∞ are slightly more tricky than the reduction from $\mathcal{SSAT}$ to CVP_1 [DKS98], and utilize an additional idea from [ABSS93].

Hardness-of-approximation results are naturally divided into those that are obtained via reduction from PCP, and those that are not. Although the best previous hardness result for SVP_∞ [ABSS93] relies on the PCP characterization of NP, our proof does not. We do, however, utilize some techniques similar to those used in the proof of the PCP characterization of NP itself. In fact, the l_∞ norm allows a cleaner construction, that avoids some of the technical complications in [DFK$^+$, DKS98, DKRS99]. Thus, we believe that SVP_∞ may be the best candidate (out of all of the lattice problems) for pushing the hardness-of-approximation factor to within polynomial range.

Structure of the Paper

Section 1 presents a variant of the $\mathcal{SSAT}$ problem from [DKS98] called $\mathcal{SSAT}_\infty$. It then proceeds with some definitions. Section 2 gives the reduction from SAT to $\mathcal{SSAT}_\infty$, whose correctness is proven in section 3. Finally, in section 4 we describe the (simple) reduction from $\mathcal{SSAT}_\infty$ to SVP_∞ and to CVP_∞, establishing the hardness of approximating SVP_∞ and CVP_∞.

1 Definitions

1.1 $\mathcal{SSAT}_\infty$

In this section we reconsider the $\mathcal{SSAT}$ problem introduced in [DKS98], and define a slightly modified version of this problem – $\mathcal{SSAT}_\infty$. Let $\Psi = \{\psi_1, \ldots, \psi_n\}$ be a system of *tests* (Boolean functions) over variables $V = \{v_1, .., v_m\}$. Denote by $\mathcal{R}_{\psi_i}$ the set of satisfying assignments for $\psi_i \in \Psi$. We recall the following definitions 1,2 and 3 from [DKS98],

Definition 1 (Super-Assignment to Tests) *A super-assignment is a function S mapping to each $\psi \in \Psi$ a value from $\mathbb{Z}^{\mathcal{R}_\psi}$. $S(\psi)$ is a vector of integer coefficients, one for each value $r \in \mathcal{R}_\psi$. Denote by $S(\psi)[r]$ the r^{th} coordinate of $S(\psi)$.*

If $S(\psi)[r] \neq \phi$, we say that the value r appears in $S(\psi)$. A *natural assignment* (an assignment in the usual sense) is identified with a super-assignment that assigns each $\psi \in \Psi$ a unit vector with a 1 in the corresponding coordinate. In this case, exactly one value appears in each $S(\psi)$.

We next define the projection of a super-assignment to a test onto each of its variables. Consistency between tests will amount to equality of projections on mutual variables.

Definition 2 (Projection) *Let S be a super-assignment to the tests. We define the projection of $S(\psi)$ on a variable x of ψ, $\pi_x(S(\psi)) \in \mathbb{Z}^{|\mathcal{F}|}$, in the natural way:*

$$\forall a \in \mathcal{F}: \qquad \pi_x(S(\psi))[a] \stackrel{def}{=} \sum_{r \in \mathcal{R}_\psi,\ r|_x = a} S(\psi)[r]$$

We shall now proceed to define the notion of consistency between tests. If the projections of two tests on each mutual variable x are equal (in other words, they both give x the same super-assignment), we say that the super-assignments of the tests are consistent (match).

Definition 3 (Consistency) *Let S be a super-assignment to the tests in Ψ. S is consistent if for every pair of tests ψ_i and ψ_j with a mutual variable x,*

$$\pi_x(S(\psi_i)) = \pi_x(S(\psi_j))$$

Given a system $\Psi = \{\psi_1, ..., \psi_n\}$, a super-assignment $S : \Psi \to \mathbb{Z}^{\mathcal{R}}$ is called *not-all-zero* if it is non-trivial on at least one test $\psi \in \Psi$ (i.e. $\exists \psi \in \Psi,\ S(\psi) \neq \vec{\mathbf{0}}$). This is a weaker requirement than the non-triviality of $\mathcal{SSAT}$. On the other hand, the norm of a super assignment S is measured by a 'stronger' measure,

$$\|S\| \stackrel{def}{=} \max_{\psi \in \Psi} (\|S(\psi)\|_2)$$

where $\|S(\psi)\|_2$ is the standard l_2 norm. The norm of a natural super-assignment is 1. Finally we define consistency of a super-assignment as in [DKRS99],

The gap of $\mathcal{SSAT}_\infty$ is formulated in terms of the norm of the minimal super-assignment that maintains consistency.

Definition 4 (*g*-$\mathcal{SSAT}_\infty$) *An instance of $\mathcal{SSAT}_\infty$ with parameter g*

$$I = \langle \Psi = \{\psi_1, .., \psi_n\}, V = \{v_1, .., v_m\}, \{\mathcal{R}_{\psi_1}, .., \mathcal{R}_{\psi_n}\} \rangle$$

consists of a set Ψ of tests *over a common set V of variables that take values in a field $\mathcal{F}$. The parameters m and $|\mathcal{F}|$ are always bounded by some polynomial in n. Each test $\psi \in \Psi$ has associated with it a list $\mathcal{R}_\psi$ of assignments to its variables, called the* satisfying assignments *or the* range *of the test ψ. The problem is to distinguish between the following two cases,*

Yes: There is a consistent natural assignment for Ψ.

No: No non-trivial consistent super-assignment is of norm $> g$.

Theorem 1 ($\mathcal{SSAT}_\infty$ **Theorem**) *$\mathcal{SSAT}_\infty$ is NP-hard for $g = n^{1/\log\log n}$.*

We conjecture that a stronger statement is true, which would imply that SVP_∞ NP-hard to approximate to within a *constant power* of the dimension.

Conjecture 2 *$\mathcal{SSAT}_\infty$ is NP-hard for $g = n^c$ for some constant $c > 0$.*

1.2 LDFs, Super-LDFs

Throughout the paper, let $\mathcal{F}$ denote a finite field $\mathcal{F} = \mathbb{Z}_p$ for some prime number $p > 1$. We adopt the following definitions from [DKS98].

Definition 5 (low degree function - $[r, d]$**-LDF)** *A function $f : \mathcal{F}^d \to \mathcal{F}$ is said to have degree r if its values are the point evaluation of a polynomial on $\mathcal{F}^d$ with degree $\leq r$ in each variable. In this case we say that f is an $[r, d]$-LDF, or $f \in LDF_{r,d}$.*

Sometimes we omit the parameters and refer simply to an LDF.

Definition 6 (low degree extension) *Let m, d be natural numbers, and let $\mathcal{H} \subset \mathcal{F}$ such that $|\mathcal{H}|^d = m$. A vector $(a_0, .., a_{m-1}) \in \mathcal{F}^m$ can be naturally identified with a function $A: \mathcal{H}^d \to \mathcal{F}$ by looking at points in $\mathcal{H}^d$ as representing numbers in base $|\mathcal{H}|$.*

There exists exactly one $[|\mathcal{H}| - 1, d]$-LDF $\hat{A} : \mathcal{F}^d \to \mathcal{F}$ that extends A. $\hat{A}$ is called the $|\mathcal{H}| - 1$ degree extension of A in $\mathcal{F}$.

A $(D+2)$-dimensional affine subspace ($(D+2)$-*cube* for short) $\mathcal{C} \subset \mathcal{F}^d$ is said to be *parallel* to the axises if it can be written as $\mathcal{C} = \boldsymbol{x} + \mathrm{span}(e_{i_1}, \ldots, e_{i_{D+2}})$, where $\boldsymbol{x} \in \mathcal{F}^d$ and $e_i \in \mathcal{F}^d$ is the i-th axis vector, $e_i = (0, .., 1, .., 0)$. We write the parameterization of the cube $\mathcal{C}$ as follows,

$$\mathcal{C}(\bar{z}) \stackrel{def}{=} \boldsymbol{x} + \sum_{j=1}^{D+2} z_j e_{i_j} \in \mathcal{F}^d \qquad \text{for } \bar{z} = (z_1, .., z_{D+2}) \in \mathcal{F}^{D+2}$$

We will need the following (simple) proposition,

Proposition 1 *Let $f : \mathcal{F}^d \to \mathcal{F}$. Suppose, for every parallel $(D+2)$-cube $\mathcal{C} \subset \mathcal{F}^d$ the function $f|_{\mathcal{C}} : \mathcal{F}^{D+2} \to \mathcal{F}$ defined by*

$$\forall \boldsymbol{x} \in \mathcal{F}^{D+2} \quad f|_{\mathcal{C}}(\boldsymbol{x}) = f(\mathcal{C}(\boldsymbol{x}))$$

is an $[r, D+2]$-LDF. Then f is an $[r, d]$-LDF.

Similar to the definition of super-assignments, we define a *super-$[r, d]$-LDF* (or a super-LDF for short) $\mathcal{G} \in \mathbb{Z}^{\mathrm{LDF}_{r,d}}$ to be a vector of integer coefficient $\mathcal{G}[P]$ per LDF $P \in \mathrm{LDF}_{r,d}$. This definition arises naturally from the fact that the tests in our final construction will range over LDFs. We further define the *norm* of a super-LDF to be the l_2 norm of the corresponding coefficient vector.

We say that an LDF $P \in \mathrm{LDF}_{r,d}$ *appears* in $\mathcal{G}$ iff $\mathcal{G}[P] \neq 0$. A point $\boldsymbol{x}$ is called *ambiguous* for a super-LDF $\mathcal{G}$, if there are two LDFs P_1, P_2 appearing in $\mathcal{G}$ such that $P_1(\boldsymbol{x}) = P_2(\boldsymbol{x})$. The following (simple) property of *low-norm* super-LDFs is heavily used in this paper.

Proposition 2 (Low Ambiguity) *Let $\mathcal{G}$ be a super-$[r, d]$-LDF of norm $\|\mathcal{G}\|_2 \leq g$. The fraction of ambiguous points for $\mathcal{G}$ is $\leq \mathrm{amb}(r, d, g) \stackrel{def}{=} \binom{g^2}{2} \frac{rd}{|\mathcal{F}|}$.*

Proof: The number of non-zero coordinates in a vector whose l_2 norm is g is $\leq g^2$. There are $\leq \binom{g^2}{2}$ pairs of LDFs appearing in $\mathcal{G}$, and each pair agrees on at most $\frac{rd}{|\mathcal{F}|}$ of the points in $\mathcal{F}^d$. ■

The following embedding-extension technique taken from [DFK+] is used in our construction,

Definition 7 (embedding extension) *Let $b \geq 2$, $k > 1$ and t be natural numbers. We define the* embedding extension mapping *$E_b : \mathcal{F}^t \to \mathcal{F}^{t \cdot k}$ as follows. E_b maps any point $\boldsymbol{x} = (\xi_1, .., \xi_t) \in \mathcal{F}^t$ to $y \in \mathcal{F}^{t \cdot k}$, $y = E_b(\boldsymbol{x}) = (\eta_1, .., \eta_{t \cdot k})$ by*

$$E_b(\xi_1, .., \xi_t) \stackrel{def}{=} \left(\xi_1, (\xi_1)^b, (\xi_1)^{b^2}, .., (\xi_1)^{b^{k-1}}, \ldots, \xi_t, (\xi_t)^b, (\xi_t)^{b^2}, .., (\xi_t)^{b^{k-1}}\right)$$

The following (simple) proposition, shows that any LDF on $\mathcal{F}^t$ can be represented by an LDF on $\mathcal{F}^{t \cdot k}$ with significantly lower degree:

Proposition 3 *Let $f : \mathcal{F}^t \to \mathcal{F}$ be a $[b^k - 1, t]$-LDF, for integers $t > 0, b > 1, k > 1$. There is a $[b - 1, t \cdot k]$-LDF $f_{ext} : \mathcal{F}^{t \cdot k} \to \mathcal{F}$ such that*

$$\forall \boldsymbol{x} \in \mathcal{F}^t : \quad f(\boldsymbol{x}) = f_{ext}(E_b(\boldsymbol{x}))$$

For any $[b-1, kt]$-LDF f, its 'restriction' to the manifold $f|_{E_b} : \mathcal{F}^t \to \mathcal{F}$ is defined as

$$\forall \boldsymbol{x} \in \mathcal{F}^t \quad f|_{E_b}(\boldsymbol{x}) \stackrel{def}{=} f(E_b(\boldsymbol{x}))$$

and is a $[b^k-1, t]$-LDF (the degree in a variable ξ_i of $f|_{E_b}$ is $(b-1)(b^0+b^1+\ldots+b^{k-1}) = b^k-1$).

Let $\widetilde{\mathcal{G}}$ be a super-$[b^k-1, t]$-LDF (i.e. a vector in $\mathbb{Z}^{\mathrm{LDF}_{r,t}}$). Its *embedding-extension* is the super-$[b-1, tk]$-LDF $\mathcal{G}$ defined by,

$$\forall f \in \mathrm{LDF}_{b-1,tk} \quad \mathcal{G}[f] \stackrel{def}{=} \widetilde{\mathcal{G}}[f|_{E_b}]$$

In a similar manner, the *restriction* $\widetilde{\mathcal{G}}$ of a super-$[b-1, tk]$-LDF $\mathcal{G}$ is a super-$[b^k-1, t]$-LDF defined by

$$\forall f \in \mathrm{LDF}_{b^k-1,t} \quad \mathcal{G}[f] \stackrel{def}{=} \widetilde{\mathcal{G}}[f_{ext}]$$

The following proposition holds (e.g. by a counting argument),

Proposition 4 *Let $\widetilde{\mathcal{G}}_1, \widetilde{\mathcal{G}}_2$ be two super-$[b^k-1, t]$-LDFs, and let $\mathcal{G}_1, \mathcal{G}_2$ be their embedding extensions (with parameter b). $\widetilde{\mathcal{G}}_1 = \widetilde{\mathcal{G}}_2$ if and only if $\mathcal{G}_1 = \mathcal{G}_2$.*

2 The Construction

We prove that $\mathcal{SSAT}_\infty$ is NP-hard via a reduction from SAT, described herein. We adopt the whole framework of the construction from [DKRS99], and refer the reader there for a more detailed exposition.

Let $\Phi = \{\varphi_1, .., \varphi_n\}$ be an instance of SAT, viewed as a set of Boolean *tests* over Boolean variables $V_\Phi = \{\mathrm{x}_1, .., \mathrm{x}_m\}$, ($m = n^c$ for some constant $c > 0$) such that each test depends on $D = O(1)$ variables. Cook's theorem [Coo71] states that it is NP-hard to decide whether there is an assignment for V_Φ satisfying all of the tests in Φ.

Starting from Φ, we shall construct an $\mathcal{SSAT}_\infty$ test-system Ψ over variables $V_\Psi \supset V_\Phi$. Our new variables V_Ψ will be non-Boolean, ranging over a field $\mathcal{F}$, with $|\mathcal{F}| = n^{c/\log\log n}$. An assignment to V_Ψ will be interpreted as an assignment to V_Φ by identifying the value $0 \in \mathcal{F}$ with the Boolean value `true` and any other non-zero value with `false`.

2.1 Constructing the CR-Forest

Before constructing the tests in Ψ, we construct the CR-forest, which is a combinatorial object holding the underlying structure of Ψ. The forest $\mathbf{F}_n(\Phi)$ will have a tree $\mathbf{T}_\varphi$ for every test $\varphi \in \Phi$. Let us (briefly) describe one tree $\mathbf{T}_\varphi$ in the forest $\mathbf{F}_n(\Phi)$.

Every tree will be of depth $K \leq \log\log n$ (however, not all of the leaves will be at the bottom level). Each node v in the tree will have a domain $\mathbf{dom}_v = \mathcal{F}^d$ of points ($\mathbf{dom}_v = \mathcal{F}^{d_0}$ in case v is the root node) associated with it. The offsprings of a non-leaf node v will be labeled each by a distinct $(D+2)$-cube $\mathcal{C}_v$ of $\mathbf{dom}_v$ (this part is slightly simpler than in [DKRS99]),

$$\mathbf{labels}(v) \stackrel{def}{=} \{\mathcal{C} \mid \mathcal{C} \text{ is a } (D+2)\text{-cube in } \mathbf{dom}_v\} .$$

The points in the domain $\mathbf{dom}_v$ of each node v will be mapped to some of Ψ's variables, by the injection $\mathbf{var}_v : \mathbf{dom}_v \to V_\Psi$. This mapping essentially describes the relation of a node to its parent, and is defined inductively as follows. For each node v, we denote by V_v the set of

'fresh new' variables mapped from $\mathbf{dom}_v$ (i.e. none of the nodes defined inductively so far have points mapped to these variables). Altogether

$$V \stackrel{def}{=} V_\Psi = \bigcup_{\substack{v \in \mathbf{T}_\varphi \\ \varphi \in \Phi}} V_v \ .$$

For the root node, $\mathbf{var}_{root_\varphi} : \mathbf{dom}_{root_\varphi} \to V_\Psi$ is defined (exactly as in [DKRS99]) by mapping $\mathcal{H}^{d_0} \subseteq \mathbf{dom}_{root_\varphi} = \mathcal{F}^{d_0}$ to V_Φ and the rest of the points to the rest of $V_{root_\varphi} \stackrel{def}{=} \hat{V}_\Phi \subset V_\Psi$ (i.e. the low-degree-extension of V_Φ). It is important that $\mathbf{var}_{root_\varphi}$ is defined independently of φ.

For a non-root node v with parent u, the points of the cube $\mathcal{C}_v \in \mathbf{labels}(u)$ labeling v are mapped into the domain $\mathbf{dom}_v$ by the embedding extension mapping, $E_{b_v} : \mathcal{C}_v \to \mathbf{dom}_v$, defined above in section 1.2 (the parameter b_v specified below depends on the specific node v, rather than just on v's level as in [DKRS99]). These points are u's points that are 'passed on' to the offspring v. We think of the point $y = E_{b_v}(\boldsymbol{x}) \in \mathbf{dom}_v$ as 'representing' the point $\boldsymbol{x} \in \mathcal{C}_v \subset \mathbf{dom}_u$, and define $\mathbf{var}_v : \mathbf{dom}_v \to V_\Psi$ as follows,

Definition 8 ($\mathbf{var}_v$, for a non-root node v) *Let v be a non-root node, let u be v's parent, and let $\mathcal{C}_v \subset \mathbf{dom}_u$ be the label attached to v. For each point $y \in E_{b_v}(\mathcal{C}_v) \subset \mathbf{dom}_v$ define $\mathbf{var}_v(y) \stackrel{def}{=} \mathbf{var}_u(E_{b_v}^{-1}(y))$, i.e. points that 'originated' from $\mathcal{C}_v$ are mapped to the previous-level variables, that their pre-images in $\mathcal{C}_v$ were mapped to. For each 'new' point $y \in \mathbf{dom}_v \setminus E_{b_v}(\mathcal{C}_v)$ we define $\mathbf{var}_v(y)$ to be a distinct variable from V_v.*

The parameters used for the embedding extension mappings E_{b_v} are $t = D + 2$, $k = d/t = a$. We set the degree of the root node $r_{root_\varphi} = |\mathcal{H}| = |\mathcal{F}|^{1/10}$ and r_v and b_v (for non-root nodes v) are defined by the following recursive formulas:

$$\begin{aligned} b_v &= \begin{cases} \sqrt[a]{r_u + 1} & \mathcal{C}_v \text{ is parallel to the axises} \\ \sqrt[a]{r_u(D+2)+1} & \text{Otherwise} \end{cases} \\ r_v &= b_v - 1 \end{aligned}$$

We stop the recursion and define a node to be a leaf (i.e. define its labels to be empty) whenever $r_v \leq 2(D+2)$. A simple calculation (to appear in the complete version) shows that b_v, r_v decrease with the level of v until for some level $K < \log\log n$, $r_v \leq 2(D+2) = O(1)$. (This may happen to some nodes sooner than others, therefore not all of the leaves are in level K).

3 Correctness of the Construction

3.1 Completeness

Lemma 3 (completeness) *If there is an assignment $\mathcal{A} : V_\Phi \to \{\texttt{true}, \texttt{false}\}$ satisfying all of the tests in Φ, then there is a natural assignment $\mathcal{A}_\Psi : V_\Psi \to \mathcal{F}$ satisfying all of the tests in Ψ.*

We extend $\mathcal{A}$ in the obvious manner, i.e. by taking its low-degree-extension (see definition 6) to the variables $\hat{V}_\Phi$, and then repeatedly taking the embedding extension of the previous-level variables, until we've assigned all of the variables in the system. More formally,

Proof: We construct an assignment $\mathcal{A}_\Psi : V_\Psi \to \mathcal{F}$ by inductively obtaining $[r_v, d]$-LDFs $P_v : \mathbf{dom}_v \to \mathcal{F}$ for every node v of every tree in the CR-forest, as follows. We first set (for every $\varphi \in \Phi$) P_{root_φ} to be the low degree extension (see definition 6) of $\mathcal{A}$ (we think of $\mathcal{A}$ as assigning each variable a value in $\{0,1\} \subset \mathcal{F}$ rather than $\{\texttt{true}, \texttt{false}\}$, see discussion in the beginning of section 2). Assume we've defined an $[r_u, d]$-LDF P_u consistently for all level-i nodes, and let v be an offspring of u, labeled by $\mathcal{C}_v$. The restriction $f = P_u|_{\mathcal{C}_v}$ of P_u to the cube $\mathcal{C}_u$ is an $[r, D+2]$-LDF where $r = r_u$ or $r = r_u(D+2)$ depending on whether $\mathcal{C}_v$ is parallel to the axises or not. f can be written as a $[\sqrt[a]{r+1} - 1,\ a \cdot (D+2)]$-LDF f_{ext} over the larger domain $\mathcal{F}^d$, as promised by proposition 3 (taking $b = \sqrt[a]{r+1}$). We define $P_v = f_{\mathbf{ext}}$ to be that $[r_v, d]$-LDF (recall that $d = a \cdot (D+2)$ and $b_v = \sqrt[a]{r+1}$).

Finally, for a variable $\mathrm{x} \in \mathbf{var}_v$, $\mathrm{x} = \mathbf{var}_v(\boldsymbol{x})$, we set $\mathcal{A}_\Psi(\mathrm{x}) \stackrel{def}{=} P_v(\boldsymbol{x})$. The construction implies that there are no collisions, i.e. $\mathrm{x}' = \mathbf{var}_{v'}(\boldsymbol{x}') = \mathbf{var}_v(\boldsymbol{x}) = \mathrm{x}$ implies $P_v(\boldsymbol{x}) = P_{v'}(\boldsymbol{x}')$. ■

3.2 Soundness

We need to show that a 'no' instance of SAT is mapped to a 'no' instance of $\mathcal{SSAT}_\infty$. We assume that the constructed $\mathcal{SSAT}_\infty$ instance has a consistent super-assignment of norm $\leq g$, and show that Φ – the SAT instance we began with – is satisfiable.

Lemma 4 (Soundness) *Let $g \stackrel{def}{=} |\mathcal{F}|^{\frac{1}{102}}$. If there exists a consistent super-assignment of norm $\leq g$ for Ψ, then Φ is satisfiable.*

Let $\mathcal{A}$ be a consistent super-assignment for Ψ, with $\|\mathcal{A}\|_\infty \leq g$. It induces (by projection) a super-assignment to the variables

$$m : V_\Psi \longrightarrow \mathbb{Z}^{|\mathcal{F}|}$$

i.e. for every variable $\mathrm{x} \in V_\Psi$, m assigns a vector $\pi_{\mathrm{x}}(\mathcal{A}(\psi))$ of integer coefficients, one per value in $\mathcal{F}$ where ψ is some test depending on x. Since $\mathcal{A}$ is consistent, m is well defined (independent of the choice of test ψ). Alternatively, we view m as a labeling of the points $\bigcup_{v \in \mathbf{F}_n(\Phi)} \mathbf{dom}_v$ by a 'super-value' – a formal linear combination of values from $\mathcal{F}$. The label of the point $\boldsymbol{x} \in \mathbf{dom}_v$ for some $v \in \mathbf{F}_n(\Phi)$, is simply $m(\mathbf{var}_v(\boldsymbol{x}))$, and with a slight abuse of notation, is sometimes denoted $m(\boldsymbol{x})$. m is used as the "underlying point super-assignment" for the rest of the proof, and will serve as an anchor by which we test consistency. We denote by $\|m\|_\infty$ the norm of m defined by $\|m\|_\infty \stackrel{def}{=} \max_{\boldsymbol{x}} \|m(\boldsymbol{x})\|_2$. Obviously, $\|\mathcal{A}\|_\infty \leq g$ implies $\|m\|_\infty \leq g$.

The central task of our proof is to show that if a tree has a non-trivial leaf, then there is a non-trivial super-LDF for the domain in the root node that is consistent with m. This is inductively shown to hold for every node in the tree (and thus for the root),

Lemma 5 *Let $u \in \mathbf{nodes}_i$ for some $0 \leq i < K$. There is a legal super-$[r_u, d]$-LDF $\mathcal{G}_u$ with $\|\mathcal{G}_u\|_2 \leq \|m\|_\infty \stackrel{def}{=} \max_{\boldsymbol{x}} \|m(\boldsymbol{x})\|_2$ such that for every $\boldsymbol{x} \in \mathbf{dom}_u$, $\pi_{\boldsymbol{x}}(\mathcal{G}_u) = m(\boldsymbol{x})$. If there is a node v in u's sub-tree for which $\mathcal{G}_v \neq \vec{\mathbf{0}}$ then $\mathcal{G}_u \neq \vec{\mathbf{0}}$.*

Proof: We prove the lemma for $i < K$ by induction on $K - i$. For nodes in level $i = K$ (or any other leaf) the lemma follows by setting $\mathcal{G}_u \stackrel{def}{=} \mathcal{A}(\psi_u)$.

Let $0 \leq i < K$, let $u \in \mathbf{nodes}_i$ be an internal node, and assume (by induction) that for every offspring v of u there is a legal super-$[r_v, d]$-LDF $\mathcal{G}_v$ over $\mathbf{dom}_v$, with $\|\mathcal{G}_v\|_2 \leq \|m\|_\infty$ such that

$$\forall \boldsymbol{x} \in \mathbf{dom}_v \quad \pi_{\boldsymbol{x}}(\mathcal{G}_v) = m(\boldsymbol{x})$$

Let $\widetilde{\mathcal{G}}_v$ be the super-$[r, D+2]$-LDF that is the restriction of $\mathcal{G}_v$ to the manifold $E_{b_v}(\mathcal{C}_v)$ as defined in section 1.2, where $r = (b_v)^a - 1 \leq r_u(D+2)$ (with equality when $\mathcal{C}_v$ isn't parallel to the axises). We know (see proposition 4) that if $\mathcal{G}_v \neq \vec{\mathbf{0}}$ then $\widetilde{\mathcal{G}}_v \neq \vec{\mathbf{0}}$ (since $r_v = b_v - 1$).

The super-LDFs $\widetilde{\mathcal{G}}_v$ have total degree $\leq (D+2)^2 \cdot r_u$. The following consistency lemma will imply the existence of a globally-consistent super-LDF $\mathcal{G}_u$,

Lemma 6 ([DKRS99]) *Let $u \in \mathbf{nodes}_i$ for some $0 \leq i < K$. If for every offspring v of u there is a super-LDF $\widetilde{\mathcal{G}}_v$ over $\mathcal{C}_v$, of total degree $\leq r$ and norm $\|\widetilde{\mathcal{G}}_v\|_2 \leq s$, such that*

$$\Pr_{x \in \mathcal{C}_v}\left(\pi_x(\widetilde{\mathcal{G}}_v) = m(x)\right) \geq 0.99$$

then there is a super-LDF $\mathcal{G}_u$ over $\mathbf{dom}_u$ of degree r and norm $\|\mathcal{G}_u\|_2 \leq 2s$ that obeys

$$\Pr_{\mathcal{C}_v}\left(\pi_{\mathcal{C}_v}(\mathcal{G}_u) = \widetilde{\mathcal{G}}_v\right) \geq 0.99$$

We apply this lemma taking $s = \|m\|_\infty$ and $r \stackrel{def}{=} (D+2)^2 \cdot r_u$. Note that in our case the inductive assumption gives

$$\forall \mathcal{C}_v \in \mathbf{labels}(u), \quad \Pr_{x \in \mathcal{C}_v}\left(\pi_x(\widetilde{\mathcal{G}}_v) = m(x)\right) = 1$$

Thus we obtain a super-LDF $\mathcal{G}_u$ over $\mathbf{dom}_u$ of total-degree r. However, any f appearing in $\mathcal{G}_u$, is actually an $[r_u, d]$-LDF. This follows by considering the set of cubes parallel to the axises in which f appears. The super-LDFs $\widetilde{\mathcal{G}}_v$ over these cubes are of degree r_u in each variable. The ambiguity of $\mathcal{G}_u$ is $\mathrm{amb}(r, d, 2\|m\|_\infty) \leq |\mathcal{F}|^{-\frac{1}{2}}$, thus the fraction of cubes parallel to each axis that may be ambiguous for f is negligible (and in particular, less than half). Proposition 1 thus implies that f is an $[r_u, d]$-LDF as claimed, and it makes sense to say that $\mathcal{G}_u$ is a super-$[r_u, d]$-LDF.

The above lemma 6 only implies consistency of $\mathcal{G}_u$ on most points $x \in \mathcal{F}^d$. Let $\mathrm{N} = \left\{x \in \mathcal{F}^d \mid \pi_x(\mathcal{G}_u) \neq m(x)\right\}$ be the set of inconsistent points. For the sake of contradiction assume $\mathrm{N} \neq \phi$, and let $x_0 \in \mathrm{N}$. Consider any cube $\mathcal{C}_v \in \mathbf{labels}(u)$ that contains x_0. We have $\pi_{x_0}(\widetilde{\mathcal{G}}_v) = m(x_0) \neq \pi_{x_0}(\mathcal{G}_u)$, so $\pi_{\mathcal{C}_v}(\mathcal{G}_u) \neq \widetilde{\mathcal{G}}_v$, therefore the super-LDF $\pi_{\mathcal{C}_v}(\mathcal{G}_u) - \widetilde{\mathcal{G}}_v$ (subtraction is defined as subtraction of two vectors in $\mathbb{Z}^{|\mathrm{LDF}_{r,D+2}|}$) is non-trivial. Proposition 2 (low-ambiguity), when applied to $\pi_{\mathcal{C}_v}(\mathcal{G}_u) - \widetilde{\mathcal{G}}_v$ implies that for *almost all* points $x \in \mathcal{C}_v$, $\pi_x(\mathcal{G}_u) \neq \pi_x(\widetilde{\mathcal{G}}_v) = m(x)$, so these points are also in N. A simple geometric argument shows that the distribution of choosing a $(D+2)$-cube $\mathcal{C}$ containing x_0, and then choosing a random point $x \in_R \mathcal{C}$ is very close to uniformly choosing a point $x \in_R \mathcal{F}^d$. We saw that a point chosen in this manner has high probability of being in N, thus N consists of (much more than) half of the points in $\mathcal{F}^d$. The fraction of $(D+2)$-cubes that don't hit a point in N is (by another simple geometric argument, relying on the fact that N is large enough) very small, and in particular, less than 0.99. Thus by lemma 6 there is a cube $\mathcal{C}_v$ for which $\pi_{\mathcal{C}_v}(\widetilde{\mathcal{G}}_v) = \widetilde{\mathcal{G}}_v$ with $\exists x_1 \in \mathrm{N} \cap \mathcal{C}_v$ and so $\pi_{x_1}(\mathcal{G}_u) = \pi_{x_1}(\widetilde{\mathcal{G}}_v) = m(x_1)$, a contradiction to $x_1 \in \mathrm{N}$. Thus $\mathrm{N} = \phi$, or

$$\forall x \in \mathcal{F}^d \quad \pi_x(\mathcal{G}_u) = m(x)$$

Finally, if $\mathcal{G}_v \neq \vec{\mathbf{0}}$ for some offspring v of u, then $\widetilde{\mathcal{G}}_v \neq \vec{\mathbf{0}}$ because of proposition 4 and since $\pi_{\mathcal{C}_v}(\mathcal{G}_u) = \widetilde{\mathcal{G}}_v$, we deduce $\mathcal{G}_u \neq \vec{\mathbf{0}}$. By proposition 2 (low-ambiguity) for most points $\|m(x)\|_2 = \|\mathcal{G}_u\|_2$, so obviously $\|\mathcal{G}_u\|_2 \leq \max_x \|m(x)\|_2 = \|m\|_\infty$. ∎

In order to complete the soundness proof, we need to find a satisfying assignment for Φ. We obtained, in lemma 5, a super-$[r_{root_\varphi}, d]$-LDF (where $r_{root_\varphi} = |\mathcal{H}|$) $\mathcal{G}_\varphi$ for each root node $root_\varphi$, such that $\forall \boldsymbol{x} \in \mathbf{dom}_{root_\varphi} = \mathcal{F}^{d_0}$, $m(\boldsymbol{x}) = \pi_{\boldsymbol{x}}(\mathcal{G}_\varphi)$. Note that indeed, for every pair of tests $\varphi \neq \varphi'$, the corresponding super-LDFs must be equal $\mathcal{G}_\varphi = \mathcal{G}_{\varphi'}$ (denote them by $\mathcal{G}$). This follows because they are point-wise equal $\pi_{\boldsymbol{x}}(\mathcal{G}_\varphi) = m(\boldsymbol{x}) = \pi_{\boldsymbol{x}}(\mathcal{G}_{\varphi'})$, and so the difference super-LDF $\mathcal{G}_\varphi - \mathcal{G}_{\varphi'}$ is trivial on every point, and must therefore (again, by proposition 2 – low-ambiguity) be trivial.

If $\mathcal{A}$ is not trivial, then there is at least one test $\psi \in \Psi$ for which $\mathcal{A}(\psi) \neq \vec{\mathbf{0}}$. Thus, by lemma 5, $\mathcal{G} = \mathcal{G}_\varphi \neq \vec{\mathbf{0}}$ (where ψ corresponds to a leaf in $\mathbf{T}_\varphi$). Take an LDF f that appears in $\mathcal{G}$, and define for every $v \in V_\Phi$, $\mathcal{A}(v) \stackrel{def}{=} f(\boldsymbol{x})$ where $\boldsymbol{x} \in \mathcal{H}^{d_0}$ is the point mapped to v. Since $\mathcal{G}$ is legal, Φ is totally satisfied by $\mathcal{A}$.

4 From $\mathcal{SSAT}_\infty$ to SVP_∞

In this section we show the reduction from $\mathcal{SSAT}_\infty$ to SVP_∞, thereby implying SVP_∞ to be NP-hard to approximate to within the same factor $n^{1/\log\log n}$ as $\mathcal{SSAT}_\infty$. This reduction follows the same lines of the reduction in [ABSS93] from Pseudo-Label-Cover to SVP_∞.

Let Ψ be an instance of $\mathcal{SSAT}_\infty$, we construct a matrix B whose columns form the basis for an SVP_∞ instance. Recall that $\Psi = \{\psi_1, \ldots, \psi_n\}$ is a set of tests over variables $V = \{v_1, .., v_m\}$, and we denoted by $\mathcal{R}_{\psi_i}$ the set of satisfying assignments for $\psi_i \in \Psi$.

The matrix B will have a column $\vec{v}_{[\psi,r]}$ for every pair of test $\psi \in \Psi$ and an assignment $r \in \mathcal{R}_\psi$ for it. There will be one additional special column $\vec{t}$. The matrix B will have two kinds of rows, consistency rows and norm-measuring rows, defined as follows.

Consistency Rows. B will have $|\mathcal{F}|$ rows for each threesome $(\psi_i, \psi_j, \boldsymbol{x})$ where $\boldsymbol{x}$ is a variable, and both ψ_i and ψ_j depend on $\boldsymbol{x}$. Only the columns $\vec{v}_{[\psi_i,\cdot]}$ of ψ_i and $\vec{v}_{[\psi_j,\cdot]}$ of ψ_j will have non-zero values in these rows.

The special column $\vec{t}$ will have g in each consistency row, and zero in the other rows.

For a pair of tests ψ_i and ψ_j that depend on a mutual variable $\boldsymbol{x}$, let's concentrate on the sub-matrix consisting of the columns of these tests, and the $|\mathcal{F}|$ rows of the threesome $(\psi_i, \psi_j, \boldsymbol{x})$. This is a pair of matrices G_1 of dimension $(|\mathcal{F}| \times |\mathcal{R}_{\psi_i}|)$ and G_2 of dimension $(|\mathcal{F}| \times |\mathcal{R}_{\psi_j}|)$. Let $r \in \mathcal{R}_{\psi_i}$ be a satisfying assignment for ψ_i and $r' \in \mathcal{R}_{\psi_j}$ be a satisfying assignment for ψ_j. The r-th column in G_1 equals g times the unit vector e_i where $i = r|_{\boldsymbol{x}}$ (i.e. a vector with zeros everywhere and a g in the $r|_{\boldsymbol{x}}$-th coordinate). The r'-th column in G_2 equals $g \cdot (\vec{1} - e_i)$ where $i = r'|_{\boldsymbol{x}}$ and $\vec{1}$ is the all-one vector (i.e. g everywhere except a zero in the $r'|_{\boldsymbol{x}}$-th coordinate).

Norm-measuring Rows. There will be a set of $\mathcal{R}_\psi$ rows designated to each test $\psi \in \Psi$, in which only *psi*'s columns have non-zero values. The columns of ψ, when restricted to these rows, will be the $(|\mathcal{R}_\psi| \times |\mathcal{R}_\psi|)$ Hadamard matrix $\mathbf{H}$ (we assume w.l.o.g. that $|\mathcal{R}_\psi|$ is a power of 2, thus such a matrix exists, see [Bol86, p. 74]). The vector $\vec{t}$, as mentioned earlier, will be zero on these rows.

Proposition 5 (Completeness) *If there is a natural assignment to Ψ, then there is a non-zero lattice vector $\vec{v} \in \mathcal{L}(B)$ with $\|\vec{v}\|_\infty = 1$.*

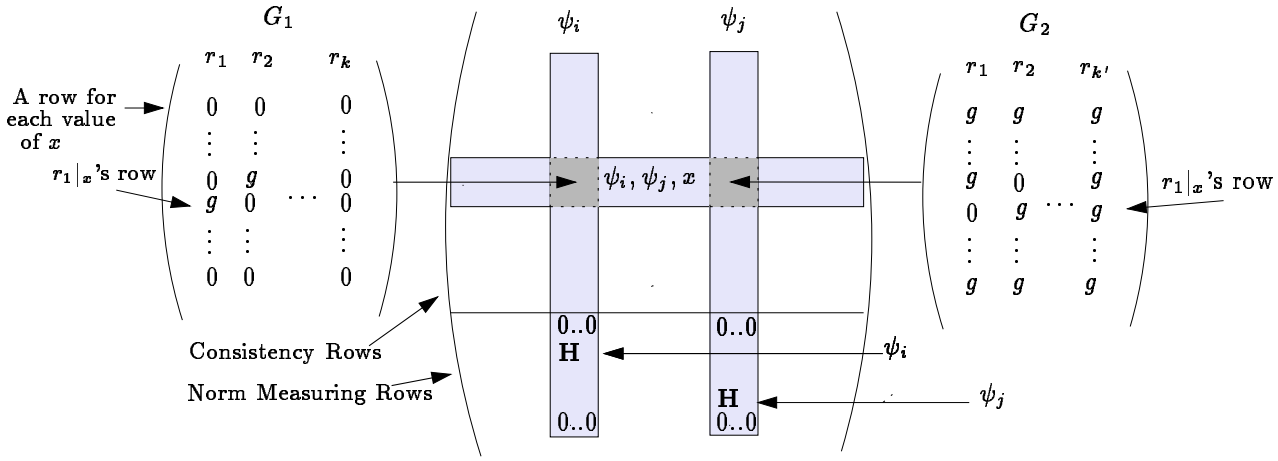

Figure 1: The matrix B

Proof: Let $\mathcal{A}$ be a consistent natural assignment for Ψ. We claim that

$$\vec{v} = \vec{t} - \sum_{\psi \in \Psi} \vec{v}_{[\psi, \mathcal{A}(\psi)]}$$

is a lattice vector with $\|\vec{v}\|_\infty = 1$. Restricting $\sum_{\psi \in \Psi} \vec{v}_{[\psi, \mathcal{A}(\psi)]}$ to an arbitrary row in the consistency rows (corresponding to a pair of tests ψ_i, ψ_j with mutual variable x), gives g, because $\mathcal{A}(\psi_i)|_x = \mathcal{A}(\psi_j)|_x$. Subtracting this from $\vec{t}$ gives zero in each consistency-row.

In the norm-measuring rows, since every test $\psi \in \Psi$ is assigned one value by $\mathcal{A}$, $\vec{v}$ restricted to ψ's rows equals some column of the Hadamard matrix which is a ± 1 matrix. Altogether, $\|\vec{v}\|_\infty = 1$ as claimed. ■

Proposition 6 (Soundness) *If there is a non-zero lattice vector $\vec{v} \in \mathcal{L}(B)$ with $\|\vec{v}\|_\infty < g$, then there is a consistent non-trivial super-assignment for Ψ, whose norm is less than g.*

Proof: Let

$$\vec{v} = c_t \cdot \vec{t} + \sum_{\psi, r} c_{[\psi, r]} \cdot \vec{v}_{[\psi, r]}$$

be a lattice vector with $\|\vec{v}\|_\infty < g$. The entries in the consistency rows of every lattice vector, are integer multiples of g. $\|\vec{v}\|_\infty < g$ implies that v is zero on these rows.

We next show that for every $\psi_i, \psi_j \in \Psi$ with mutual variable x, if $r|_x = r'|_x$ then $c_{[\psi_i, r]} = c_{[\psi_j, r']}$. This follows by restricting our attention to the rows of the threesome (ψ_i, ψ_j, x), and noticing that any zero linear combination of the vectors $\left\{e_i,\ \vec{1} - e_i,\ \vec{1}\right\}_i$ must give e_i the same coefficient as $\vec{1} - e_i$, because the vectors $\{e_i\}_i$ are linearly independent.

Define a super-assignment to Ψ by setting for each $\psi \in \Psi$ and $r \in \mathcal{R}_\psi$, $\mathcal{A}(\psi)[r] \stackrel{def}{=} c_{[\psi, r]}$. $\mathcal{A}$ is consistent by the above and not-all-zero because $\vec{v}$ is non-trivial (if only c_t was non-zero, then $\|\vec{v}\|_\infty = g$).

The norm of $\mathcal{A}$ is defined as

$$\|\mathcal{A}\|_\infty = \max_{\psi \in \Psi} (\|\mathcal{A}(\psi)\|_2) \ .$$

The vector $\vec{v}$ restricted to the norm-measuring rows of ψ is exactly $\mathbf{H}\,\mathcal{A}(\psi)$. Now $\|\frac{1}{\sqrt{|\mathcal{R}_\psi|}}\mathbf{H}\,\mathcal{A}(\psi)\|_2 = \|\mathcal{A}(\psi)\|_2$ because $\frac{1}{\sqrt{|\mathcal{R}_\psi|}}\mathbf{H}$ is a $(|\mathcal{R}_\psi| \times |\mathcal{R}_\psi|)$ orthonormal matrix. Since for every $z \in \mathbb{R}^n$, $\|z\|_\infty \geq \|z\|_2/\sqrt{n}$, we obtain $\|\mathbf{H}\mathcal{A}(\psi)\|_\infty \geq \|\mathcal{A}(\psi)\|_2$. Hence $\|\mathcal{A}\|_\infty \leq \|v\|_\infty < g$ as claimed. ■

Finally, if Ψ is a $\mathcal{SSAT}_\infty$ no instance, then the norm of any consistent super-assignment $\mathcal{A}$ must be at least g, and so the norm of the shortest lattice vector in $\mathcal{L}(B)$, must also be at least g. This completes the proof of the reduction.

The reduction to CVP_∞ is quite obvious, and is omitted.

References

[ABSS93] S. Arora, L. Babai, J. Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes and linear equations. In *Proc. 34th IEEE Symp. on Foundations of Computer Science*, pages 724–733, 1993.

[AD97] Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 284–293, El Paso, Texas, 4–6 May 1997.

[Ajt96] M. Ajtai. Generating hard instances of lattice problems. In *Proc. 28th ACM Symp. on Theory of Computing*, pages 99–108, 1996.

[Ajt97] M. Ajtai. The shortest vector problem in l_2 is NP-hard for randomized reductions. manuscript, May 1997.

[Bab86] L. Babai. On Lovász's lattice reduction and the nearest lattice point problem. *Combinatorica*, 6:1–14, 1986.

[Bol86] B. Bollobás. *Combinatorics*. Cambridge University Press, 1986.

[CN98] J.Y. Cai and A. Nerurkar. Approximating the SVP to within a factor $(1 + 1/\dim^{\varepsilon})$ is NP-hard under randomized reductions. In *Proc. of the 13th Annual IEEE Conference on Computational Complexity*, pages 46–55. 1998.

[Coo71] S. Cook. The complexity of theorem-proving procedures. In *Proc. 3rd ACM Symp. on Theory of Computing*, pages 151–158, 1971.

[DFK+] I. Dinur, E. Fischer, G. Kindler, R. Raz, and S. Safra. PCP characterizations of NP: Towards a polynomially-small error-probability. Manuscript, to appear in STOC99.

[DKRS99] I. Dinur, G. Kindler, R. Raz, and S. Safra. Approximating-CVP to within almost-polynomial factors is NP-hard. Manuscript, 1999.

[DKS98] I. Dinur, G. Kindler, and S. Safra. Approximating-CVP to within almost-polynomial factors is NP-hard. In *Proc. 39th IEEE Symp. on Foundations of Computer Science*, 1998.

[FT85] András Frank and Éva Tardos. An application of simultaneous approximation in combinatorial optimization. In *26th Annual Symposium on Foundations of Computer Science*, pages 459–463, Portland, Oregon, 21–23 October 1985. IEEE.

[GG98] O. Goldreich and S. Goldwasser. On the limits of non-approximability of lattice problems. In *Proc. 30th ACM Symp. on Theory of Computing*, pages 1–9, 1998.

[LLL82] A.K. Lenstra, H.W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261:513–534, 1982.

[LO85] J. C. Lagarias and A. M. Odlyzko. Solving low-density subset sum problems. *Journal of the ACM*, 32(1):229–246, January 1985.

[Mic98] D. Micciancio. The shortest vector in a lattice is hard to approximate to within some constant. In *Proc. 39th IEEE Symp. on Foundations of Computer Science*, 1998.

[Sch85] C.P. Schnorr. A hierarchy of polynomial-time basis reduction algorithms. In *Proceedings of Conference on Algorithms, Pécs (Hungary)*, pages 375–386. North-Holland, 1985.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'