Lattice Problems in $\mathsf{NP} \cap \mathsf{coNP}$

Dorit Aharonov *   Oded Regev †

July 17, 2004

Abstract

We show that the problems of approximating the shortest and closest vector in a lattice to within a factor of $\sqrt{n}$ lie in NP intersect coNP. The result (almost) subsumes the three mutually-incomparable previous results regarding these lattice problems: Banaszczyk [7], Goldreich and Goldwasser [13], and Aharonov and Regev [2]. Our technique is based on a simple fact regarding succinct approximation of functions using their Fourier transform over the lattice. This technique might be useful elsewhere – we demonstrate this by giving a simple and efficient algorithm for one other lattice problem (CVPP) improving on a previous result of Regev [25]. An interesting fact is that our result emerged from a "dequantization" of our previous quantum result in [2]. This route to proving purely classical results might be beneficial elsewhere.

*School of Computer Science and Engineering, The Hebrew University, Jerusalem, Israel. doria@cs.huji.ac.il. Research supported by ISF grant 032-9738.

†Department of Computer Science, Tel-Aviv University, Tel-Aviv 69978, Israel. Work supported by an Alon Fellowship and the Army Research Office grant DAAD19-03-1-0082.

1 Introduction

A lattice is the set of all integer combinations of n linearly independent vectors $v_1, \ldots, v_n$ in $\mathbb{R}^n$. These vectors are known as a *basis* of the lattice. The study of lattices originated some 200 years ago by Gauss [12], who gave an algorithm to find the shortest vector in a two-dimensional lattice. Since then, lattices have been shown to be pervasive in mathematics, and many different problems can be phrased as questions about lattices, such as integer programming [17], factoring polynomials with rational coefficients [22], integer relation finding [15], integer factoring and Diophantine approximation [27]. Recently, the study of lattices gained a lot of attention in the computer science community due to the fact that lattice problems were shown by Ajtai [3] to possess a particularly desirable property for cryptography: worst case to average case reducibility.

Two lattice problems have been widely studied. The first is the Shortest Vector Problem (SVP): given a basis $v_1, \ldots, v_n$ of a lattice, find the shortest nonzero lattice point in the Euclidean norm. The second is the Closest Vector Problem (CVP): given a basis $v_1, \ldots, v_n$ of a lattice and a target vector $v \in \mathbb{R}^n$ find the closest lattice point to v in the Euclidean norm. Both problems are known to be NP-complete [4, 28]. In light of this, and the importance of lattice problems in mathematics, a very interesting question is the study of the approximation version of these problems. The parameter of interest here is the factor of approximation β. The problem GapSVP_β is the following: Given a basis $v_1, \ldots, v_n$, decide whether the l_2 norm of the shortest nonzero vector in the lattice is smaller than 1 or larger than β. The problem GapCVP_β is: Given a basis $v_1, \ldots, v_n$ and an extra vector $v \in \mathbb{R}^n$, decide whether the distance of v from the lattice is smaller than 1 or larger than β. The best inapproximability result for CVP is due to Dinur et al. [9] where it is shown that approximating CVP to within $n^{c/\log\log n}$ is NP-hard for some $c > 0$. For SVP, Khot [19] recently showed that for any $\varepsilon > 0$ obtaining approximation factors below $2^{(\log n)^{1/2-\varepsilon}}$ is hard unless $\mathsf{NP} \subseteq \mathsf{BPTIME}(2^{\mathrm{poly}(\log n)})$; this improves on a previous result of Micciancio [23]. The best probabilistic polynomial time approximation algorithm due to Ajtai et al. [6] obtains a $2^{O(n\log\log n/\log n)}$-approximation factor for both problems; it is based on the deterministic polynomial time $2^{O(n(\log\log n)^2/\log n)}$-approximation algorithm by Schnorr [26].

The complexity of lattice problems in the range of polynomial approximation factors is of particular interest. For example, Ajtai's seminal work [3] is based on the hardness of approximation in this region (see also [5, 24]). A sequence of incomparable results gave upper bounds on the complexity of lattice problems in the polynomial approximation region. Banaszczyk [7] showed that GapCVP_n is in $\mathsf{NP} \cap \mathsf{coNP}$, improving on the previous result of $\mathsf{GapCVP}_{n^{1.5}} \in \mathsf{NP} \cap \mathsf{coNP}$ by Lagarias, Lenstra and Schnorr [21]. We note that containment in NP is trivial, and the difficult part is showing the containment in coNP, i.e., showing the existence of a succinct proof that a vector is far from any lattice point. Goldreich and Goldwasser [13] gave an upper bound on the complexity of the harder problem $\mathsf{GapCVP}_{\sqrt{n}}$, but their upper bound is weaker: they showed containment in $\mathsf{NP} \cap \mathsf{coAM}$, which means that instead of showing the existence of a succinct proof that a vector is far from any lattice point, they gave an interactive proof of two rounds to that effect. Recently, the current authors showed [2] that there exists a succinct quantum proof (namely a superposition on polynomially many quantum bits) to the fact that a vector is far from the lattice. This result holds only with additional constraints on the lattice.

In this paper we prove a result that essentially subsumes all three results mentioned above. We prove:

Theorem 1.1 *There exists $c > 0$ such that $\mathsf{GapCVP}_{c\sqrt{n}}$ is in $\mathsf{NP} \cap \mathsf{coNP}$.*

Of the three results, the only result that Theorem 1.1 does not completely subsume is the result of Goldreich and Goldwasser [13] since there is an area of gaps, between $\sqrt{n/\log n}$ and $\sqrt{n}$ for which our result does not apply, and so containment in $\mathsf{NP} \cap \mathsf{coNP}$ is not known to hold.

There is a known approximation preserving reduction from GapSVP to GapCVP [14], which we include for completeness in Appendix A. Using this reduction, we obtain:

Corollary 1.2 *There exists $c > 0$ such that $\mathsf{GapSVP}_{c\sqrt{n}}$ is in $\mathsf{NP} \cap \mathsf{coNP}$.*

We summarize the current complexity of lattice problems as a function of the approximation ratio β in Figure 1.

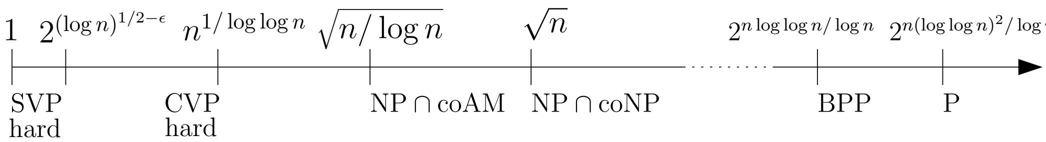

Figure 1: The complexity of lattice problems (some constants omitted)

1.1 Proof Overview

As mentioned before, the containment in NP is trivial and it suffices to prove that $\mathsf{GapCVP}_{\sqrt{n}}$ is in coNP. To show this we construct an NP verifier that given a polynomial witness, verifies that v is *far* from the lattice. There are three steps to this proof.

1. **Define** f

   In this part we define a function $f : \mathbb{R}^n \mapsto \mathbb{R}^+$ that is periodic over the lattice L, i.e., for all $x \in \mathbb{R}^n$ and $y \in L$ we have $f(x) = f(x + y)$. For any lattice L, the function f satisfies the following two properties: it is non-negligible (i.e., larger than some $1/\text{poly}(n)$) for any point that lies within distance $\sqrt{\log n}$ from a lattice point, and is exponentially small at distance $\geq \sqrt{n}$ from the lattice. Note that $f(v)$ indicates whether v is far or close to the lattice.

2. **Encode** f

   We show that there exists a succinct description (which we denote by W) of a function f_W that approximates f at *any* point in $\mathbb{R}^n$ to within polynomially small additive error. We now use W as the witness in the NP proof.

3. **Verify** f

   We construct an efficient NP verifier that, given a witness W, verifies that v is *far* from the lattice. The verifier verifies first that $f_W(v)$ is small, and also that $f_W(x) \geq 1/2$ for any x which is close to the lattice.

Step 1 The function f already appeared in [7], and in fact, the two properties mentioned in Step 1 were already proven there. The function is defined as a sum of Gaussians centered around each lattice point.

Step 2 This is the main innovative step: the function f, defined on exponentially many points (if we consider some finite precision) can be approximated pointwise, to a polynomially small additive precision, by a polynomial size circuit. We show this by using a useful property of f: its Fourier series $\hat{f}$ is a probability measure over the dual lattice (defined as the set of all points in $\mathbb{R}^n$ with integer inner product with all lattice points). In other words, its Fourier series is non-negative and the sum of all of its entries is 1. This allows us to view f as an expectation of a random variable, and so by the Hoeffding bound (a generalization of the Chernoff bound), polynomially many samples from the distribution on the dual lattice given by $\hat{f}$ would suffice. This leads us to the following lemma. We will later define ℓ as some polynomial in n and L_ℓ as a very fine grid in $\mathbb{R}^n$. For now, one can think of the lemma as applying to any $x \in \mathbb{R}^n$ and not only to $x \in L_\ell$.

Lemma 1.3 (The Pointwise Approximation Lemma) *Let L be an n-dimensional lattice, and let f be a function from $\mathbb{R}^n$ to $\mathbb{R}$ that is periodic over L and whose Fourier series $\hat{f}$ is a probability measure over the* *dual lattice L^*. For any constant $c > 0$ define N to be $n^{2c+2}\ell$. Let $w_1, \ldots, w_N$ be vectors in the dual lattice chosen randomly and independently from the distribution $\hat{f}$. Then with probability at least $3/4$,*

$$f_W(x) \stackrel{def}{=} \frac{1}{N}\sum_{i=1}^{N}\cos(2\pi\langle x, w_i\rangle) \tag{1}$$

satisfies that $|f_W(x) - f(x)| \le n^{-c}$ for all $x \in L_\ell$.

Remark: For future applications, we note that the lemma can be generalized in two ways. First, the requirement that the Fourier transform is a probability measure can be relaxed – we need only require that the sum of the absolute values of the Fourier coefficients of f is polynomially bounded. Moreover, the lemma can be extended to work over any Abelian group, where the average is taken over randomly chosen characters of the group. The proofs of both extensions are straightforward generalizations of our proof.

We note that a related lemma regarding approximation of functions using the Fourier transform was proven in a paper by Kushilevitz and Mansour [20], and was used in various other contexts. There, however, the notion of approximation in l_2 is used, while here we need the notion of approximation in l_∞, i.e., pointwise. A good approximation in l_∞ cannot be achieved by the technique of [20], which is based on truncating small Fourier coefficients.

Given this lemma, it is natural to define the witness as the list $w_1, \ldots, w_N$ of vectors in the dual lattice; this list is also referred to as W.

Step 3 Here we construct an efficient NP verifier that, given W, verifies that a point is *far* from the lattice. Given a lattice L and a vector v, it accepts if the distance of v from L is greater than $\sqrt{n}$ and rejects if this distance is less than $1/100$. This will show that $\mathsf{GapCVP}_{100\sqrt{n}}$ is in coNP (after appropriate rescaling).

The verifier starts by computing $f_W(v)$, as defined in Equation 1. The verifier rejects if he sees that $f_W(v)$ is at least, say, $1/2$. We can do this since in case the distance of v from L is greater than $\sqrt{n}$ we know that $f(v)$ is exponentially small and hence $f_W(v)$ must be at most $1/\text{poly}(n) < 1/2$ (assuming the witness W is chosen from $\hat{f}$ as it should be).

This verifier, however, is clearly not strong enough: the prover can 'cheat' by sending w_i's that have nothing to do with $\hat{f}$ and for which $f_W(v)$ is small even though v is within distance $1/100$ of the lattice. One might try to avoid such cheating by verifying that f_W is close to f everywhere, or, equivalently, that the w_i's were indeed chosen from the correct distribution $\hat{f}$. We do not know how to construct such a verifier. Instead, we verify that the w_i's 'look like' vectors chosen from $\hat{f}$, according to some simple statistical tests. We will later see that these tests suffice.

But how do vectors chosen from $\hat{f}$ look like? We identify two important properties. First, by definition we see that all the w_i's are in L^*. Second, we show that with high probability, for any unit vector $u \in \mathbb{R}^n$ it holds that the average over i of the square of the projection of w_i on u is at most 10. Intuitively, this follows from the fact that the length of the w_i's is roughly $\sqrt{n}$ and that they are not concentrated in any particular direction. The proof uses another lemma due to Banaszczyk [7].

It turns out that the verifier can check these two properties efficiently. The first property is easy to check by, say, solving linear equations. But how can we check the second property efficiently? It seems that we have to check it for all vectors u. It turns out (and this is not too difficult to see) that we can equivalently check that the largest eigenvalue of the $n \times n$ matrix $W \cdot W^T$, where W is the $n \times N$ matrix whose columns are the vectors $w_1, \ldots, w_N$, is at most $10N$. Computing the eigenvalues of this matrix can be done in polynomial time.

We now claim that any witness $w_1, \ldots, w_N$ that passes the above tests is good enough; namely, it satisfies that $f_W(x) \ge 1/2$ for all points x within distance $1/100$ from the lattice. By the definition of f_W we see that the first test guarantees that the function f_W is periodic on L. Hence it is enough to show that $f_W(x) \ge 1/2$

for any x satisfying $\|x\| \leq 1/100$. For such x, the second test implies that for most i's, $|\langle x, w_i \rangle|$ is small. Therefore, for such x most of the cos's in the definition of $f_W(x)$ are close to 1. This implies that $f_W(x)$ is greater than $1/2$ and soundness follows.

1.2 Another Application: The Closest Vector Problem with Preprocessing

Steps 1 and 2 imply that important information regarding the lattice can be encoded in a short description, though this description may be very hard to find. Note that this description is independent of the target vector v. Hence, if we had infinite time to preprocess the lattice before seeing the vector v, we could prepare the approximating function f_W and then calculate $f_W(v)$ when given v in polynomial time. This is exactly the setting in the Closest Vector Problem with Preprocessing (CVPP). The problem is defined as follows: given a lattice, we are allowed to preprocess it and to output a polynomially long description, without any computational restrictions on the preprocessing phase. Then, given a preprocessed lattice and a query point $v \in \mathbb{R}^n$, the algorithm is supposed to approximate the distance of v from the lattice. The motivation for this problem comes from cryptography and coding theory. See [10] for a more precise definition and a further discussion and references. The best known inapproximability result is that CVPP is NP-hard to approximate to within a factor of $\sqrt{3}$ [25], and the best polynomial time approximation algorithm is for a factor n [25]. Steps 1 and 2 in our proof immediately imply an efficient $\sqrt{n/\log n}$ approximation algorithm for CVPP:

Theorem 1.4 *For any constant $c > 0$, the problem $\mathsf{GapCVPP}_{c\sqrt{n/\log n}}$ can be solved in polynomial time.*

Note that by using standard methods, a solution to a gap problem can be converted to a solution to the corresponding approximation problem. Hence, the above theorem implies that for any constant $c > 0$ there exists a $c\sqrt{n/\log n}$ approximation algorithm for CVPP.

1.3 Speculation

Note that Step 3 is not the best that one can hope for: the function f has the property that it is non-negligible in the $\sqrt{\log n}$ vicinity of lattice points. Yet, we are only able to verify that the given function f_W is non-negligible in a *constant* distance. It is possible that the verification procedure can be improved so that it includes the $\sqrt{\log n}$ vicinity of lattice points. This would imply the following speculation:

Speculation 1.5 *$\mathsf{GapCVP}_{\sqrt{n/\log n}}$ is in $\mathsf{NP} \cap \mathsf{coNP}$.*

Recall that this problem is currently known to be in $\mathsf{NP} \cap \mathsf{coAM}$ [13]. The factor $\sqrt{n/\log n}$ arises naturally in both our work (from properties of Gaussians) and in [13] (from properties of intersections of high dimensional spheres). We note that going below $\sqrt{n/\log n}$ would probably require some substantially new ideas, and in fact, might be impossible; it may be the case that this is where the NP-hardness is manifested.

1.4 Relation to Quantum Computation

It is intriguing to note that our result emerged from a "dequantization" of a quantum result [2], in which we showed that $\mathsf{coGapSVP}_{\sqrt{n}}$ is contained in the quantum analogue of the class NP, called QMA, in which both witness and verifier are quantum. In the dequantization process we replaced both witness and verifier by classical objects. This result thus continues an existing thread of quantum-inspired purely-classical results (e.g., [18, 1]). We would like to emphasize, however, that the proof we present in the present paper is completely classical, and bares little resemblance to the original quantum proof. In fact, the new proof is stronger and holds not only for SVP but also for CVP.

1.5 Organization

The rest of the paper is organized as follows. Section 2 gives the basic notations and definitions. In Section 3 we define f and prove its required properties. In Section 4 we prove the pointwise approximation lemma, show that f satisfies the conditions of the lemma, and deduce that there exists a polynomial size circuit that approximates f. In Section 5 we show how the previous two sections imply an improved algorithm for CVPP. In Section 6 we complete the proof of the main theorem. For the sake of completeness, we add two known results in the appendices: Appendix A gives the reduction from GapSVP_β to GapCVP_β, whereas Appendix B shows why our results (as well as previous results) imply that the lattice problems we are considering are unlikely to be NP-hard.

2 Preliminaries

2.1 Lattices

For an introduction to lattices, see [24]. A lattice in $\mathbb{R}^n$ is defined as the set of all integer combinations of n linearly independent vectors. This set of vectors is known as a basis of the lattice and is not unique. Given a basis $(v_1, \ldots, v_n)$ of a lattice L, the fundamental parallelepiped is defined as

$$\mathcal{P}(v_1, \ldots, v_n) = \left\{ \sum_{i=1}^{n} x_i v_i \mid x_i \in [0,1) \right\}.$$

Note that a lattice has a different fundamental parallelepiped for each possible basis. However, everything we do is independent of the basis, and so we will use the notation $\mathcal{P}(L)$ instead of $\mathcal{P}(v_1, \ldots, v_n)$. We denote by $\det(L)$ the volume of the fundamental parallelepiped of L or equivalently, the determinant of the matrix whose columns are the basis vectors of the lattice (again, this is independent of the basis). For a point $x \in \mathbb{R}^n$ we define $d(x, L)$ as the minimum of $\|x - y\|$ over all $y \in L$.

For any n-dimensional lattice L, the dual lattice of L, denoted L^*, is an n-dimensional lattice defined as the set of all points in $\mathbb{R}^n$ with integer inner products with all lattice points:

$$L^* = \{y \mid y \in \mathbb{R}^n,\ \forall x \in L. \langle x, y \rangle \in \mathbb{Z}\}.$$

2.2 Shortest and Closest Vector in a Lattice

The shortest (non-zero) vector of L is the vector $x \in L$, such that $\|x\| \neq 0$ and is minimal. The following is the gap version of the shortest vector problem:

Definition 2.1 ($\mathsf{coGapSVP}$) *For any gap parameter $\beta = \beta(n)$ the promise problem $\mathsf{coGapSVP}_\beta$ is defined as follows. The input is a basis for a lattice L. It is a* YES *instance if the length of the shortest vector is more than β. It is a* NO *instance if the length of the shortest vector is at most* 1.

We also define the gap version of the closest vector problem.

Definition 2.2 ($\mathsf{coGapCVP}$) *For any gap parameter $\beta = \beta(n)$ the promise problem $\mathsf{coGapCVP}_\beta$ is defined as follows. The input is a basis for a lattice L and a vector v. It is a* YES *instance if $d(v, L) > \beta$. It is a* NO *instance if $d(v, L) \leq 1$.*

Notice that we can replace the values β and 1 by, say, $\beta/100$ and $1/100$ respectively without really affecting the complexity of the problems. This follows from an easy reduction that simply rescales the input by a factor of 100.

2.3 Precision Issues

Each vector in the input basis $v_1, \ldots, v_n$ is always given with polynomially many bits. Without loss of generality, we assume that the target vector v is given to us in the form $\sum a_i v_i$ where each $0 \le a_i < 1$ is represented by at most $\ell = \mathrm{poly}(n)$ bits. To this end we define, for a given lattice L, a refined lattice $L_\ell = L/2^\ell$. In other words, L_ℓ is given by all integer combinations of the basis vectors $\frac{1}{2^\ell}v_1, \ldots, \frac{1}{2^\ell}v_n$. Notice that we have $v \in L_\ell$.

2.4 Fourier Series of Periodic Functions

A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is said to be *periodic over a lattice* L if $f(x) = f(x+y)$ holds for all $x \in \mathbb{R}^n$ and for all $y \in L$. Let f be such a periodic function. Then define the Fourier coefficient of f at $w \in L^*$, denoted by $\hat{f}(w)$, to be

$$\hat{f}(w) = \frac{1}{\det(L)} \int_{z \in \mathcal{P}(L)} e^{-2\pi i \langle w,z \rangle} f(z) dz.$$

Note that the above definition is independent of the basis due to the definition of the parallelepiped, and the periodicity of $e^{-i\langle w,z\rangle} f(z)$ over the lattice. The following fact is well known from standard Fourier analysis:

Fact 2.3 *If $\sum_{w \in L^*} |\hat{f}(w)|$ converges, then for all $x \in \mathbb{R}^n$,*

$$f(x) = \sum_{w \in L^*} \hat{f}(w) e^{2\pi i \langle w,x \rangle}.$$

This sum is known as the Fourier series of f evaluated at x.

2.5 The Gaussian

The Fourier transform of a function $h : \mathbb{R}^n \mapsto \mathbb{R}$ is defined to be

$$\hat{h}(w) = \int_{\mathbb{R}^n} h(x) e^{-2\pi i \langle x,w \rangle} dx.$$

If $h : \mathbb{R}^n \mapsto \mathbb{R}$ is a Gaussian,

$$h(x) = e^{-\pi \|x\|^2},$$

then its Fourier transform turns out to be of a very nice form – it is exactly another Gaussian:

$$\hat{h}(w) = e^{-\pi \|w\|^2}.$$

2.6 Some Useful Lemmas

The following technical claim shows that all the sums that we use are well defined.

Claim 2.4 *For any n-dimensional lattice L and for any $x \in \mathbb{R}^n$, the sum $\sum_{y \in L} e^{-\pi\|x-y\|^2}$ is finite.*

Proof: Notice that

$$1 = \int_{y \in \mathbb{R}^n} e^{-\pi\|x-y\|^2} = \lim_{m \mapsto \infty} \left(\sum_{y \in L/m} e^{-\pi\|x-y\|^2} \right) \det(L/m)$$

where L/m denotes the lattice scaled down by a factor m. Hence, there exists m_0 such that

$$2 \ge \left(\sum_{y \in L/m_0} e^{-\pi\|x-y\|^2} \right) \det(L/m_0).$$

Hence, $\sum_{y \in L} e^{-\pi\|x-y\|^2} \le \sum_{y \in L/m_0} e^{-\pi\|x-y\|^2}$ is finite. ■

We now quote two lemmas due to Banaszczyk [7] that we use throughout the proof.

Lemma 2.5 *(Lemma 1.5 in [7]) For any n-dimensional lattice L, $x \in \mathbb{R}^n$ and $c > \frac{1}{\sqrt{2\pi}}$, one has*

$$\frac{\sum_{y\in L, \|x-y\|>c\sqrt{n}} e^{-\pi\|x-y\|^2}}{\sum_{y\in L} e^{-\pi\|y\|^2}} \le 2(c\sqrt{2\pi e}\cdot e^{-\pi c^2})^n = 2^{-\Omega(n)}.$$

This lemma was used in [7] to show several tight connections between a lattice and its dual (these are known as 'transference theorems'). Its proof is non-trivial; for another proof, see Štefankovič's thesis [29].

Lemma 2.6 *(Lemma 1.3 in [7]) For any n-dimensional lattice L and any unit vector $u \in \mathbb{R}^n$ we have*

$$\frac{\sum_{y\in L} \langle y,u\rangle^2 e^{-\pi\|y\|^2}}{\sum_{y\in L} e^{-\pi\|y\|^2}} \le \frac{1}{2\pi}.$$

To get some intuition on this bound, let us mention that we can get arbitrarily close to $\frac{1}{2\pi}$ by choosing L to be a very dense lattice. In fact, it can be easily checked that we obtain an equality if we replace the above sum with an integral.

2.7 The Hoeffding Bound

We will use Hoeffding's inequality [16] which states the following. Let $X_1, \ldots, X_N$ be N identically distributed independent random variables, such that for all i, $X_i \in [a,b]$. Then $S_N = \sum_i X_i$ satisfies that

$$\Pr(|S_N - \mathrm{E}(S_N)| \ge N\varepsilon) \le 2e^{-N\varepsilon^2/(b-a)^2}. \tag{2}$$

3 Define f

We define the function $g : \mathbb{R}^n \to \mathbb{R}$ as

$$g(x) = \sum_{y\in L} e^{-\pi\|x-y\|^2}.$$

This sum is finite by Claim 2.4. We also define

$$f(x) = \frac{g(x)}{g(0)}.$$

The following lemmas show that the value of f indicates the distance from the lattice.

Lemma 3.1 *Let $c > \frac{1}{\sqrt{2\pi}}$ be any constant. Then for any $x \in \mathbb{R}^n$, if $d(x,L) \ge c\sqrt{n}$ then $f(x) \le 2^{-\Omega(n)}$.*

Proof: The proof follows trivially from Lemma 2.5. ■

Lemma 3.2 *Let $c > 0$ be any constant. Then for any $x \in \mathbb{R}^n$, if $d(x,L) \le c\sqrt{\log n}$ then $f(x) > n^{-10c^2}$.*

Proof: Notice that because of the periodicity of f over the lattice, it is sufficient to prove that if $\|x\| \le c\sqrt{\log n}$ then $f(x) > n^{-10c^2}$. We can therefore complete the proof by showing that for any $x \in \mathbb{R}^n$, $f(x) \ge e^{-\pi\|x\|^2}$. This is done as follows:

$$\begin{aligned} g(x) &= \sum_{y\in L} e^{-\pi\|x-y\|^2} = \frac{1}{2}\sum_{y\in L}[e^{-\pi\|x-y\|^2} + e^{-\pi\|x+y\|^2}] \\ &= e^{-\pi\|x\|^2}\sum_{y\in L} e^{-\pi\|y\|^2}\frac{1}{2}[e^{-2\pi\langle x,y\rangle} + e^{2\pi\langle x,y\rangle}] \\ &\ge e^{-\pi\|x\|^2}\sum_{y\in L} e^{-\pi\|y\|^2} = e^{-\pi\|x\|^2} g(0) \end{aligned}$$

where the last inequality follows from the fact that for any real r, $r + \frac{1}{r} \ge 2$. ■

4 Encode f

Claim 4.1 *The Fourier series of f is given by*

$$\hat{f}(w) = \frac{e^{-\pi\|w\|^2}}{\sum_{z\in L^*} e^{-\pi\|z\|^2}}.$$

Proof: By definition of g and the Fourier series,

$$\hat{g}(w) = \frac{1}{\det(L)} \int_{x\in\mathcal{P}(L)} \left(\sum_{y\in L} e^{-\pi\|x-y\|^2} \right) e^{-2\pi i\langle x,w\rangle} dx$$

for any $w \in L^*$. By the definition of L^*, we have $\langle x, w\rangle = \langle x-y, w\rangle \bmod 1$ for any $y \in L$ and so

$$\begin{aligned} \hat{g}(w) &= \frac{1}{\det(L)} \int_{x\in\mathcal{P}(L)} \left(\sum_{y\in L} e^{-\|x-y\|^2} e^{-2\pi i\langle x-y,w\rangle} \right) dx \\ &= \frac{1}{\det(L)} \int_{z\in\mathbb{R}^n} e^{-\pi\|z\|^2} e^{-2\pi i\langle z,w\rangle} dz. \end{aligned}$$

This is simply the definition of the Fourier transform of a Gaussian divided by $\det(L)$, and hence we have (see Section 2)

$$\hat{g}(w) = \frac{1}{\det(L)} e^{-\pi\|w\|^2}.$$

To derive $\hat{f}(w)$ we have to divide by $g(0)$. By Claim 2.4 we can apply Fact 2.3 and so we have that

$$g(0) = \sum_{w\in L^*} \hat{g}(w)$$

which gives us the desired result. ∎

Corollary 4.2 *The Fourier series of f is a probability measure on the dual lattice (i.e., it is non-negative and the sum over all points in the dual lattice is* 1*).*

We are thus in a situation which satisfies the conditions of Lemma 1.3. It remains to prove the lemma.

Proof of Lemma 1.3: By the conditions of the lemma, the sum of the Fourier series of f converges to 1. Hence we can apply Fact 2.3 and we have:

$$f(x) = \sum_{w\in L^*} \hat{f}(w) e^{2\pi i\langle w,x\rangle} = \sum_{w\in L^*} \hat{f}(w) \cos(2\pi\langle w, x\rangle)$$

where the last equality follows from the fact that both f and $\hat{f}$ are real, and so the imaginary part cancels out. Hence $f(x)$ can be seen as the expectation of $\cos(2\pi\langle w, x\rangle)$ (whose values range between -1 and 1), where w is chosen according to the probability measure $\hat{f}$:

$$f(x) = \mathrm{E}_{w\sim\hat{f}} \cos(2\pi\langle w, x\rangle)$$

Let $x \in \mathbb{R}^n$. By Hoeffding's bound, Equation 2, we have that the probability that the mean of N samples is not within a window of n^{-c} of the correct expectation is $2^{-\Omega(N/n^{2c})}$. We now want to show that this holds simultaneously for all $x \in L_\ell$. Since f is periodic over the lattice, it suffices to consider x in $\mathcal{P}(L) \cap L_\ell$. By definition of L_ℓ, there are exactly $2^{\ell n}$ such points. Hence, by the union bound, the probability that the approximation is within n^{-c} window of the correct expectation at all points in L_ℓ simultaneously is at least $1 - 2^{n\ell} 2^{-\Omega(N/n^{2c})}$. Since $N = n^{2c+2}\ell$ we get exponentially good confidence. ∎

Applying the lemma in our case implies that with high probability, f_W approximates f everywhere in L_ℓ to within polynomial precision. In particular, since $v \in L_\ell$, we have that $f_W(v)$ approximates $f(v)$ to within polynomial precision.

Remark: In fact, the above lemma is stronger than what we need for our main application, namely for the proof of Theorem 1.1. We will only need the lemma to hold for any *given* x, but not necessarily *simultaneously* for all $x \in L_\ell$, and so for our main application the final union bound in the proof is unnecessary. However, for the CVPP application, which follows next, we need the full strength of the above lemma.

5 Interlude: The Closest Vector Problem with Preprocessing

Proof of Theorem 1.4: Given a lattice L, the preprocessing step outputs a polynomially long sequence $w_1, \ldots, w_N$ such that the function f_W defined by them approximates f at any point in L_ℓ to within $\frac{1}{4}n^{-10/c^2}$. Such a sequence exists by Lemma 1.3. Given a vector v and the preprocessed lattice $w_1, \ldots, w_N$, the computation step involves a simple computation of $f_W(v)$. If its value is more than $\frac{1}{2}n^{-10/c^2}$ then we decide that $d(x, L) \le \sqrt{\log n}/c$; otherwise, we decide that $d(x, L) > \sqrt{n}$. The correctness of the algorithm follows from Lemmas 3.1,3.2 and 1.3. ■

6 Verify f_W

In this section we prove Theorem 1.1 by showing that $\mathsf{GapCVP}_{100\sqrt{n}}$ is in coNP. We do this by providing a coNP verifier for a rescaled problem, where the NO instances have distance at most $1/100$ from L, and the YES instances have distance at least $\sqrt{n}$ from L. The witness is a sequence of vectors $w_1, \ldots, w_N$, where N is chosen to be a large enough polynomial in n, say, $N = n^4\ell$. It will be convenient to refer to the witness, equivalently, as an $n \times N$ matrix W whose columns correspond to $w_1, \ldots, w_N$.

The verifier performs three tests and accepts if and only if all of them are satisfied:

(a) Checks that the w_i's are in the dual lattice L^*,

(b) Checks that $f_W(v) < 1/2$,

(c) Checks that the maximal eigenvalue of the $n \times n$ positive semidefinite matrix WW^T is at most $10N$.

It is easy to see that the verifier can be implemented in polynomial time.

6.1 Soundness

Assume that v is a NO instance, i.e., its distance from L is at most $1/100$ and assume that tests $(a), (c)$ accept. We will show that test (b) must reject. First, since test (a) accepts, we have that f_W is periodic over L. Let $\tau(v)$ denote the vector given by v minus the lattice point closest to v. Notice that $\|\tau(v)\| \le 1/100$. Since f_W is periodic on the lattice, $f_W(v) = f_W(\tau(v))$. It thus suffices to prove that $f_W(\tau(v)) \ge 1/2$, or, for that matter, that $f_W(x) \ge 1/2$ for all x in a ball of radius $1/100$ around the origin.

This is done as follows. Let x be such that $\|x\| \le 1/100$. Since test (c) passes, we have that

$$\frac{1}{N}\sum_{j=1}^{N} \langle x, w_j \rangle^2 = \frac{1}{N} x^T W W^T x \le \frac{1}{N}\frac{10N}{10000} = \frac{1}{1000}$$

where the inequality follows by expressing x in the eigenvector basis of WW^T. Hence, by a Markov argument it follows that for at most $1/10$ of the j's, $\langle x, w_j \rangle^2 \ge 1/100$. For the rest of the j's, $|\langle x, w_j \rangle| < 1/10$ which implies that $\cos(2\pi\langle x, w_j\rangle) > \cos(2\pi/10)$. We have that

$$f_W(x) = \frac{1}{N}\sum_{j=1}^{N}\cos(2\pi\langle x, w_j\rangle) > -\frac{1}{10} + \frac{9}{10}\cos(2\pi/10) \geq \frac{1}{2}.$$

6.2 Completeness

Suppose v is a YES instance, i.e., its distance from L is at least $\sqrt{n}$. We show that a random witness chosen according to $\hat{f}$ satisfies each of the above tests with probability at least 3/4. Clearly, this implies the existence of a witness that satisfies all tests. Test (a) is always satisfied because $\hat{f}$'s support is on L^*.

Claim 6.1 *The probability that a random witness generated by the above procedure satisfies test (b) is more than $3/4$, i.e., $f_W(v) < 1/2$ with probability at least $3/4$.*

Proof: The proof follows trivially from Lemma 3.1 combined with Lemma 1.3. ■

Claim 6.2 *The probability that a random witness generated by the above procedure satisfies test (c) is at least $3/4$.*

Proof: Proving that the largest eigenvalue of the matrix $W \cdot W^T$ is at most $10N$, is equivalent to showing that for all unit vectors $u \in \mathbb{R}^n$ simultaneously we have

$$\frac{1}{N}\sum_{i=1}^{N}\langle u, w_i\rangle^2 \leq 10.$$

We start by showing that this holds with high probability for a single unit vector u, and then extend by standard arguments first to an ε-net in the set of all directions, and then to a general direction.

<u>Proof for a single unit vector:</u> Recall that $\hat{f}$ is the Gaussian distribution over the dual lattice. We will use two facts. First, according to Lemma 2.5, the probability that the norm of a vector chosen from $\hat{f}$ is more than, say, $\sqrt{n}$, is $2^{-\Omega(n)}$. Second, Lemma 2.6 states that for any unit vector u, the average norm squared of the projection on u of a vector w chosen from $\hat{f}$ is at most $\frac{1}{2\pi}$:

$$\mathrm{E}_{w\sim\hat{f}}\langle u, w\rangle^2 \leq \frac{1}{2\pi}.$$

Consider the distribution D which we get by conditioning $\hat{f}$ on the vector w being not longer than $\sqrt{n}$. We have:

$$\mathrm{E}_{w\sim\hat{f}}\langle u, w\rangle^2 \geq \Pr_{w\sim\hat{f}}(\|w\| \leq \sqrt{n})\ \mathrm{E}_{w\sim D}\langle u, w\rangle^2.$$

Combining the two aforementioned facts, we get that the average norm squared of the projection on u of a vector w chosen from D is

$$\mathrm{E}_{w\sim D}\langle u, w\rangle^2 \leq \frac{1}{2\pi(1-2^{-\Omega(n)})} \leq 1.$$

Let $w_1, \ldots, w_N$ be chosen from the distribution $\hat{f}$. With probability at least $1 - 2^{-\Omega(n)}$, none of them is longer than $\sqrt{n}$. Hence, from now on we can condition on this event. Notice that now each w_i is distributed according to D. Hence, by Hoeffding's inequality (2), for any unit vector u we obtain that

$$\frac{1}{N}\sum_{i=1}^{N}\langle u, w_i\rangle^2 \leq 2 \tag{3}$$

holds with probability at least $1 - 2^{-\Omega(N/n^2)} \geq 1 - 2^{-\Omega(n^2)}$ on the choice of the w_i's.

Proof for all unit vectors in an ε-net: Now, consider an ε-net A on the unit sphere with parameter $\varepsilon = 1/(2n)$, i.e., a set of points such that for any point on the sphere there exists a point in A within distance at most ε. There exists such a net A with $2^{O(n)}$ points (see, e.g., [11]). For completeness, let us sketch a construction of a net with $2^{O(n \log n)}$ points; this will be enough for our needs. Let C be $[-1, 1]^n$, i.e., the n dimensional cube of edge length 2. Notice that C contains the unit sphere. Partition C into $(2\sqrt{n}/\varepsilon)^n = 2^{O(n \log n)}$ small cubes of edge length $\varepsilon/\sqrt{n}$. For each small cube that intersects the n dimensional sphere, choose any point in the intersection and add it to A. It is easy to see that the collection of these points constitutes an ε-net on the sphere, since any point in the sphere belongs to one of the small cubes, and the diameter of each small cube is exactly ε. We now apply the union bound on all points in the ε-net A. It follows that Equation 3 holds with probability at least $1 - 2^{-\Omega(n^2)}$, for all u in the net A *simultaneously*.

Proof for all unit vectors: Next, we want to show that if this holds for all points in A, then a slightly weaker version of Equation 3 holds for *all* directions. For this we approximate a general direction u' by a point in A. Let $u \in A$ be the closest point to u' in A. Notice that $\|u - u'\| \le \varepsilon$. Thus,

$$\left| \frac{1}{N} \sum_{i=1}^{N} \langle u', w_i \rangle^2 - \frac{1}{N} \sum_{i=1}^{N} \langle u, w_i \rangle^2 \right| \le \frac{1}{N} \sum_{i=1}^{N} |\langle u' - u, w_i \rangle \langle u' + u, w_i \rangle| \le 2\varepsilon \max_i \|w_i\|^2 \le 2\varepsilon n = 1$$

where in the last inequality we used the fact that the w_i's are chosen from D and hence their length is at most $\sqrt{n}$. This yields that with probability $1 - 2^{-\Omega(n^2)}$ on the choice of the w_i's from D it holds that

$$\frac{1}{N} \sum_{i=1}^{N} \langle u, w_i \rangle^2 \le 2 + 1 \le 10$$

for all unit vectors u. ■

References

[1] S. Aaronson. Lower bounds for local search by quantum arguments. In *Proc. 36th ACM Symp. on Theory of Computing (STOC)*, pages 465–474, 2004.

[2] D. Aharonov and O. Regev. A lattice problem in quantum NP. In *Proc. 44th Annual IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 210–219, 2003.

[3] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proc. 28th ACM Symp. on Theory of Computing (STOC)*, pages 99–108, 1996.

[4] M. Ajtai. The shortest vector problem in l_2 is NP-hard for randomized reductions (extended abstract) 10-19. In *Proc. 30th ACM Symp. on Theory of Computing (STOC)*, pages 10–19, 1998.

[5] M. Ajtai and C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proc. 29th ACM Symp. on Theory of Computing (STOC)*, pages 284–293, 1997.

[6] M. Ajtai, R. Kumar, and D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *Proc. 33rd ACM Symp. on Theory of Computing*, pages 601–610, 2001.

[7] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(4):625–635, 1993.

[8] J.-Y. Cai and A. Nerurkar. A note on the non-NP-hardness of approximate lattice problems under general Cook reductions. *Inform. Process. Lett.*, 76(1-2):61–66, 2000.

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

[10] U. Feige and D. Micciancio. The inapproximability of lattice and coding problems with preprocessing. In *Computational Complexity*, pages 44–52, 2002.

[11] J. Friedman, J. Kahn, and E. Szemerédi. On the second eigenvalue of random regular graphs. In *Proc. 21st Annual Symp. on Theory of Computing (STOC)*, pages 587–598, 1989.

[12] C. F. Gauss. *Disquisitiones Arithmeticae*. Gerh. Fleischer Iun, 1801.

[13] O. Goldreich and S. Goldwasser. On the limits of nonapproximability of lattice problems. *J. Comput. System Sci.*, 60(3):540–563, 2000.

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

[15] J. Håstad, B. Just, J. C. Lagarias, and C.-P. Schnorr. Polynomial time algorithms for finding integer relations among real numbers. *SIAM J. Comput.*, 18(5):859–881, 1989.

[16] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[17] R. Kannan. Improved algorithms for integer programming and related lattice problems. In *Proc.* 15^{th} *Symp. Theory. of Comp.*, pages 193–206, 1983.

[18] I. Kerenidis and R. de Wolf. Exponential lower bound for 2-query locally decodable codes via a quantum argument. In *Proc. 35th ACM Symp. on Theory of Computing (STOC)*, pages 106–115, 2003.

[19] S. Khot. BCH codes, augmented tensor products and hardness of the shortest vector problem in lattices. In *Proc. 45th Annual IEEE Symp. on Foundations of Computer Science (FOCS)*, 2004.

[20] E. Kushilevitz and Y. Mansour. Learning decision trees using the fourier spectrum. *SIAM J. Comput.*, 22(6):1331–1348, 1993.

[21] J. C. Lagarias, H. W. Lenstra, Jr., and C.-P. Schnorr. Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. *Combinatorica*, 10(4):333–348, 1990.

[22] A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261:515–534, 1982.

[23] D. Micciancio. The shortest vector problem is NP-hard to approximate to within some constant. *SIAM Journal on Computing*, 30(6):2008–2035, March 2001. Preliminary version in FOCS 1998.

[24] D. Micciancio and S. Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers, Boston, Massachusetts, March 2002.

[25] O. Regev. Improved inapproximability of lattice and coding problems with preprocessing. In *Proc. of 18th IEEE Annual Conference on Computational Complexity (CCC)*, pages 363–370, 2003.

[26] C.-P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theoretical Computer Science*, 53(2-3):201–224, 1987.

[27] C.-P. Schnorr. Factoring integers and computing discrete logarithms via diophantine approximation. In *Proc. of Eurocrypt '91*, volume 547, pages 171–181. Springer-Verlag, 1991.

[28] P. van Emde Boas. Another NP-complete problem and the complexity of computing short vectors in a lattice. Technical report, University of Amsterdam, Department of Mathematics, Netherlands, 1981. Technical Report 8104.

[29] D. Štefankovič. Fourier transforms in computer science. Master's Thesis, University of Chicago, Department of Computer Science, TR-2002-03.

A Reducing GapSVP to GapCVP

Lemma A.1 *If for some $\beta = \beta(n)$, GapCVP_β is in coNP then so is GapSVP_β.*

Proof: Consider an instance of GapSVP_β given by the lattice L whose basis is $(b_1, \ldots, b_n)$. We map it to n instances of GapCVP_β where the i'th instance, $i = 1, \ldots, n$, is given by the lattice L_i spanned by $(b_1, \ldots, b_{i-1}, 2b_i, b_{i+1}, \ldots, b_n)$ and the target vector b_i. In the following we show that this mapping has the property that if L is a YES instance of GapSVP_β then at least one of (L_i, b_i) is a YES instance of GapCVP_β and if L is a NO instance then all n instances (L_i, b_i) are NO instances. This will complete the proof of the lemma since a NO witness for L can be given by n NO witnesses for (L_i, b_i).

Consider the case where L is a YES instance. In other words, if

$$u = a_1 b_1 + a_2 b_2 + \ldots + a_n b_n$$

denotes the shortest vector, then its length is at most 1. Notice that not all the a_i's are even for otherwise the vector $u/2$ is a shorter lattice vector. Let j be such that a_j is odd. Then the distance of b_j from the lattice B_j is at most $\|u\| \leq 1$ since $b_j + u \in B_j$. Hence, (L_j, b_j) is a YES instance of GapCVP_β. Now consider the case where L is a NO instance of GapSVP_β. That is, the length of the shortest vector is at least β. Since for any $i \in [n]$, $b_i \notin L_i$ this implies that $d(b_i, L_i) \geq \beta$. Hence, (L_i, b_i) is a NO instance of GapCVP_β. ■

B $\mathsf{GapSVP}_{\sqrt{n}}$ and $\mathsf{GapCVP}_{\sqrt{n}}$ are unlikely to be NP-hard

It is easy to see that our main result implies that if $\mathsf{GapSVP}_{\sqrt{n}}$ or $\mathsf{GapCVP}_{\sqrt{n}}$ are NP-hard under Karp reductions then the polynomial hierarchy collapses. In this section we show that the same is true for Cook reductions. This does not follow immediately from our main result because we are dealing with promise problems. The difficulty arises because a Cook reduction might perform queries that are neither a YES instance nor a NO instance. This issue can be resolved by using the fact that not only $\mathsf{GapCVP}_{\sqrt{n}} \in \mathsf{NP}$ but also $\mathsf{CVP} \in \mathsf{NP}$ (and similarly for SVP). In other words, no promise is needed in order to show that a point is close to the lattice. In the following, we will show a proof that holds for any problem with the above properties. We remark that a similar proof has already appeared before in [24, 8] and we repeat it here mainly for completeness.

Lemma B.1 *Let $\Pi = (\Pi_{\mathsf{YES}}, \Pi_{\mathsf{NO}})$ be a promise problem and let Π_{MAYBE} denote all instances outside $\Pi_{\mathsf{YES}} \cup \Pi_{\mathsf{NO}}$. Assume that Π is in coNP and that the (non-promise) problem $\Pi' = (\Pi_{\mathsf{YES}} \cup \Pi_{\mathsf{MAYBE}}, \Pi_{\mathsf{NO}})$ is in NP. Then, if Π is NP-hard under Cook reductions then $\mathsf{NP} \subseteq \mathsf{coNP}$ and the polynomial hierarchy collapses.*

Proof: Assume there exists a Cook reduction from, say, SAT to Π. That is, there exists a polynomial time procedure T that solves SAT given access to an oracle to Π. Notice that while the oracle is guaranteed to answer YES on queries from Π_{YES} and NO on queries from Π_{NO}, its answers on queries from Π_{MAYBE} are arbitrary and should not affect the output of T.

Since $\Pi \in \mathsf{coNP}$, there exists a verifier V_1 and a witness $w_1(x)$ for every $x \in \Pi_{\mathsf{NO}}$ such that V_1 accepts $(x, w_1(x))$. Moreover, V_1 rejects (x, w) for any $x \in \Pi_{\mathsf{YES}}$ and any w. Similarly, since $\Pi' \in \mathsf{NP}$, there exists a verifier V_2 and a witness $w_2(x)$ for every $x \in \Pi_{\mathsf{YES}} \cup \Pi_{\mathsf{MAYBE}}$ such that V_2 accepts $(x, w_2(x))$. Moreover, V_2 rejects (x, w) for any $x \in \Pi_{\mathsf{NO}}$ and any w.

We would like to show that SAT is in coNP. Let Φ be a SAT instance and let $x_1, \ldots, x_k$ be the set of oracle queries which T performs on input Φ. Our witness consists of k pairs, one for each x_i. For $x_i \in \Pi_{\mathsf{NO}}$ we include the pair $(\mathsf{NO}, w_1(x))$ and for $x_i \in \Pi_{\mathsf{YES}} \cup \Pi_{\mathsf{MAYBE}}$ we include the pair $(\mathsf{YES}, w_2(x))$. The verifier simulates T; for each query x_i that T performs, the verifier reads pair corresponding to x_i in the witness. If the pair is of the form (YES, w) then the verifier checks that $V_2(x_i, w)$ accepts and then returns YES to T. Similarly, if the pair is of the form (NO, w) then the verifier checks that $V_1(x_i, w)$ accepts and then returns NO to T. If any of the calls to V_1 or V_2 rejects, then the verifier rejects. Finally, if T outputs that Φ is satisfiable, the verifier rejects and otherwise it accepts.

The completeness follows easily. More specifically, if Φ is unsatisfiable then the witness described above will cause the verifier to accept. In order to prove soundness, assume that Φ is satisfiable and let us show that the verifier rejects. Notice that for each query $x_i \in \Pi_{\mathsf{NO}}$ the witness must include a pair of the form (NO, w) because otherwise V_2 would reject. Similarly, for each query $x_i \in \Pi_{\mathsf{YES}}$ the witness must include a pair of the form (YES, w) because otherwise V_1 would reject. This implies that T receives the correct answers for all of its queries inside $\Pi_{\mathsf{NO}} \cup \Pi_{\mathsf{YES}}$ and must therefore output the correct answer, i.e., that Φ is satisfiable and then the verifier rejects. ■