Almost perfect lattices, the covering radius problem, and applications to Ajtai's connection factor *

Daniele Micciancio

University of California, San Diego
9500 Gilman Dr., Mail code 0114
La Jolla, CA 92093, USA
Email: daniele@cs.ucsd.edu

Abstract

Lattices have received considerable attention as a potential source of computational hardness to be used in cryptography, after a breakthrough result of Ajtai (STOC 1996) connecting the average-case and worst-case complexity of various lattice problems. The purpose of this paper is twofold. On the expository side, we present a rigorous self contained proof of results along the lines of Ajtai's seminal work. At the same time, we explore to what extent Ajtai's original results can be quantitatively improved. As a by-product, we define a random class of lattices such that computing short nonzero vectors in the class with non-negligible probability is at least as hard as approximating the length of the shortest nonzero vector in *any* n-dimensional lattice within worst-case approximation factors $\gamma(n) = n^3\omega(\sqrt{\log n \log\log n})$. This improves previously known best connection factor $\gamma(n) = n^{4+\epsilon}$ (Cai and Nerurkar, FOCS 1997). We also show how our reduction implies the existence of collision resistant cryptographic hash functions based on the worst-case inapproximability of the shortest vector problem within the same factors $\gamma(n) = n^3\omega(\sqrt{\log n \log\log n})$.

In the process we distill various new lattice problems that might be of independent interest, related to the covering radius, the bounded distance decoding problem, approximate counting of lattice points inside convex bodies, and the efficient construction of lattices with good geometric and algorithmic decoding properties. We also show how further investigation of these new lattice problems might lead to even stronger connections between the average-case and worst-case complexity of the shortest vector problem, possibly leading to connection factors as low as $\gamma(n) = n^{1.5}\omega(\log n)$.

1 Introduction

It has long been realized that the relevant notion of hardness in cryptography is *average-case* hardness: if the key of a cryptographic function is chosen at random, then no probabilistic polynomial time algorithm can break the scheme with non-negligible probability. In the past few years, computational problems on point lattices have attracted considerable interest for their potential cryptographic applications because of the following remarkable discovery of Ajtai [2]: a certain natural computational problem (namely, finding small nonzero solutions to a suitably chosen random system of homogeneous linear equations) is at least as hard on the *average* as the *worst-case* instance of various lattice problems, e.g., approximating the length of the shortest nonzero vector in a lattice within a worst-case factor $\gamma(n)$ polynomial in the dimension n of the lattice. This immediately gives provably secure cryptographic functions based on the conjectured *worst-case*

*Preliminary versions of this paper appeared in the proceedings of STOC 2002 and CCC 2002 with the title "Improved cryptographic hash functions with worst-case/average-case connection" [35], and in Chapter 8 of the book "Complexity of lattice problems: a cryptographic perspective" [36]. Research supported in part by NSF Career Award CCR-0093029 and a Sloan Research Fellowship. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

intractability of lattice problems.[1] Moreover, since the set of integer solutions of a linear system forms a lattice, the result in [2] can also be regarded as a connection between the complexity of finding short nonzero vectors in suitably chosen random lattices *on the average*,[2] and the complexity of approximating the length of the shortest nonzero vector (as well as solving various other lattice problems) in any lattice *in the worst case*.

We remark that building cryptographic functions that are as hard to break as the *worst-case* instance of the underlying mathematical problem is especially important in the case of lattices because lattice approximation algorithms (like the LLL algorithm [27]) are believed to perform much better on the average than their worst-case theoretical upper bounds. Moreover, since lattice problems get easier and easier as the approximation factor $\gamma(n)$ increases, determining the smallest worst-case inapproximability factor $\gamma(n)$ that implies the average-case hardness of solving Ajtai's random equations, is both a theoretically interesting and a practically important problem, and it has been the subject of subsequent work by Cai and Nerurkar [11] and Micciancio [35] in a preliminary version of this paper.

Our contribution. The purpose of this paper is twofold. First of all, we give a full, self contained proof of Ajtai's original result [2] and a detailed account of all relevant techniques introduced in subsequent improvements by Cai and Nerurkar [11]. Previous papers [2, 11] only appeared in the form of extended abstracts or technical reports that left to the reader the burden of reconstructing the details of many technical steps. In this paper we develop a number of elementary, still useful, general techniques that allow both to cover all the steps in great detail, and at the same time also offer a cleaner high level picture of the proof. Secondly, we explore to what extent the worst-case inapproximability factors $\gamma(n)$ for lattice problems (shown to imply the average-case hardness of solving random linear equations in [2, 11]) can be further reduced. In the process, we introduce and start investigating various new lattice problems that might be of independent interest, and are discussed in more detail in the following subsections. These technical contributions are summarized as follows. On the average-case complexity side, we introduce a kind of lattices (that we call *almost perfect* in analogy with perfect codes), and use them to define a *new random class* of linear equations such that finding small solutions on the average is potentially harder than in the random class proposed by Ajtai. On the worst-case complexity side, we consider various *new lattice problems*, like approximating the *covering radius* of a lattice. Using these new problems, we are able to improve the connection factor for the shortest vector problem established in [2, 11]. Specifically, we show that finding small solutions to our random equations (with non-negligible probability), is at least as hard as the *worst-case* instance of

- approximating the length of the shortest nonzero vector in any n-dimensional lattice within a factor $\gamma(n) = \tau(n) \cdot n^{2.5}\omega(\log n)$, where $\tau(n) \in [1, \sqrt{n}]$ is a parameter that depends on the almost perfect lattices used in the construction and $\omega(\log n)$ is an arbitrary superlogarithmic function.

Even for $\tau(n) = \sqrt{n}$ this improves the connection factor $\gamma(n) = n^{4+\epsilon}$ of [11] by more than a factor n. (See Subsection 1.3 for details about the results of [11].) We remark that function $\tau(n)$ is a parameter that depends on the construction of the random equations, and it equals $\sqrt{n}$ in Ajtai's construction as studied in [2, 11]. In this paper we give a better construction showing that smaller values of $\tau(n)$ are possible. The improvement we present is quantitatively modest (namely, $\tau(n) = \sqrt{n \log\log n / \log n}$), but qualitatively interesting, as it shows that Ajtai's class of random equations is not necessarily optimal and there is room to hope that more substantial improvements are possible.

We also relate the average case complexity of computing small solutions to our random equations to other lattice problems, like the *worst-case* instances of

[1] In particular, Ajtai [2] showed that if no algorithm can efficiently approximate the length of the shortest nonzero vector in any n-dimensional lattice within (worst-case) polynomial approximation factors $\gamma(n) = n^{O(1)}$, then *one-way functions* exist. Subsequently, Goldreich, Goldwasser and Halevi [18] observed that under essentially the same assumptions as Ajtai's, one can prove the existence of *collision resistant hash functions*, a particularly useful kind of one-way function families with many applications in cryptography.

[2] For clarity, in the rest of the paper we always refer to this average-case problem as "finding small solutions to random equations", while lattices are used only to describe worst-case problems. However, we remark that the two formulations are equivalent, and all results discussed in this paper can be regarded as connections between the average-case and worst-case complexity of (different) lattice problems.

- computing maximal sets of linearly independent vectors that are within a factor $\gamma(n) = \tau(n) \cdot n^2 \cdot \omega(\log n)$ from the shortest,[3]
- approximating within a factor $\gamma(n) = \tau(n) \cdot n^2 \cdot \omega(\log n)$ the covering radius of any n-dimensional lattice, i.e., the maximum possible distance $\rho(\mathcal{L}(\mathbf{B})) = \max_{\mathbf{t}} \text{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B}))$ where $\mathbf{t}$ ranges over the entire space spanned by $\mathbf{B}$,
- finding, given an n-dimensional lattice basis $\mathbf{B}$ and a target point $\mathbf{t}$, a lattice point whose distance from the target $\mathbf{t}$ is at most $\gamma(n) = \tau(n) \cdot n^2 \cdot \omega(\log n)$ times the covering radius of the lattice.

Even for $\tau(n) = \sqrt{n}$, the first relation improves previously known best connection factor $n^{3+\epsilon}$ of [11] by more than $\sqrt{n}$. (See Subsection 1.3 for details about the results of [11].) The other two relations are the first results explicitly connecting the complexity of finding small solutions to random equations to the covering radius problem. Although neither problem has been previously considered in computational complexity, they are both natural computational problems on lattices that might be of independent interest. The last problem is a "guaranteed distance" variant of the well studied closest vector problem, where the error instead of being measured with respect to the distance of the given target, is measured with respect to the worst case distance over all possible target vectors.

All our results are obtained as corollaries to a main theorem that shows that finding small solutions to our random equations is at least as hard as the worst-case instance of finding maximal sets of linearly independent vectors of length at most $\gamma(n) = \tau(n)\sqrt{n} \cdot \omega(\log n)$ times a new lattice invariant that we call the *generalized uniform radius*. Notice how this factor is extremely small: depending on the value of $\tau(n)$, $\gamma(n)$ can be as small as $\sqrt{n} \cdot \omega(\log n)$. This suggests that further investigation of almost perfect lattices and the connection between the uniform radius and other lattice invariants might lead to even stronger connections between the average-case and worst-case complexity of computing short lattice vectors. In particular, we conjecture that there exist random classes of linear equations such that finding small solutions on the average is at least as hard as approximating the length of the shortest nonzero vector in any n-dimensional lattice within a factor $\gamma(n) = n^{1.5} \cdot \omega(\log n)$ in the worst case.

In the following subsections we give a more detailed description of the new lattice problems introduced in this paper, and then review previous work in related areas.

1.1 New lattice problems

A lattice is the set of intersection points of a regular, but not necessarily orthogonal, n-dimensional grid. Mathematically, it can be described as the set of all integer linear combination $x_1\mathbf{b}_1 + \cdots + x_n\mathbf{b}_n$ (with $x_1, \ldots, x_n \in \mathbb{Z}$) of a sequence of linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ in Euclidean space $\mathbb{R}^m$. The simplest example is the *integer lattice* $\mathbb{Z}^n$, i.e., the set of all n-dimensional vectors with integer coordinates. Two fundamental constants associated to any lattice are the *packing radius* and the *covering radius*: the packing radius is the largest radius such that (open) spheres centered at distinct lattice points do not intersect, and the covering radius is the smallest radius such that (closed) spheres centered at all lattice points cover the entire space. Equivalently, the *packing radius* can be defined as the largest r such that any (open) sphere of radius r (not necessarily centered around a lattice point) contains *at most one* lattice point. Similarly, the *covering radius* can be defined as the smallest r such that any (closed) sphere of radius r contains *at least one* lattice point. In this paper we introduce a new quantity, the *uniform radius*, defined as the smallest r such that all spheres of radius r contain *approximately the same number* of lattice points. (See Section 3 for a formal definition.) For technical reasons, we introduce also a variant of the uniform radius, the *generalized uniform radius*, which considers not only spheres, but arbitrary convex bodies.

Of all these quantities, only the *packing radius* has received considerable attention from a computational complexity point of view. It is easy to see that for any lattice, the packing radius equals half the length of the shortest nonzero lattice vector, so (approximately) computing the packing radius is computationally equivalent to computing the (approximate, within the same approximation factor) length of the shortest nonzero lattice vector. (See Subsection 2.3 for a discussion of the computational complexity of this problem.)

[3]Here, and throughout the rest of the paper, the length of a finite set of (linearly independent) vectors is defined as the maximum length of the vectors in the set.

Determining the *covering radius* of a lattice is a classic problem in the geometry of numbers, but it has received so far very little attention from a computational complexity point of view. We suggest that the covering radius is, by itself, an interesting problem to be studied as a potential source of computational hardness. We conjecture that computing the covering radius is NP-hard. We remark that even for the exact version of the problem, no NP-hardness result is currently known. However, the exact solution to the covering radius problem is not even known to be computable in *non-deterministic* polynomial time (NP), and the analogous problem for linear codes is known to be complete for the second level of the polynomial hierarchy [30], a class of problems presumably much harder than NP-complete ones.

The problem of estimating the *(generalized) uniform radius*, has been implicitly considered before in connection with vector quantization[4] [29], and volume estimation problems [25], but only for the special case of the integer lattice $\mathbb{Z}^n$ and specific convex bodies (spheres or polyhedra). In this paper we generalize this natural geometric problem to arbitrary lattices and convex bodies, and show that the generalized uniform radius is always within a factor $O(n)$ from the covering radius.

1.2 Almost perfect lattices

The packing radius and covering radius have been extensively studied in coding theory. *Codes* are sets of strings (called *codewords*) of some fixed length n over a finite alphabet Σ, with the (Hamming) distance between strings measured as the number of positions in which the two strings differ. Similarly to lattices, the packing radius and covering radius of a code are defined as the largest and smallest radii such that the Hamming spheres centered at codewords are disjoint or cover the entire space Σ^n, respectively. A code is called *perfect* if the packing radius equals the covering radius. In other words, the code is perfect if it is possible to partition the entire space Σ^n with equal (Hamming) spheres centered at the codewords. Interestingly, perfect codes are rare but do exist (see [21, Section 5]). However, the same is not true for lattices: it is not possible to partition the Euclidean space $\mathbb{R}^n$ with equal spheres of radius bounded away from 0. However, one can attempt to partition the space with almost spherical bodies. Any lattice naturally defines a partition of space into regions, the *Voronoi cells*, obtained by mapping each point in space to the closest lattice point (with ties broken in some standard way). It is easy to see that each Voronoi cell contains an open sphere of radius equal to the packing radius, and is completely contained in a closed sphere of radius equal to the covering radius. The covering radius is always at least as large as the packing radius, and the smaller is the gap between the two radii, the closer are the Voronoi cells to perfect spheres. We say that a lattice is τ-perfect if the covering radius is at most τ times the packing radius. We are interested in lattices that are τ-perfect for $\tau > 1$ as small as possible. Notice that the integer lattice $\mathbb{Z}^n$ is $\tau(n)$-perfect for $\tau(n) = \sqrt{n}$, so — in trying to minimize $\tau(n)$ — we may assume without loss of generality that $\tau(n) \in [1, \sqrt{n}]$. We say that a sequence of lattices is *almost perfect* if it is $\tau(n)$-perfect for some function $\tau(n) = o(\sqrt{n})$ asymptotically smaller than $\sqrt{n}$. Ideally, we would like to find almost perfect lattices with $\tau(n) = O(1)$ equal to a constant independent of the dimension.

Another fundamental problem in coding theory is the maximum likelihood decoding: given a target point, find the codeword closest to the target. The analogous problem on lattices, called the closest vector problem, is: given a lattice and a target vector, find the lattice point closest to it. Differently from lattices, in coding theory most work has focused on finding efficient decoding algorithms for specific codes, whereas in the closest vector problem the lattice is usually considered as part of the input. In this paper, we consider the lattice decoding problem for specific lattices. We say that a lattice is *easily decodable* if there is an efficient algorithm that on input a target point, outputs the lattice point closest to the target. (Formally, we need to consider a sequence of lattices in higher and higher dimension. See Section 4 for details.) For example, the integer lattice $\mathbb{Z}^n$ is easily decodable: given a target point $\mathbf{y} \in \mathbb{Q}^n$, the closest lattice point is easily found by rounding each coordinate of $\mathbf{y}$ to the closest integer.

The random classes of equations defined in this paper are based on easily decodable $\tau(n)$-perfect lattices, and the smaller is $\tau(n)$, the harder is to find small solutions to the random equations. So, it is natural to ask what is the smallest value of $\tau(n)$ for which we can efficiently build *easily decodable* $\tau(n)$-perfect lattices. It is known [40, 10] that very good almost perfect lattices exist, achieving constant $\tau(n) = O(1)$.

[4] Vector quantization is the problem of mapping arbitrary real vectors to a discrete set of points (e.g., the points of a lattice) in such a way that each vector is mapped to a nearby point.

Unfortunately, the proofs in [40, 10] do not give an efficient procedure to build and decode these lattices. Various examples of easily decodable lattices are given in [12], but none of them is almost perfect, i.e., they only achieve $\tau(n) = \Theta(\sqrt{n})$. It is natural to ask if almost perfect easily decodable lattices exist at all. In this paper we initiate the study of almost perfect lattices from a computational point of view, and we give the first efficient construction of easily decodable almost perfect lattices with $\tau(n) = O(\sqrt{n \log\log n / \log n})$.

Our almost perfect lattices allow to slightly improve (by a multiplicative factor $O(\sqrt{\log n / \log\log n})$) the worst-case/average-case connection factor $\gamma(n)$ for all lattice problems considered in this paper. Although not substantial, this improvement in the connection factor is qualitatively important because it shows that there are random classes of linear equations for which finding small solutions is potentially harder than for the random class originally considered by Ajtai. Moreover, it suggests that it might be possible to find even better easily decodable almost perfect lattices that allow to further reduce the connection factors for all lattice problems considered in this paper by almost $\sqrt{n}$.

1.3 Related work

This work directly builds upon techniques of Ajtai [2], Cai and Nerurkar [11] and Goldreich, Goldwasser and Halevi [18], and it is the final version of [35]. In [2] Ajtai showed[5] that if one can efficiently find small solutions to random linear equations on the average with non-negligible probability $\delta(n) = 1/n^{O(1)}$, then one can efficiently approximate the length of the shortest nonzero vector in any n-dimensional lattice within a (worst-case) polynomial factor $\gamma(n) = n^{O(1)}/\delta(n)$, where the smaller the success probability $\delta(n)$, the larger the approximation factor $\gamma(n)$.[6] Moreover, even for large values of $\delta(n)$ (say $\delta(n) = 1/2$), the factor $\gamma(n)$ given by [2] is rather large.[7] Following Ajtai's seminal work, Cai and Nerurkar [11] showed that finding small solutions to Ajtai's random linear equations (with non-negligible probability $\delta(n) = 1/n^{O(1)}$) is at least as hard as computing maximal sets of linearly independent vectors that are within a worst-case factor $n^{3+\epsilon}$ from the shortest[8] (for any fixed $\epsilon > 0$, independently of the success probability $\delta(n)$). It immediately follows (using standard relations between lattice problems [26, 7]) that Ajtai's random problem is at least as hard to solve on the average as approximating the length of the shortest nonzero vector in any n-dimensional lattice within a factor $\gamma(n) = n^{4+\epsilon}$ in the worst case.[9]

The question of determining under what conditions the number of lattice points inside a convex body $\mathcal{Q}$ is roughly proportional to the volume has been extensively studied, but mostly for the case of the integer lattice $\mathbb{Z}^n$. For example Mazo and Odlyzko [29] study the problem when $\mathcal{Q}$ is a sphere of radius r, in connection with universal quantization and low density subset sum problems. (See [29] and references therein for a description of these problems.) In particular they show that for $r = O(\sqrt{n})$ the number of integer lattice points in the sphere can deviate from the expected value by factors exponential in n, but claim that if $r = n^{1/2+\epsilon}$ (for any $\epsilon > 0$) then the number of integer lattice points in the sphere is always asymptotic to the volume, no matter where the center is located. A different class of convex bodies is considered by Kannan and Vempala in [25], but, as usual, only for the special case of the integer lattice $\mathbb{Z}^n$. In [25] $\mathcal{Q}$ is an

[5] To be precise, [2] only proves the result for $\delta(n) = 1/2$, and remarks that the proof can be generalized to any non-negligible $\delta(n) = 1/n^{O(1)}$.

[6] It should be remarked that, as observed in [2], setting $\delta(n) = 1/2$ already gives weak one-way functions, which can be transformed (using standard techniques, see [16]) into strong one-way functions based on the worst-case hardness of approximating the shortest vector problem within a fixed polynomial factor $\gamma(n) = n^{O(1)}$. However, in order to argue that no efficient algorithm can solve Ajtai's original problem with non-negligible probability, [2] seems to require that no efficient algorithm can approximate the worst-case lattice problems within *any* polynomial factors.

[7] No specific value of $\gamma(n)$ is given in [2], but [11] estimates a factor $\gamma(n) = n^8$ can be derived from the proof.

[8] Cai and Nerurkar [11] only prove the result for the shortest basis problem, but it is easy to modify their proof to yield a result for the shortest independent vector problem. (See footnote 9 for more information.)

[9] To be precise, Cai and Nerurkar [11] only claim a $\gamma(n) = n^{5+\epsilon}$ connection factor, which is proved in three steps: (1) first they use small solutions (say, of size $O(n)$) of random linear equations to find linearly independent lattice vectors within a $n^{3+\epsilon}$ factor from the shortest *basis* in the lattice, (2) then, they use these linearly independent vectors to get a $n^{3.5+\epsilon}$ approximation to the shortest basis problem, and (3) finally, they connect the shortest basis problem to the shortest vector problem loosing additional $n^{1.5}$ factor. We observe that the linear equations of [11] have solutions as small as $n^{0.5+\epsilon}$, which, if used in the proof, result in linearly independent lattice vectors within a $n^{2.5+\epsilon}$ factor from the shortest lattice basis. Then, we observe that the same technique used in [11] to transform these vector into a $n^{3+\epsilon}$-approximate solution to the shortest basis problem, can also be used to show that the original vectors are a $n^{3+\epsilon}$-approximate solution to the shortest independent vectors problem. This allows to use known results [26, 7] to solve the shortest vector problem by loosing only an additional factor n (instead of $n^{1.5}$), leading to a $n^{4+\epsilon}$-approximate solution to the shortest vector problem.

n-dimensional convex polytope with m facets, and the result is that the number of integer lattice points in $\mathcal{Q}$ is approximately proportional to the volume provided that $\mathcal{Q}$ contains a sphere of radius $\Omega(n \cdot \sqrt{\log m})$.[10] A result for arbitrary convex bodies is proved by Dyer, Frieze and Kannan [14] who show that the number of integer lattice points in $\mathcal{Q}$ is approximately proportional to the volume of $\mathcal{Q}$, provided $\mathcal{Q}$ contains a sphere of radius $\Omega(n^{1.5})$. In Section 3 we generalize the result of [14] to arbitrary lattices, and show that the number of lattice points in $\mathcal{Q}$ is approximately proportional to the volume provided that $\mathcal{Q}$ contains a sphere of radius $\Omega(n)$ times bigger than the covering radius of the lattice (see Theorem 3.6).

The covering radius problem has been extensively studied from a mathematical point of view, leading for example to the transference theorems of Banaszczyk [7], but it has received little or no attention from a computational perspective. Two relevant results about the covering radius problem are McLoughlin's proof [30] that the analogous problem on linear codes is hard for the second level of the polynomial hierarchy, and Kannan's algorithm [24] showing that a variant of the covering radius problem for lattices (where the norm defined by an input parallelotope is used, instead of the usual Euclidean norm) can be solved in polynomial time for any fixed dimension. Partly motivated by our work [35], some initial progress towards understanding the computational complexity of (approximately) computing the covering radius of lattices and linear codes has been recently made by Guruswami, Micciancio and Regev [20]. The reader is referred to that paper and Subsection 2.3 for further information about the computational complexity of the covering radius problem.

The problem of decoding specific lattices has been considered in coding theory, for example in connection with vector quantization. In [12] Conway and Sloane give polynomial time decoding algorithms for the root lattices A_n, D_n and their duals A_n^*, D_n^*, as well as various other low dimensional lattices.[11] From a computational complexity point of view, the problem has been considered under the name *closest vector problem with preprocessing*. Adapting similar results of Bruck and Naor [9] and Lobstein [28] for coding and subset-sum problems, Micciancio [31] showed that there are sequences of lattices such that solving the closest vector problem is NP-hard. These results have been improved by Feige and Micciancio [15] and then Regev [38] to show that (unless P = NP) there are lattices and codes that cannot be efficiently decoded even approximately for any constant approximation factor smaller than $\sqrt{3}$. Notice that the goal of [31, 15, 38] is opposite to ours: while [31, 15, 38] give explicit constructions of lattices that cannot be easily decoded, in this paper we search for explicit constructions of easily decodable lattices.

Almost perfect lattices have been extensively studied from a mathematical point of view. In particular, Rogers [40] proved that there exist $\tau(n)$-perfect lattices for $\tau(n) < 3$ and any dimension n, and Butler [10] improved the result to $\tau(n) = 2 + o(1)$. Our exponential time construction of almost perfect lattices in Theorem 4.4 is essentially an algorithmic variant of Rogers' proof.

In this paper we consider the worst case complexity of computing short vectors (as well as solving other computational lattice approximation problems) in *any* lattice. In a recent breakthrough paper [39], Regev has given encryption schemes and collision resistant hash functions that are as hard to break as computing shortest nonzero vectors in lattices with *special structure*. The results proved in [39] achieve approximation factors $O(n^{1.5})$ which are smaller than any other known reduction, but only for lattices where the shortest vector is *unique* is some technical sense. This special structure is common in the construction of lattice based public key encryption schemes [4], but does not seem necessary to build one-way or collision resistant hash functions. In Section 8 we explain how the techniques presented in this paper might lead to one-way and collision resistant hash functions that are as hard to break as solving the shortest vector problem (or other lattice problems) in *any* lattice, within approximation factors similar to those established in [39] for the special class of lattices possessing unique shortest vectors.

Another relevant paper establishing results similar to ours, but for a special class of lattices is [34], where the goal is to obtain hard-on-average problems with very compact representation, rather than improving the connection factor. In [34] Micciancio shows that if approximating the shortest vector problem (or various other lattice problems) is hard in the worst case over a class of lattices with a special *cyclic* property, then one can define random linear equations with only $\omega(\log n)$ variables that are hard to solve on the average. This yields random equations (and cryptographic one-way functions) with a much smaller representation

[10] As a side remark, the motivation to study this problem in [25] is somehow opposite to ours, as they count the number of lattice points in a polytope to estimate its volume. Here, we try to get a bound on the number of lattice points, for convex bodies $\mathcal{Q}$ of known volume.

[11] Asymptotically, only results for infinite families of lattices are interesting because the closest vector problem is known to be solvable in polynomial time in any fixed dimension [23].

size than those considered in this paper, possibly leading to practical and provably secure lattice based cryptographic functions.

1.4 Outline

The rest of the paper is organized as follows. In Section 2 we introduce basic definitions and notation used throughout the paper, and give background about lattice problems and computational complexity. In Section 3 we define the (generalized) uniform radius and relate it to other lattice quantities. In Section 4 we initiate the algorithmic study of almost perfect lattices and present the first polynomial time construction of easily decodable almost perfect lattices. These lattices are used in Section 5 to define a new random class of equations that generalizes Ajtai's one. In Section 6 we prove the main technical result of the paper: finding small solutions to the random linear equations of Section 5 is at least as hard as finding short (relative to the generalized uniform radius) linearly independent lattice vectors in the worst case. In Section 7 we relate this problem to other well known lattice problems, like approximating the length of the shortest vector in a lattice. Section 8 concludes with a brief summary of our main results, and some open problems whose solution would allow to further improve the connection factors established in this paper.

2 Preliminaries

In this section, we introduce the notation that will be used in the rest of the paper, and then we briefly recall basic notions about lattices, statistical distance, and iterative reductions. For more background material about lattices the reader is referred to the book [36].

2.1 Notation

For any finite set S, the size of S is denoted $\#S$. For any real x, $\lfloor x \rfloor$ denotes the largest integer not greater than x, and $\lfloor x \rceil = \lfloor x + 1/2 \rfloor$ is the rounding of x to the closest integer. For any string s, the length of s is denoted $|s|$. For any positive real $\epsilon > 0$, we write $[1 \pm \epsilon]$ to denote the interval $[1 - \epsilon, 1 + \epsilon]$. Arithmetic operations on intervals are defined in the obvious way by extending the standard arithmetic operations pointwise, e.g., $x \cdot [1 \pm \epsilon]$ denotes the interval $[x - \epsilon x, x + \epsilon x]$. Notice that $a \in b \cdot [1 \pm \epsilon]$ if and only if the relative additive error $|a - b|/|b|$ is at most ϵ. For any two positive reals $a, b \geq 0$, we write $a \gtrsim b$ if $a \geq (1/2) \cdot b$, and $a \lesssim b$ if $a \leq (3/2) \cdot b$. We say that a is approximately equal to b (written $a \approx b$), if both $a \lesssim b$ and $a \gtrsim b$, i.e., $a \in b \cdot [1 \pm 1/2]$. Notice that $a \approx b$ is not a symmetric relation, i.e., $a \approx b$ does not imply $b \approx a$. For any $a, b, c \geq 0$, if $a \approx c$ and $b \approx c$, then a and b are within a factor 3 one from the other, i.e.,

$$\forall a, b, c.\ (a \approx c) \wedge (b \approx c) \Rightarrow (a/3 \leq b \leq 3a). \tag{2.1}$$

In the paper we use the standard asymptotic notation for functions. For any two positive real functions f, g, we write $f = O(g)$ or $g = \Omega(f)$ if there exists a constant $c > 0$ such that $f(x) \leq c \cdot g(x)$ for all sufficiently large x. We write $f = \Theta(g)$ if $f = O(g)$ and $f = \Omega(g)$. We write $f = o(g)$ or $g = \omega(f)$ if $f(x) < c \cdot g(x)$ for all $c > 0$ and all sufficiently large x. We also use notation $O(f)$, (resp. $\Omega(f), o(f), \omega(f)$) to denote the class of all functions g such that $g = O(f)$, (resp. $g = \Omega(f)$, etc.), or an arbitrary, but fixed, function from that class. For example, we write $O(1)$ to denote an arbitrary constant, or $n^{O(1)}$ to denote an arbitrary polynomially bounded function of n. A function f is negligible if $f(n) = n^{-\omega(1)}$, i.e., $f(n)$ is asymptotically smaller than any inverse polynomial in n.

Let $\mathbb{R}$, $\mathbb{Q}$ and $\mathbb{Z}$ be the sets of the reals, the rationals and the integers, respectively. The m-dimensional Euclidean space is denoted $\mathbb{R}^m$. We use bold lower case letters (e.g., $\mathbf{x}$) to denote vectors, and bold upper case letters (e.g., $\mathbf{M}$) to denote matrices. The n-dimensional identity matrix, i.e., the $n \times n$ diagonal matrix with 1's on the diagonal, is denoted $\mathbf{I}_n$, or simply $\mathbf{I}$ when the dimension is clear from the context. The columns of a matrix $\mathbf{M}$ are usually denoted by the corresponding lowercase letters, e.g., $\mathbf{m}_1, \ldots, \mathbf{m}_n$. As an exception, the standard unit vectors, i.e., the columns of the identity matrix $\mathbf{I}_n$, are denoted $\mathbf{e}_1, \ldots, \mathbf{e}_n$. If $\mathcal{Q} \subseteq \mathbb{R}^n$ is an arbitrary region of space, and $\mathbf{x} \in \mathbb{R}^n$ is a vector, $\mathcal{Q} + \mathbf{x} = \{\mathbf{y} + \mathbf{x} : \mathbf{y} \in \mathcal{Q}\}$ denotes a copy of $\mathcal{Q}$ shifted by $\mathbf{x}$. The ℓ_2 norm of a vector $\mathbf{x} \in \mathbb{R}^n$ is $\|\mathbf{x}\| = \sqrt{\sum x_i^2}$, and the associated distance is $\mathrm{dist}(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$. For a matrix $\mathbf{M} = [\mathbf{m}_1, \ldots, \mathbf{m}_n]$, we define $\|\mathbf{M}\| = \max_i \|\mathbf{m}_i\|$, where $\mathbf{m}_i$ are the columns of $\mathbf{M}$. For vector

$\mathbf{v} \in \mathbb{R}^n$ and set $\mathcal{Q} \subseteq \mathbb{R}^n$, let $\mathrm{dist}(\mathbf{v}, \mathcal{Q}) = \inf_{\mathbf{w} \in \mathcal{Q}} \|\mathbf{v} - \mathbf{w}\|$ be the distance between $\mathbf{v}$ and $\mathcal{Q}$. For vector $\mathbf{v} \in \mathbb{R}^n$ and real r, let $\mathcal{B}(\mathbf{v}, r) = \{\mathbf{w} \in \mathbb{R}^n : \mathrm{dist}(\mathbf{v}, \mathbf{w}) < r\}$ be the open ball of radius r centered in $\mathbf{v}$, and $\bar{\mathcal{B}}(\mathbf{v}, r) = \{\mathbf{w} \in \mathbb{R}^n : \mathrm{dist}(\mathbf{v}, \mathbf{w}) \leq r\}$ its topological closure. When the center $\mathbf{v} = \mathbf{0}$ is the origin, we simply write $\mathcal{B}(r)$ and $\bar{\mathcal{B}}(r)$. We often use matrix notation to denote sets of vectors. For example, matrix $\mathbf{S} \in \mathbb{R}^{m \times n}$ represents the set of m-dimensional vectors $\{\mathbf{s}_1, \ldots, \mathbf{s}_n\}$, where $\mathbf{s}_1, \ldots, \mathbf{s}_n$ are the columns of $\mathbf{S}$. The linear space spanned by a set of vectors $\mathbf{S}$ is denoted $\mathrm{span}(\mathbf{S}) = \{\sum_i x_i \cdot \mathbf{s}_i : \forall i. x_i \in \mathbb{R}\}$. For any set of linearly independent vectors $\mathbf{S}$, we define the centered half open parallelepiped $\mathcal{P}(\mathbf{S}) = \{\mathbf{S}\mathbf{x} : -1/2 \leq x_i < 1/2\}$.

2.2 Lattices

An m-dimensional *lattice* is the set of all integer combinations $\{\sum_{i=1}^n x_i \mathbf{b}_i : x_i \in \mathbb{Z}\}$ of n linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ in $\mathbb{R}^m$ $(m \geq n)$. The set of vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ is called a *basis* for the lattice, and the integer $n = \dim(\mathrm{span}(\mathbf{B}))$ is called the *rank* of the lattice. If the rank n equals the dimension m, then the lattice is called *full rank* or *full dimensional*. Lattices are infinite Abelian groups with respect to the vector addition operation, and can be equivalently defined as discrete additive subgroups of $\mathbb{R}^m$. A basis can be compactly represented by the matrix $\mathbf{B} = [\mathbf{b}_1 | \ldots | \mathbf{b}_n] \in \mathbb{R}^{m \times n}$ having the basis vectors as columns. The lattice generated by $\mathbf{B}$ is denoted $\mathcal{L}(\mathbf{B})$. Notice that $\mathcal{L}(\mathbf{B}) = \{\mathbf{B}\mathbf{x} : \mathbf{x} \in \mathbb{Z}^n\}$, where $\mathbf{B}\mathbf{x}$ is the usual matrix-vector multiplication. We use notation $\mathcal{L}(\mathbf{B})$ to denote the set $\{\mathbf{B}\mathbf{x} : \mathbf{x} \in \mathbb{Z}^n\}$ even when vectors $\mathbf{B}$ are not linearly independent. The dual of a lattice $\mathcal{L}(\mathbf{B})$, is the set

$$\mathcal{L}(\mathbf{B})^* = \{\mathbf{x} \in \mathrm{span}(\mathcal{L}(\mathbf{B})) : \forall \mathbf{y} \in \mathcal{L}(\mathbf{B}). \langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{Z}\}$$

of all vectors in the linear span of $\mathcal{L}(\mathbf{B})$ that have integer scalar product with all lattice vectors. The dual of a lattice is a lattice, and a possible basis for the dual of $\mathcal{L}(\mathbf{B})$ is given by $\mathbf{B}^* = \mathbf{B}(\mathbf{B}^T\mathbf{B})^{-1}$, where $\mathbf{B}^T$ is the transpose of $\mathbf{B}$. (Notice that if $\mathbf{B}$ is a basis, it has full column rank and the square matrix $\mathbf{B}^T\mathbf{B}$ is invertible.)

The *minimum distance* of a lattice $\mathcal{L}(\mathbf{B})$, (denoted $\lambda_1(\mathcal{L}(\mathbf{B}))$), is the minimum distance between any two (distinct) lattice points and equals the length of the shortest nonzero lattice vector:

$$\lambda_1(\mathcal{L}(\mathbf{B})) = \min\{\mathrm{dist}(\mathbf{x}, \mathbf{y}) : \mathbf{x} \neq \mathbf{y} \in \mathcal{L}(\mathbf{B})\} = \min\{\|\mathbf{x}\| : \mathbf{x} \in \mathcal{L}(\mathbf{B}) \setminus \{\mathbf{0}\}\}.$$

This definition can be generalized to define the ith successive minimum as the smallest λ_i such that $\bar{\mathcal{B}}(\lambda_i)$ contains i linearly independent lattice points:

$$\lambda_i(\mathcal{L}(\mathbf{B})) = \min\{r : \dim(\mathrm{span}(\mathcal{L}(\mathbf{B}) \cap \bar{\mathcal{B}}(r))) \geq i\}$$

Another important constant associated to a lattice is the covering radius: the covering radius $\rho(\mathcal{L}(\mathbf{B}))$ of a lattice is the maximum distance $\mathrm{dist}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ when $\mathbf{x}$ ranges over the linear span of $\mathbf{B}$:

$$\rho(\mathcal{L}(\mathbf{B})) = \max_{\mathbf{x} \in \mathrm{span}(\mathbf{B})} \{\mathrm{dist}(\mathbf{x}, \mathcal{L}(\mathbf{B}))\}.$$

A sublattice of $\mathcal{L}(\mathbf{B})$ is a lattice $\mathcal{L}(\mathbf{S})$ such that $\mathcal{L}(\mathbf{S}) \subseteq \mathcal{L}(\mathbf{B})$. $\mathcal{L}(\mathbf{S})$ is a *full rank* sublattice of $\mathcal{L}(\mathbf{B})$ if it has the same rank as $\mathcal{L}(\mathbf{B})$. The determinant of a (rank n) lattice $\det(\mathcal{L}(\mathbf{B}))$ is the (n-dimensional) volume of the fundamental parallelepiped $\mathcal{P}(\mathbf{B})$, and it does not depend on the choice of the basis $\mathbf{B}$. If $\mathcal{L}(\mathbf{B})$ is full dimensional, then $\det(\mathcal{L}(\mathbf{B}))$ equals the absolute value of the determinant of the $n \times n$ basis matrix $|\det(\mathbf{B})|$. Hadamard's bound gives a simple way to bound the determinant of a lattice as $\det(\mathcal{L}(\mathbf{B})) \leq \prod_i \|\mathbf{b}_i\|$. Hadamard's bound can be much larger than the actual value of the determinant, and it equals the determinant if and only if the basis $\mathbf{B}$ is orthogonal. Minkowski's first theorem (see [36, pp. 11–14]) implies that any rank n lattice $\mathcal{L}(\mathbf{B})$ contains a nonzero vector of length at most

$$\lambda_1(\mathcal{L}(\mathbf{B})) \leq \sqrt{n} \det(\mathcal{L}(\mathbf{B}))^{1/n}. \tag{2.2}$$

The Voronoi cells of a lattice, defined below, play an important role in our proofs. For uniformity, and by analogy with the definition of the half-open parallelepiped, we first define the half-open Voronoi cells. However, we remark that in the rest of the paper we only need the more standard notions of open and closed Voronoi cells.

Definition 2.1 *Let $\preceq$ be the total order on $\mathbb{R}^n$ where $\mathbf{x} \preceq \mathbf{y}$ if and only if $\|\mathbf{x}\| < \|\mathbf{y}\|$ or $\|\mathbf{x}\| = \|\mathbf{y}\|$ and $\mathbf{x}$ precedes $\mathbf{y}$ lexicographically, i.e., $x_i < y_i$ for the first coordinate i such that $x_i \neq y_i$. For any lattice $\mathcal{L}(\mathbf{B})$ and lattice point $\mathbf{x} \in \mathcal{L}(\mathbf{B})$, the (half-open)* Voronoi *cell of $\mathbf{x}$ is the set*

$$\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B})) = \{\mathbf{z} \in \mathrm{span}(\mathcal{L}(\mathbf{B})) : \forall \mathbf{y} \in \mathcal{L}(\mathbf{B}).(\mathbf{z} - \mathbf{x}) \preceq (\mathbf{z} - \mathbf{y})\}.$$

The closed (resp. open) Voronoi cell $\bar{\mathcal{V}}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ (resp. $\mathcal{V}^o(\mathbf{x}, \mathcal{L}(\mathbf{B}))$) is defined as the topological closure (resp. interior) of $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$.

For simplicity, the Voronoi cell of the origin $\mathbf{x} = \mathbf{0}$ is denoted $\mathcal{V}(\mathcal{L}(\mathbf{B}))$. Notice that the Voronoi cell of the integer lattice equals the half-open unit cube: $\mathcal{V}(\mathbb{Z}^n) = \mathcal{P}(\mathbf{I}_n)$. We need some simple properties about Voronoi cells, as listed below. All properties are easily verified and their proof is left to the reader.

Proposition 2.2 *For any lattice $\mathcal{L}(\mathbf{B})$, the Voronoi cells of $\mathcal{L}(\mathbf{B})$ satisfy the following properties:*

- *The half-open Voronoi cells form a partition of* $\mathrm{span}(\mathcal{L}(\mathbf{B}))$*, i.e., for any $\mathbf{y} \in \mathrm{span}(\mathcal{L}(\mathbf{B}))$ there exists a unique lattice point $\mathbf{x} \in \mathcal{L}(\mathbf{B})$ such that $\mathbf{y} \in \mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$.*
- *All Voronoi cells $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ (with $\mathbf{x} \in \mathcal{L}(\mathbf{B})$) are shifted copies $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B})) = \mathcal{V}(\mathcal{L}(\mathbf{B})) + \mathbf{x}$ of the fundamental cell associated to the origin.*
- *Each cell $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ contains the open sphere $\mathcal{B}(\mathbf{x}, \lambda_1(\mathcal{L}(\mathbf{B}))/2)$ and it is contained in the closed sphere $\bar{\mathcal{B}}(\mathbf{x}, \rho(\mathcal{L}(\mathbf{B})))$.*
- *The volume of a Voronoi cell equals* $\mathrm{vol}(\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}))) = \det(\mathcal{L}(\mathbf{B}))$.
- *The open Voronoi cell $\mathcal{V}^o(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ is the set of all points in* $\mathrm{span}(\mathcal{L}(\mathbf{B}))$ *that are strictly closer to $\mathbf{x}$ than to any other lattice point.*
- *The closed Voronoi cell $\bar{\mathcal{V}}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ is the set of all points in* $\mathrm{span}(\mathcal{L}(\mathbf{B}))$ *that are at least as close to $\mathbf{x}$ than to any other lattice point.*
- *The cells $\mathcal{V}^o(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ and $\bar{\mathcal{V}}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ are convex and symmetric about their center $\mathbf{x}$.*

2.3 Computational problems on lattices

When discussing computational issues related to lattices, it is customary to assume that the lattices are represented by a basis matrix $\mathbf{B}$ and that $\mathbf{B}$ has integer entries. Other representations are possible, e.g., a sublattice of $\mathbb{Z}^n$ can be defined as the set of integer solutions to a system of homogeneous modular linear equations. These alternative representations are computationally equivalent to giving a basis, i.e., for example, given a system of homogeneous modular linear equations one can compute in polynomial time a basis for the corresponding lattice.

In this paper we consider the following problems on lattices. All problems are defined in their approximation version, where the approximation factor $\gamma(n)$ can be a function of the rank n of the lattice. The exact version of the problems corresponds to approximation factor $\gamma(n) = 1$.

Definition 2.3 *The* Shortest Vector Problem *(*SVP*), given a lattice basis $\mathbf{B}$, asks for a nonzero lattice vector $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ of length at most $\gamma(n) \cdot \lambda_1(\mathcal{L}(\mathbf{B}))$, where n is the rank of $\mathbf{B}$ and $\gamma(n) \geq 1$ is the approximation factor. The problem can be defined also in a* length estimation *version, where given a basis $\mathbf{B}$, one only has to find a value $\hat{\lambda}_1$ such that $\lambda_1(\mathcal{L}(\mathbf{B})) \leq \hat{\lambda}_1 \leq \gamma(n) \cdot \lambda_1(\mathcal{L}(\mathbf{B}))$. The promise problem*[12] *naturally associated to the length estimation version of* SVP *(denoted* GapSVP$_\gamma$*) is, given $(\mathbf{B}, d)$ where $\mathbf{B}$ is a lattice basis and d is a (rational) number, decide if $\lambda_1(\mathcal{L}(\mathbf{B})) \leq d$ or $\lambda_1(\mathcal{L}(\mathbf{B})) > \gamma(n) \cdot d$.*

[12] Promise problems are a natural generalization of decision problems where one is asked whether a given input satisfies one of two mutually exclusive properties. (E.g., tell if a given input lattice contains a short nonzero vector, or it does not.) However, differently from decision problems, the two properties are not necessarily exhaustive. The problem is, under the promise that the given input satisfies one of the two conditions, tell which of the two properties is satisfied. If the input satisfies neither property, then any answer is acceptable.

The promise problem is easily shown to be equivalent to the length estimation version of SVP. (See for example [36, pp. 20-21].) However, the promise and length estimation problems are not known to be equivalent to the search (vector finding) version of SVP for $\gamma(n) > 1$, i.e., given an oracle to (approximately) compute the length of the shortest nonzero vector in any lattice, it is not clear how to find short lattice vectors.[13] The shortest vector problem is NP-hard (under randomized reductions), even in its promise version, for any constant approximation factor $\gamma(n) < \sqrt{2}$ [3, 33]. The promise version of the problem is clearly solvable in NP. For $\gamma(n) = \Omega(\sqrt{n/\log n})$ the problem is in coAM [17], and for $\gamma(n) = \Omega(\sqrt{n})$ it is also in coNP [1]. (See [26, 7] for earlier results with approximation factor $\gamma(n) = \Omega(n)$.) Finally, when $\gamma(n) = e^{\Omega(n \log\log n/\log n)}$ the problem can be solved in random polynomial time [5], and deterministic polynomial time solutions are known only for $\gamma(n) = e^{\Omega(n(\log\log n)^2/\log n)}$ [41].

Definition 2.4 *The* Shortest Independent Vectors Problem *(*SIVP*), given a lattice basis* $\mathbf{B}$ *of rank* n*, asks for a set of* n *linearly independent lattice vectors* $\mathbf{S} \subseteq \mathcal{L}(\mathbf{B})$ *such that* $\|\mathbf{S}\| \leq \gamma(n) \cdot \lambda_n(\mathcal{L}(\mathbf{B}))$*. The problem can be defined also in a* length estimation *version, where given a basis* $\mathbf{B}$*, one only has to find a value* $\hat{\lambda}_n$ *such that* $\lambda_n(\mathcal{L}(\mathbf{B})) \leq \hat{\lambda}_n \leq \gamma(n) \cdot \lambda_n(\mathcal{L}(\mathbf{B}))$*. The promise problem naturally associated to the length estimation version of* SIVP *(denoted* GAPSIVP$_\gamma$*) is, given* $(\mathbf{B}, d)$ *where* $\mathbf{B}$ *is a lattice basis and* d *is a (rational) number, decide if* $\lambda_n(\mathcal{L}(\mathbf{B})) \leq d$ *or* $\lambda_n(\mathcal{L}(\mathbf{B})) > \gamma(n) \cdot d$*.*

SIVP is NP-hard (as usual, already in its promise version) for quasi-polynomial approximation factors $\gamma(n) = n^{1/O(\log\log n)}$ [8, 13]. The (promise version of) SIVP is clearly in NP. For $\gamma(n) = \Omega(\sqrt{n/\log n})$ the problem is in coAM [20, 17], and for $\gamma(n) = \Omega(\sqrt{n})$ it is also in coNP [20, 1]. On the algorithmic side, it is possible to reduce approximating SIVP within a factor $\sqrt{n} \cdot \gamma(n)$ to approximating SVP within a factor $\gamma(n)$, where both SIVP and SVP are considered in their search version. (See for example [36, Chapter 7].) For the promise version of the problems, the transference theorems of [26, 7] immediately give a reduction from GAPSIVP$_{n \cdot \gamma(n)}$ to (the complement of) GAPSVP$_{\gamma(n)}$. These reductions from SIVP to SVP immediately give deterministic polynomial time algorithms for approximating SIVP within factors $\gamma(n) = e^{O(n(\log\log n)^2/\log n)}$, and probabilistic polynomial time algorithms for $\gamma(n) = e^{O(n(\log\log n)/\log n)}$.

Definition 2.5 *The* Covering Radius Problem *(*CRP*), given a lattice basis* $\mathbf{B}$*, asks for a value* $\hat{\rho}$ *such that* $\rho(\mathcal{L}(\mathbf{B})) \leq \hat{\rho} \leq \gamma(n) \cdot \rho(\mathcal{L}(\mathbf{B}))$*. The promise problem naturally associated to* CRP*, (denoted* GAPCRP$_\gamma$*) is, given* $(\mathbf{B}, d)$ *where* $\mathbf{B}$ *is a lattice basis and* d *is a (rational) number, decide if* $\rho(\mathcal{L}(\mathbf{B})) \leq d$ *or* $\rho(\mathcal{L}(\mathbf{B})) > \gamma(n) \cdot d$*.*

Currently, no NP-hardness result is known for CRP. However, we do not even know how to solve the problem (in its exact version, i.e., for $\gamma(n) = 1$) in non-deterministic polynomial time (NP), and the analogous problem for linear codes is known to be hard for the second level of the polynomial hierarchy [30]. So, we can reasonably conjecture that the same is true for the covering radius problem on lattices.

Conjecture 2.6 *The exact version of the covering radius problem on lattices* GAPCRP$_1$ *is* Π_2*-hard.*

Recently, [20] has shown that GAPCRP$_\gamma$ is in AM for $\gamma = 2$, in coAM for $\gamma(n) = \Omega(\sqrt{n/\log(n)})$, and in NP $\cap$ coNP for $\gamma(n) = \Omega(\sqrt{n})$. The problem can also be approximated within $\gamma = 1 + \epsilon$ (for any constant $\epsilon > 0$) in random exponential time [20], $\gamma(n) = e^{O(n\log\log n)/\log n)}$ in random polynomial time, and $\gamma(n) = e^{O(n(\log\log n)^2/\log n)}$ in deterministic polynomial time.

Definition 2.7 *The* Closest Vector Problem *(*CVP*), given a lattice basis* $\mathbf{B}$ *and target vector* $\mathbf{t}$*, asks for a lattice point* $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ *such that* $\mathrm{dist}(\mathbf{t}, \mathbf{v}) \leq \gamma(n) \cdot \mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B}))$*. The problem can be defined also in a* distance estimation *version, where given a basis* $\mathbf{B}$ *and target* $\mathbf{t}$*, one only has to find a value* $\hat{d}$ *such that* $\mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B})) \leq \hat{d} \leq \gamma(n) \cdot \mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B}))$*. The promise problem naturally associated to* CVP*, (denoted* GAPCVP$_\gamma$*) is, given* $(\mathbf{B}, \mathbf{t}, d)$ *where* $\mathbf{B}$ *is a lattice basis,* $\mathbf{t}$ *a target vector, and* d *is a (rational) number, decide if* $\mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B})) \leq d$ *or* $\mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B})) > \gamma(n) \cdot d$*.*

[13] A reduction for the exact case ($\gamma = 1$) is given in [22]. This is the only direct reduction known to date. Technically, a (trivial) reduction between the two problems also exists for approximation factors γ for which approximating λ_1 is NP-hard or finding short vectors is solvable in polynomial time. No reduction is known for any other intermediate approximation factor.

The closest vector problem is known to be at least as hard as the shortest vector problem [19] for any approximation factor $\gamma(n)$. Moreover, it is NP-hard for quasi-polynomial approximation factors $\gamma(n) = n^{1/O(\log\log n)}$ [13]. For $\gamma(n) = \Omega(\sqrt{n/\log n})$ the problem is in coAM [17], and for $\gamma(n) = \Omega(\sqrt{n})$ it is also in coNP [1]. (See [26, 7, 38] for earlier results with approximation factor $\gamma(n) = \Omega(n)$.) Finally, the problem can be approximated in deterministic polynomial time within $\gamma(n) = e^{\Omega(n(\log\log n)^2/\log n)}$ [41, 22].

In the closest vector problem, the target point $\mathbf{t}$ can be arbitrarily far from the lattice. In coding theory, Vardy [42] has considered a variant of the closest vector problem where the distance of the target from the code is guaranteed to be less than the packing radius of the code. This problem (called the *bounded distance decoding* problem, BDD) is interesting because decoding within the packing radius, if solvable, has unique solution. (For this reason, the packing radius is sometime called also the "unique decoding" radius.) For lattices, the analogous problem would be the following: given a lattice $\mathbf{B}$ and a point $\mathbf{t}$ within distance $d < \lambda_1(\mathcal{L}(\mathbf{B}))/2$ from $\mathcal{L}(\mathbf{B})$, find the (unique) lattice point within distance d from $\mathbf{t}$. In general we can consider a similar problem for values of d different from $\lambda_1(\mathcal{L}(\mathbf{B}))/2$, although when $d \geq \lambda_1(\mathcal{L}(\mathbf{B}))/2$ the solution is not necessarily unique. We consider the case when $d = \rho(\mathcal{L}(\mathbf{B}))$ equals the covering radius of the lattice. This case is interesting because there is always a lattice point within distance $\rho(\mathcal{L}(\mathbf{B}))$ from the target. Below we formally define an approximation version of this problem. Since for any lattice $\mathcal{L}(\mathbf{B})$ and target $\mathbf{t}$, there is always a lattice point within distance $\rho(\mathcal{L}(\mathbf{B}))$ from $\mathbf{t}$, we do not define distance estimation or promise versions of this problem.

Definition 2.8 *The* Guaranteed Distance Decoding *problem* (GDD_γ)*, given a lattice* $\mathbf{B}$ *and a target point* $\mathbf{t} \in \mathrm{span}(\mathbf{B})$*, asks for a lattice point* $\mathbf{x} \in \mathcal{L}(\mathbf{B})$ *such that* $\mathrm{dist}(\mathbf{t}, \mathbf{x}) \leq \gamma(n)\rho(\mathcal{L}(\mathbf{B}))$*, where* n *is the rank of the lattice.*

The following relations are known among the parameters of a lattice $\mathcal{L}(\mathbf{B})$.

Proposition 2.9 *For any rank* n *lattice* $\mathbf{B}$*,*

$$\lambda_1(\mathcal{L}(\mathbf{B})) \leq \lambda_n(\mathcal{L}(\mathbf{B})) \leq 2\rho(\mathcal{L}(\mathbf{B})) \leq \sqrt{n}\lambda_n(\mathcal{L}(\mathbf{B})). \tag{2.3}$$

Moreover, if $\mathcal{L}(\mathbf{B})^*$ *is the dual lattice of* $\mathcal{L}(\mathbf{B})$*, then*

$$1 \leq \lambda_1(\mathcal{L}(\mathbf{B}))2\rho(\mathcal{L}(\mathbf{B})^*) \leq n \tag{2.4}$$

and

$$1 \leq \lambda_1(\mathcal{L}(\mathbf{B}))\lambda_n(\mathcal{L}(\mathbf{B})^*) \leq n. \tag{2.5}$$

Proof: See [36, Theorem 7.9] for (2.3) and [7] for (2.4) and (2.5). □

2.4 Lattices and Groups

Let $\mathcal{L}(\mathbf{L})$ be a lattice. Any sublattice $\mathcal{L}(\mathbf{M}) \subseteq \mathcal{L}(\mathbf{L})$ defines a natural equivalence relation on $\mathcal{L}(\mathbf{L})$ as follows: two lattice points $\mathbf{x}, \mathbf{y} \in \mathcal{L}(\mathbf{L})$ are equivalent (written $\mathbf{x} \equiv_\mathbf{M} \mathbf{y}$) if and only if $\mathbf{x} - \mathbf{y} \in \mathcal{L}(\mathbf{M})$. The reader can easily check that $\equiv_\mathbf{M}$ is an equivalence relation, i.e., it is reflexive ($\mathbf{x} \equiv_\mathbf{M} \mathbf{x}$), symmetric ($\mathbf{x} \equiv_\mathbf{M} \mathbf{y} \Leftrightarrow \mathbf{y} \equiv_\mathbf{M} \mathbf{x}$) and transitive ($\mathbf{x} \equiv_\mathbf{M} \mathbf{y} \wedge \mathbf{y} \equiv_\mathbf{M} \mathbf{z} \Rightarrow \mathbf{x} \equiv_\mathbf{M} \mathbf{z}$). The $\equiv_\mathbf{M}$-equivalence class of $\mathbf{x} \in \mathcal{L}(\mathbf{L})$ (denoted $[\mathbf{x}]_\mathbf{M}$) is the set of all $\mathbf{y} \in \mathcal{L}(\mathbf{L})$ such that $\mathbf{x} \equiv_\mathbf{M} \mathbf{y}$. The quotient $\mathcal{L}(\mathbf{L})/\mathcal{L}(\mathbf{M}) = \{[\mathbf{x}]_\mathbf{M} : \mathbf{x} \in \mathcal{L}(\mathbf{L})\}$ is the set of all $\equiv_\mathbf{M}$-equivalence classes of $\mathcal{L}(\mathbf{L})$. The equivalence relation $\equiv_\mathbf{M}$ is a congruence with respect to the addition operation, i.e., if $\mathbf{x} \equiv_\mathbf{M} \mathbf{x}'$ and $\mathbf{y} \equiv_\mathbf{M} \mathbf{y}'$, then $(\mathbf{x} + \mathbf{y}) \equiv_\mathbf{M} (\mathbf{x}' + \mathbf{y}')$. It follows that for any two equivalence classes $[\mathbf{x}]_\mathbf{M}$ and $[\mathbf{y}]_\mathbf{M}$, the sum $[\mathbf{x}]_\mathbf{M} + [\mathbf{y}]_\mathbf{M} = [\mathbf{x} + \mathbf{y}]_\mathbf{M}$ is well defined, i.e., it does not depend on the choice of representatives $\mathbf{x}$, $\mathbf{y}$, and the quotient $\mathcal{L}(\mathbf{L})/\mathcal{L}(\mathbf{M})$ is an additive group with the sum operation just described. Notice that if $\mathcal{L}(\mathbf{L})$ is regarded as an Abelian group, then sublattice $\mathcal{L}(\mathbf{M})$ is a subgroup of $\mathcal{L}(\mathbf{L})$ and $(\mathcal{L}(\mathbf{L})/\mathcal{L}(\mathbf{M}), +)$ is just the standard quotient group.

Group $\mathcal{L}(\mathbf{L})/\mathcal{L}(\mathbf{M})$ is finite if and only if $\mathcal{L}(\mathbf{M})$ is a full rank sublattice of $\mathcal{L}(\mathbf{L})$, in which case, the cardinality of the group is

$$\#(\mathcal{L}(\mathbf{L})/\mathcal{L}(\mathbf{M})) = \frac{\det(\mathcal{L}(\mathbf{M}))}{\det(\mathcal{L}(\mathbf{L}))}.$$

Elements of this group can be represented using several standard techniques, e.g., selecting a unique representative from each equivalence class. It is easy to see that for every equivalence class $[\mathbf{x}]_{\mathbf{M}}$ there exists a unique element $\mathbf{x}' \in \mathcal{L}(\mathbf{L}) \cap \mathcal{P}(\mathbf{M})$ such that $\mathbf{x} \equiv_{\mathbf{M}} \mathbf{x}'$. So, a possible set of (unique) representatives is given by the set

$$\mathcal{L}(\mathbf{L}) \cap \mathcal{P}(\mathbf{M})$$

of all lattice points that belong to the half open parallelepiped $\mathcal{P}(\mathbf{M})$. Given an arbitrary lattice point $\mathbf{x} \in \mathcal{L}(\mathbf{B})$, the corresponding representative can be efficiently computed as follows: write $\mathbf{x}$ as $\mathbf{Mz}$, let $z'_i = \lfloor z_i \rceil$ for all $i = 1, \ldots, n$, and set $\mathbf{x}' = \mathbf{M}(\mathbf{z} - \mathbf{z}')$.

The representation of group elements using vectors in $\mathcal{P}(\mathbf{M}) \cap \mathcal{L}(\mathbf{L})$, although polynomial, is not very efficient. In particular, the number of bits necessary to store a single group element can be much larger than $\log_2 \#G$. Other more efficient ways to represent group elements are possible, for example using the Hermite Normal Form, or Smith Normal Form. These representations allow to store group elements using only $\log |G|$ bits, and perform the group operations in linear time. The techniques described in this paper are largely independent from the way group elements are represented, so we do not elaborate on this any further, and refer the reader to [32, 36] for more details.

Later in this paper we need to sample elements from group $G = \mathcal{L}(\mathbf{L})/\mathcal{L}(\mathbf{M})$ uniformly at random. This can be easily done using an elementary group theoretic technique described in the following proposition.

Proposition 2.10 *Let G be a finite Abelian group and $g_1, \ldots, g_n$ a generating set for G. Then, if $d_1, \ldots, d_n$ are chosen uniformly at random in $\{1, \ldots, \#G\}$, then the group element*

$$g = \sum_{i=1}^{n} d_i g_i$$

is distributed uniformly at random over G.

Proof: Since elements $g_1, \ldots, g_n$ generate the entire group, we know that for any group element $a \in G$ there exists an integer vector $\mathbf{d}_a = [d_{a,1}, \ldots, d_{a,n}]$ such that $\sum_{i=1}^{n} d_{a,i} g_i = a$. Let K be the set of all integer vectors $\mathbf{d} = [d_1, \ldots, d_n]$ such that $\sum d_i g_i = 0$ (in G). Notice that for any $a \in G$ and $\mathbf{d} \in \mathbb{Z}^n$, $\sum_i d_i g_i = a$ if and only if $\mathbf{d} \in K + \mathbf{d}_a$. Therefore, if $d_1, \ldots, d_n$ are chosen uniformly at random in $\{1, \ldots, \#G\}$, then the probability that $\sum_i d_i g_i = a$ equals exactly the size of $(K + \mathbf{d}_a) \cap \{1, \ldots, \#G\}^n$ divided by $(\#G)^n$. Since K is periodic modulo $\#G$ (i.e., it is invariant under translations by vectors in $\#G \cdot \mathbb{Z}^n$), all sets $(K + \mathbf{d}_a) \cap \{1, \ldots, \#G\}^n$ have the same size, and the probability that $\sum_i d_i g_i = a$ is the same for all $a \in G$. $\square$

Of particular interest in this paper are quotient groups $G = \mathcal{L}(\mathbf{L})/\mathcal{L}(\mathbf{M})$ where $\mathbf{M}$ defines an almost *orthogonal* sublattice of $\mathcal{L}(\mathbf{L})$. The following lemma gives a possible way to build almost orthogonal sublattices for any input lattice $\mathcal{L}(\mathbf{L})$.

Lemma 2.11 *Let $\mathcal{L}(\mathbf{B})$ be a lattice of rank n, σ a positive real, and D be a decoding procedure that on input a vector $\mathbf{y} \in \mathrm{span}(\mathbf{B})$ returns a lattice point $D(\mathbf{y}) \in \mathcal{L}(\mathbf{B})$ such that $\mathrm{dist}(D(\mathbf{y}), \mathbf{y}) \leq \sigma$. For any $\alpha \geq 2\sqrt{n} \cdot \sigma$, one can efficiently find (with n calls to D) a basis of a full rank sublattice $\mathbf{S} \subset \mathcal{L}(\mathbf{B})$ such that for all $\mathbf{x} \in \mathbb{R}^n$*

$$\|\mathbf{Sx}\| \approx \alpha \cdot \|\mathbf{x}\|.$$

Proof: Let $\mathbf{s}_i = D(\alpha \cdot \mathbf{t}_i)$, where $\mathbf{t}_1, \ldots, \mathbf{t}_n$ are an orthonormal basis of $\mathrm{span}(\mathbf{B})$, e.g., if $\mathcal{L}(\mathbf{B})$ is a full rank lattice, set $\mathbf{t}_i = \mathbf{e}_i$. Clearly $\mathbf{s}_i \in \mathcal{L}(\mathbf{B})$ for all $i = 1, \ldots, n$. Let $\mathbf{x} \in \mathbb{R}^n$ be an arbitrary vector. We want to prove that $\|\mathbf{Sx}\| \approx \alpha \cdot \|\mathbf{x}\|$. We know that $\mathbf{s}_i = \alpha \cdot \mathbf{t}_i + \mathbf{r}_i$ where $\|\mathbf{r}_i\| = \|D(\alpha \cdot \mathbf{t}_i) - \alpha \cdot \mathbf{t}_i\| \leq \sigma$. Therefore,

$$\|\mathbf{Sx}\| = \|(\alpha \cdot \mathbf{T} + \mathbf{R})\mathbf{x}\| = \|\alpha \cdot \mathbf{Tx} + \mathbf{Rx}\|.$$

By triangle inequality, and using $\|\mathbf{Tx}\| = \|\mathbf{x}\|$ (which follows from the fact that $\mathbf{T}$ is an orthonormal set of vectors) we get

$$\alpha \cdot \|\mathbf{x}\| - \|\mathbf{Rx}\| \leq \|\mathbf{Sx}\| \leq \alpha \cdot \|\mathbf{x}\| + \|\mathbf{Rx}\|.$$

So, we need to prove that $\|\mathbf{Rx}\| \leq \frac{\alpha}{2}\|\mathbf{x}\|$. By the triangle inequality and Cauchy-Schwarz,

$$\|\mathbf{Rx}\| \leq \sum_{i=1}^{n} \|\mathbf{r}_i\| \cdot |x_i| \leq \sigma \cdot \sum_{i=1}^{n} |x_i| \leq \sqrt{n}\sigma \cdot \|\mathbf{x}\| \leq \frac{\alpha}{2} \cdot \|\mathbf{x}\|.$$

This proves that $\|\mathbf{Sx}\| \approx \alpha \cdot \|\mathbf{x}\|$. The linear independence of vectors $\mathbf{S}$ immediately follows because if $\mathbf{S}$ were linearly dependent, then one could find a nonzero vector $\mathbf{x}$ such that $\mathbf{Sx} = \mathbf{0}$, contradicting $\|\mathbf{Sx}\| \approx \alpha \cdot \|\mathbf{x}\| > 0$. □

So far, we have shown how to use lattices and sublattices to define finite Abelian groups. It is also possible to use finite Abelian groups to define lattices.

Proposition 2.12 *Let G be a finite Abelian group, and $g_1, \ldots, g_n$ a sequence of elements of G. Then, the set*

$$\Lambda(g_1, \ldots, g_n) = \left\{ \mathbf{x} \in \mathbb{Z}^n : \sum_{i=1}^{n} x_i g_i = 0 \right\}$$

is a lattice, and its determinant satisfies $\det(\Lambda(g_1, \ldots, g_n)) \leq \#G$, with equality if and only if $g_1, \ldots, g_n$ generate the entire group G.

Proof: $\Lambda(g_1, \ldots, g_n)$ is a lattice because it is an additive subgroup of $\mathbb{Z}^n$. Let G' be the subgroup generated by $g_1, \ldots, g_n$. Notice that the quotient $\mathbb{Z}^n/\Lambda(g_1, \ldots, g_n)$ is isomorphic to G', with isomorphism given by $\phi([\mathbf{x}]) = \sum_i x_i g_i$. It follows that the size of G' is $\det(\Lambda(g_1, \ldots, g_n))/\det(\mathbb{Z}^n) = \det(\Lambda(g_1, \ldots, g_n))$, and $\det(\Lambda(g_1, \ldots, g_n)) = \#G' \leq \#G$. □

2.5 Statistical distance

The statistical distance is a measure of how two probability distributions are far apart from each other, and it is a convenient tool in the analysis of randomized algorithms and reductions. In this subsection we define the statistical distance and state some simple facts that will be used in the analysis of the algorithms in this paper. All the properties of the statistical distance stated in this subsection are easily verified. For more details the reader is referred to [36, Chapter 8].

Definition 2.13 *Let X and Y be two discrete random variables over a (countable) set A. The statistical distance between X and Y is the quantity*

$$\Delta(X, Y) = \frac{1}{2} \sum_{a \in A} |\Pr\{X = a\} - \Pr\{Y = a\}|.$$

We say that two random variables X, Y are identically distributed (written $X \equiv Y$) if and only if $\Pr\{X = a\} = \Pr\{Y = a\}$ for every $a \in A$. The reader can easily check that the statistical distance satisfies the usual properties of distance functions, i.e., $\Delta(X, Y) \geq 0$ (with equality if and only if $X \equiv Y$), $\Delta(X, Y) = \Delta(Y, X)$, and $\Delta(X, Z) \leq \Delta(X, Y) + \Delta(Y, Z)$.

The following proposition shows that applying a (possibly randomized) function to two distributions does not increase the statistical distance.

Proposition 2.14 *Let X, Y be two random variables over a common set A. For any (possibly randomized) function f with domain A, the statistical distance between $f(X)$ and $f(Y)$ is at most*

$$\Delta(f(X), f(Y)) \leq \Delta(X, Y) \tag{2.6}$$

Another useful property of the statistical distance is the following.

Proposition 2.15 *Let $X_1, \ldots, X_k$ and $Y_1, \ldots, Y_k$ be two lists of totally independent random variables. Then*

$$\Delta((X_1, \ldots, X_k), (Y_1, \ldots, Y_k)) \leq \sum_{i=1}^{k} \Delta(X_i, Y_i). \tag{2.7}$$

The next proposition and corollary show how to use the statistical distance to estimate expectations and probabilities.

Proposition 2.16 *If X and Y are random variables over set A and $f: A \to [a,b]$ is a real valued function, then*

$$|\operatorname{Exp}[f(X)] - \operatorname{Exp}[f(Y)]| \leq |b-a| \cdot \Delta(X,Y) \tag{2.8}$$

As a corollary, we immediately obtain the following.

Corollary 2.17 *If X and Y are random variables over set A and $p: A \to \{0,1\}$ is a predicate, then*

$$|\Pr[p(X)=1] - \Pr[p(Y)=1]| \leq \Delta(X,Y). \tag{2.9}$$

The following proposition gives a standard amplification technique that allows to generate almost uniform samples from a group by adding a relatively small number of independent samples that are not too far from uniform.

Proposition 2.18 *Let $(G,+)$ be a finite group and let $A_1, \ldots, A_k$ be k independent (but possibly not identically distributed) random variables over G such that $\Pr\{A_i = g\} \approx 1/\#G$ for all $i = 1, \ldots, k$ and any $g \in G$. Then, for any $g \in G$,*

$$\Pr\left\{\sum_{i=1}^{k} A_i = g\right\} \in \frac{1}{\#G} \cdot \left[1 \pm \frac{1}{2^k}\right].$$

In particular, the statistical distance between the sum $A = \sum_{i=1}^{k} A_i$ and the uniform distribution U over G is at most

$$\Delta\left(\sum_{i=1}^{k} A_i, U\right) \leq \frac{1}{2^{k+1}}.$$

The next proposition gives a way to estimate how much a given distribution is affected by conditioning.

Proposition 2.19 *For any random variable X over set A, and event Y, the statistical distance between distribution X and the conditional distribution of X given Y is exactly half the expected relative additive error of Y given X, i.e.,*

$$\Delta(X,(X|Y)) = \frac{1}{2}\operatorname*{Exp}_X\left[\left|\frac{\Pr\{Y \mid X\}}{\Pr\{Y\}} - 1\right|\right] \leq \frac{1}{2}\max_{a \in A}\left|\frac{\Pr\{Y|X=a\}}{\Pr\{Y\}} - 1\right|.$$

2.6 Iterative algorithms and reductions

Many lattice algorithms work by first computing a relatively poor solution to the problem in question, and then iteratively improving it until a solution meeting some desired condition is found. Examples of such iterative algorithms are the LLL basis reduction algorithm [27], and Ajtai worst- to average-case reduction and variants [2, 11, 35, 34]. Many other fundamental algorithms can also be described as an iterative process, e.g., Euclid's algorithm starts from two input numbers a, b, and iteratively computes smaller and smaller numbers until the greatest common divisor of a and b is found.

The high level structure of many iterative algorithms is the same and it can be formulated abstractly without any reference to the specific problem in question. Although the method is now standard and it has been repeatedly used to solve many lattice problems, it has never been explicitly formulated. In order to avoid unnecessary repetitions, and highlight both the similarities and differences among all these algorithms, below we give an abstract formulation of this iterative method, and present a general proof that the method implies standard polynomial time algorithms and reductions.

Computational problems can be abstractly defined by giving a binary relation consisting of all problem-solution pairs.

Definition 2.20 *The language associated to a binary relation R is the set L_R of all strings x such that $(x,w) \in R$ for some w. The problem defined by relation R is: given a string $x \in L_R$, find a w such that $(x,w) \in R$. For any $(x,w) \in R$, we say that w is a solution to problem x.*

Since no polynomial time algorithm (or reduction) can possibly output a solution w of size superpolynomial in $|x|$, it is usually assumed that there exists a polynomial p such that $|w| \leq p(|x|)$ for all $(x, w) \in R$. This is the case for all problems considered in this paper. However, we remark that in general relation R is not required to be polynomial time computable. For example, in the γ approximate shortest vector problem, membership in the associated relation $R = \{(\mathbf{B}, \mathbf{v})\colon \mathbf{v} \in \mathcal{L}(\mathbf{B}) \setminus \{\mathbf{0}\} \wedge \|\mathbf{v}\| \leq \gamma \cdot \lambda_1(\mathcal{L}(\mathbf{B}))\}$ is not polynomial time computable for $\gamma < \sqrt{2}$ (unless NP = RP, [33]). If membership in R is polynomial time computable (and there exists a polynomial p such that $|w| \leq p(|x|)$ for all $(x, w) \in R$), then R is an NP-relation.

A deterministic algorithm $\mathcal{F}$ solves problem R if for any $x \in L_R$, the output $w = \mathcal{F}(x)$ satisfies $(x, w) \in R$. If $\mathcal{F}$ is a randomized algorithm, then we say that $\mathcal{F}$ solves R *in the worst-case* with probability p (possibly a function of the input length) if for any $x \in L_R$, the probability (over the internal randomness of $\mathcal{F}$ alone) that $(x, \mathcal{F}(x)) \in R$ is at least $p(|x|)$. Algorithm $\mathcal{F}$ solves R *on the average* if $(x, \mathcal{F}(x)) \in R$ with probability at least $p(|x|)$, when the probability is computed over the internal randomness of $\mathcal{F}$ and the random selection of the input x according to some specified distribution over all strings in L_R of length $|x|$.

For all worst-case problems considered in this paper, given two tentative solutions w_0, w_1 for a problem x, it is possible to efficiently select the "best" of the two, i.e., there is a polynomial time algorithm that selects a w_i such that if $(x, w_0) \in R$ or $(x, w_1) \in R$, then $(x, w_i) \in R$. It follows that any algorithm that solves the problem with non-negligible probability $p(|x|) \geq |x|^{-O(1)}$, can be easily transformed into an algorithm that solves the problem with probability exponentially close to 1, by repeatedly executing the basic algorithm a polynomial (e.g., $|x| \cdot p(|x|)$) number of times and selecting the best solution. So, we describe any such algorithm as solving the problem with *high probability*, without explicitly stating the exact value of the success probability. Notice, however, that this applies only to (randomized) algorithms that solve a problem in the *worst-case*. The success probability amplification technique we just described may not work when applied to an algorithm that solves a problem with non-negligible probability, but only *on the average*, when the input is chosen at random.

Before we give the general definition of iterative reduction (or algorithm), we illustrate it with a familiar example. Consider Euclid's algorithm. The input is a pair of numbers $x = (a, b)$, and we want to find the greatest common divisor $w = \gcd(a, b)$. The algorithm works iteratively, maintaining at each iteration a pair $s = (s_0, s_1)$ such that $s_0 \geq s_1$ and $\gcd(s_0, s_1) = w$. At every iteration, if $s_1 \neq 0$ the state is updated from (s_0, s_1) to $(s_1, s_0 \bmod s_1)$. Polynomial time termination is guaranteed because at every iteration the function $f(s_0, s_1) = s_0 \cdot s_1 + 1$ decreases at least by a factor 2. When $s_1 = 0$ no more progress is possible, and the algorithm terminates with output s_0. In the case of lattice problems, the input is usually a lattice basis $x = \mathbf{B}$, and the algorithm maintains a set of linearly independent vectors $s = \mathbf{S} \subset \mathcal{L}(\mathbf{B})$, or a basis for the input lattice $\mathcal{L}(\mathbf{S}) = \mathcal{L}(\mathbf{B})$. Typically, the goal is to find a set of short vectors $\mathbf{S}$. This set is found by initially setting $\mathbf{S}$ to the input basis, and then iteratively applying an algorithm that on input $\mathbf{B}$ and $\mathbf{S}$ outputs a better set $\mathbf{S}'$. This general idea is formalized in the following definition. (See Figure 1 for a pictorial representation of the intended use of iterative reductions, and the proof of Theorem 2.22 below for further explanations.)

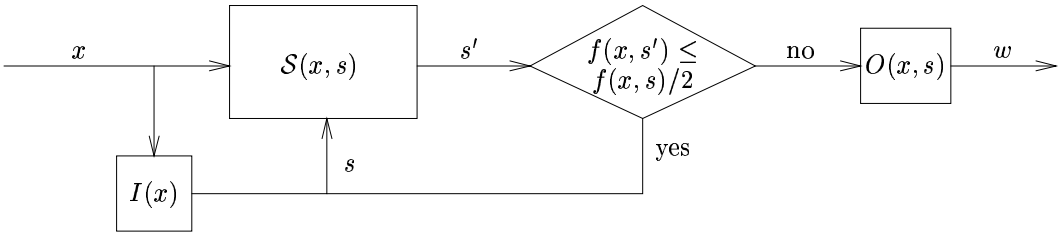

Figure 1: Iterative reduction

Definition 2.21 *An* iterative reduction *from problem P to a target problem P' is a tuple $(R, f, I, O, \mathcal{S})$ where*

- *R is a relation (defining the set $R_x = \{s\colon (x, s) \in R\}$ of valid internal states) such that $|s| \leq p(|x|)$ for some polynomial p and all $(x, s) \in R$.*

- $f : R \to \mathbb{Q}^+$ *(the* progress function*) is a deterministic polynomial time computable function, mapping pairs* $(x, s) \in R$ *to positive rational numbers* $f(x, s) \in \mathbb{Q}^+$*,*
- I *(the* initialization function*) is a deterministic polynomial time computable function mapping each problem instance* $x \in L_R$ *to an initial state* $I(x)$ *satisfying* $(x, I(x)) \in R$*,*
- O *(the* output function*) is a deterministic polynomial time computable function mapping valid pairs* $(x, s) \in R$ *to tentative solutions* $O(x, s)$*, possibly satisfying* $(x, O(x, s)) \in P$*,*
- $\mathcal{S}^{(\cdot)}$ *(the* iterative step*) is a (probabilistic) polynomial time oracle algorithm such that for any oracle* $\mathcal{F}$ *solving problem* P' *(with high probability), and for all* $(x, s) \in R$*, the output* $s' = \mathcal{S}^{\mathcal{F}}(x, s)$ *satisfies*
  - $(x, s') \in R$*, and*
  - *if* $P(x, O(x, s))$ *is false, then* $f(x, s') \leq f(x, s)/2$ *with high probability (over the random choices of* $\mathcal{S}$ *and* $\mathcal{F}$*).*

The following theorem shows that iterative reductions easily imply the existence of standard randomized Cook reductions. We remark that the notion of iterative reduction formulated in Definition 2.21 and used in Theorem 2.22 below, considers both P and P' as *worst-case* problems. The definition can be easily extended to the case where P' is an average case problem, however this is not needed in this paper. In Section 6 we will show that the iterative reduction implicit in Ajtai's worst-case to average-case connection and variants can be easily factored into an iterative reduction between two worst-case problems, and a worst- to average-case reduction that involve not iteration.

Theorem 2.22 *If there is an iterative reduction from problem* P *to problem* P'*, then there is a polynomial time reduction from solving* P *(in the worst case and with high probability) to solving* P' *(in the worst case and with high probability).*

Proof: Let $(R, f, I, O, \mathcal{S})$ be an iterative reduction from P to P'. We define a standard polynomial time reduction between the two problems. Assume that given oracle access to a solution $\mathcal{F}$ to problem P', $\mathcal{S}^{\mathcal{F}}$ satisfies the last condition in the definition with non-negligible probability $\delta(|x|)$. The reduction works as follows (see Figure 1):

1. On input x, compute $s = I(x)$.
2. Compute $\mathcal{S}^{\mathcal{F}}(x, s) = s'$ and check if $f(x, s') \leq f(x, s)/2$.
3. If the check succeeds, replace s with s' and repeat the previous step.
4. If the check fails, leave s unchanged, and repeat up to $|x|/\delta(|x|)$ times.
5. If the check fails for $|x|/\delta(|x|)$ consecutive times, then terminate and output $w = O(x, s)$.

It is immediate to verify that the algorithm satisfies the invariant $(x, s) \in R$ at all iterations. We need to prove that the running time is polynomial in $|x|$ and that the final output satisfies $P(x, w)$ with high probability. For the running time, notice that since $(x, s) \in R$, $|s|$ is bounded by a fixed polynomial in $|x|$ in all iterations, and all steps can be performed in polynomial time. We need to bound the number of iterations. Let p be a polynomial such that $|s| \leq p(|x|)$ for all $(x, s) \in R$, and let q be a polynomial bounding the running time of f. It follows that the initial value of $f(x, s)$ is at most $2^{q(|x|+p(|x|))}$, and that $f(x, s)$ is always at least $2^{-q(|x|+p(|x|))}$. Since at every successful iteration $f(x, s)$ decreases by a factor 2, and the algorithm terminates upon failing $|x|/\delta(|x|)$ consecutive times, the maximum number of iterations is at most $2q(|x| + p(|x|)) \cdot |x|/\delta(|x|)$, which is polynomial in $|x|$ for any non-negligible function $\delta(\cdot)$.

Finally, let's prove correctness. Consider the conditional probability of terminating with an incorrect answer, given that the algorithm performs exactly t update operations. We know that $(x, s) \in R$ right after the last successful update, and assume the algorithm outputs an incorrect answer $w = O(x, s)$, i.e., $P(x, w)$ is false. But we know that when $(x, s) \in R$ and $P(x, O(x, s))$ is false, $\mathcal{S}^{\mathcal{F}}(x, s)$ produces a new s' satisfying condition $f(x, s') \leq f(x, s)/2$ with probability at least $\delta(|x|)$. So, the probability that $\mathcal{S}$ fails $|x|/\delta(|x|)$ consecutive times and terminates with output $w = O(x, s)$ is at most $(1 - \delta(|x|))^{|x|/\delta(|x|)} \leq e^{-|x|}$. Adding up the conditional probabilities for all possible values of t, we get that the overall probability of terminating with the wrong answer is at most $2q(|x| + p(|x|))/e^{|x|}$ which is exponentially small. □

3 Covering radius and uniform radius

Let $\mathcal{L}(\mathbf{B})$ be an n-dimensional lattice and let $\mathcal{Q}$ be a convex body in $\mathbb{R}^n$. It can be shown that if we consider a randomly shifted copy of the body $\mathcal{Q}+\mathbf{x}$ (where $\mathbf{x}$ is chosen uniformly at random[14]), then the expected number of lattice points in $\mathcal{Q}+\mathbf{x}$ equals exactly

$$\operatorname*{Exp}_{\mathbf{x}}[\#(\mathcal{L}(\mathbf{B})\cap(\mathcal{Q}+\mathbf{x}))]=\frac{\mathrm{vol}(\mathcal{Q})}{\det(\mathcal{L}(\mathbf{B}))}.$$

In particular, if $\mathcal{Q}$ is a sphere of radius r, then

$$\operatorname*{Exp}_{\mathbf{x}}[\#(\mathcal{L}(\mathbf{B})\cap\mathcal{B}(\mathbf{x},r))]=\frac{\mathrm{vol}(\mathcal{B}(r))}{\det(\mathcal{L}(\mathbf{B}))}.$$

This corresponds to the intuition that the determinant $\det(\mathcal{L}(\mathbf{B}))$ is the inverse of the density of lattice points in space. Notice that the actual number of lattice points in a specific $\mathcal{Q}$ may deviate arbitrarily from the expectation, even for the special case of spherical $\mathcal{Q}$. Consider for example a lattice generated by two orthogonal vectors $\mathbf{e}_1$ and $D\mathbf{e}_2$, where D is a large constant. Notice that the determinant of the lattice is D, so on the average we would expect to find $\mathrm{vol}(\mathcal{Q})/D$ lattice points inside $\mathcal{Q}$. Now, let $\mathcal{Q}=\mathcal{B}(\mathbf{x},\sqrt{D})$ be the open disc of radius $\sqrt{D}$. The area of $\mathcal{Q}$ is $\mathrm{vol}(\mathcal{Q})=\pi D$, so on the average we would expect to find π lattice points in $\mathcal{Q}$. However, if $\mathbf{x}=0$, the number of lattice points in $\mathcal{Q}$ is $2\lceil\sqrt{D}\rceil-1$. Even worse, if $\mathbf{x}=(D/2)\mathbf{e}_2$, then $\mathcal{Q}$ does not contain any lattice point at all.

We define the *uniform radius* of a lattice as the smallest value $r=\zeta(\mathcal{L}(\mathbf{B}))$ such that any sphere $\mathcal{B}(\mathbf{x},r)$ contains a number of lattice points close to the expected value.

Definition 3.1 *For any n-dimensional lattice $\mathcal{L}(\mathbf{B})$, the* uniform radius $\zeta(\mathcal{L}(\mathbf{B}))$ *is the smallest positive real r such that*

$$\#(\mathcal{L}(\mathbf{B})\cap\mathcal{B}(\mathbf{x},r))\approx\frac{\mathrm{vol}(\mathcal{B}(r))}{\det(\mathcal{L}(\mathbf{B}))}$$

for any $\mathbf{x}\in\mathrm{span}(\mathcal{L}(\mathbf{B}))$.

The following proposition shows that the uniform radius $\zeta(\mathcal{L}(\mathbf{B}))$ is at least as large as the covering radius $\rho(\mathcal{L}(\mathbf{B}))$. Later we will also show that the uniform radius is never much bigger than that.

Proposition 3.2 *For any lattice $\mathcal{L}(\mathbf{B})$, $\rho(\mathcal{L}(\mathbf{B}))<\zeta(\mathcal{L}(\mathbf{B}))$.*

Proof: The proof is immediate because for $r\le\rho(\mathcal{L}(\mathbf{B}))$ any sphere of radius r centered in a deep hole (i.e., a point in space at distance $\rho(\mathcal{L}(\mathbf{B}))$ from the lattice) does not contain any lattice point. □

The uniform radius can be used to estimate the number of lattice points contained in a sphere. Later in this paper, we need to estimate the number of lattice points inside arbitrary convex bodies. So, we generalize the definition of the uniform radius to arbitrary convex bodies.

Definition 3.3 *For any n-dimensional lattice $\mathcal{L}(\mathbf{B})$, the* generalized uniform radius $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ *is the smallest positive real r such that for any convex body $\mathcal{Q}$ containing a sphere $\mathcal{B}(\mathbf{x},r)\subseteq\mathcal{Q}$ of radius r, the number of lattice points inside the body satisfies*

$$\#(\mathcal{L}(\mathbf{B})\cap\mathcal{Q})\approx\frac{\mathrm{vol}(\mathcal{Q})}{\det(\mathcal{L}(\mathbf{B}))}.$$

Clearly, the generalized uniform radius is at least as large as the uniform radius: for any lattice $\mathcal{L}(\mathbf{B})$, $\zeta(\mathcal{L}(\mathbf{B}))\le\hat{\zeta}(\mathcal{L}(\mathbf{B}))$. In particular, $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ is always larger than the covering radius $\rho(\mathcal{L}(\mathbf{B}))$. We bound the (generalized) uniform radius from above, and show that for any lattice $\mathcal{L}(\mathbf{B})$, the (generalized) uniform

[14] Intuitively, we would like to choose $\mathbf{x}$ uniformly at random from $\mathbb{R}^n$, but this is not possible because $\mathbb{R}^n$ has infinite measure. This problem is easily solved observing that it is enough to choose $\mathbf{x}$ uniformly at random from the fundamental region $\mathcal{P}(\mathbf{B})$ of the lattice, because the lattice repeats identically when translated by $\mathbf{Bx}$ for $\mathbf{x}\in\mathbb{Z}^n$.

radius is not much larger than the covering radius. Specifically, we show that $\hat{\zeta}(\mathcal{L}(\mathbf{B})) = O(n) \cdot \rho(\mathcal{L}(\mathbf{B}))$. A similar result was proved by Dyer, Frieze and Kannan in [14], for the special case of $\mathcal{L}(\mathbf{B}) = \mathbb{Z}^n$. We observe that the proof of [14] is a general volume argument and it does not use any special property of lattice $\mathbb{Z}^n$. So, it can be easily adapted to arbitrary lattices. Below we recall two simple geometric lemmas proved in [14], and then use them to prove the bound on $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$.

Lemma 3.4 ([14, Proposition 1]) *Suppose $\mathcal{Q}$ is a convex body in $\mathbb{R}^n$ containing the unit ball $\mathcal{B}(1)$, and let $\epsilon > 0$ be any positive real. Then all points within distance ϵ from $\mathcal{Q}$ belong to $(1+\epsilon)\mathcal{Q}$.*

Lemma 3.5 ([14, Proposition 2]) *Suppose $\mathcal{Q}$ is a convex body in $\mathbb{R}^n$ containing the unit ball $\mathcal{B}(1)$, and let $0 < \epsilon \leq 1$. Then, all points within distance ϵ from $(1-\epsilon)\mathcal{Q}$ belong to $\mathcal{Q}$.*

We can now prove the bound on the uniform radius in terms of the covering radius.

Theorem 3.6 *For any n-dimensional lattice $\mathcal{L}(\mathbf{B})$,*

$$\hat{\zeta}(\mathcal{L}(\mathbf{B})) \leq 3n\rho(\mathcal{L}(\mathbf{B})).$$

Proof: Let $\mathcal{L}(\mathbf{B})$ be a full rank lattice in $\mathbb{R}^n$ with covering radius $\rho(\mathcal{L}(\mathbf{B}))$, and let $\mathcal{Q}$ be a convex body containing a sphere of radius $r = 3n\rho(\mathcal{L}(\mathbf{B}))$. We want to prove that $\#(\mathcal{L}(\mathbf{B}) \cap \mathcal{Q}) \approx \frac{\mathrm{vol}(\mathcal{Q})}{\det(\mathcal{L}(\mathbf{B}))}$. Let $\mathcal{L}(\mathbf{B}') = \mathcal{L}(\mathbf{B})/r$ and $\mathcal{Q}' = \mathcal{Q}/r$ be scaled versions of $\mathcal{L}(\mathbf{B})$ and $\mathcal{Q}$, and consider the set of points

$$S = \mathcal{L}(\mathbf{B}') \cap \mathcal{Q}' = \frac{\mathcal{L}(\mathbf{B}) \cap \mathcal{Q}}{r}.$$

Clearly, $\#S = \#(\mathcal{L}(\mathbf{B}') \cap \mathcal{Q}') = \#(\mathcal{L}(\mathbf{B}) \cap \mathcal{Q})$. We want to prove that

$$\#S \approx \frac{\mathrm{vol}(\mathcal{Q}')}{\det(\mathcal{L}(\mathbf{B}'))} = \frac{\mathrm{vol}(\mathcal{Q})}{\det(\mathcal{L}(\mathbf{B}))}.$$

Consider the union of all Voronoi cells $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))$ with centers $\mathbf{x} \in S$. Notice that all points $\mathbf{y} \in \mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))$ are within distance $\rho(\mathcal{L}(\mathbf{B}')) = \rho(\mathcal{L}(\mathbf{B}))/r$ from $\mathbf{x}$. Moreover $\mathcal{Q}'$ contains a sphere of radius 1. Therefore, by Lemma 3.4, for all $\mathbf{x} \in S \subset \mathcal{Q}'$ and $\mathbf{y} \in \mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))$, we have $\mathbf{y} \in \mathcal{Q}' \cdot (1 + \rho(\mathcal{L}(\mathbf{B}))/r)$, i.e., $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}')) \subseteq (1 + \rho(\mathcal{L}(\mathbf{B}))/r) \cdot \mathcal{Q}'$. (Scaling, this time, performed using as origin the center of the unit sphere contained in $\mathcal{Q}'$.) Since Voronoi cells are disjoint and have the same volume, we have

$$\begin{aligned}
\#S &= \frac{\sum_{\mathbf{x} \in S} \mathrm{vol}\left(\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))\right)}{\mathrm{vol}(\mathcal{V}(\mathcal{L}(\mathbf{B}')))} \\
&= \frac{\mathrm{vol}\left(\bigcup_{\mathbf{x} \in S} \mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))\right)}{\mathrm{vol}(\mathcal{V}(\mathcal{L}(\mathbf{B}')))} \\
&\leq \frac{\mathrm{vol}(\mathcal{Q}' \cdot (1 + \rho(\mathcal{L}(\mathbf{B}))/r))}{\det(\mathcal{L}(\mathbf{B}'))} \\
&= \left(1 + \frac{\rho(\mathcal{L}(\mathbf{B}))}{r}\right)^n \frac{\mathrm{vol}(\mathcal{Q}')}{\det \mathcal{L}(\mathbf{B}')}.
\end{aligned}$$

Finally, using the assumption $r \geq 3n\rho(\mathcal{L}(\mathbf{B}))$, we get

$$\begin{aligned}
\left(1 + \frac{\rho(\mathcal{L}(\mathbf{B}))}{r}\right)^n &< \left(1 + \frac{1}{3n-1}\right)^n \\
&= \frac{1}{\left(1 - \frac{1}{3n}\right)^n} \\
&\leq \frac{1}{1 - \frac{1}{3}} = \frac{3}{2}.
\end{aligned}$$

This proves the upper bound $\#S \lesssim \mathrm{vol}(\mathcal{Q}')/\det \mathcal{L}(\mathbf{B}')$.

We now turn to the lower bound. Let S' be the set of all lattice points $\mathbf{x} \in \mathcal{L}(\mathbf{B}')$ such that the Voronoi cell $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))$ intersects $(1-\rho(\mathcal{L}(\mathbf{B}))/r)\mathcal{Q}'$. Notice that if $\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))$ intersects $(1-\rho(\mathcal{L}(\mathbf{B}))/r)\mathcal{Q}'$, then $\mathbf{x}$ must be within distance $\rho(\mathcal{L}(\mathbf{B}')) = \rho(\mathcal{L}(\mathbf{B}))/r$ from $(1-\rho(\mathcal{L}(\mathbf{B}))/r) \cdot \mathcal{Q}'$. So, by Lemma 3.5, $\mathbf{x} \in \mathcal{Q}'$. This proves that $S' \subseteq S$, and $\#S \geq \#S'$. Since Voronoi cells cover $\mathbb{R}^n$, $(1-\rho(\mathcal{L}(\mathbf{B}))/r)\mathcal{Q}'$ is fully contained in the union $\bigcup_{\mathbf{x}\in S'} \mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}'))$, and

$$\begin{aligned}
\#S' &= \frac{\sum_{\mathbf{x}\in S'} \mathrm{vol}(\mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}')))}{\mathrm{vol}(\mathcal{V}(\mathcal{L}(\mathbf{B}')))} \\
&= \frac{\mathrm{vol}(\bigcup_{\mathbf{x}\in S'} \mathcal{V}(\mathbf{x}, \mathcal{L}(\mathbf{B}')))}{\mathrm{vol}(\mathcal{V}(\mathcal{L}(\mathbf{B}')))} \\
&\geq \frac{\mathrm{vol}((1-\rho(\mathcal{L}(\mathbf{B}))/r)\mathcal{Q}')}{\det(\mathcal{L}(\mathbf{B}'))} \\
&= \left(1-\frac{\rho(\mathcal{L}(\mathbf{B}))}{r}\right)^n \frac{\mathrm{vol}(\mathcal{Q}')}{\det(\mathcal{L}(\mathbf{B}'))}.
\end{aligned}$$

Using the assumption $r \geq 3n\rho(\mathcal{L}(\mathbf{B}))$, we immediately get

$$\left(1-\frac{\rho(\mathcal{L}(\mathbf{B}))}{r}\right)^n > \left(1-\frac{1}{2n}\right)^n \geq 1-\frac{1}{2}.$$

This proves the lower bound $\#S \gtrsim \mathrm{vol}(\mathcal{Q}')/\det(\mathcal{L}(\mathbf{B}'))$, and completes the proof of the theorem. □

Using inequality $\rho(\mathcal{L}(\mathbf{B})) \leq \sqrt{n} \cdot \lambda_n(\mathcal{L}(\mathbf{B}))/2$ from (2.3) we can bound $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ in terms of $\lambda_n(\mathcal{L}(\mathbf{B}))$:

$$\hat{\zeta}(\mathcal{L}(\mathbf{B})) \leq \frac{3}{2} n^{1.5} \lambda_n(\mathcal{L}(\mathbf{B})). \tag{3.1}$$

Similarly, using transference theorem (2.4), we can bound $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ in terms of the length of the shortest nonzero vector in the dual lattice:

$$\hat{\zeta}(\mathcal{L}(\mathbf{B})) \leq \frac{3}{2} n^2 \frac{1}{\lambda_1(\mathcal{L}(\mathbf{B})^*)}. \tag{3.2}$$

These bounds can be used to relate the average-case complexity of finding small solutions to random equations to the worst-case complexity of approximating SIVP or GapSVP.

It would be interesting to improve bounds (3.1) and (3.2). In particular, is it true that $\hat{\zeta}(\mathcal{L}(\mathbf{B})) = O(n) \cdot \lambda_n(\mathcal{L}(\mathbf{B}))$ for any n-dimensional lattice $\mathcal{L}(\mathbf{B})$? Is it true that $\hat{\zeta}(\mathcal{L}(\mathbf{B})) = O(n)/\lambda_1(\mathcal{L}(\mathbf{B})^*)$? Proving these improved bounds would immediately result in a reduction of the connection factor for SIVP by a factor $O(\sqrt{n})$, and for GapSVP by a factor $O(n)$.

4 Easily decodable almost perfect lattices

We are interested in lattices that have both good algorithmic and geometric properties. Algorithmically, we would like lattices where the closest vector problem can be efficiently solved. Notice that, despite the NP-hardness of CVP, the closest vector problem may be efficiently solvable for *specific* lattices. For example, in the integer lattice $\mathbb{Z}^n$, a lattice vector $\mathbf{x} \in \mathbb{Z}^n$ closest to a given target $\mathbf{t} \in \mathbb{Q}^n$ can be easily found rounding each coordinate of $\mathbf{t}$ to the closest integer $x_i = \lfloor t_i \rceil$. Since for any fixed dimension the closest vector problem can be solved in polynomial time, in order to properly formulate this problem one needs to consider not a single lattice, but an infinite sequence of lattices in increasing dimension. For simplicity, in the definition below we focus on full rank lattices, although this restriction is not necessary.

Definition 4.1 *Let $\{\mathbf{L}_n\}_{n\geq 1}$ be a sequence of full rank lattices $\mathcal{L}(\mathbf{L}_n) \subseteq \mathbb{R}^n$. We say that the sequence $\{\mathbf{L}_n\}_{n\geq 1}$ is* easily decodable *if there exists a polynomial time algorithm $\mathrm{CVP}_{\mathbf{L}}$ such that for any $n \geq 1$ and $\mathbf{t} \in \mathbb{Q}^n$, $\mathrm{CVP}_{\mathbf{L}}(\mathbf{t})$ outputs a lattice vector in $\mathcal{L}(\mathbf{L}_n)$ closest to $\mathbf{t}$.*

The simplest example of easily decodable sequence of lattices is given by the integer lattices $\mathbb{Z}^n$ defined by matrices $\mathbf{L}_n = \mathbf{I}_n$. Other easily decodable lattices considered in [12] are the *root lattices* A_n, D_n, and their duals D_n^* and A_n^*.[15]

From a geometric point of view, we would like the Voronoi cells of the lattice to be as spherical as possible. Remember that the Voronoi cell $\mathcal{V}(\mathcal{L}(\mathbf{L}))$ contains a sphere $\mathcal{B}(\lambda_1(\mathcal{L}(\mathbf{L}))/2)$ with radius equal to the packing radius, and is completely contained in a sphere $\bar{\mathcal{B}}(\rho(\mathcal{L}(\mathbf{L})))$ with radius equal to the covering radius. So, the closer is the covering radius to the packing radius, the better are the Voronoi cells approximated by spheres. This motivates the following definition.

Definition 4.2 *For any $\tau \geq 1$, a lattice $\mathcal{L}(\mathbf{L})$ is τ-perfect if*

$$\rho(\mathcal{L}(\mathbf{L})) \leq \tau \cdot \left(\frac{\lambda_1(\mathcal{L}(\mathbf{L}))}{2} \right).$$

For any function $\tau(n)$, a sequence of (full rank) lattices $\{\mathbf{L}_n\}_{n\geq 1}$ (where n is the dimension of $\mathcal{L}(\mathbf{L}_n)$) is $\tau(n)$-perfect if $\mathcal{L}(\mathbf{L}_n)$ is $\tau(n)$-perfect for any $n \geq 1$. A sequence of (full rank) lattices $\{\mathbf{L}_n\}_{n\geq 1}$ is almost perfect if it is $\tau(n)$-perfect for some $\tau(n) = o(\sqrt{n})$.

We are interested in sequences of lattices such that $\tau(n)$ is as small as possible. Moreover, we would like the lattices to be easily decodable. The integer lattice $\mathbb{Z}^n$, as well as all other sequences A_n, A_n^*, D_n, D_n^* of easily decodable lattices considered in [12], are $\tau(n)$-perfect for $\tau(n) = \Theta(\sqrt{n})$, so they are not almost perfect. It is natural to ask if non-trivial easily decodable almost perfect lattices (i.e., $\tau(n)$-perfect lattices with $\tau(n) = o(\sqrt{n})$) exist, or the almost perfectness and easy decodability requirements are incompatible.

In this section we start the algorithmic study of almost perfect lattices and give the first efficient construction of non-trivial easily decodable almost perfect lattices. Our lattices are $\tau(n)$-perfect for $\tau(n) = \sqrt{n \log\log n / \log n} = o(\sqrt{n})$. Although this is not a substantial improvement over $\tau(n) = \Theta(\sqrt{n})$ from a quantitative point of view, it is qualitatively interesting because it shows that non-trivial easily decodable almost perfect lattices exist.

We first present a construction of 3-perfect lattices such that the construction and the decoding algorithm run in exponential time $n^{O(n)}$. Then we show how to use small dimensional lattices obtained using this construction to efficiently construct $O(\sqrt{n \log\log n / \log n})$-perfect lattices such that the closest vector problem can be solved in polynomial time. The construction is based on the following simple lemma.

Lemma 4.3 *For any lattice $\mathcal{L}(\mathbf{B})$, there exist a lattice vector $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ such that* $\text{dist}(\mathbf{v}/3, \mathcal{L}(\mathbf{B})) \geq (2/3)\rho(\mathcal{L}(\mathbf{B}))$. *In particular, if $\rho(\mathcal{L}(\mathbf{B})) \geq 3 \cdot \lambda_1(\mathcal{L}(\mathbf{B}))/2$ then* $\text{dist}(\mathbf{v}/3, \mathcal{L}(\mathbf{B})) \geq \lambda_1(\mathcal{L}(\mathbf{B}))$.

Proof: Let $\mathbf{h}$ be a deep hole, i.e., a point in $\text{span}(\mathcal{L}(\mathbf{B}))$ at distance $\rho(\mathcal{L}(\mathbf{B}))$ from $\mathcal{L}(\mathbf{B})$. Consider the point $3\mathbf{h}$, and let $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ be a lattice point closest to $3\mathbf{h}$. By definition of covering radius, it must be $\|\mathbf{v} - 3\mathbf{h}\| \leq \rho(\mathcal{L}(\mathbf{B}))$. Therefore, dividing by 3 we get $\|\mathbf{v}/3 - \mathbf{h}\| \leq \rho(\mathcal{L}(\mathbf{B}))/3$, and by triangle inequality

$$\text{dist}(\mathbf{v}/3, \mathcal{L}(\mathbf{B})) \geq \text{dist}(\mathbf{h}, \mathcal{L}(\mathbf{B})) - \text{dist}(\mathbf{v}/3, \mathbf{h}) \geq \rho(\mathcal{L}(\mathbf{B})) - \frac{1}{3}\rho(\mathcal{L}(\mathbf{B})) = \frac{2}{3} \cdot \rho(\mathcal{L}(\mathbf{B})).$$

□

We use the lemma to give an algorithmic construction of τ-perfect lattices with $\tau < 3$. The following theorem is essentially an algorithmic variant of the proof of existence given in [40]. Both the procedure to build the lattice and the one to decode it run in time $n^{O(n)}$. It should be noted that for any n-dimensional lattice, in principle the closest vector problem can always be solved in time $n^{O(n)}$ [23]. However, the algorithm of [23] for general lattices is rather complex. In the theorem below we show how to build a lattice $\mathcal{L}(\mathbf{B})$ together with some (polynomial size) auxiliary information $\mathbf{V}$ that allows to solve the closest vector problem in lattice $\mathcal{L}(\mathbf{B})$, still in time $n^{O(n)}$ as in [23], but with a much simpler algorithm.

[15] Conway and Sloane [12] also describe other efficient decoding algorithms for specific lattices, but $\mathbb{Z}^n, A_n, A_n^*, D_n, D_n^*$ are the only infinite sequences of lattices considered for which the problem of efficient decoding admits an interesting asymptotic formulation.

Theorem 4.4 *There is an algorithm running in time $n^{O(n)}$ that on input n outputs an n-dimensional 3-perfect lattice $\mathbf{L}_n$. Moreover, the sequence of lattices $\{\mathbf{L}_n\}_{n\geq 1}$ is decodable in time $n^{O(n)}$, i.e., there is an algorithm $\mathrm{CVP}_{\mathbf{L}}$ running in time $n^{O(n)}$ that on input a vector $\mathbf{t} \in \mathbb{Q}^n$ outputs a lattice vector $\mathrm{CVP}_{\mathbf{L}}(\mathbf{t}) \in \mathcal{L}(\mathbf{L}_n)$ closest to $\mathbf{t}$.*

Proof: The algorithm starts from an arbitrary n-dimensional easily decodable lattice $\mathcal{L}(\mathbf{B}_0)$, e.g., the integer lattice $\mathcal{L}(\mathbf{B}_0) = \mathbb{Z}^n$ generated by the identity matrix $\mathbf{B}_0 = \mathbf{I}$. Notice that the closest vector in $\mathbb{Z}^n$ to a target $\mathbf{t}$ can be easily found by rounding each coordinate of $\mathbf{t}$ to the closest integer. Below we assume that $\mathbf{B}_0 = \mathbf{I}$ and, in particular, $\det(\mathcal{L}(\mathbf{B}_0)) = 1$ and $\lambda_1(\mathcal{L}(\mathbf{B}_0)) = 1$, but the construction works for any easily decodable lattice.

Starting from $\mathbf{B}_0$, we iteratively build a sequence of lattice bases $\mathbf{B}_i$ and auxiliary vectors $\mathbf{v}_i$ for $i = 1, \ldots, m$ for some $m = O(n \log n)$ to be determined. The final output are basis $\mathbf{B} = \mathbf{B}_m$ and set of vectors $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$. For each $k = 1, \ldots, m$, vector $\mathbf{v}_k$ and basis $\mathbf{B}_k$ are computed as follows:

1. For any vector $\mathbf{s} \in \{-1, 0, +1\}^n$, let $\mathbf{t} = (1/3)\mathbf{B}_{k-1}\mathbf{s}$ and compute the distance of $\mathbf{t}$ from the lattice $\mathcal{L}(\mathbf{B}_{k-1})$. (We will show below how this can be done in time $n^{O(1)} \cdot 3^k$.)
2. If for all $\mathbf{s} \in \{-1, 0, +1\}^n$, $\mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B}_{k-1})) < 1$, then set $m = k - 1$, and terminate with output $\mathbf{B} = \mathbf{B}_{k-1}$ and $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_{k-1}]$.
3. Otherwise (if $\mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B}_{k-1})) \geq 1$ for some $\mathbf{s}$) proceed as follows. Notice that since $\mathrm{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B}_{k-1})) > 0$, it must be $\mathbf{s} \neq \mathbf{0}$.
4. Let $i \in \{1, \ldots, n\}$ such that $s_i \neq 0$.
5. Set $\mathbf{v}_k = \mathbf{t}$.
6. Set $\mathbf{B}_k$ to the matrix obtained by replacing the ith vector in $\mathbf{B}_{k-1}$ with $\mathbf{v}_k$.

The algorithm uses a procedure to find closest vectors in lattice $\mathcal{L}(\mathbf{B}_k)$. We will show that the maximum number of iterations performed by the algorithm is $m \leq (n/2)\log_3 n = O(n \log n)$, and that for any k, the closest vector problem in $\mathcal{L}(\mathbf{B}_k)$ can be solved in time $n^{O(1)} \cdot 3^k$. It follows that the total running time of the algorithm is

$$O(m \cdot n^{O(1)} \cdot 3^m) = n^{O(n)}$$

and that the closest vector problem in $\mathcal{L}(\mathbf{B})$ can also be solved in time $n^{O(n)}$.

The correctness of the algorithm is based on the fact that for any k,

- Vector $3\mathbf{v}_k$ belongs to the lattice $\mathcal{L}(\mathbf{B}_{k-1})$.
- $\mathbf{B}_k$ is a basis for the lattice generated by $[\mathbf{B}_{k-1}|\mathbf{v}_k]$.
- The shortest vector in $\mathcal{L}(\mathbf{B}_k)$ has length 1.

The first property immediately follows by construction. For the second property, it is clear that $\mathcal{L}(\mathbf{B}_k)$ is a subset of $\mathcal{L}([\mathbf{B}_{k-1}|\mathbf{v}_k])$. In order to prove $\mathcal{L}(\mathbf{B}_k) = \mathcal{L}([\mathbf{B}_{k-1}|\mathbf{v}_k])$ we only need to show that the ith vector of $\mathbf{B}_{k-1}$ (namely, $\mathbf{B}_{k-1}\mathbf{e}_i$) belongs to $\mathcal{L}(\mathbf{B}_k)$. Notice that $3\mathbf{v}_k = \mathbf{B}_{k-1}\mathbf{s} = \sum_j s_j \mathbf{B}_{k-1}\mathbf{e}_j$. So, $s_i \cdot \mathbf{B}_{k-1}\mathbf{e}_i = 3\mathbf{v}_k - \sum_{j\neq i} s_j \mathbf{B}_{k-1}\mathbf{e}_j = 3\mathbf{B}_k\mathbf{e}_i - \sum_{j\neq i} s_j \mathbf{B}_k\mathbf{e}_j$ belongs to $\mathcal{L}(\mathbf{B}_k)$. Since $s_i = \pm 1$, also $\mathbf{B}_{k-1}\mathbf{e}_i = \pm(s_i \cdot \mathbf{B}_{k-1}\mathbf{e}_i)$ belongs to $\mathcal{L}(\mathbf{B}_k)$. Now, let's get to the third property. Consider any nonzero vector in $\mathcal{L}(\mathbf{B}_k)$. Since $\mathcal{L}(\mathbf{B}_k) = \mathcal{L}([\mathbf{B}_{k-1}|\mathbf{v}_k])$, any such a vector can be written as $\mathbf{B}_{k-1}\mathbf{x} + \mathbf{v}_k \cdot y$. Moreover, since $3\mathbf{v}_k \in \mathcal{L}(\mathbf{B}_{k-1})$, we can assume without loss of generality that $y \in \{-1, 0, +1\}$. So, the length of $\mathbf{B}_{k-1}\mathbf{x} + \mathbf{v}_k \cdot y$ is at least the minimum of $\lambda_1(\mathcal{L}(\mathbf{B}_{k-1}))$ (if $y = 0$) or $\mathrm{dist}(\pm\mathbf{v}_k, \mathcal{L}(\mathbf{B}_{k-1}))$ (if $y = \pm 1$). But $\lambda_1(\mathcal{L}(\mathbf{B}_{k-1})) \geq 1$ by induction, and $\mathrm{dist}(\pm\mathbf{v}_k, \mathcal{L}(\mathbf{B}_{k-1})) = \mathrm{dist}(\mathbf{v}_k, \mathcal{L}(\mathbf{B}_{k-1})) \geq 1$ by construction. It follows that $\lambda_1(\mathcal{L}(\mathbf{B}_k)) \geq 1$.

It is also easy to see that for any k, the determinant of lattice $\mathcal{L}(\mathbf{B}_k)$ equals $\det(\mathcal{L}(\mathbf{B}_k)) = 3^{-k}\det(\mathcal{L}(\mathbf{B}_0)) = 3^{-k}$ because each $\mathbf{B}_k$ can be obtained from $\mathbf{B}_{k-1}$ by first performing some elementary integer column operations, and then dividing a column by 3. We can now prove that the algorithm performs at most $m = O(n \log n)$ iterations. Since $\lambda_1(\mathcal{L}(\mathbf{B}_k)) = 1$ and $\det(\mathcal{L}(\mathbf{B}_k)) = 3^{-k}$, by Minkowski's first theorem (2.2),

$$1 = \lambda_1(\mathcal{L}(\mathbf{B}_k)) \leq \sqrt{n}\det(\mathcal{L}(\mathbf{B}_k))^{1/n} = \sqrt{n}3^{-(k/n)}.$$

It follows that

$$k \leq (1/2)n \log_3 n = O(n \log n)$$

is an upper bound on the maximum number of iterations. (It can also be shown by a volume argument that $m = \Theta(n \log n)$ iterations are required in order to reach termination.)

Next we prove that upon termination $\rho(\mathcal{L}(\mathbf{L}_n)) < 3 \cdot \lambda_1(\mathcal{L}(\mathbf{L}_n))/2$. We show that if $\rho(\mathcal{L}(\mathbf{L}_n)) \geq (3/2) \cdot \lambda_1(\mathcal{L}(\mathbf{L}_n))$, then the algorithm certainly performs one more iteration. By Lemma 4.3, if $\rho(\mathcal{L}(\mathbf{L}_n)) \geq (3/2)\lambda_1(\mathcal{L}(\mathbf{L}_n))$ then there exists a vector $\mathbf{v} = \mathbf{B}_{k-1}\mathbf{x} \in \mathcal{L}(\mathbf{B}_{k-1})$ such that

$$\text{dist}(\mathbf{v}/3, \mathcal{L}(\mathbf{B}_{k-1})) \geq \lambda_1(\mathcal{L}(\mathbf{B}_{k-1})) \geq 1.$$

Let $\mathbf{s} \in \{-1, 0, +1\}^n$ be such that $\mathbf{s} \equiv \mathbf{x} \pmod 3$, i.e., $(\mathbf{s} - \mathbf{x})/3 \in \mathbb{Z}^n$. We claim that the distance of $\mathbf{t} = (1/3)\mathbf{B}_{k-1}\mathbf{s}$ from the lattice $\mathcal{L}(\mathbf{B}_{k-1})$ is at least 1. Notice that

$$\mathbf{t} = (1/3)\mathbf{B}_{k-1}\mathbf{s} = \mathbf{B}_{k-1}\mathbf{x}/3 + \mathbf{B}_{k-1}(\mathbf{s} - \mathbf{x})/3 \in \mathbf{v}/3 + \mathcal{L}(\mathbf{B}_{k-1}).$$

It follows that $\text{dist}(\mathbf{t}, \mathcal{L}(\mathbf{B}_{k-1})) = \text{dist}(\mathbf{v}/3, \mathcal{L}(\mathbf{B}_{k-1})) \geq 1$, and therefore the algorithm does not terminate at iteration k.

We conclude the proof of the theorem by giving a simple algorithm to solve the closest vector problem in $\mathcal{L}(\mathbf{B}_k)$ in time $n^{O(1)} \cdot 3^k \leq n^{O(n)}$. Notice that any lattice point in $\mathcal{L}(\mathbf{B}_k)$ can be written as $\mathbf{B}_0\mathbf{x} + [\mathbf{v}_1, \ldots, \mathbf{v}_k]\mathbf{y}$ where $\mathbf{x} \in \mathbb{Z}^n$ and $\mathbf{y} \in \{-1, 0, +1\}^k$. So, in order to find the lattice point closest to some target $\mathbf{t}$, we can consider all vectors of the form $\mathbf{t} - [\mathbf{v}_1, \ldots, \mathbf{v}_k]\mathbf{y}$ and compute their distance from $\mathcal{L}(\mathbf{B}_0)$. Let $\mathbf{y}$ such that $\text{dist}(\mathbf{t} - [\mathbf{v}_1, \ldots, \mathbf{v}_k]\mathbf{y}, \mathcal{L}(\mathbf{B}_k))$ is minimized, and let $\mathbf{B}_0\mathbf{x}$ the lattice vector closest to $\mathbf{t} - [\mathbf{v}_1, \ldots, \mathbf{v}_k]\mathbf{y}$. The lattice vector in $\mathcal{L}(\mathbf{B}_k)$ closest to $\mathbf{t}$ is $\mathbf{B}_0\mathbf{x} + [\mathbf{v}_1, \ldots, \mathbf{v}_k]\mathbf{y}$. □

The theorem gives an algorithmic construction of almost perfect lattices and an algorithm to solve the closest vector problem, however the running time is huge. The next theorem shows how to use these lattices for small values of n to get a construction that runs in polynomial time.

Theorem 4.5 *There exists a family of $\tau(n)$-perfect easily decodable lattices with $\tau(n) = O(\sqrt{n \log \log n / \log n})$.*

Proof: In order to keep the construction polynomial in n, we use Theorem 4.4 to build a 3-perfect lattice $\mathbf{M}$ in dimension $m = \log n / \log \log n$. Notice that such a lattice can be constructed in time

$$2^{O(m \log m)} = 2^{O(\log n \log(\log n / \log \log n) / \log \log n)} = n^{O(1)}.$$

Moreover, the closest vector problem in this lattice can also be solved in time $2^{O(m \log m)} = n^{O(1)}$.

Now set $\mathcal{L}(\mathbf{L}_n)$ to the direct sum of (n/m) copies of $\mathcal{L}(\mathbf{M})$, i.e. the lattice generated by the block diagonal matrix with n/m blocks, all equal to $\mathbf{M}$. The lattice vector in $\mathcal{L}(\mathbf{L}_n)$ closest to a target $\mathbf{t}$ is easily found by breaking $\mathbf{t}$ into n/m blocks, each with m coordinates in it, and finding the $\mathcal{L}(\mathbf{M})$ vector closest to each block. Moreover, the length of the shortest nonzero vector in $\mathcal{L}(\mathbf{L}_n)$ is $\lambda_1(\mathcal{L}(\mathbf{M}))$ because vectors from different copies of $\mathbf{M}$ are orthogonal. Finally, the covering radius of $\mathcal{L}(\mathbf{L}_n)$ is $\sqrt{n/m}$ times $\rho(\mathcal{L}(\mathbf{M}))$. So, $\mathcal{L}(\mathbf{L}_n)$ is $\tau(n)$-perfect for

$$\tau(n) = \frac{\sqrt{n/m}\rho(\mathcal{L}(\mathbf{M}))}{\lambda_1(\mathcal{L}(\mathbf{M}))/2} \leq 3\sqrt{n/m} = O(\sqrt{n \log \log n / \log n}).$$

□

5 A generalized class of random equations

In this section we define a class of random equations that generalizes Ajtai's one. Ajtai's problem can be described as finding a small (e.g., with respect to the bound proved in Theorem 5.5 below) nonzero integer solution to a homogeneous linear equation in $m(n)$ variables with coefficients chosen uniformly at random from the group $G_n = \mathbb{Z}_{q(n)}^n$ of n-dimensional vectors modulo $q(n)$, for appropriately chosen functions $q(n)$ and $m(n)$. Here we consider equations with coefficients in a group G_n possibly different from $\mathbb{Z}_{q(n)}^n$. In general, we define the following problem.

Definition 5.1 *Let $\{G_n\}$ be a sequence of finite Abelian groups, and $m(n)$ and $\beta(n)$ two arbitrary (polynomial time computable) functions. For any $m(n)$-dimensional vector $\mathbf{g} = [g_1, \ldots, g_{m(n)}]^T \in G_n^{m(n)}$, define the set of solutions to the associated homogeneous linear equation*

$$\Lambda(\mathbf{g}) = \left\{ \mathbf{z} \in \mathbb{Z}^{m(n)} \colon \sum_{i=1}^{m(n)} z_i \cdot g_i = 0 \right\}. \tag{5.1}$$

The Homogeneous Small Integer Solution *problem* $\mathrm{HSIS}_{G,m,\beta}$ *(in the ℓ_2 norm) is: given a (random) homogeneous linear equation $\mathbf{g} \in G_n^{m(n)}$, find a nonzero solution $\mathbf{z} \in \Lambda(\mathbf{g}) \setminus \{\mathbf{0}\}$ of ℓ_2 norm at most $\|\mathbf{z}\| \leq \beta(n)$.*

Of course, the problem is interesting only when a solution of length at most $\beta(n)$ exists. Below we define a family of groups (that includes Ajtai's groups as a special case), and give conditions under which a solution of length at most $\beta(n)$ is guaranteed to exist.

Our groups are parametrized by an easily decodable family of lattices $\{\mathcal{L}(\mathbf{L}_n)\}_{n>0}$ and a function $\alpha(n)$, and each group $G_n = \mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n))$ is defined as the quotient of $\mathcal{L}(\mathbf{L}_n)$ modulo an appropriately chosen full dimensional sublattice $\mathcal{L}(\mathbf{M}_n) \subseteq \mathcal{L}(\mathbf{L}_n)$.

Definition 5.2 *For any easily decodable family of lattices $\{\mathcal{L}(\mathbf{L}_n)\}_{n>0}$ (with decoding algorithm $\mathrm{CVP}_{\mathbf{L}}$) and function $\alpha(n)$ satisfying $\alpha(n) \geq 2\sqrt{n}\rho(\mathcal{L}(\mathbf{L}_n))$, define the sequence of quotient groups*

$$G_n = \mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n)) = \mathcal{L}(\mathbf{L}_n)/\mathcal{L}(\mathbf{M}_n) \tag{5.2}$$

where for any $n > 0$, $\mathcal{L}(\mathbf{M}_n)$ is the full rank sublattice obtained by applying Lemma 2.11 to lattice $\mathcal{L}(\mathbf{L}_n)$, value $\alpha(n)$ and decoding procedure $\mathrm{CVP}_{\mathbf{L}}$.

From Lemma 2.11 we immediately obtain the following corollary.

Corollary 5.3 *For any family of lattices $\{\mathcal{L}(\mathbf{L}_n)\}_{n>0}$ and function $\alpha(n)$ satisfying the conditions in Definition 5.2, the groups $\mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n)) = \mathcal{L}(\mathbf{L}_n)/\mathcal{L}(\mathbf{M}_n)$ can be computed in time polynomial in n and the matrix $\mathbf{M}_n$ satisfies*

$$\forall \mathbf{x} \in \mathbb{R}^n . \|\mathbf{M}_n \mathbf{x}\| \approx \alpha(n) \cdot \|\mathbf{x}\|.$$

Proof: Matrix $\mathbf{M}_n$ is polynomial time computable because lattice $\mathcal{L}(\mathbf{L}_n)$ is easily decodable, so the decoding procedure $\mathrm{CVP}_{\mathbf{L}}$ runs in polynomial time. In order to bound $\|\mathbf{M}_n \mathbf{x}\|$, simply observe that $\mathcal{L}(\mathbf{L}_n)$, $\alpha(n)$ and $\mathrm{CVP}_{\mathbf{L}}$ satisfy the conditions of Lemma 2.11 because $\alpha(n) \geq 2\sqrt{n}\rho(\mathcal{L}(\mathbf{L}_n)) \geq 2\sqrt{n}\,\mathrm{dist}(\mathbf{x}, \mathrm{CVP}_{\mathbf{L}}(\mathbf{x}))$ for all $\mathbf{x} \in \mathbb{R}^n$. □

Notice that if $\mathcal{L}(\mathbf{L}_n) = \mathbb{Z}^n$ is the integer lattice, and $\alpha(n) = q(n)$, then Definition 5.2 gives matrix $\mathbf{M}_n = q(n) \cdot \mathbf{I}$ and Ajtai's group $\mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n)) = \mathbb{Z}_{q(n)}^n$ as a special case. We will see that this choice of group G_n is not the best possible for our analysis, and setting $\mathcal{L}(\mathbf{L}_n)$ to an almost perfect lattice leads to better results. In the rest of this section, we prove that for any $\mathbf{g} \in G_n^{m(n)}$ and all sufficiently large $m(n)$, the set $\Lambda(\mathbf{g})$ always contains small nonzero solutions. The main result of this paper (proved in Section 6) is that although these small solutions are guaranteed to exist, they are computationally hard to find when $\mathbf{g}$ is chosen uniformly at random.

We know from Proposition 2.12 that $\Lambda(\mathbf{g})$ is a lattice with determinant at most $\det(\Lambda(\mathbf{g})) \leq \#G_n$. We show that $\Lambda(\mathbf{g})$ always contains small solutions by bounding the size of group G_n, and then applying Minkowski's first theorem.

Lemma 5.4 *For any n, the group $G_n = \mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n))$ defined in Definition 5.2 has size at most*

$$\#G_n \leq \left(\frac{3\alpha(n)\sqrt{n}}{2\lambda_1(\mathcal{L}(\mathbf{L}_n))} \right)^n.$$

Proof: The size of the group is $\#G_n = \det(\mathcal{L}(\mathbf{M}_n))/\det(\mathcal{L}(\mathbf{L}_n))$. We bound the two determinants separately. By Corollary 5.3, the columns of $\mathbf{M}_n$ have length at most

$$\|\mathbf{M}_n\mathbf{e}_i\| \leq (3/2)\alpha(n) \cdot \|\mathbf{e}_i\| = 3\alpha(n)/2.$$

Therefore, by Hadamard's inequality

$$\det(\mathcal{L}(\mathbf{M}_n)) \leq (3\alpha(n)/2)^n.$$

We bound the determinant of $\mathcal{L}(\mathbf{L}_n)$ using Minkowski's inequality (2.2) $\lambda_1(\mathcal{L}(\mathbf{L}_n)) \leq \sqrt{n}\det(\mathcal{L}(\mathbf{L}_n))^{1/n}$. Solving for $\det(\mathcal{L}(\mathbf{L}_n))$, we get that the determinant of $\mathcal{L}(\mathbf{L}_n)$ is at least $(\lambda_1(\mathcal{L}(\mathbf{L}_n))/\sqrt{n})^n$. Combining the two bounds, we get that group G_n has cardinality

$$\#G_n = \frac{\det(\mathcal{L}(\mathbf{M}_n))}{\det(\mathcal{L}(\mathbf{L}_n))} \leq \left(\frac{3\alpha(n)\sqrt{n}}{2\lambda_1(\mathcal{L}(\mathbf{L}_n))}\right)^n. \tag{5.3}$$

□

A bound on the size of the smallest nonzero solution to equation $\mathbf{g}$ easily follows from Proposition 2.12 and Minkowski's first theorem (2.2).

Theorem 5.5 *For any equation $\mathbf{g} \in G_n^{m(n)}$ in $m(n) = \Omega(n \log n)$ variables with coefficients in a group G_n of size $\#G_n \leq n^{O(n)}$ (e.g., $G_n = \mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n))$ for some $\alpha(n) = n^{O(1)} \cdot \lambda_1(\mathcal{L}(\mathbf{L}_n))$,) there exists a nonzero solution $\mathbf{z} \in \Lambda(\mathbf{g})$ of length at most $\|\mathbf{z}\| = O(\sqrt{m(n)})$.*

6 The worst- to average-case reduction

In this section we prove the main technical result of the paper. Namely, we show that finding short solutions to random linear equations as defined in Section 5 (on the average and with non-negligible probability) is at least as hard as finding linearly independent vectors of length not much bigger than the generalized uniform radius in any lattice (in the worst case and with high probability). Formally, we prove the hardness of the homogeneous small integer solution problem of Definition 5.1 over groups $\mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n))$, by reduction from the following variant of SIVP where the quality of a solution $\|\mathbf{S}\|$, instead of being measured with respect to the size of the smallest possible solution $\lambda_n(\mathcal{L}(\mathbf{B}))$, is measured with respect to some other parameter of interest $\phi(\mathcal{L}(\mathbf{B}))$.

Definition 6.1 *Let ϕ be an arbitrary function mapping lattices to positive reals. The* Generalized Independent Vectors Problem $\mathrm{GIVP}_\gamma^\phi$*, given a lattice basis $\mathbf{B}$ of rank n, asks for a set of n linearly independent lattice vectors $\mathbf{S} \subset \mathcal{L}(\mathbf{B})$ such that $\|\mathbf{S}\| \leq \gamma(n) \cdot \phi(\mathcal{L}(\mathbf{B}))$.*

Notice that SIVP_γ is a special case of $\mathrm{GIVP}_\gamma^\phi$, where $\phi = \lambda_n$. Here we are interested in $\mathrm{GIVP}_\gamma^{\hat{\zeta}}$, i.e., the problem of finding a maximal independent set of lattice vectors that are not much longer than the generalized uniform radius.

The reduction is performed in two steps. First we reduce $\mathrm{GIVP}_\gamma^\phi$ to an intermediate problem. Next, we reduce this intermediate problem to the problem of solving random instances of $\mathrm{HSIS}_{G,m,\beta}$. We remark that the intermediate problem is a worst-case one, i.e., the first part of the reduction is a standard worst-case to worst-case probabilistic Cook reduction. Only the second part of the reduction, from the intermediate problem to the problem of solving random equations, is a worst- to average-case reduction. The advantage of introducing the intermediate problem is that the first part of the reduction (which involves solving many instances of the target problem) is a standard reduction where all problems are solved in the worst-case. Once the GIVP problem has been reduced to the intermediate problem, the worst-case to average-case reduction can be expressed in a conceptually simpler setting where a single (worst-case) instance of the intermediate problem is reduced to a single (random) instance of the average-case problem.

The rest of this section is organized as follows. In Subsection 6.1 we reduce GIVP to the intermediate problem, and give sufficient conditions for the solutions of the latter. In Subsection 6.2 we present the worst-

Symbol	Explanation	Reference
$\mathbf{L}_n$	easily decodable $\tau(n)$-perfect lattice	Theorem 6.5
$\mathbf{M}_n$	almost orthogonal sublattice of $\mathcal{L}(\mathbf{L}_n)$	Equation (6.2)
$\mathbf{B}$	GIVP input lattice	
$\mathbf{C}$	almost orthogonal sublattice of $\mathcal{L}(\mathbf{B})$	Equation (6.3)
$\mathbf{S}$	full rank sublattice of $\mathcal{L}(\mathbf{B})$	
G_n	Abelian group ($\mathcal{L}(\mathbf{L}_n)$ modulo $\mathcal{L}(\mathbf{M}_n)$)	Definition 5.1
ψ	Linear function mapping $\mathbf{M}_n$ to $\mathbf{C}$	Equation (6.4)
$(\mathbf{w}_{i,j}, \mathbf{v}_{i,j})$	vectors in $\mathcal{L}(\mathbf{L}_n) \times \mathcal{L}(\mathbf{B})$ output by the sampling algorithm	Lemma 6.6
$a_{i,j}$	group element $(\mathbf{w}_{i,j} \bmod \mathbf{M}_n) \in G_n$	
a_i	sum of $a_{i,j}$ for $j = 1, \ldots, k(n)$	
$k(n)$	Number of samples used to generate each a_i	Equation (6.7)
$\mathbf{a}$	homogeneous linear equation over G_n (input to $\mathcal{F}$)	
$\Lambda(\mathbf{a})$	set of solutions to equation $\mathbf{a}$	Definition 5.1
$\mathbf{z}$	solution to equation $\mathbf{a}$ output by $\mathcal{F}$	
n	rank of $\mathbf{L}_n, \mathbf{M}_n, \mathbf{B}, \mathbf{C}$ and $\mathbf{S}$	
$m(n)$	number of variables in $\mathbf{a}$	
$\alpha(n)$	scaling factor used in the definition of $\mathbf{M}_n$	Equation (6.1)
$\beta(n)$	length of the solution $\mathbf{z}$ returned by $\mathcal{F}$	Theorem 6.5
$\gamma(n)$	GIVP approximation factor	Theorem 6.5
$\tau(n)$	upper bound on $2\rho(\mathcal{L}(\mathbf{B}))/\lambda_1(\mathcal{L}(\mathbf{B}))$	
$\mathbf{y}_{i,j}$	offset vector $\mathbf{v}_{i,j} - \psi(\mathbf{w}_{i,j})$	
$\mathbf{s}$	output of $\mathcal{A}$	Equation (6.5)

Table 1: Symbols used in the reduction from $\mathrm{GIVP}_\gamma^{\hat{\xi}}$ to $HSIS_{G,m,\beta}$.

to average-case reduction from the intermediate problem to HSIS (Theorem 6.5). The reduction is based on a sampling procedure (Lemma 6.6) that is described and analyzed in Subsection 6.3. Three technical lemmas (Lemmas 6.8, 6.9 and 6.10) used in the proof of Theorem 6.5 are proved in Subsection 6.4 after establishing some important properties of the sampling procedure. For reference, the notation and symbols used in the reduction are listed in Table 1.

6.1 The intermediate problem

In this subsection, we define the intermediate problem, reduce GIVP to it, and present sufficient conditions for its solution. The intermediate problem is essentially an incremental version of GIVP, where given a set of linearly independent vectors $\mathbf{S}$, one has to find a single slightly shorter lattice vector.

Definition 6.2 *The* Incremental Generalized Independent Vectors Problem *($\textsc{IncGIVP}_\gamma^\phi$), given a rank n lattice basis $\mathbf{B}$, a set of n linearly independent vectors $\mathbf{S} \subset \mathcal{L}(\mathbf{B})$ satisfying $\|\mathbf{S}\| > \gamma(n) \cdot \phi(\mathcal{L}(\mathbf{B}))$, and an $(n-1)$-dimensional hyperplane $\mathcal{H} \subset \mathrm{span}(\mathbf{B})$, asks for a lattice vector $\mathbf{s} \in \mathcal{L}(\mathbf{B}) \setminus \mathcal{H}$ such that $\|\mathbf{s}\| \leq \|\mathbf{S}\|/2$.*

The following theorem shows that $\mathrm{GIVP}_\gamma^\phi$ is easily reducible to $\textsc{IncGIVP}_\gamma^\phi$.

Theorem 6.3 *For any functions ϕ and γ, there is a probabilistic polynomial time reduction from solving $\mathrm{GIVP}_\gamma^\phi$ in the worst case (with high probability), to solving $\textsc{IncGIVP}_\gamma^\phi$ in the worst-case (with high probability).*

Proof: We give an iterative reduction (see Subsection 2.6) from $\mathrm{GIVP}_\gamma^\phi$ to $\textsc{IncGIVP}_\gamma^\phi$. By Theorem 2.22, this immediately implies a standard (probabilistic) Cook reduction between the two problems. Let $\mathcal{A}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ be a probabilistic algorithm solving $\textsc{IncGIVP}_\gamma^\phi$ in the worst case with non-negligible probability $\delta(n)$. The iterative reduction $(R, f, I, O, \mathcal{S})$ is defined as follows. Relation R is the set of all $(\mathbf{B}, \mathbf{S})$ where $\mathbf{B}$ is the GIVP input lattice, and $\mathbf{S} \subset \mathcal{L}(\mathbf{B})$ is a maximal set of linearly independent lattice vectors such that

$\|\mathbf{S}\| \leq \|\mathbf{B}\|$. (This condition is introduced to make sure that the size of $\mathbf{S}$ is polynomial in the input size.) Initially, $\mathbf{S}$ is set to the input basis $I(\mathbf{B}) = \mathbf{B}$. Upon termination, the iterative reduction outputs the current set $O(\mathbf{B}, \mathbf{S}) = \mathbf{S}$. Progress at each iteration is measured by the function $f(\mathbf{B}, \mathbf{S}) = \prod_{i=1}^{n} \|\mathbf{s}_i\|^2$. Notice that function f is polynomial time computable. In order to complete the iterative reduction we give a probabilistic polynomial time oracle algorithm $\mathcal{S}^{\mathcal{A}}$ (the iterative step) that on input a rank n lattice basis $\mathbf{B}$ and n linearly independent lattice vectors $\mathbf{S} \subset \mathcal{L}(\mathbf{B})$ such that $\gamma(n) \cdot \phi(\mathcal{L}(\mathbf{B})) < \|\mathbf{S}\| \leq \|\mathbf{B}\|$, finds a set of linearly independent lattice vectors $\mathbf{S}'$ such that $\|\mathbf{S}'\| \leq \|\mathbf{B}\|$ and $f(\mathbf{S}') \leq f(\mathbf{S})/2$ with non-negligible probability $\delta(n)$.

Algorithm $\mathcal{S}^{\mathcal{A}}(\mathbf{B}, \mathbf{S})$ works as follows. Let i be the index of a longest vector in $\mathbf{S}$, i.e., $\|\mathbf{s}_i\| = \|\mathbf{S}\|$, and let $\mathcal{H} = \text{span}(\mathbf{s}_1, \ldots, \mathbf{s}_{i-1}, \mathbf{s}_{i+1}, \ldots, \mathbf{s}_n)$ be the $(n-1)$-dimensional hyperplane spanned by the other vectors. The iterative step $\mathcal{S}$ computes $\mathbf{s} = \mathcal{A}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ and checks that vector $\mathbf{s}$ satisfies $\mathbf{s} \in \mathcal{L}(\mathbf{B}) \setminus \mathcal{H}$ and $\|\mathbf{s}\| \leq \|\mathbf{S}\|/2$. If so, then $\mathcal{S}$ replaces $\mathbf{s}_i$ with $\mathbf{s}$, and outputs $\mathbf{S}' = [\mathbf{s}_1, \ldots, \mathbf{s}_{i-1}, \mathbf{s}, \mathbf{s}_{i+1}, \ldots, \mathbf{s}_n]$. Otherwise $\mathcal{S}$ simply outputs the input set $\mathbf{S}' = \mathbf{S}$. Notice that in both cases, the output $\mathbf{S}'$ satisfies the relation $(\mathbf{B}, \mathbf{S}') \in R$, i.e., $\mathbf{S}' \subset \mathcal{L}(\mathbf{B})$ is a set of n linearly independent lattice vectors and $\|\mathbf{S}'\| \leq \|\mathbf{S}\| \leq \|\mathbf{B}\|$. Moreover, if algorithm $\mathcal{A}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ successfully returned a vector $\mathbf{s} \in \mathcal{L}(\mathbf{B}) \setminus \mathcal{H}$ satisfying $\|\mathbf{s}\| \leq \|\mathbf{S}\|/2$, then

$$f(\mathbf{B}, \mathbf{S}') = f(\mathbf{B}, \mathbf{S}) \frac{\|\mathbf{s}\|^2}{\|\mathbf{s}_i\|^2} = f(\mathbf{B}, \mathbf{S}) \frac{\|\mathbf{s}\|^2}{\|\mathbf{S}\|^2} \leq \frac{f(\mathbf{B}, \mathbf{S})}{4}.$$

This proves that the iterative step succeeds at least with the same (non-negligible) probability $\delta(n)$ of algorithm $\mathcal{A}$. $\square$

The following lemma establishes sufficient conditions for the solution of IncGIVP.

Lemma 6.4 *Let $\mathcal{A}^{(\cdot)}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ be a probabilistic polynomial time oracle algorithm that on input a rank n lattice basis $\mathbf{B}$, a full rank subset $\mathbf{S} \subset \mathcal{L}(\mathbf{B})$ and an $(n-1)$-dimensional hyperplane $\mathcal{H} \subset \text{span}(\mathbf{B})$, makes a single oracle call $\mathbf{a} \in G_n^{m(n)}$, and (provided the query is answered with a valid solution $\mathbf{z} \in \Lambda(\mathbf{a})$) outputs a lattice vector $\mathbf{s} \in \mathcal{L}(\mathbf{B})$. Assume that for any input $(\mathbf{B}, \mathbf{S}, \mathcal{H})$ such that $\|\mathbf{S}\| > \gamma(n) \cdot \phi(\mathcal{L}(\mathbf{B}))$, the vectors $\mathbf{a}, \mathbf{z}, \mathbf{s}$ produced by $\mathcal{A}^{(\cdot)}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ satisfy the following properties*

- *the statistical distance between the query $\mathbf{a}$ and a uniformly distributed $\mathbf{u} \in G_n^{m(n)}$ is negligible, i.e.,*

$$\Delta(\mathbf{a}, \mathbf{u}) = n^{-\omega(1)},$$

- *for any $\hat{\mathbf{a}} \in G_n^{m(n)}$ and $\hat{\mathbf{z}} \in \Lambda(\hat{\mathbf{a}}) \setminus \{\mathbf{0}\}$, the conditional probability that $\mathbf{s} \notin \mathcal{H}$ is at least*

$$\Pr\{\mathbf{s} \notin \mathcal{H} \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}\} = \Omega(1),$$

- *for any $\hat{\mathbf{a}} \in G_n^{m(n)}$ and $\hat{\mathbf{z}} \in \Lambda(\hat{\mathbf{a}})$, the conditional expectation of $\|\mathbf{s}\|$ is at most*

$$\text{Exp}[\|\mathbf{s}\| \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] = o\left(\frac{\|\hat{\mathbf{z}}\| \cdot \|\mathbf{S}\|}{\beta(n)}\right).$$

Then, for any randomized procedure $\mathcal{F}$ that solves $\text{HSIS}_{G,m,\beta}$ with non-negligible probability $\delta(n)$, $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ solves $\text{IncGIVP}_{\gamma}^{\phi}$ with high probability[16] *$\Omega(\delta(n))$.*

Proof: Let $\mathcal{F}$ be a randomized procedure that solves $\text{HSIS}_{G,m,\beta}$ with non-negligible probability $\delta(n)$. We want to prove that, for any valid input, $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ solves $\text{IncGIVP}_{\gamma}^{\phi}$ with probability $\Omega(\delta(n))$. Namely, we want to prove that for any rank n lattice basis $\mathbf{B}$, full rank subset $\mathbf{S} \subset \mathcal{L}(\mathbf{B})$ such that $\|\mathbf{S}\| > \gamma(n) \cdot \phi(\mathcal{L}(\mathbf{B}))$, and $(n-1)$-dimensional hyperplane $\mathcal{H} \subset \text{span}(\mathbf{B})$, procedure $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ outputs a lattice vector $\mathbf{s} \in \mathcal{L}(\mathbf{B}) \setminus \mathcal{H}$ of length $\|\mathbf{s}\| \leq \|\mathbf{S}\|/2$ with non-negligible probability $\Omega(\delta(n))$.

[16] Remember that, since $\mathcal{A}$ solves IncGIVP in the *worst-case*, and given a vector $\mathbf{s}$ it is easy to check if $\mathbf{s}$ is a correct solution to an IncGIVP instance, the success probability of $\mathcal{A}$ can be efficiently boosted from any non-negligible fraction to exponentially close to one.

Assume without loss of generality that $\mathcal{F}(\mathbf{a})$ always returns a (possibly zero) solution $\mathcal{F}(\mathbf{a}) \in \Lambda(\mathbf{a})$ of length $\|\mathcal{F}(\mathbf{a})\| \leq \beta(n)$. The assumption on $\mathcal{F}$ is that $\Pr\{\mathcal{F}(\mathbf{u}) \neq \mathbf{0}\} = \delta(n)$ when $\mathbf{u} \in G_n^{m(n)}$ is chosen uniformly at random. Since $\mathcal{F}(\mathbf{a})$ always returns a valid solution $\mathbf{z} \in \Lambda(\mathbf{a})$, the output vector $\mathbf{s}$ is guaranteed to belong to the lattice $\mathcal{L}(\mathbf{B})$. We need to bound the probability that $\mathbf{s}$ also satisfies $\mathbf{s} \notin \mathcal{H}$ and $\|\mathbf{s}\| \leq \|\mathbf{S}\|/2$. Consider an execution of $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S}, \mathcal{H}) = \mathbf{s}$, and let $\mathcal{F}(\mathbf{a}) = \mathbf{z}$ be the oracle call made by $\mathcal{A}$. Conditioning on the value of $\mathbf{a}$ and $\mathbf{z}$, and restricting our attention to the nonzero solutions $\mathbf{z} \neq \mathbf{0}$, we get

$$\begin{aligned}
&\Pr\{\mathbf{s} \notin \mathcal{H} \wedge \|\mathbf{s}\| \leq \|\mathbf{S}\|/2\} = \\
&= \sum_{\hat{\mathbf{a}}, \hat{\mathbf{z}}} \Pr\{\mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\} \cdot \Pr\{\mathbf{s} \notin \mathcal{H} \wedge \|\mathbf{s}\| \leq \|\mathbf{S}\|/2 \mid \mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\} \\
&\geq \sum_{\hat{\mathbf{a}}, \hat{\mathbf{z}}: \hat{\mathbf{z}} \neq \mathbf{0}} \Pr\{\mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\} \cdot \left(\Pr\{\mathbf{s} \notin \mathcal{H} \mid \mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\} - \Pr\{\|\mathbf{s}\| > \|\mathbf{S}\|/2 \mid \mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\}\right),
\end{aligned}$$

where the summations range over all $\hat{\mathbf{a}} \in G_n^{m(n)}$ and $\hat{\mathbf{z}} \in [\mathcal{F}(\hat{\mathbf{a}})] \subseteq \Lambda(\hat{\mathbf{a}}) \cap \mathcal{B}(\beta(n))$. By assumption on $\mathcal{A}$, for any $\hat{\mathbf{a}} \in G_n^{m(n)}$ and $\hat{\mathbf{z}} \in \Lambda(\hat{\mathbf{a}})$ such that $0 < \|\hat{\mathbf{z}}\| \leq \beta(n)$, the two conditional probabilities in the last expression satisfy

$$\Pr\{\mathbf{s} \notin \mathcal{H} \mid \mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\} = \Omega(1),$$

and, using Markov inequality,

$$\begin{aligned}
\Pr\{\|\mathbf{s}\| > \|\mathbf{S}\|/2 \mid \mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\} &\leq \frac{\mathrm{Exp}[\|\mathbf{s}\| \mid \mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}]}{\|\mathbf{S}\|/2} \\
&\leq o\left(\frac{2\|\hat{\mathbf{z}}\| \cdot \|\mathbf{S}\|}{\|\mathbf{S}\| \cdot \beta(n)}\right) = o(1).
\end{aligned}$$

Adding up for all possible values of $\hat{\mathbf{a}}$ and $\hat{\mathbf{z}} \neq \mathbf{0}$ we get

$$\begin{aligned}
\Pr\{\mathbf{s} \notin \mathcal{H} \wedge \|\mathbf{s}\| \leq \|\mathbf{S}\|/2\} &\geq \sum_{\hat{\mathbf{a}}, \hat{\mathbf{z}}: \hat{\mathbf{z}} \neq \mathbf{0}} \Pr\{\mathbf{a} = \hat{\mathbf{a}} \wedge \mathbf{z} = \hat{\mathbf{z}}\} \cdot (\Omega(1) - o(1)) \\
&= \Omega(\Pr\{\mathbf{z} \neq \mathbf{0}\}).
\end{aligned}$$

Notice that $\mathbf{z} = \mathcal{F}(\mathbf{a})$ and $\Pr\{\mathcal{F}(\mathbf{u}) \neq \mathbf{0}\} = \delta(n)$ when $\mathbf{u} \in G_n^{m(n)}$ is uniformly distributed. By assumption, the statistical distance $\Delta(\mathbf{a}, \mathbf{u})$ between $\mathbf{a}$ and $\mathbf{u}$ is negligible. Therefore, by Corollary 2.17,

$$\begin{aligned}
\Pr\{\mathbf{z} \neq \mathbf{0}\} &= \Pr\{\mathcal{F}(\mathbf{a}) \neq \mathbf{0}\} \\
&\geq \Pr\{\mathcal{F}(\mathbf{u}) \neq \mathbf{0}\} - \Delta(\mathbf{a}, \mathbf{u}) \\
&\geq \delta(n) - n^{-\omega(1)}.
\end{aligned}$$

So, for all non-negligible $\delta(n)$, $\Pr\{\mathbf{s} \notin \mathcal{H} \wedge \|\mathbf{s}\| \leq \|\mathbf{S}\|/2\} \geq \Omega(\delta(n) - n^{-\omega(1)}) = \Omega(\delta(n))$. $\square$

6.2 The main reduction

In this subsection we show that for appropriate choice of groups G_n, and parameters $\beta(n), m(n), \gamma(n)$, there is a reduction from solving $\textsc{IncGIVP}_\gamma^{\hat{\zeta}}$ in the worst case to solving $\mathrm{HSIS}_{G,m,\beta}$ on the average.

Theorem 6.5 *Let $\tau(n) \geq 1$ such that there exists an easily decodable family of $\tau(n)$-perfect lattices. Then, for any $\beta(n) \geq 1$, $m(n) = n^{O(1)}$ and $\gamma(n) = \beta(n)\tau(n) \cdot \sqrt{\omega(\log n)}$, there is a sequence of efficiently computable Abelian groups G_n of size $\#G_n \leq (n^{1.5}\gamma(n)/8)^n$ such that solving $\textsc{IncGIVP}_\gamma^{\hat{\zeta}}$ in the worst case with high probability reduces to solving $\mathrm{HSIS}_{G,m,\beta}$ on the average with non-negligible probability.*

Proof: Let $\{\mathcal{L}(\mathbf{L}_n)\}$ a family of easily decodable $\tau(n)$-perfect lattices. For any $\beta(n) \geq 1$ and $m(n) = n^{O(1)}$, let $\gamma(n) = \beta(n)\tau(n) \cdot \sqrt{\omega(\log n)}$ and

$$\alpha(n) = \frac{n\lambda_1(\mathcal{L}(\mathbf{L}_n))\gamma(n)}{12}. \tag{6.1}$$

Notice that from the definition of $\alpha(n)$ and $\gamma(n)$, and the assumption that $\mathcal{L}(\mathbf{L}_n)$ is $\tau(n)$-perfect, we get

$$\alpha(n) = \frac{n\lambda_1(\mathcal{L}(\mathbf{L}_n))\beta(n)\tau(n)\sqrt{\omega(\log n)}}{12} \geq \left(\frac{\beta(n)\sqrt{n\cdot\omega(\log n)}}{12}\right) 2\sqrt{n}\rho(\mathcal{L}(\mathbf{L}_n)) \geq 2\sqrt{n}\rho(\mathcal{L}(\mathbf{L}_n)).$$

So, $\alpha(n)$ satisfies the condition in Definition 5.2, and we can define a full rank subset $\mathbf{M}_n \subseteq \mathcal{L}(\mathbf{L}_n)$, and quotient group $G_n = \mathcal{G}(\mathcal{L}(\mathbf{L}_n), \alpha(n)) = \mathcal{L}(\mathbf{L}_n)/\mathcal{L}(\mathbf{M}_n)$ such that Corollary 5.3 and Lemma 5.4 hold true, i.e.,

$$\forall \mathbf{x} \in \mathbb{R}^n . \|\mathbf{M}_n\mathbf{x}\| \approx \alpha(n) \cdot \|\mathbf{x}\| \tag{6.2}$$

and group G_n has size at most

$$\#G_n \leq \left(\frac{3\alpha(n)\sqrt{n}}{2\lambda_1(\mathcal{L}(\mathbf{L}_n))}\right)^n = \left(\frac{n^{1.5}\gamma(n)}{8}\right)^n.$$

We define a probabilistic polynomial time oracle algorithm $\mathcal{A}^{(\cdot)}$ satisfying the conditions in Lemma 6.4 with $\phi = \hat{\zeta}$. It follows from Lemma 6.4 that $\mathcal{A}^{(\cdot)}$ is a probabilistic polynomial time worst-case to average-case reduction from $\text{IncGIVP}_\gamma^{\hat{\zeta}}$ to $\text{HSIS}_{G,m,\beta}$. The intuition behind procedure $\mathcal{A}^{(\cdot)}$ is the following. (See Figure 2.) Map $\mathcal{L}(\mathbf{M}_n)$ to a sublattice $\mathcal{L}(\mathbf{C}) = \psi(\mathcal{L}(\mathbf{M}_n)) \subset \mathcal{L}(\mathbf{B})$ using a linear function ψ with small

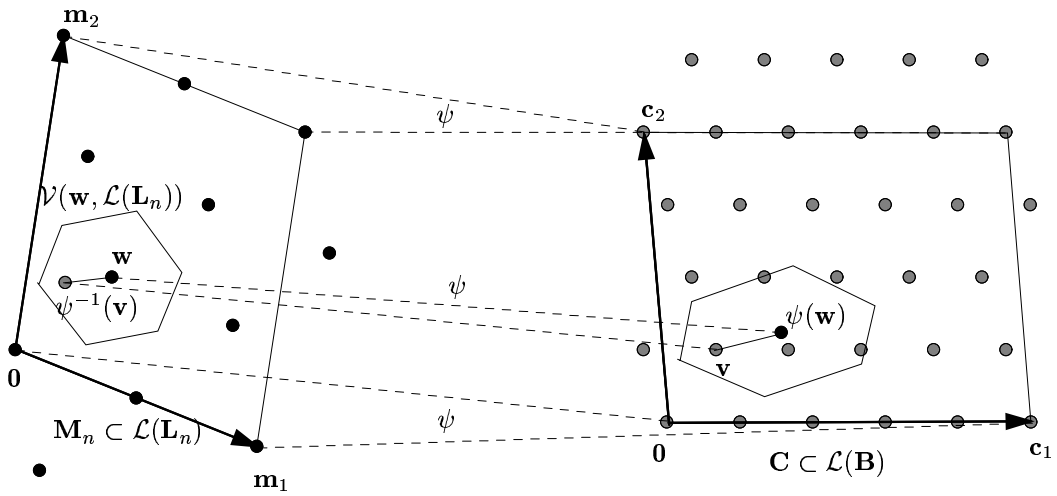

Figure 2: Sampling lattice points

distortion, i.e., a function that approximately preserves distance ratios. One possible way to achieve this is to map the almost orthogonal set $\mathbf{M}_n$ to an almost orthogonal subset $\mathbf{C} = \psi(\mathbf{M}_n) \subset \mathcal{L}(\mathbf{B})$ and extend ψ to span$(\mathbf{M}_n)$ by linearity. Now, consider the Voronoi cells $\mathcal{V}(\mathbf{w}, \mathcal{L}(\mathbf{L}_n))$ of the $\tau(n)$-perfect lattice $\mathcal{L}(\mathbf{L}_n)$. Function ψ maps each cell to a corresponding region $\psi(\mathcal{V}(\mathbf{w}, \mathcal{L}(\mathbf{L}_n)))$ centered around $\psi(\mathbf{w})$. Partition the points of $\mathcal{L}(\mathbf{B})$ into subsets, according to these regions. Pick $m(n)$ points $\mathbf{v}_i \in \mathcal{L}(\mathbf{B})$ at random, and map each of them to the center $\psi(\mathbf{w}_i)$ of the corresponding region. Notice that each region $\psi(\mathcal{V}(\mathbf{w}, \mathcal{L}(\mathbf{L}_n)))$ is associated to a group element $[\mathbf{w}]_{\mathbf{M}_n} \in G_n$. So, the points $\mathbf{v}_i$ define $m(n)$ group elements $a_i = [\mathbf{w}_i]_{\mathbf{M}_n} \in G_n$. Use $\mathcal{F}$ to find a small nonzero solution $\mathbf{z} = \mathcal{F}(\mathbf{a})$ to equation $\mathbf{a} = [a_1, \ldots, a_{m(n)}]$. The output of $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S}, \mathcal{H})$ is vector $\mathbf{s} = \sum_i z_i(\mathbf{v}_i - \psi(\mathbf{w}_i))$. Notice that $\mathbf{s} \in \mathcal{L}(\mathbf{B})$ because $\sum z_i\mathbf{v}_i$ is an integer combination of lattice vectors, and $\sum_i z_i\mathbf{w}_i \in \mathcal{L}(\mathbf{M}_n) \subseteq \psi^{-1}(\mathcal{L}(\mathbf{B}))$. (See Lemma 6.7 for details.) Before moving to the actual proof, we informally explain why vectors $\mathbf{a}, \mathbf{z}, \mathbf{s}$ are expected to satisfy the three conditions in Lemma 6.4. (1) Vector $\mathbf{a}$ is distributed almost uniformly at random because coefficients $a_i = [\mathbf{w}_i]_{\mathbf{M}_n}$ are chosen independently, and each region $\psi(\mathcal{V}(\mathbf{w}, \mathcal{L}(\mathbf{L}_n)))$ contains roughly the same number of lattice points from $\mathcal{L}(\mathbf{B})$. (2) Vector $\mathbf{s}$ does not belong to any fixed hyperplane $\mathcal{H}$ with high probability because each $\mathbf{v}_i - \psi(\mathbf{w}_i)$ is somehow randomly distributed within $\psi(\mathcal{V}(\mathcal{L}(\mathbf{L}_n)))$. (3) Finally, $\mathbf{s}$ is short because it is a small combination of short

vectors $\mathbf{v}_i - \psi(\mathbf{w}_i)$, each one lying within the region $\psi(\mathcal{V}(\mathcal{L}(\mathbf{L}_n)))$. This is an oversimplified description of the reduction. For example, Lemma 6.4 requires distribution $\mathbf{a}$ to be extremely close to uniform. In order to ensure the almost uniform distribution of $\mathbf{a}$, we will need to slightly modify the above procedure by sampling many points $(\mathbf{w}_{i,j}, \mathbf{v}_{i,j})$ and adding up the corresponding $a_{i,j} = [\mathbf{w}_{i,j}]_{\mathbf{M}_n}$ to obtain group elements $a_i = \sum_j a_{i,j}$ whose distribution is extremely close to uniform.

We now give a detailed description of procedure $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S}, \mathcal{H})$. Notice that the procedure outlined above does not use the input hyperplane $\mathcal{H}$, and condition $\mathbf{s} \notin \mathcal{H}$ holds with high probability for any fixed hyperplane $\mathcal{H}$. Therefore, below we simply write $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S})$ instead of $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S}, \mathcal{H})$, to emphasize the fact that $\mathcal{A}$ does not use the input hyperplane $\mathcal{H}$.

Procedure $\mathcal{A}^{(\cdot)}(\mathbf{B}, \mathbf{S})$ works as follows. First of all, notice that using Babai's nearest plane algorithm [6], matrix $\mathbf{S}$ allows to approximate any vector $\mathbf{x} \in \text{span}(\mathbf{B})$ with a lattice point $\mathbf{y} \in \mathcal{L}(\mathbf{S}) \subseteq \mathcal{L}(\mathbf{B})$ within distance $\sigma = (\sqrt{n}/2)\|\mathbf{S}\|$ from $\mathbf{x}$.[17] Therefore, using Lemma 2.11, we can find an almost orthogonal sublattice $\mathcal{L}(\mathbf{C}) \subset \mathcal{L}(\mathbf{B})$ such that

$$\forall \mathbf{x} \in \mathbb{R}^n . \|\mathbf{C}\mathbf{x}\| \approx n\|\mathbf{S}\| \cdot \|\mathbf{x}\|. \tag{6.3}$$

Let $\psi(\mathbf{x}) = \mathbf{C}\mathbf{M}_n^{-1}\mathbf{x}$ be the linear transformation that maps $\mathbf{m}_i$ to $\mathbf{c}_i$ for all $i = 1, \ldots, n$. Combining (6.2) and (6.3), and using (2.1), we get

$$\forall \mathbf{x} \in \mathbb{R}^n . \frac{1}{3}\left(\frac{n\|\mathbf{S}\|}{\alpha(n)}\right) \cdot \|\mathbf{x}\| \leq \|\psi(\mathbf{x})\| \leq 3\left(\frac{n\|\mathbf{S}\|}{\alpha(n)}\right) \cdot \|\mathbf{x}\| \tag{6.4}$$

i.e., the linear function ψ is close to an orthogonal transformation that scales all distances by a factor $n\|\mathbf{S}\|/\alpha(n)$.

Notice that $\mathcal{L}(\mathbf{M}_n)$ is a common sublattice of both $\mathcal{L}(\mathbf{L}_n)$ and $\psi^{-1}(\mathcal{L}(\mathbf{B}))$. The following lemma shows how to use function ψ together with decoding algorithm $\text{CVP}_{\mathbf{L}}$ to simultaneously sample from groups $G_n = \mathcal{L}(\mathbf{L}_n)/\mathcal{L}(\mathbf{M}_n)$ and $\mathcal{L}(\mathbf{B})/\mathcal{L}(\mathbf{C}) \equiv \psi^{-1}(\mathcal{L}(\mathbf{B}))/\mathcal{L}(\mathbf{M}_n)$.

Lemma 6.6 *There is a sampling algorithm that on input two rank n lattices $\mathbf{L}_n$ and $\mathbf{B}$, a full rank sublattice $\mathbf{M}_n \subset \mathcal{L}(\mathbf{L}_n)$ and a non-singular linear transformation ψ such that $\mathbf{C} = \psi(\mathbf{M}_n) \subset \mathcal{L}(\mathbf{B})$, outputs two vectors $\mathbf{w} \in \mathcal{L}(\mathbf{L}_n)$ and $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ such that the following holds:*

1. *The group element $[\mathbf{v}]_{\mathbf{C}}$ is uniformly distributed over $\mathcal{L}(\mathbf{B})/\mathcal{L}(\mathbf{C})$.*
2. *$\psi^{-1}(\mathbf{v}) \in \bar{\mathcal{V}}(\mathbf{w}, \mathcal{L}(\mathbf{L}_n))$, or, equivalently, $\mathbf{v} - \psi(\mathbf{w}) \in \psi(\bar{\mathcal{V}}(\mathcal{L}(\mathbf{L}_n)))$.*
3. *The distribution of $\mathbf{v} - \psi(\mathbf{w})$ is symmetric about the origin, and, in particular, $\text{Exp}[\mathbf{v} - \psi(\mathbf{w})] = 0$.*
4. *$\mathbf{w} \in \mathcal{P}(\mathbf{M}_n)$.*

Moreover, if lattice $\mathbf{L}_n$ is easily decodable, then the sampling procedure runs in polynomial time.

The actual properties of the sampling algorithm are not important at this point, and the proof of Lemma 6.6 is deferred to Subsection 6.3. All that matters for now is that the sampling algorithm generates pairs of vectors $(\mathbf{w}, \mathbf{v}) \in \mathcal{L}(\mathbf{L}_n) \times \mathcal{L}(\mathbf{B})$. Below we describe how to use any such sampling procedure to compute a lattice vector $\mathbf{s} \in \mathcal{L}(\mathbf{B})$. After defining the full rank sublattice $\mathbf{C} \subset \mathcal{L}(\mathbf{S})$ and linear function $\psi(\mathbf{M}_n) = \mathbf{C}$ satisfying (6.4), algorithm $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S})$ proceeds as follows:

1. Run the sampling procedure of Lemma 6.6, $m(n) \cdot k(n)$ times (where $k(n) = \omega(\log n)$ is a superlogarithmic function to be specified) to generate vectors $\mathbf{w}_{i,j} \in \mathcal{L}(\mathbf{L}_n)$ and $\mathbf{v}_{i,j} \in \mathcal{L}(\mathbf{B})$, for $i = 1, \ldots, m(n)$ and $j = 1, \ldots, k(n)$.
2. Let $a_{i,j} = [\mathbf{w}_{i,j}]_{\mathbf{M}_n} \in G_n$ be the group elements corresponding to lattice points $\mathbf{w}_{i,j}$ and, for every $i = 1, \ldots, m(n)$, define group element $a_i = \sum_{j=1}^{k(n)} a_{i,j}$.
3. Use oracle $\mathcal{F}$ to compute a nonzero solution $\mathbf{z} = \mathcal{F}(\mathbf{a}) \in \Lambda(\mathbf{a}) \setminus \{\mathbf{0}\}$ to equation $\mathbf{a} = [a_1, \ldots, a_{m(n)}]$.

[17] This is not a particularly critical part of the reduction, and using poorer rounding procedures (e.g., rounding off the coordinates of $\mathbf{x}$ with respect to basis $\mathbf{S}$ to the closest integers as done in [2]) results in substantially the same connection factors as using Babai's nearest plane algorithm [6].

4. For any i, j, let $\mathbf{y}_{i,j} = \mathbf{v}_{i,j} - \psi(\mathbf{w}_{i,j})$, and output

$$\mathbf{s} = \sum_{i=1}^{m(n)} z_i \sum_{j=1}^{k(n)} \mathbf{y}_{i,j}. \tag{6.5}$$

Notice that randomness is used twice in the routine: first in step 1 and then in step 3. In step 1, randomness is used to run the sampling procedure $m(n) \cdot k(n)$ times and generate a random equation $\mathbf{a}$ to be passed as input to $\mathcal{F}$. In step 3 randomness is used to run the probabilistic procedure $\mathcal{F}$ on input $\mathbf{a}$ to compute a solution $\mathbf{z}$. Since $\mathcal{F}$ is only guaranteed to work *on the average*, it is important that both the input $\mathbf{a}$ and the internal randomness of $\mathcal{F}$ are chosen (almost) uniformly and independently at random. We remark that, although the value of $\mathbf{z}$ depends both on the randomness used by the sampling procedure and that used directly by $\mathcal{F}$, the two procedures use independent sources of randomness. So, for example, given the value of $\mathbf{a}$, the conditional distribution of $\mathbf{z}$ is independent from the conditional distribution of the samples $(\mathbf{w}_{i,j}, \mathbf{v}_{i,j})$. We will use this fact in the probabilistic analysis of the success probability of the reduction.

In the following lemma we prove that algorithm $\mathcal{A}^{\mathcal{F}}$ is correct, i.e., the output vector $\mathbf{s}$ belongs to lattice $\mathcal{L}(\mathbf{B})$, provided query $\mathbf{a}$ is answered with a valid solution $\mathbf{z} \in \Lambda(\mathbf{a})$.

Lemma 6.7 *Let $\mathbf{s}$ be the output vector defined in (6.5). If $\mathbf{z} \in \Lambda(\mathbf{a})$, then $\mathbf{s} \in \mathcal{L}(\mathbf{B})$.*

Proof: Define the vector

$$\mathbf{w} = \sum_{i=1}^{m(n)} z_i \sum_{j=1}^{k(n)} \mathbf{w}_{i,j}.$$

Using the definition of $\mathbf{y}_{i,j}$ and the linearity of ψ, we get

$$\mathbf{s} = \sum_{i=1}^{m(n)} z_i \sum_{j=1}^{k(n)} \mathbf{y}_{i,j} = \sum_{i,j} z_i(\mathbf{v}_{i,j} - \psi(\mathbf{w}_{i,j})) = \left(\sum_{i,j} z_i \mathbf{v}_{i,j}\right) - \psi(\mathbf{w}).$$

The first term $\sum_{i,j} z_i \mathbf{v}_{i,j}$ clearly belongs to $\mathcal{L}(\mathbf{B})$ because it is an integer linear combination of lattice vectors $\mathbf{v}_{i,j} \in \mathcal{L}(\mathbf{B})$. We need to prove that also the second term $\psi(\mathbf{w})$ belongs to $\mathcal{L}(\mathbf{B})$. We show that $\mathbf{w} \in \mathcal{L}(\mathbf{M}_n)$. Since ψ maps $\mathcal{L}(\mathbf{M}_n)$ to $\mathcal{L}(\mathbf{C})$, it follows that $\psi(\mathbf{w}) \in \mathcal{L}(\mathbf{C}) \subseteq \mathcal{L}(\mathbf{B})$.

Remember that $\mathbf{z} = \mathcal{F}(\mathbf{a}) \in \Lambda(\mathbf{a})$, i.e., $\sum_i z_i a_i = 0$ (in G_n). Since all $\mathbf{w}_{i,j}$ belong to $\mathcal{L}(\mathbf{L}_n)$, $\mathbf{w}$ is also a lattice point of $\mathcal{L}(\mathbf{L}_n)$ and $[\mathbf{w}]_{\mathbf{M}_n} \in G_n$. The group element corresponding to lattice vector $\mathbf{w}$ is

$$[\mathbf{w}]_{\mathbf{M}_n} = \sum_{i=1}^{m(n)} z_i \sum_{j=1}^{k(n)} [\mathbf{w}_{i,j}]_{\mathbf{M}_n} = \sum_i z_i \sum_j a_{i,j} = \sum_i z_i a_i = 0.$$

Since G_n is the quotient of $\mathcal{L}(\mathbf{L}_n)$ modulo $\mathcal{L}(\mathbf{M}_n)$, this proves that $\mathbf{w} \in \mathcal{L}(\mathbf{M}_n)$. □

The following three lemmas show that, provided $\alpha(n)$ is in a prescribed range, procedure $\mathcal{A}$ satisfies the conditions in Lemma 6.4. The lemmas are proved in Subsection 6.4, after establishing some useful properties of the sampling procedure in Subsection 6.3.

The first lemma shows that the equation $\mathbf{a}$ passed as input to oracle $\mathcal{F}$ is almost uniformly distributed.

Lemma 6.8 *If $n\|\mathbf{S}\|\lambda_1(\mathcal{L}(\mathbf{L}_n)) \geq 6\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ and equation (6.4) holds true, then the statistical distance between vector $\mathbf{a}$ (passed as input to $\mathcal{F}$ during the execution of $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S})$) and a uniformly distributed $\mathbf{u} \in G_n^{m(n)}$ is at most $\Delta(\mathbf{a}, \mathbf{u}) \leq m(n)/2^{k(n)+1}$. In particular, for any polynomially bounded $m(n) = n^{O(1)}$ and superlogarithmic function $k(n) = \omega(\log n)$, the statistical distance $\Delta(\mathbf{a}, \mathbf{u}) = n^{-\omega(1)}$ is negligible.*

The other two lemmas show that the output vector $\mathbf{s}$ of procedure $\mathcal{A}^{\mathcal{F}}$ is sufficiently random and usually short, even after conditioning on the input and output values of oracle $\mathcal{F}$.

Lemma 6.9 *Assume* $n\|\mathbf{S}\|\lambda_1(\mathcal{L}(\mathbf{L}_n)) \geq 12\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ *and equation (6.4) holds true. Then, for any* $\hat{\mathbf{a}} \in G_n^{m(n)}$, $\hat{\mathbf{z}} \in \Lambda(\hat{\mathbf{a}}) \setminus \{\mathbf{0}\}$, *and* $(n-1)$*-dimensional hyperplane* $\mathcal{H} \subset \mathrm{span}(\mathbf{B})$,

$$\Pr\{\mathbf{s} \notin \mathcal{H} \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}\} = \Omega(1).$$

Lemma 6.10 *If* $n\|\mathbf{S}\|\lambda_1(\mathcal{L}(\mathbf{L}_n)) \geq 6\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$, *equation (6.4) holds true, and function* $\alpha(n)$ *satisfies* $\alpha(n) = \omega(n\sqrt{k(n)}\beta(n)\rho(\mathcal{L}(\mathbf{L}_n)))$, *then for any* $\hat{\mathbf{a}} \in G_n^{m(n)}$ *and* $\hat{\mathbf{z}} \in \Lambda(\hat{\mathbf{a}})$,

$$\mathrm{Exp}[\|\mathbf{s}\| \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] = o\left(\frac{\|\hat{\mathbf{z}}\| \cdot \|\mathbf{S}\|}{\beta(n)}\right) \cdot \sqrt{1 + \frac{m(n)k(n)}{2^{k(n)}}}.$$

In particular, for any polynomially bounded $m(n) = n^{O(1)}$ *and superlogarithmic function* $k(n) = \omega(\log n)$,

$$\mathrm{Exp}[\|\mathbf{s}\| \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] = o\left(\frac{\|\hat{\mathbf{z}}\| \cdot \|\mathbf{S}\|}{\beta(n)}\right).$$

We complete the proof of the theorem by showing that if $k(n)$ is appropriately chosen, then the hypotheses of Lemmas 6.8, 6.9 and 6.10 are satisfied. Notice that from the definition of $\alpha(n) = n\lambda_1(\mathcal{L}(\mathbf{L}_n))\gamma(n)/12$ and the assumption that $\|\mathbf{S}\| > \gamma(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$, we immediately get

$$12\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B})) = n\lambda_1(\mathcal{L}(\mathbf{L}_n))\gamma(n)\hat{\zeta}(\mathcal{L}(\mathbf{B})) < n\lambda_1(\mathcal{L}(\mathbf{L}_n))\|\mathbf{S}\|. \tag{6.6}$$

So, the first condition in Lemmas 6.8, 6.9 and 6.10 is satisfied. We already observed that (6.4) follows from (6.2) and (6.3). In order to satisfy the third hypothesis of Lemma 6.10, we set

$$k(n) = \frac{\gamma(n) \cdot \sqrt{\log n}}{\beta(n) \cdot \tau(n)} = \omega(\log n). \tag{6.7}$$

Solving (6.7) for $\gamma(n) = k(n)\beta(n)\tau(n)/\sqrt{\log n}$ and substituting this value in the definition of $\alpha(n)$, we get

$$\begin{aligned} \alpha(n) &= \frac{n\lambda_1(\mathcal{L}(\mathbf{L}_n))}{12} \frac{k(n)\beta(n)\tau(n)}{\sqrt{\log n}} \\ &\geq \omega(n\sqrt{k(n)}\beta(n)\rho(\mathcal{L}(\mathbf{L}_n))), \end{aligned}$$

where we have used the perfectness condition $\rho(\mathcal{L}(\mathbf{L}_n)) \leq \tau(n)\lambda_1(\mathcal{L}(\mathbf{L}_n))/2$ and the fact that $k(n)/\sqrt{\log n} = \sqrt{k(n)/\log n}\sqrt{k(n)} = \omega(\sqrt{k(n)})$. This proves that for any polynomially bounded function $m(n) = n^{O(1)}$, and $k(n)$ as defined in (6.7), the hypotheses of Lemmas 6.8, 6.9 and Lemma 6.10 are satisfied, and algorithm $\mathcal{A}$ satisfies the conditions in Lemma 6.4. Therefore, for any (possibly probabilistic) oracle $\mathcal{F}$ solving $\mathrm{HSIS}_{G,m,\beta}$ on the average with non-negligible probability, $\mathcal{A}^{\mathcal{F}}$ solves $\textsc{IncGIVP}_\gamma^{\hat{\zeta}}$ in the worst case with high probability. □

6.3 The sampling procedure

In this subsection we give a simple sampling procedure that satisfies the conditions in Lemma 6.6. Then, we establish some additional properties of the output of the sampling procedure that will be useful in Subsection 6.4. The sampling procedure is illustrated in Figure 2.

Proof [of Lemma 6.6]: We first show how to achieve the first two properties in the lemma. Choose integers

$$d_1, \ldots, d_n \in \{1, \ldots, \det(\mathcal{L}(\mathbf{C}))/\det(\mathcal{L}(\mathbf{B}))\}$$

uniformly at random and let $\mathbf{v}'' = \sum_i d_i\mathbf{b}_i \in \mathcal{L}(\mathbf{B})$. By Proposition 2.10, $[\mathbf{v}'']_{\mathbf{C}}$ is distributed uniformly at random in $\mathcal{L}(\mathbf{B})/\mathcal{L}(\mathbf{C})$. Then, compute $\mathbf{w}'' = \mathrm{CVP}_{\mathbf{L}}(\psi^{-1}(\mathbf{v}''))$. Clearly, $\psi^{-1}(\mathbf{v}'')$ belongs to the Voronoi cell $\bar{\mathcal{V}}(\mathbf{w}'', \mathcal{L}(\mathbf{L}_n))$. So, the pair $(\mathbf{v}'', \mathbf{w}'')$ satisfies the first two properties.

Now, choose $b \in \{0,1\}$ uniformly at random and set $\mathbf{v}' = (-1)^b \mathbf{v}''$ and $\mathbf{w}' = (-1)^b \mathbf{w}''$. Clearly, for any $\mathbf{v}''$ and $\mathbf{w}''$, the distribution of $\mathbf{v}' - \psi(\mathbf{w}') = (-1)^b(\mathbf{v}'' - \psi(\mathbf{w}''))$ is symmetric about the origin. So, $(\mathbf{v}', \mathbf{w}')$ satisfies the third property. We need to check that the first two properties are preserved. Since $[\mathbf{v}'']_{\mathbf{C}}$ is uniformly distributed, also $[-\mathbf{v}'']_{\mathbf{C}} = -[\mathbf{v}'']_{\mathbf{C}}$ is uniform. It follows that $[\mathbf{v}']_{\mathbf{C}}$ is uniformly distributed because $\mathbf{v}'$ is a convex combination of distributions $\mathbf{v}''$ and $-\mathbf{v}''$. Finally, since closed Voronoi cells of a lattice are symmetric,

$$\mathbf{v}' - \psi(\mathbf{w}') = (-1)^b(\mathbf{v}'' - \psi(\mathbf{w}'')) \in (-1)^b \bar{\mathcal{V}}(\mathcal{L}(\mathbf{L}_n)) = \bar{\mathcal{V}}(\mathcal{L}(\mathbf{L}_n)).$$

This proves that $(\mathbf{v}', \mathbf{w}')$ satisfies the first three properties in the lemma.

In order to achieve also the fourth property, set $\mathbf{w} = (\mathbf{w}' \bmod \mathbf{M}_n)$ and $\mathbf{v} = (\mathbf{v}' - \psi(\mathbf{w}' - \mathbf{w}))$. Property $\mathbf{w} \in \mathcal{P}(\mathbf{M}_n)$ immediately follows by the definition of $\mathbf{w}$. We show that the first three properties are preserved. By the definition of $\mathbf{v}$, we have $\mathbf{v}' - \mathbf{v} = \psi(\mathbf{w}' - \mathbf{w})$ and $\mathbf{v} - \psi(\mathbf{w}) = \mathbf{v}' - \psi(\mathbf{w}')$. So, the second and third properties are satisfied because they only depend on $\mathbf{v} - \psi(\mathbf{w})$. In order to prove the first property, notice that $\mathbf{w}' - \mathbf{w} = \mathbf{w}' - (\mathbf{w}' \bmod \mathbf{M}_n) \in \mathcal{L}(\mathbf{M}_n)$. Therefore, $\mathbf{v}' - \mathbf{v} \in \psi(\mathcal{L}(\mathbf{M}_n)) = \mathcal{L}(\mathbf{C})$, and $[\mathbf{v}]_{\mathbf{C}} = [\mathbf{v}']_{\mathbf{C}}$, proving that $[\mathbf{v}]_{\mathbf{C}}$ is distributed identically to $[\mathbf{v}']_{\mathbf{C}}$. □

The sampling procedure produces vectors $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ such that $[\mathbf{v}]_{\mathbf{C}}$ is distributed uniformly at random over the group $\mathcal{L}(\mathbf{B})/\mathcal{L}(\mathbf{C})$. However, vector $\mathbf{v}$ (before the reduction modulo $\mathbf{C}$) is not necessarily uniformly distributed over any set of lattice vectors. (This is due to lattice points $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ such that $\psi^{-1}(\mathbf{v})$ lies on the boundary of Voronoi cells $\mathcal{V}(\mathbf{w}, \mathcal{L}(\mathbf{L}_n))$.) In the next lemma, we give simple upper and lower bounds on the probability of outputting any specific vector $\mathbf{v} \in \mathcal{L}(\mathbf{B})$.

Lemma 6.11 *Let $(\mathbf{w}, \mathbf{v})$ be generated according to a sampling procedure of Lemma 6.6. Then, for any $\hat{\mathbf{v}} \in \mathcal{L}(\mathbf{B})$, $\Pr\{\mathbf{v} = \hat{\mathbf{v}}\} \leq \det(\mathcal{L}(\mathbf{B}))/\det(\mathcal{L}(\mathbf{C}))$. Moreover, if $\psi^{-1}(\hat{\mathbf{v}})$ belongs to the interior of a Voronoi cell $\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))$ for some $\hat{\mathbf{w}} \in \mathcal{L}(\mathbf{L}_n) \cap \mathcal{P}(\mathbf{M}_n)$, then $\Pr\{\mathbf{v} = \hat{\mathbf{v}}\} = \det(\mathcal{L}(\mathbf{B}))/\det(\mathcal{L}(\mathbf{C}))$.*

Proof: The upper bound is easy: for any $\hat{\mathbf{v}} \in \mathcal{L}(\mathbf{B})$,

$$\Pr\{\mathbf{v} = \hat{\mathbf{v}}\} \leq \Pr\{[\mathbf{v}]_{\mathbf{C}} = [\hat{\mathbf{v}}]_{\mathbf{C}}\} = \det(\mathcal{L}(\mathbf{B}))/\det(\mathcal{L}(\mathbf{C}))$$

because $[\mathbf{v}]_{\mathbf{C}}$ is uniformly distributed over a set $\mathcal{L}(\mathbf{B})/\mathcal{L}(\mathbf{C})$ of size $\det(\mathcal{L}(\mathbf{C}))/\det(\mathcal{L}(\mathbf{B}))$.

Now assume $\psi^{-1}(\hat{\mathbf{v}}) \in \mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))$ for some $\hat{\mathbf{w}} \in \mathcal{L}(\mathbf{L}_n) \cap \mathcal{P}(\mathbf{M}_n)$. We claim that if $[\mathbf{v}]_{\mathbf{C}} = [\hat{\mathbf{v}}]_{\mathbf{C}}$, then $\mathbf{v} = \hat{\mathbf{v}}$, and therefore

$$\Pr\{\mathbf{v} = \hat{\mathbf{v}}\} \geq \Pr\{[\mathbf{v}]_{\mathbf{C}} = [\hat{\mathbf{v}}]_{\mathbf{C}}\} = \det(\mathcal{L}(\mathbf{B}))/\det(\mathcal{L}(\mathbf{C})).$$

Let $[\mathbf{v}]_{\mathbf{C}} = [\hat{\mathbf{v}}]_{\mathbf{C}}$, i.e., $\mathbf{v} - \hat{\mathbf{v}} \in \mathcal{L}(\mathbf{C})$. It follows that vector

$$\mathbf{y} = \psi^{-1}(\mathbf{v}) - \psi^{-1}(\hat{\mathbf{v}}) = \psi^{-1}(\mathbf{v} - \hat{\mathbf{v}})$$

belongs to lattice $\psi^{-1}(\mathcal{L}(\mathbf{C})) = \mathcal{L}(\mathbf{M}_n)$. Since $\mathcal{L}(\mathbf{M}_n)$ is a sublattice of $\mathcal{L}(\mathbf{L}_n)$, and $\hat{\mathbf{w}} \in \mathcal{L}(\mathbf{L}_n)$, also $\hat{\mathbf{w}} + \mathbf{y}$ is a lattice point in $\mathcal{L}(\mathbf{L}_n)$. Consider the open Voronoi cells $\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))$ and $\mathcal{V}^o(\hat{\mathbf{w}} + \mathbf{y}, \mathcal{L}(\mathbf{L}_n))$. Using the definition of $\mathbf{y}$ and the hypothesis $\psi^{-1}(\hat{\mathbf{v}}) \in \mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))$, we get

$$\psi^{-1}(\mathbf{v}) = \psi^{-1}(\hat{\mathbf{v}}) + \mathbf{y} \in \mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)) + \mathbf{y} = \mathcal{V}^o(\hat{\mathbf{w}} + \mathbf{y}, \mathcal{L}(\mathbf{L}_n)),$$

i.e., $\psi^{-1}(\mathbf{v})$ is closer to $\hat{\mathbf{w}} + \mathbf{y}$ than to any other lattice point in $\mathcal{L}(\mathbf{L}_n)$. But we know from Lemma 6.6 that $\psi^{-1}(\mathbf{v})$ belongs to the Voronoi cell $\bar{\mathcal{V}}(\mathbf{w}, \mathcal{L}(\mathbf{L}_n))$, i.e., $\psi^{-1}(\mathbf{v})$ is at least as close to $\mathbf{w} \in \mathcal{L}(\mathbf{L}_n)$ as to any other lattice point. Therefore, it must be $\mathbf{w} = \hat{\mathbf{w}} + \mathbf{y}$. We also know that both $\mathbf{w}$ and $\hat{\mathbf{w}}$ belong to $\mathcal{P}(\mathbf{M}_n)$, and $\mathbf{y} \in \mathcal{L}(\mathbf{M}_n)$. So, $\mathbf{w} = \hat{\mathbf{w}} + \mathbf{y}$ is possible only if $\mathbf{y} = \mathbf{0}$, which implies $\mathbf{v} = \hat{\mathbf{v}}$. □

Lemma 6.11 can be used to establish two important properties of the sampling algorithm of Lemma 6.6. The distribution $[\mathbf{v}]_{\mathbf{C}}$ produced by the sampling algorithm is uniform. However, $[\mathbf{w}]_{\mathbf{M}_n}$ is not in general uniformly distributed over G_n. The first property is that, provided $\|\mathbf{S}\|$ is large enough, the distribution of $[\mathbf{w}]_{\mathbf{M}_n}$ is relatively close to uniform.

Lemma 6.12 *Let $(\mathbf{w}, \mathbf{v})$ be generated according to the sampling procedure of Lemma 6.6. If $n\|\mathbf{S}\|\lambda_1(\mathcal{L}(\mathbf{L}_n)) \geq 6\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ and equation (6.4) holds true, then for any group element $g \in G_n$,*

$$\Pr\{[\mathbf{w}]_{\mathbf{M}_n} = g\} \approx \frac{1}{\#G_n}.$$

Proof: Fix group element g, and let $\hat{\mathbf{w}}$ be the unique lattice point in $\mathcal{L}(\mathbf{L}_n) \cap \mathcal{P}(\mathbf{M}_n)$ such that $[\hat{\mathbf{w}}]_{\mathbf{M}_n} = g$. Since $\mathbf{w} \in \mathcal{P}(\mathbf{M}_n)$, $[\mathbf{w}]_{\mathbf{M}_n} = g$ if and only if $\mathbf{w} = \hat{\mathbf{w}}$. We estimate the probability that $\mathbf{w} = \hat{\mathbf{w}}$.

Notice that if $\mathbf{v} \in \psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$, then $\mathbf{w} = \hat{\mathbf{w}}$. Therefore,

$$\Pr\{\mathbf{w} = \hat{\mathbf{w}}\} \geq \sum_{\hat{\mathbf{v}} \in \psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))) \cap \mathcal{L}(\mathbf{B})} \Pr\{\mathbf{v} = \hat{\mathbf{v}}\}.$$

By Lemma 6.11, for any $\hat{\mathbf{v}} \in \psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))) \cap \mathcal{L}(\mathbf{B})$, $\Pr\{\mathbf{v} = \hat{\mathbf{v}}\} = \det(\mathcal{L}(\mathbf{B}))/\det(\mathcal{L}(\mathbf{C}))$. So,

$$\Pr\{\mathbf{w} = \hat{\mathbf{w}}\} \geq \frac{\det(\mathcal{L}(\mathbf{B}))}{\det(\mathcal{L}(\mathbf{C}))} \cdot \#(\psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))) \cap \mathcal{L}(\mathbf{B})).$$

Similarly, if $\mathbf{w} = \hat{\mathbf{w}}$, then $\mathbf{v} \in \psi(\bar{\mathcal{V}}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$. Therefore,

$$\Pr\{\mathbf{w} = \hat{\mathbf{w}}\} \leq \sum_{\hat{\mathbf{v}} \in \psi(\bar{\mathcal{V}}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))) \cap \mathcal{L}(\mathbf{B})} \Pr\{\mathbf{v} = \hat{\mathbf{v}}\} \leq \frac{\det(\mathcal{L}(\mathbf{B}))}{\det(\mathcal{L}(\mathbf{C}))} \cdot \#(\psi(\bar{\mathcal{V}}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))) \cap \mathcal{L}(\mathbf{B})).$$

In order to complete the proof, we need to estimate the number of lattice points from $\mathcal{L}(\mathbf{B})$ that belong to $\psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$ and $\psi(\bar{\mathcal{V}}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$. Since $\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))$ contains an open sphere of radius $\lambda_1(\mathcal{L}(\mathbf{L}_n))/2$, using (6.4) we get that $\psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$ (and therefore, also $\psi(\bar{\mathcal{V}}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$) contains a sphere of radius

$$\frac{n\|\mathbf{S}\|}{3\alpha(n)} \cdot \frac{\lambda_1(\mathcal{L}(\mathbf{L}_n))}{2} \geq \hat{\zeta}(\mathcal{L}(\mathbf{B})).$$

Therefore, by the definition of $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$, the number of lattice points in $\psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$ (and $\psi(\bar{\mathcal{V}}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))$) is approximately equal to

$$\frac{\mathrm{vol}(\psi(\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\mathcal{L}(\mathbf{B}))} = \frac{\mathrm{vol}(\psi(\bar{\mathcal{V}}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\mathcal{L}(\mathbf{B}))} = \frac{\mathrm{vol}(\psi(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\mathcal{L}(\mathbf{B}))}.$$

Combining this estimate with the upper and lower bounds on the probability that $\mathbf{w} = \hat{\mathbf{w}}$, we get

$$\begin{aligned}
\Pr\{[\mathbf{w}]_{\mathbf{M}_n} = g\} &\approx \frac{\det(\mathcal{L}(\mathbf{B}))}{\det(\mathcal{L}(\mathbf{C}))} \cdot \frac{\mathrm{vol}(\psi(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\mathcal{L}(\mathbf{B}))} \\
&= \frac{\mathrm{vol}(\psi(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\psi(\mathcal{L}(\mathbf{M}_n)))} \\
&= \frac{\mathrm{vol}(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))}{\det(\mathcal{L}(\mathbf{M}_n))} \\
&= \frac{\det(\mathcal{L}(\mathbf{L}_n))}{\det(\mathcal{L}(\mathbf{M}_n))} = \frac{1}{\#G_n}.
\end{aligned}$$

□

The second property implies that, provided $\|\mathbf{S}\|$ is large enough, the distribution of $\mathbf{v} - \psi(\mathbf{w})$, as generated by the sampling procedure, is not concentrated over any fixed $(n-1)$-dimensional hyperplane. In fact, we prove a stronger property, and show that $\mathbf{v} - \psi(\mathbf{w})$ belongs to any of the two half-spaces defined by the hyperplane with high probability. This is true even for the conditional distribution of $\mathbf{v} - \psi(\mathbf{w})$ given $\mathbf{w}$.

Lemma 6.13 *Let $(\mathbf{w}, \mathbf{v})$ be generated according to the sampling procedure of Lemma 6.6. If $n\|\mathbf{S}\|\lambda_1(\mathcal{L}(\mathbf{L}_n)) \geq 12\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ and equation (6.4) holds true, then for any $\mathbf{h} \in \mathrm{span}(\mathbf{B}) \setminus \{\mathbf{0}\}$ and $g \in G_n$,*

$$\Pr\{\mathbf{h}^T \cdot (\mathbf{v} - \psi(\mathbf{w})) > 0 \mid [\mathbf{w}]_{\mathbf{M}_n} = g\} \geq \frac{1}{6}.$$

Proof: Fix group element g, and let $\hat{\mathbf{w}}$ be the unique lattice point in $\mathcal{L}(\mathbf{L}_n) \cap \mathcal{P}(\mathbf{M}_n)$ such that $[\hat{\mathbf{w}}]_{\mathbf{M}_n} = g$. Since Lemma 6.6 guarantees $\mathbf{w} \in \mathcal{P}(\mathbf{M}_n)$, condition $[\mathbf{w}]_{\mathbf{M}_n} = g$ is equivalent to $\mathbf{w} = \hat{\mathbf{w}}$. Let $\mathcal{Q} = \{\mathbf{x} \in \mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)) : \mathbf{h}^T \cdot \psi(\mathbf{x} - \hat{\mathbf{w}}) > 0\}$ be one of the two (open) halves of the Voronoi cell $\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))$ defined by the hyperplane $\mathbf{h}^T \cdot \psi(\mathbf{x}) = \mathbf{h}^T \cdot \psi(\hat{\mathbf{w}})$. (See Figure 3.) First we estimate the probability that $\mathbf{v} \in \psi(\mathcal{Q})$.

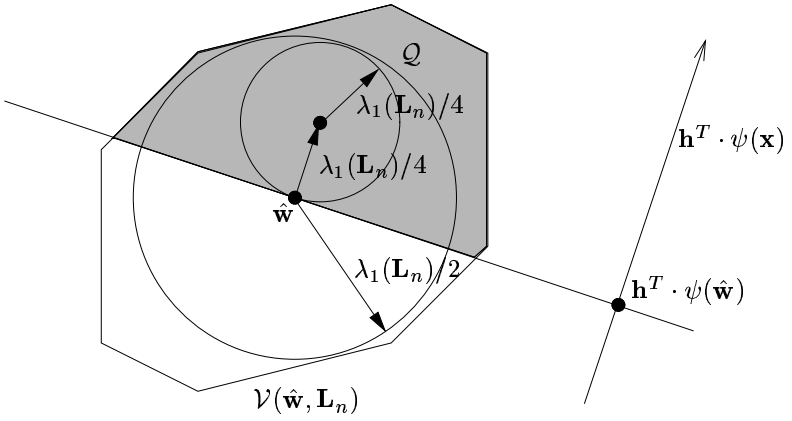

Figure 3: The conditional distribution of sampled lattice points

Since $\mathcal{Q}$ is contained in the open Voronoi cell $\mathcal{V}^o(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))$, by Lemma 6.11,

$$\Pr\{\mathbf{v} \in \psi(\mathcal{Q})\} = \sum_{\hat{\mathbf{v}} \in \psi(\mathcal{Q}) \cap \mathcal{L}(\mathbf{B})} \Pr\{\mathbf{v} = \hat{\mathbf{v}}\} = \frac{\det(\mathcal{L}(\mathbf{B}))}{\det(\mathcal{L}(\mathbf{C}))} \cdot \#(\psi(\mathcal{Q}) \cap \mathcal{L}(\mathbf{B})).$$

Notice that $\mathcal{Q}$ contains an open sphere of radius $\lambda_1(\mathcal{L}(\mathbf{L}_n))/4$. (See Figure 3.) Therefore, by (6.4), $\psi(\mathcal{Q})$ contains a sphere of radius

$$\frac{n\|\mathbf{S}\|}{3\alpha(n)} \cdot \frac{\lambda_1(\mathcal{L}(\mathbf{L}_n))}{4} \geq \hat{\zeta}(\mathcal{L}(\mathbf{B})).$$

By definition of $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$, the number of lattice points in $\psi(\mathcal{Q})$ satisfies

$$\#(\psi(\mathcal{Q}) \cap \mathcal{L}(\mathbf{B})) \approx \frac{\mathrm{vol}(\psi(\mathcal{Q}))}{\det(\mathcal{L}(\mathbf{B}))} = \frac{1}{2} \frac{\mathrm{vol}(\psi(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\mathcal{L}(\mathbf{B}))}$$

and

$$\begin{aligned}
\Pr\{\mathbf{v} \in \psi(\mathcal{Q})\} &\approx \frac{1}{2} \frac{\det(\mathcal{L}(\mathbf{B}))}{\det(\mathcal{L}(\mathbf{C}))} \cdot \frac{\mathrm{vol}(\psi(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\mathcal{L}(\mathbf{B}))} \\
&= \frac{1}{2} \frac{\mathrm{vol}(\psi(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n))))}{\det(\psi(\mathcal{L}(\mathbf{M}_n)))} \\
&= \frac{1}{2} \frac{\mathrm{vol}(\mathcal{V}(\hat{\mathbf{w}}, \mathcal{L}(\mathbf{L}_n)))}{\det(\mathcal{L}(\mathbf{M}_n))} \\
&= \frac{1}{2 \cdot \#G_n}.
\end{aligned}$$

Notice that if $\mathbf{v} \in \psi(\mathcal{Q})$ then $\mathbf{w} = \hat{\mathbf{w}}$ and $\mathbf{h}^T \cdot (\mathbf{v} - \psi(\hat{\mathbf{w}})) > 0$. Therefore,

$$\Pr\{[\mathbf{w}]_{\mathbf{M}_n} = g \wedge \mathbf{h}^T \cdot (\mathbf{v} - \psi(\hat{\mathbf{w}})) > 0\} \geq \Pr\{\mathbf{v} \in \psi(\mathcal{Q})\} \geq \frac{1}{4 \cdot \#G_n}.$$

We can now compute the conditional probability,

$$\begin{aligned}
\Pr\{\mathbf{h}^T \cdot (\mathbf{v} - \psi(\mathbf{w})) > 0 \mid [\mathbf{w}]_{\mathbf{M}_n} = g\} &= \frac{\Pr\{\mathbf{h}^T \cdot (\mathbf{v} - \psi(\mathbf{w})) > 0 \wedge [\mathbf{w}]_{\mathbf{M}_n} = g\}}{\Pr\{[\mathbf{w}]_{\mathbf{M}_n} = g\}} \\
&\geq \frac{1}{4 \cdot \#G_n \cdot \Pr\{[\mathbf{w}]_{\mathbf{M}_n} = g\}}.
\end{aligned}$$

Using Lemma 6.12, $\Pr\{[\mathbf{w}]_{\mathbf{M}_n} = g\} \lesssim 1/\#G_n$, i.e., $\#G_n \cdot \Pr\{[\mathbf{w}]_{\mathbf{M}_n} = g\} \leq 3/2$. Substituting in the previous inequality we get

$$\Pr\{\mathbf{h}^T \cdot (\mathbf{v} - \psi(\mathbf{w})) > 0 \mid [\mathbf{w}]_{\mathbf{M}_n} = g\} \geq \frac{1}{6}.$$

□

6.4 Proofs of the lemmas

In this subsection we prove Lemmas 6.8, 6.9 and 6.10 used in the analysis the algorithm $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S})$ in Subsection 6.2. The intuition behind Lemma 6.8 is that since group elements $a_{i,j}$ are independent and not too far from uniformly distributed, their sums $a_i = \sum_j a_{i,j}$ are extremely close to uniform.

Proof [of Lemma 6.8]: First consider the distribution of a single group element $a_{i,j} = [\mathbf{w}_{i,j}]_{\mathbf{M}_n}$ as output by the sampling procedure. Since $n\|\mathbf{S}\|\lambda_1(\mathcal{L}(\mathbf{L}_n)) \geq 6\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ and (6.4) holds true by assumption, Lemma 6.12 tells us that for any group element $g \in G_n$,

$$\Pr\{a_{i,j} = g\} \approx \frac{1}{\#G_n}.$$

So, the probability distribution of each $a_{i,j}$ is not too far from uniform. Adding up a relatively small number of $a_{i,j}$ we get a group element $a_i = \sum_{j=1}^{k(n)} a_{i,j}$ which is almost uniformly distributed. In particular, by Proposition 2.18, the statistical distance between a_i and a uniformly distributed $u_i \in G$ is at most

$$\Delta(a_i, u_i) \leq \frac{1}{2^{k(n)+1}}. \tag{6.8}$$

Since the random variables a_i are independent, by Proposition 2.15 the statistical distance between vector $\mathbf{a} = [a_1, \ldots, a_{m(n)}]^T$ and a uniformly distributed $\mathbf{u} \in G_n^{m(n)}$ is at most

$$\Delta(\mathbf{a}, \mathbf{u}) \leq \sum_{i=1}^{m(n)} \Delta(a_i, u_i) \leq \frac{m(n)}{2^{k(n)+1}} = \frac{n^{O(1)}}{n^{\omega(1)}} = n^{-\omega(1)}. \tag{6.9}$$

□

The intuition behind Lemma 6.9 is that since each $\mathbf{y}_{i,j} = \mathbf{v}_{i,j} - \psi(\mathbf{w}_{i,j})$ has the property described in Lemma 6.13, then also s (which is a nonzero linear combination of the $\mathbf{y}_{i,j}$'s) has a similar property.

Proof [of Lemma 6.9]: Fix $\hat{\mathbf{a}} \in G_n^{m(n)}$, $\hat{\mathbf{z}} \in \Lambda(\hat{\mathbf{a}}) \setminus \{\mathbf{0}\}$, and $(n-1)$-dimensional hyperplane $\mathcal{H} \subset \mathrm{span}(\mathbf{B})$. Since $\hat{\mathbf{z}} \neq \mathbf{0}$, there exists a coordinate i such that $\hat{z}_i \neq 0$. Assume without loss of generality that $\hat{z}_1 \neq 0$.

The output of $\mathcal{A}^{\mathcal{F}}(\mathbf{B}, \mathbf{S})$ is a random variable that depends on the randomness of oracle $\mathcal{F}$ and the randomness used during the execution of the sampling procedure in the computation of $(\mathbf{w}_{i,j}, \mathbf{v}_{i,j})$ for $i = 1, \ldots, m(n)$ and $j = 1, \ldots, k(n)$. Fix the randomness of $\mathcal{F}$ and sampling procedure, except for $(i,j) = (1,1)$. Finally, for this remaining run of the sampling procedure, fix the value of $\mathbf{w}_{1,1}$ and consider the conditional distribution of $\mathbf{v}_{1,1}$. Notice that this uniquely determines

- the values of $\mathbf{v}_{i,j}$ for all $(i,j) \neq (1,1)$,
- the values of $\mathbf{w}_{i,j}$ for all i, j,
- the value of $\mathbf{a} = \sum_{i,j} [\mathbf{w}_{i,j}]_{\mathbf{M}_n} \mathbf{e}_i$, and
- the value of $\mathbf{z} = \mathcal{F}(\mathbf{a})$.

We prove a stronger statement than the one in the lemma. Namely, we show that the conditional probability

$$\Pr\{\mathbf{s} \notin \mathcal{H} \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}, \forall (i,j).\mathbf{w}_{i,j} = \hat{\mathbf{w}}_{i,j}, \forall (i,j) \neq (1,1).\mathbf{v}_{i,j} = \hat{\mathbf{v}}_{i,j}\}$$

is at least $1/6$. Notice that this probability depends only on the conditional distribution of $\mathbf{v}_{1,1}$, because all other vectors are fixed by conditioning. Averaging over all cases such that $\mathbf{a} = \hat{\mathbf{a}}$ and $\mathbf{z} = \hat{\mathbf{z}}$, we get that $\Pr\{\mathbf{s} \notin \mathcal{H} \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}\} \geq 1/6$.

For any fixed values $\hat{\mathbf{w}}_{i,j}$, $\hat{\mathbf{v}}_{i,j}$ and $\hat{\mathbf{a}}$, define the vector

$$\mathbf{y} = \sum_{(i,j)\neq(1,1)} \hat{z}_i \cdot (\hat{\mathbf{v}}_{i,j} - \psi(\hat{\mathbf{w}}_{i,j})).$$

Notice that, given $\mathbf{z} = \hat{\mathbf{z}}$, $\mathbf{w}_{i,j} = \hat{\mathbf{w}}_{i,j}$ for all i, j, and $\mathbf{v}_{i,j} = \hat{\mathbf{v}}_{i,j}$ for all $(i,j) \neq (1,1)$,

$$\mathbf{s} = \sum_{i,j} z_i \cdot (\mathbf{v}_{i,j} - \psi(\mathbf{w}_{i,j})) = \mathbf{y} + \hat{z}_1 \cdot (\mathbf{v}_{1,1} - \psi(\hat{\mathbf{w}}_{1,1})).$$

We want to bound the conditional probability that $\mathbf{s} \in \mathcal{H}$, or, equivalently,

$$\mathbf{v}_{1,1} - \psi(\hat{\mathbf{w}}_{1,1}) \in \frac{\mathcal{H} - \mathbf{y}}{\hat{z}_1} = \mathcal{H}'.$$

Let $\mathbf{h} \in \text{span}(\mathbf{B})$ be a vector orthogonal to $\mathcal{H}'$ such that $\mathbf{h}^T \cdot \mathbf{x} \leq 0$ for any $\mathbf{x} \in \mathcal{H}'$. (Notice that since $\mathbf{h}$ is orthogonal to $\mathcal{H}'$, the function $\mathbf{x} \mapsto \mathbf{h}^T \cdot \mathbf{x}$ is constant over $\mathcal{H}'$.) Since $\mathbf{h}^T \cdot \mathbf{x} > 0$ implies $\mathbf{x} \notin \mathcal{H}'$ for all vectors $\mathbf{x}$, the conditional probability that $\mathbf{v}_{1,1} - \psi(\hat{\mathbf{w}}_{1,1}) \notin \mathcal{H}'$ is at least as big as the conditional probability that $\mathbf{h}^T \cdot (\mathbf{v}_{1,1} - \psi(\hat{\mathbf{w}}_{1,1})) > 0$. Since $n\|\mathbf{S}\|\lambda_1(\mathcal{L}(\mathbf{L}_n)) \geq 12\alpha(n)\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ and (6.4) holds true by assumption, Lemma 6.13 tells us that the latter probability is at least $1/6$. □

The intuition behind Lemma 6.10 is that vector $\mathbf{s}$ is short because it is a linear combination (with small coefficients z_i) of short vectors $\mathbf{y}_{i,j}$ lying within $\psi(\bar{\mathcal{V}}(\mathcal{L}(\mathbf{L}_n)))$. A bound on the length of $\|\mathbf{s}\|$ can be easily computed using the triangle inequality. This would lead to a result similar to the one in Theorem 6.5, but with a larger (by approximately $\sqrt{m(n) \cdot k(n)}$) value of $\gamma(n)$. Here we use cancellations between the $\mathbf{y}_{i,j}$ vectors to prove a better bound.

Proof [of Lemma 6.10]: First, we prove an upper bound on the length of $\mathbf{y}_{i,j} = \mathbf{v}_{i,j} - \psi(\mathbf{w}_{i,j})$ for all $(\mathbf{w}_{i,j}, \mathbf{v}_{i,j})$ in the range of the sampling algorithm of Lemma 6.6. We know from Lemma 6.6 that $\psi^{-1}(\mathbf{v}_{i,j}) \in \bar{\mathcal{V}}(\mathbf{w}_{i,j}, \mathcal{L}(\mathbf{L}_n))$, and therefore $\|\psi^{-1}(\mathbf{v}_{i,j}) - \mathbf{w}_{i,j}\| \leq \rho(\mathcal{L}(\mathbf{L}_n))$. Using (6.4) we immediately get that $\mathbf{y}_{i,j} = \psi(\psi^{-1}(\mathbf{v}_{i,j}) - \mathbf{w}_{i,j})$ has length at most

$$\|\mathbf{y}_{i,j}\| \leq \frac{3n\|\mathbf{S}\| \cdot \rho(\mathcal{L}(\mathbf{L}_n))}{\alpha(n)}. \tag{6.10}$$

Substituting $\alpha(n) = \omega(n\sqrt{k(n)}\beta(n)\rho(\mathcal{L}(\mathbf{L}_n)))$ into (6.10) we get

$$\|\mathbf{y}_{i,j}\| \leq \frac{3\|\mathbf{S}\|}{\omega(\sqrt{k(n)}\beta(n))} = o\left(\frac{\|\mathbf{S}\|}{\sqrt{k(n)}\beta(n)}\right). \tag{6.11}$$

By triangle inequality and Cauchy-Schwarz, it immediately follows that

$$\|\mathbf{s}\| \leq \sum_{i=1}^{m(n)} |z_i| \sum_{j=1}^{k(n)} \|\mathbf{y}_{i,j}\| \leq \sqrt{m(n)k(n)} \cdot o\left(\frac{\|\mathbf{S}\| \cdot \|\mathbf{z}\|}{\beta(n)}\right).$$

We want to prove a better (probabilistic) bound by computing the conditional expectation of $\|\mathbf{s}\|^2$. Using the definition of $\mathbf{s}$ (6.5), we get

$$\begin{aligned}
\text{Exp}[\|\mathbf{s}\|^2 \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] &= \text{Exp}[\langle \sum_{i=1}^{m(n)} \hat{z}_i \sum_{j=1}^{k(n)} \mathbf{y}_{i,j}, \sum_{i'=1}^{m(n)} \hat{z}_{i'} \sum_{j'=1}^{k(n)} \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] \\
&= \sum_{i,i'} \hat{z}_i \hat{z}_{i'} \sum_{j,j'} \text{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] \\
&\leq \sum_{i,i'} |\hat{z}_i \hat{z}_{i'}| \sum_{j,j'} |\text{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}]|.
\end{aligned}$$

We bound each term $\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}]$ separately. Notice that, given $\mathbf{a} = \hat{\mathbf{a}}$, vector $\mathbf{z} = \mathcal{F}(\mathbf{a}) = \mathcal{F}(\hat{\mathbf{a}})$ depends only on the randomness of oracle $\mathcal{F}$. Therefore, given $\mathbf{a} = \hat{\mathbf{a}}$, $\mathbf{z}$ is independent from $\mathbf{y}_{i,j}$ and $\mathbf{y}_{i',j'}$, and

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] = \mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}]. \tag{6.12}$$

We will bound the value of (6.12) and show that

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}] = \begin{cases} o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right) & \text{if } (i,j) = (i',j') \\ o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)} \cdot \frac{1}{2^{k(n)}}\right) & \text{otherwise} \end{cases}. \tag{6.13}$$

Using (6.13) in the expression for $\mathrm{Exp}[\|\mathbf{s}\|^2 \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}]$ we get

$$\begin{aligned} \mathrm{Exp}[\|\mathbf{s}\|^2 \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] &\leq \left(\sum_{i,j} \hat{z}_i^2 + \sum_{(i,j) \neq (i',j')} \frac{|\hat{z}_i| \cdot |\hat{z}_{i'}|}{2^{k(n)}} \right) \cdot o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right) \\ &\leq \left(k(n) \cdot \|\hat{\mathbf{z}}\|^2 + \frac{k(n)^2 \cdot (\sqrt{m(n)}\|\hat{\mathbf{z}}\|)^2}{2^{k(n)}} \right) \cdot o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right) \\ &\leq \left(1 + \frac{m(n)k(n)}{2^{k(n)}}\right) \cdot o\left(\frac{\|\mathbf{S}\|^2 \|\hat{\mathbf{z}}\|^2}{\beta(n)^2}\right) \end{aligned}$$

It follows from the concavity of the square root function that

$$\begin{aligned} \mathrm{Exp}[\|\mathbf{s}\| \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] &= \mathrm{Exp}[\sqrt{\|\mathbf{s}\|^2} \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}] \\ &\leq \sqrt{\mathrm{Exp}[\|\mathbf{s}\|^2 \mid \mathbf{a} = \hat{\mathbf{a}}, \mathbf{z} = \hat{\mathbf{z}}]} \\ &\leq o\left(\frac{\|\mathbf{S}\| \cdot \|\hat{\mathbf{z}}\|}{\beta(n)}\right) \cdot \sqrt{1 + \frac{m(n)k(n)}{2^{k(n)}}}. \end{aligned}$$

In order to complete the proof of the lemma, we need to prove bound (6.13). The case $(i,j) = (i',j')$ immediately follows from (6.11):

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}] \leq \max \|\mathbf{y}_{i,j}\|^2 = o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right)$$

where the maximum is over all $\mathbf{y}_{i,j}$ in the support of the sampling algorithm. Now assume $i = i'$, but $j \neq j'$, and let's bound $\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}]$. Notice that a_h is independent from $\mathbf{y}_{i,j}$ and $\mathbf{y}_{i,j'}$ for all $h \neq i$. Therefore

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}] = \mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle \mid a_i = \hat{a}_i].$$

We want to bound the conditional expectation of $\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle$ given $a_i = \hat{a}_i$. Notice that vectors $\mathbf{y}_{i,j}$ and $\mathbf{y}_{i,j'}$ are statistically independent (because they come from different runs of the sampling procedure), and symmetrically distributed (by Lemma 6.6). Therefore,

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle] = \langle \mathrm{Exp}[\mathbf{y}_{i,j}], \mathrm{Exp}[\mathbf{y}_{i,j'}] \rangle = \langle \mathbf{0}, \mathbf{0} \rangle = 0. \tag{6.14}$$

Using (6.14), we immediately get

$$|\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle \mid a_i = \hat{a}_i]| = |\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle \mid a_i = \hat{a}_i] - \mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle]|. \tag{6.15}$$

Notice that, by (6.11), for any $\mathbf{y}_{i,j}$ and $\mathbf{y}_{i,j'}$ in the range of the sampling procedure,

$$|\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle| \leq \|\mathbf{y}_{i,j}\| \cdot \|\mathbf{y}_{i,j'}\| = o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right).$$

Therefore, by Proposition 2.16, the difference (6.15) is at most

$$2 \cdot o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right) \cdot \Delta((\mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{y}_{i,j}, \mathbf{y}_{i,j'})),$$

where $(\mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \mid a_i = \hat{a}_i)$ is the conditional distribution of $(\mathbf{y}_{i,j}, \mathbf{y}_{i,j'})$ given a_i. In the following lemma, we bound the statistical distance $\Delta((\mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{y}_{i,j}, \mathbf{y}_{i,j'}))$.

Lemma 6.14 *The statistical distance between* $(\mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \mid a_i = \hat{a}_i)$ *and* $(\mathbf{y}_{i,j}, \mathbf{y}_{i,j'})$ *is at most*

$$\Delta((\mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{y}_{i,j}, \mathbf{y}_{i,j'})) \leq \frac{5}{2(2^{k(n)} - 1)}.$$

Proof: Notice that $(\mathbf{y}_{i,j}, \mathbf{y}_{i,j'}) = (\mathbf{v}_{i,j} - \psi(\mathbf{w}_{i,j}), \mathbf{v}_{i,j'} - \psi(\mathbf{w}_{i,j'}))$ is a randomized function of $(\mathbf{w}_{i,j}, \mathbf{w}_{i,j'})$, where $\mathbf{v}_{i,j}$ and $\mathbf{v}_{i,j'}$ are computed according to the conditional distribution of the sampling algorithm (given $\mathbf{w} = \mathbf{w}_{i,j}$ or $\mathbf{w} = \mathbf{w}_{i,j'}$). Therefore, by Proposition 2.14,

$$\Delta((\mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{y}_{i,j}, \mathbf{y}_{i,j'})) \leq \Delta((\mathbf{w}_{i,j}, \mathbf{w}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{w}_{i,j}, \mathbf{w}_{i,j'})).$$

By Proposition 2.19, the distance between $(\mathbf{w}_{i,j}, \mathbf{w}_{i,j'} \mid a_i = \hat{a}_i)$ and $(\mathbf{w}_{i,j}, \mathbf{w}_{i,j'})$ is at most

$$\begin{aligned}
\Delta((\mathbf{w}_{i,j}, \mathbf{w}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{w}_{i,j}, \mathbf{w}_{i,j'})) &\leq \frac{1}{2} \max_{\hat{\mathbf{w}}, \hat{\mathbf{w}}'} \left| \frac{\Pr\{a_i = \hat{a}_i \mid \mathbf{w}_{i,j} = \hat{\mathbf{w}}, \mathbf{w}_{i,j'} = \hat{\mathbf{w}}'\}}{\Pr\{a_i = \hat{a}_i\}} - 1 \right| \\
&= \frac{1}{2} \max_{\hat{\mathbf{w}}, \hat{\mathbf{w}}'} \left| \frac{\Pr\{\sum_{h \notin \{j,j'\}} a_{i,h} = \hat{a}_i - [\hat{\mathbf{w}}]_{\mathbf{M}_n} - [\hat{\mathbf{w}}']_{\mathbf{M}_n}\}}{\Pr\{\sum_{h=1}^{k(n)} a_{i,h} = \hat{a}_i\}} - 1 \right|.
\end{aligned}$$

By Proposition 2.18, the probability at the denominator equals $(1/\#G_n)(1+\epsilon)$ for some $|\epsilon| \leq 2^{-k(n)}$. Similarly, the probability at the numerator equals $(1/\#G_n)(1+\epsilon')$ for some $|\epsilon'| \leq 2^{-(k(n)-2)}$. It follows that

$$\begin{aligned}
\Delta((\mathbf{w}_{i,j}, \mathbf{w}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{w}_{i,j}, \mathbf{w}_{i,j'})) &\leq \frac{1}{2} \left| \frac{(1/\#G_n)(1+\epsilon')}{(1/\#G_n)(1+\epsilon)} - 1 \right| \\
&= \frac{1}{2} \left| \frac{\epsilon' - \epsilon}{1+\epsilon} \right| \\
&\leq \frac{1}{2} \frac{|\epsilon'| + |\epsilon|}{1 - |\epsilon|} \\
&\leq \frac{5}{2(2^{k(n)} - 1)}.
\end{aligned}$$

□

Using Lemma 6.14, we get that

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}] \leq 2 \cdot o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right) \cdot \frac{5}{2(2^{k(n)} - 1)} = o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)} \cdot \frac{1}{2^{k(n)}}\right),$$

proving (6.13) for the case when $i = i'$ and $j \neq j'$.

The case when $i \neq i'$ is similar. Consider the conditional distribution of $\mathbf{y}_{i,j}, \mathbf{y}_{i',j'}$ given $\mathbf{a} = \hat{\mathbf{a}}$. Notice that a_h is independent from $\mathbf{y}_{i,j}$ and $\mathbf{y}_{i',j'}$ for all $h \notin \{i, i'\}$. Therefore

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}] = \mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid a_i = \hat{a}_i, a_{i'} = \hat{a}_{i'}].$$

Moreover, $\mathbf{y}_{i,j}$ and a_i are independent from $\mathbf{y}_{i',j'}$ and $a_{i'}$ because they come from different runs of the sampling algorithm. Therefore,

$$\begin{aligned}
\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid a_i = \hat{a}_i, a_{i'} = \hat{a}_{i'}] &= \mathrm{Exp}[\langle (\mathbf{y}_{i,j} \mid a_i = \hat{a}_i), (\mathbf{y}_{i',j'} \mid a_{i'} = \hat{a}_{i'}) \rangle] \\
&= \langle \mathrm{Exp}[\mathbf{y}_{i,j} \mid a_i = \hat{a}_i], \mathrm{Exp}[\mathbf{y}_{i',j'} \mid a_{i'} = \hat{a}_{i'}] \rangle,
\end{aligned}$$

where, as usual, $(\mathbf{y}_{i,j} \mid a_i = \hat{a}_i)$ (resp. $(\mathbf{y}_{i',j'} \mid a_{i'} = \hat{a}_{i'})$) is the conditional distribution of $\mathbf{y}_{i,j}$ given a_i (resp. $\mathbf{y}_{i',j'}$ given $a_{i'}$). Let $\mathbf{y} = \mathrm{Exp}[\mathbf{y}_{i',j'} \mid a_{i'} = \hat{a}_{i'}]$. We know from (6.11) that $\|\mathbf{y}\| = o(\|\mathbf{S}\|/(\beta(n)\sqrt{k(n)}))$. Since $\mathbf{y}_{i,j}$ is symmetrically distributed, $\langle \mathrm{Exp}[\mathbf{y}_{i,j}], \mathbf{y} \rangle = \langle \mathbf{0}, \mathbf{y} \rangle = 0$, and

$$\begin{aligned}
\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid a_i = \hat{a}_i, a_{i'} = \hat{a}_{i'}] &= \langle \mathrm{Exp}[\mathbf{y}_{i,j} \mid a_i = \hat{a}_i], \mathbf{y} \rangle \\
&= \langle \mathrm{Exp}[\mathbf{y}_{i,j} \mid a_i = \hat{a}_i], \mathbf{y} \rangle - \langle \mathrm{Exp}[\mathbf{y}_{i,j}], \mathbf{y} \rangle \\
&= \mathrm{Exp}[\langle (\mathbf{y}_{i,j} \mid a_i = \hat{a}_i), \mathbf{y} \rangle] - \mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y} \rangle].
\end{aligned}$$

Notice that, by (6.11), for all $\mathbf{y}_{i,j}$ in the range of the sampling algorithm

$$|\langle \mathbf{y}_{i,j}, \mathbf{y} \rangle| \leq \|\mathbf{y}_{i,j}\| \cdot \|\mathbf{y}\| = o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right).$$

Therefore, by Proposition 2.16, the difference between $\mathrm{Exp}[\langle (\mathbf{y}_{i,j} \mid a_i = \hat{a}_i), \mathbf{y} \rangle]$ and $\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y} \rangle]$ is at most

$$2 \cdot o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right) \cdot \Delta((\mathbf{y}_{i,j} \mid a_i = \hat{a}_i), (\mathbf{y}_{i,j})).$$

Since (for any j') vector $\mathbf{y}_{i,j}$ is a function of $(\mathbf{y}_{i,j}, \mathbf{y}_{i,j'})$, by Proposition 2.14 and Lemma 6.14,

$$\Delta((\mathbf{y}_{i,j} \mid a_i = \hat{a}_i), (\mathbf{y}_{i,j})) \leq \Delta((\mathbf{y}_{i,j}, \mathbf{y}_{i,j'} \mid a_i = \hat{a}_i), (\mathbf{y}_{i,j}, \mathbf{y}_{i,j'})) \leq \frac{5}{2(2^{k(n)} - 1)}.$$

So, also in this case we have

$$\mathrm{Exp}[\langle \mathbf{y}_{i,j}, \mathbf{y}_{i',j'} \rangle \mid \mathbf{a} = \hat{\mathbf{a}}] \leq 2 \cdot o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)}\right) \cdot \frac{5}{2(2^{k(n)} - 1)} = o\left(\frac{\|\mathbf{S}\|^2}{\beta(n)^2 k(n)} \cdot \frac{1}{2^{k(n)}}\right).$$

□

7 Applications

In the previous section we proved (Theorems 6.3 and 6.5) that the problem of finding n linearly independent lattice vectors of length not much bigger than the generalized uniform radius (in the worst case) reduces to the problem of finding small integer solutions to random linear equations on the average. In this section we show how this result can be reformulated as a connection between the average-case and worst-case complexity of various lattice approximation problems. As usual, we refer to the average-case problem as the problem of finding a nonzero integer solution to a random linear equation, but we stress that this is equivalent to finding (approximately) shortest vectors in a random lattice.

We also show that our results imply the existence of provably secure cryptographic (collision resistant) hash functions based on the worst-case hardness of lattice approximation problems.

Corollary 7.1 *Let $\tau(n) = n^{O(1)}$ be a function such that there exists a family of $\tau(n)$-perfect easily decodable lattices. For every polynomially bounded functions $\mu(m) = m^{O(1)}$ and $m(n) = \Omega(n \log n)$, there exists a sequence of groups $\{G_n\}$ of size $\#G_n = n^{O(n)}$ such that the following is true. If there is a probabilistic polynomial time algorithm $\mathcal{F}$ that on input a uniformly chosen random equation $\mathbf{g} \in G_n^{m(n)}$, outputs with non-negligible probability a nonzero solution $\mathcal{F}(\mathbf{g}) \in \Lambda(\mathbf{g})$ of length within a factor $\mu(m(n))$ from the shortest (or, more generally, within a factor $\mu(m(n))$ from Minkowski's bound (2.2)), then there is a probabilistic polynomial time algorithm that on input any rank n lattice basis $\mathbf{B}$ solves, in the worst case and with high probability, any of the following problems, where $\omega(1)$ is an arbitrary superconstant and polynomially bounded function of n:*

1. [SIVP] *Find a set $\mathbf{S} \subseteq \mathcal{L}(\mathbf{B})$ of n linearly independent vectors such that*

$$\|\mathbf{S}\| \leq \omega(1) \cdot \mu(m(n)) \cdot n^{1.5} \cdot \tau(n) \cdot \sqrt{m(n) \cdot \log n} \cdot \lambda_n(\mathcal{L}(\mathbf{B})).$$

2. [GapSVP] *Compute an approximation $\hat{\lambda}_1$ such that*

$$\frac{\lambda_1(\mathcal{L}(\mathbf{B}))}{\omega(1) \cdot \mu(m(n)) \cdot n^2 \cdot \tau(n) \cdot \sqrt{m(n) \cdot \log n}} \leq \hat{\lambda}_1 \leq \lambda_1(\mathcal{L}(\mathbf{B})).$$

3. [GapCRP] *Compute an approximation $\hat{\rho}$ such that*

$$\rho(\mathcal{L}(\mathbf{B})) \leq \hat{\rho} \leq \omega(1) \cdot \mu(m(n)) \cdot n^{1.5} \cdot \tau(n) \cdot \sqrt{m(n) \cdot \log n} \cdot \rho(\mathcal{L}(\mathbf{B})).$$

4. [GDD] *Given also a target vector* $\mathbf{t} \in \mathrm{span}(\mathbf{B})$*, find a lattice vector* $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ *such that*

$$\|\mathbf{v} - \mathbf{t}\| \leq \omega(1) \cdot \mu(m(n)) \cdot n^{1.5} \cdot \tau(n) \cdot \sqrt{m(n) \cdot \log n} \cdot \rho(\mathcal{L}(\mathbf{B})).$$

Proof: Let $\beta(n) = \sqrt{\omega(1) \cdot m(n)} \cdot \mu(m(n))$, $\gamma(n) = \beta(n)\tau(n)\sqrt{\omega(1) \cdot \log n}$ and G_n be as defined in Theorem 6.5. Notice that $\gamma(n) \leq \omega(1) \cdot \sqrt{m(n) \cdot \log n} \cdot \tau(n) \cdot \mu(m(n)) = n^{O(1)}$ and $\#G_n \leq (n^{1.5}\gamma(n)/8)^n = n^{O(n)}$. Therefore, by Theorem 5.5, any equation $\mathbf{g} \in G_n^{m(n)}$ has a nonzero solution of length at most $O(\sqrt{m(n)})$. Let $\mathcal{F}(\cdot)$ be a probabilistic polynomial time algorithm to find nonzero solutions of length within a factor $\mu(m(n))$ from the shortest (with non-negligible probability). We know that, when successful, $\mathcal{F}(\cdot)$ finds solutions of length at most

$$\|\mathcal{F}(\mathbf{g})\| \leq \mu(m(n))O(\sqrt{m(n)}) \leq \beta(n).$$

So, algorithm $\mathcal{F}(\cdot)$ solves $\mathrm{HSIS}_{G,m,\beta}$ on the average with non-negligible probability. Combining $\mathcal{F}$ with the reductions from Theorems 6.5 and 6.3, we get a polynomial time algorithm $\mathcal{S}(\cdot)$ that solves $\mathrm{GIVP}_\gamma^{\hat{\zeta}}$ (in the worst case and with high probability) for approximation factor

$$\gamma(n) = \beta(n) \cdot \tau(n) \cdot \sqrt{\omega(1) \cdot \log n} = \omega(1) \cdot \sqrt{m(n) \cdot \log n} \cdot \mu(m(n)) \cdot \tau(n).$$

We show how to use this algorithm to solve all the problems in the conclusion of the theorem.

1. [SIVP] Just run $\mathbf{S} = \mathcal{S}(\mathbf{B})$ and output $\mathbf{S}$. By (3.1),

$$\|\mathbf{S}\| \leq \gamma(n) \cdot \hat{\zeta}(\mathcal{L}(\mathbf{B})) \leq \frac{3}{2} n^{1.5} \cdot \gamma(n) \cdot \lambda_n(\mathcal{L}(\mathbf{B})) = \frac{3}{2}\omega(1) \cdot n^{1.5} \cdot \tau(n) \cdot \mu(m(n)) \cdot \sqrt{m(n) \cdot \log n} \cdot \lambda_n(\mathcal{L}(\mathbf{B})).$$

2. [GapSVP] On input basis $\mathbf{B}$, run $\mathbf{S} = \mathcal{S}(\mathbf{B}^*)$ where $\mathbf{B}^*$ is the basis of the dual lattice, and output $1/\|\mathbf{S}\|$. By (2.5),

$$\|\mathbf{S}\| \geq \lambda_n(\mathcal{L}(\mathbf{B})^*) \geq \frac{1}{\lambda_1(\mathcal{L}(\mathbf{B}))}.$$

   Also, by (3.2),

$$\|\mathbf{S}\| \leq \gamma(n) \cdot \hat{\zeta}(\mathcal{L}(\mathbf{B})^*) \leq \frac{3}{2} n^2 \cdot \gamma(n)/\lambda_1(\mathcal{L}(\mathbf{B})) = \frac{3}{2}\omega(1) \cdot n^2 \cdot \tau(n) \cdot \mu(m(n)) \cdot \sqrt{m(n) \cdot \log n}/\lambda_1(\mathcal{L}(\mathbf{B})).$$

3. [GapCRP] This time, we run $\mathbf{S} = \mathcal{S}(\mathbf{B})$ and output $\sqrt{n}\|\mathbf{S}\|/2$. By (2.3),

$$\sqrt{n}\|\mathbf{S}\|/2 \geq (\sqrt{n}/2)\lambda_n(\mathcal{L}(\mathbf{B})) \geq \rho(\mathcal{L}(\mathbf{B})).$$

   Moreover, by Theorem 3.6,

$$\sqrt{n}\|\mathbf{S}\|/2 \leq \frac{3}{2} n^{1.5} \cdot \gamma(n) \cdot \rho(\mathcal{L}(\mathbf{B})) = \frac{3}{2}\omega(1) \cdot n^{1.5} \cdot \tau(n) \cdot \mu(m(n)) \cdot \sqrt{m(n) \cdot \log n} \cdot \rho(\mathcal{L}(\mathbf{B})).$$

4. [GDD] In order to find a lattice point close to target $\mathbf{t}$, we first run $\mathbf{S} = \mathcal{S}(\mathbf{B})$ and then execute Babai's nearest plane algorithm [6] using sublattice $\mathbf{S}$ and target $\mathbf{t}$. The result is a point within distance $\sqrt{n}\|\mathbf{S}\|/2$ from the target. As in the proof for the covering radius problem, this bound satisfies

$$\sqrt{n}\|\mathbf{S}\|/2 \leq \frac{3}{2}\omega(1) \cdot n^{1.5} \cdot \tau(n) \cdot \mu(m(n)) \cdot \sqrt{m(n) \cdot \log n} \cdot \rho(\mathcal{L}(\mathbf{B})).$$

□

Notice that in the proof of Corollary 7.1, the definition of group G_n implicitly depends on the function $m(n)$. This is because in Theorem 6.5 the definition of group G_n depends on the value of $\alpha(n)$, which in turn, depends (via $\gamma(n)$) on the value of $\beta(n)$. Moreover, the definition of $\beta(n)$ in the proof of Corollary 7.1 depends on $\mu(m(n))$. So, unless $\mu(\cdot)$ is a constant function, group G_n can be selected only after the value of $m(n)$ has been chosen. The following corollary immediately follows from Corollary 7.1 by setting $m(n) = \Theta(n \log n)$ and $\mu(m) = 1$, and observing that the definition of group G_n does not depend on $m(n)$ when $\mu(m)$ is constant.

Corollary 7.2 *Let $\tau(n) = n^{O(1)}$ be a function such that there exists a family of $\tau(n)$-perfect easily decodable lattices. For every superlogarithmic function $\omega(\log n)$, there exists a sequence of groups $\{G_n\}$ of size $\#G_n = n^{O(n)}$ such that for any $m(n) = \Theta(n \log n)$, the following is true. If there is a probabilistic polynomial time algorithm $\mathcal{F}(\cdot)$ that on input a uniformly chosen random equation $\mathbf{g} \in G_n^{m(n)}$, outputs with non-negligible probability a shortest nonzero solution $\mathcal{F}(\mathbf{g}) \in \Lambda(\mathbf{g})$ (or, more generally, a solution satisfying Minkowski's bound (2.2) $\|\mathcal{F}(\mathbf{g})\| \leq \sqrt{m(n)} \cdot \det(\Lambda(\mathbf{g}))^{1/m(n)}$), then there is a probabilistic polynomial time algorithm that on input any rank n lattice basis $\mathbf{B}$ solves, in the worst case and with high probability, any of the following problems:*

1. [SIVP] *Find a maximal set of linearly independent vectors of length within $n^2 \cdot \tau(n) \cdot \omega(\log n)$ from the shortest.*
2. [GapSVP] *Approximate $\lambda_1(\mathcal{L}(\mathbf{B}))$ within a factor $n^{2.5} \cdot \tau(n) \cdot \omega(\log n)$.*
3. [GapCRP] *Approximate $\rho(\mathcal{L}(\mathbf{B}))$ within a factor $n^2 \cdot \tau(n) \cdot \omega(\log n)$.*
4. [GDD] *Given also a target vector $\mathbf{t} \in \text{span}(\mathbf{B})$, find a lattice vector $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ within distance $n^2 \cdot \tau(n) \cdot \omega(\log n) \cdot \rho(\mathcal{L}(\mathbf{B}))$ from $\mathbf{t}$.*

We now turn to the construction of collision resistant hash functions. Following [18], for any $\mathbf{g} \in G_n^{m(n)}$, define function $h_{\mathbf{g}}: \{0,1\}^{m(n)} \to G_n$ by

$$h_{\mathbf{g}}(\mathbf{x}) = \sum_{i=1}^{n} g_i x_i.$$

Notice that function $h_{\mathbf{g}}$ maps $m(n)$ bits to $\log_2 \#G$ bits. If $m(n) > \log_2 \#G$, then the function compresses the input $\mathbf{x}$, and collisions $h_{\mathbf{g}}(\mathbf{x}) = h_{\mathbf{g}}(\mathbf{y})$ (for $\mathbf{x} \neq \mathbf{y}$) are guaranteed to exist by the pigeon hole principle. We prove that, if the key $\mathbf{g}$ is chosen at random, then these collisions are computationally hard to find.

Corollary 7.3 *Let $\tau(n) = n^{O(1)}$ be a function such that there exists a family of $\tau(n)$-perfect easily decodable lattices. For every superlogarithmic function $\omega(\log n)$, there exists a sequence of groups $\{G_n\}$ of size $\#G_n = n^{O(n)}$ such that the following is true. Assume no probabilistic polynomial time algorithm can solve problems* SIVP, GapSVP, GapCRP *or* GDD *(in the worst case and with high probability) within the factors specified in Corollary 7.2. Then for any $c > 1$ and $m(n) = \max\{c \cdot \log_2 \#G_n, \Theta(n \log n)\}$, there exist no probabilistic polynomial time algorithm that on input a random key $\mathbf{g} \in G_n^{m(n)}$, outputs with non-negligible probability an $h_{\mathbf{g}}$-collision, i.e., two binary vectors $\mathbf{x} \neq \mathbf{y}$ such that $h_{\mathbf{g}}(\mathbf{x}) = h_{\mathbf{g}}(\mathbf{y})$.*

Proof: Notice that $m(n) \geq c \cdot \log_2 \#G_n$, so function $h_{\mathbf{g}}$ is a hash function with compression ratio c. Assume, for contradiction, that $\mathcal{F}(\mathbf{g}) = (\mathbf{x}, \mathbf{y})$ is a collision finder algorithm with non-negligible success probability, and notice that if $\mathcal{F}$ is successful, then $\mathbf{x} - \mathbf{y} \in \Lambda(\mathbf{g}) \setminus \{\mathbf{0}\}$ is a nonzero solution to equation $\mathbf{g}$ of length at most

$$\|\mathbf{x} - \mathbf{y}\| \leq \sqrt{m(n)}.$$

Since $\Lambda(\mathbf{g})$ is a sublattice of $\mathbb{Z}^n$, $\det(\Lambda(\mathbf{g})) \geq \det(\mathbb{Z}^n) = 1$, and solution $\mathbf{x} - \mathbf{y} \in \Lambda(\mathbf{g})$ satisfies Minkowski's bound (2.2)

$$\|\mathbf{x} - \mathbf{y}\| \leq \sqrt{m(n)} \leq \sqrt{m(n)} \det(\Lambda(\mathbf{g}))^{1/m(n)}.$$

In order to apply Corollary 7.2 and get a contradiction we only need to show that $m(n) = \Theta(n \log n)$. The lower bound $m(n) = \Omega(n \log n)$ immediately follows from the definition of $m(n) \geq \Theta(n \log n)$. The upper bound $m(n) = O(n \log n)$ follows from the fact that $\#G_n = n^{O(1)}$. This proves that $m(n) = \Theta(n \log n)$, and by Corollary 7.2 there exist probabilistic polynomial time algorithms to approximately solve SIVP, GapSVP, GapCRP and GDD in the worst case and with high probability. □

We conclude the section with two remarks about the choice of the groups G_n in the previous corollaries.

Remark 7.4 *It can be shown that the groups G_n defined in the proofs of Corollaries 7.1, 7.2 and 7.3, have size $\#G_n = n^{\Theta(n)}$. In particular, in Corollary 7.3, we could have simply defined $m(n) = c \log_2 \#G_n$, instead of $\max\{c \log_2 \#G_n, \Theta(n \log n)\}$, because $c \log_2 \#G_n = \Theta(n \cdot \log n)$.*

Remark 7.5 *Corollaries 7.1, 7.2 and 7.3 are pretty flexible in terms of the choice of group G_n. The only properties required for the proof to go through are that G_n is a group of size $n^{\Theta(n)}$ that can be represented as the quotient of an easily decodable $\tau(n)$-perfect lattice $\mathcal{L}(\mathbf{L}_n)$ modulo a sublattice $\mathcal{L}(\mathbf{M}_n)$ such that (6.2) holds true.*

8 Conclusion and open problems

We related the computational complexity of finding (approximately) shortest nonzero integer solutions to random linear equations with coefficients in a suitably chosen group (on the average and with non-negligible probability) to the worst-case complexity of approximating various lattice problems. Since the set of integer solutions to a homogeneous linear equation forms a lattice, the result can be interpreted as a connection between the average-case and worst-case complexity of various lattice problems. The connection immediately gives also provably secure cryptographic hash functions that are as hard to break on the average as the worst case complexity of approximating various lattice problems within polynomial factors. The worst-case approximation factors achieved depend on the class of easily decodable lattices used in the definition of the class of equations (or cryptographic hash functions). In particular, if $\tau(n)$-perfect easily decodable lattices are used, then finding shortest solutions to random equations (or finding collisions to hash functions) is at least as hard as approximating the length of the shortest vector in any lattice, in the worst case, within a factor $\gamma(n) = n^{2.5}\tau(n)\omega(\log n)$. Even for $\tau(n) = \sqrt{n}$ (which corresponds, as a special case, to Ajtai's random class of equations,) this improves previously known best connection factor of [11] by more than $O(n)$. We also showed that finding shortest solutions to random equations is at least as hard as approximating within a factor $n^2\tau(n)\omega(\log n)$ any of the following problems:

- [SIVP] computing a maximal set of shortest linearly independent vectors,
- [GAPCRP] computing the covering radius, and
- [GDD] computing a lattice vector within distance $\max_{\mathbf{x}} \mathrm{dist}(\mathbf{x}, \mathcal{L}(\mathbf{B}))$ from a given target,

improving [11] in the case of SIVP by more than $O(\sqrt{n})$, and connecting the average case complexity of solving random equations to two new computational problems on lattices that might be of independent interest.

We also gave polynomial time constructions of easily decodable $\tau(n)$-perfect lattices with $\tau(n) = o(\sqrt{n})$. These constructions allow to achieve approximation factors $n^{2.5}\omega(\sqrt{\log n \log\log n})$ (for SIVP, GAPCRP and GDD) and $n^3\omega(\sqrt{\log n \log\log n})$ (for SVP). While this improvement over $\tau(n) = \sqrt{n}$ is not substantial, it suggests that further investigation of almost perfect lattices might allow to find easily decodable $\tau(n)$-perfect lattices with much smaller $\tau(n)$, e.g., $\tau(n) = n^\epsilon$ or even $\tau(n) = O(1)$. This would immediately reduce the approximation factor for all the above problems by about $\sqrt{n}$.

Another possible source of improvement are better bounds relating the fundamental constants associated to a lattice. Our main theorem (Theorem 6.5) shows that finding short solutions on the average is at least as hard as finding vectors that are not much longer than a new lattice quantity called the generalized uniform radius. All other results are obtained by first relating the generalized uniform radius to the covering radius (Theorem 3.6), and then bounding the covering radius in terms of other lattice constants using standard transference theorems and other well known bounds (Proposition 2.9). In particular, (3.1) and (3.2) show that the generalized uniform radius $\hat{\zeta}(\mathcal{L}(\mathbf{B}))$ is at most $O(n^{1.5})$ times $\lambda_n(\mathcal{L}(\mathbf{B}))$ or at most $O(n^2)$ times $1/\lambda_1(\mathcal{L}(\mathbf{B})^*)$. It would be interesting to improve (3.1) and (3.2) to show, for example, that

$$\hat{\zeta}(\mathcal{L}(\mathbf{B})) \leq O(n)\lambda_n(\mathcal{L}(\mathbf{B})) \tag{8.1}$$

and

$$\hat{\zeta}(\mathcal{L}(\mathbf{B})) \leq O(n)/\lambda_1(\mathcal{L}(\mathbf{B})^*). \tag{8.2}$$

Whether these bounds hold true is a natural geometric question, and proving them would be of independent interest. Moreover, it would allow to reduce the approximation factors for SIVP and SVP by $O(\sqrt{n})$ and $O(n)$, respectively. Together with the construction of better almost perfect easily decodable lattices,

this would immediately improve the approximation factors for both SVP and SIVP to just $n^{1.5}\omega(\log n)$. Connections with such small approximation factors are currently known only for restrictions of the (worst-case) shortest vector problem to lattices with special structure where the shortest vector is unique is some technical sense [39].

Notice that by (2.5), bound (8.2) would also imply (8.1). Also, (8.2), if correct, would be asymptotically optimal because Conway and Thompson (see [37]) showed that there exist self dual lattices such that $\rho(\mathcal{L}(\mathbf{B}))\cdot\lambda_1(\mathcal{L}(\mathbf{B})^*) \geq O(n)$, and by Proposition 3.2 $\rho(\mathcal{L}(\mathbf{B})) \leq \zeta(\mathcal{L}(\mathbf{B})) \leq \hat{\zeta}(\mathcal{L}(\mathbf{B}))$. We conjecture that (8.2) holds true, and that there exist classes of random equations such that finding shortest nonzero solutions on the average (with non-negligible probability) is at least as hard as approximating the length of the shortest nonzero vector (or finding a maximal set of shortest linearly independent vectors) in any n-dimensional lattice within a factor $n^{1.5}\omega(\log n)$.

9 Acknowledgments

The author would like to thank Ravi Kannan and Alex Vardy for interesting discussions and pointers to relevant references, Oded Regev and an anonymous reviewer for their many and valuable comments that helped to simplify and improve the presentation, and Oded Goldreich for his invaluable feedback about the paper.

References

[1] D. Aharonov and O. Regev. Lattice problems in NP $\cap$ coNP. Manuscript, 2004.

[2] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing - STOC '96*, pages 99–108, Philadelphia, PA, USA, May 1996. ACM.

[3] M. Ajtai. The shortest vector problem in l_2 is NP-hard for randomized reductions (extended abstract). In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing - STOC '98*, pages 10–19, Dallas, TX, USA, May 1998. ACM.

[4] M. Ajtai and C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing - STOC '97*, pages 284–293, El Paso, TX, USA, May 1997. ACM.

[5] M. Ajtai, R. Kumar, and D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing - STOC '01*, pages 266–275, Heraklion, Cre., Greece, July 2001. ACM.

[6] L. Babai. On Lovasz' lattice reduction and the nearest lattice point problem. *Combinatorica*, 6(1):1–13, 1986. Preliminary version in STACS 1985.

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

[8] J. Blömer and J.-P. Seifert. On the complexity of computing short linearly independent vectors and short bases in a lattice. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing - STOC '99*, pages 711–720, Atlanta, GA, USA, May 1999. ACM.

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

[10] G. J. Butler. Simultaneous packing and covering in euclidean space. *Proceedings of the London Mathematical Society*, 25:721–735, 1972.

[11] J.-Y. Cai and A. P. Nerurkar. An improved worst-case to average-case connection for lattice problems (extended abstract). In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science - FOCS '97*, pages 468–477, Miami Beach, FL, USA, Oct. 1997. IEEE.

[12] J. H. Conway and N. J. A. Sloane. *Sphere Packings, Lattices and Groups*. Springer Verlag, 3rd edition, 1998.

[13] 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. Preliminary version in FOCS 1998.

[14] M. Dyer, A. Frieze, and R. Kannan. A random polynomial-time algorithm for approximating the volume of convex bodies. *Journal of the ACM*, 38(1):1–17, Jan. 1991.

[15] U. Feige and D. Micciancio. The inapproximability of lattice and coding problems with preprocessing. *Journal of Computer and System Sciences*, 2004. (To appear. Preliminary version in CCC 2002.).

[16] O. Goldreich. *Foundation of Cryptography - Basic Tools*. Cambridge Unievrsity Press, 2001.

[17] O. Goldreich and S. Goldwasser. On the limits of nonapproximability of lattice problems. *Journal of Computer and System Sciences*, 60(3):540–563, 2000. Preliminary version in STOC 1998.

[18] O. Goldreich, S. Goldwasser, and S. Halevi. Collision-free hashing from lattice problems. Technical Report TR96-056, Electronic Colloquium on Computational Complexity (ECCC), 1996.

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

[20] V. Guruswami, D. Micciancio, and O. Regev. The comlexity of the covering radius problem on lattices and codes. In *Proceedings of the 19th Annual IEEE Conference on Computational Complexity - CCC '04*, Amherst, MA, USA, June 2004. IEEE. To appear.

[21] I. Honkala and A. Tietäväinen. *Handbook of Coding Theory*, volume 2, chapter 13: "Codes and number theory", pages 1141–1194. Elsevier, 1998.

[22] R. Kannan. *Annual reviews of computer science*, volume 2, chapter "Algorithmic geometry of numbers", pages 231–267. Annual Review Inc., Palo Alto, California, 1987.

[23] R. Kannan. Minkowski's convex body theorem and integer programming. *Mathematics of operation research*, 12(3):415–440, Aug. 1987.

[24] R. Kannan. Lattice translates of a polytope and the Frobenius problem. *Combinatorica*, 12(2):161–177, 1992.

[25] R. Kannan and S. Vempala. Sampling lattice points. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing - STOC '97*, pages 696–700, El Paso, TX, USA, May 1997. ACM.

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

[27] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261:513–534, 1982.

[28] A. Lobstein. The hardness of solving subset sum with preprocessing. *IEEE Transactions on Information Theory*, 36(4):943–946, July 1990.

[29] J. E. Mazo and A. M. Odlyzko. Lattice points in high dimensional spheres. *Monatshefte fuer Mathematik*, 110:47–61, 1990.

[30] A. McLoughlin. The complexity of computing the covering radius of a code. *IEEE Transactions on Information Theory*, 30:800–804, Nov. 1984.

[31] D. Micciancio. The hardness of the closest vector problem with preprocessing. *IEEE Transactions on Information Theory*, 47(3):1212–1215, Mar. 2001.

[32] D. Micciancio. Improving lattice based cryptosystems using the Hermite normal form. In J. Silverman, editor, *Cryptography and Lattices Conference - CaLC '01*, volume 2146 of *Lecture Notes in Computer Science*, pages 126–145, Providence, RI, USA, Mar. 2001. Springer.

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

[34] D. Micciancio. Generalized compact knapsaks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In *Proceedings of the 43rd Annual Symposium on Foundations of Computer Science - FOCS '02*, pages 356–365, Vancouver, BC, Canada, Nov. 2002. IEEE.

[35] D. Micciancio. Improved cryptographic hash functions with worst-case/average-case connection. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing - STOC '02*, pages 609–618, Montréal, Qué., Canada, May 2002. ACM.

[36] 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, Mar. 2002.

[37] J. Milnor and D. Husemoller. *Symmetric Bilinear Forms*. Springer-Verlag, 1973.

[38] O. Regev. Improved inapproximability of lattice and coding problems with preprocessing. In *Proceedings of the 18th Annual IEEE Conference on Computational Complexity - CCC '03*, pages 315 –322, Århus, Denmark, July 2003. IEEE.

[39] O. Regev. New lattice based cryptographic constructions. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing - STOC '03*, pages 407–426, San Diego, CA, USA, June 2003. ACM.

[40] C. A. Rogers. A note on coverings and packings. *Journal of the London Mathematical Society*, 25:327–331, 1950.

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

[42] A. Vardy. Algorithmic complexity in coding theory and the minimum distance problem. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing - STOC '97*, pages 92–109, El Paso, TX, USA, May 1997. ACM.