Algorithmic Number Theory – Proceedings of ANTS-VI
(June 13–18, 2004, Burlington, U.S.A.)
D. Buell (Ed.), vol. ???? of Lecture Notes in Computer Science, pages ???–???

Low-Dimensional Lattice Basis Reduction Revisited (Extended Abstract)

Phong Q. Nguyen[1] and Damien Stehlé[2]

[1] CNRS/École normale supérieure, Département d'informatique,
45 rue d'Ulm, 75230 Paris Cedex 05, France.
pnguyen@di.ens.fr and http://www.di.ens.fr/~pnguyen/
[2] LORIA/INRIA Lorraine, 615 rue du J. botanique, 54602 Villers-lès-Nancy, France.
stehle@loria.fr and http://www.loria.fr/~stehle/

Abstract. Most of the interesting algorithmic problems in the geometry of numbers are NP-hard as the lattice dimension increases. This article deals with the low-dimensional case. We study a greedy lattice basis reduction algorithm for the Euclidean norm, which is arguably the most natural lattice basis reduction algorithm, because it is a straightforward generalization of the well-known two-dimensional Gaussian algorithm. Our results are two-fold. From a mathematical point of view, we show that up to dimension four, the output of the greedy algorithm is optimal: the output basis reaches all the successive minima of the lattice. However, as soon as the lattice dimension is strictly higher than four, the output basis may not even reach the first minimum. More importantly, from a computational point of view, we show that up to dimension four, the bit-complexity of the greedy algorithm is quadratic without fast integer arithmetic: this allows to compute various lattice problems (*e.g.* computing a Minkowski-reduced basis and a closest vector) in quadratic time, without fast integer arithmetic, up to dimension four, while all other algorithms known for such problems have a bit-complexity which is at least cubic. This was already proved by Semaev up to dimension three using rather technical means, but it was previously unknown whether or not the algorithm was still polynomial in dimension four. Our analysis, based on geometric properties of low-dimensional lattices and in particular Voronoï cells, arguably simplifies Semaev's analysis in dimensions two and three, unifies the cases of dimensions two, three and four, but breaks down in dimension five.

1 Introduction

A *lattice* is a discrete subgroup of $\mathbb{R}^n$. Any lattice L has a *lattice basis*, i.e. a set $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d\}$ of linearly independent vectors such that the lattice is the set of all integer linear combinations of the $\boldsymbol{b}_i$'s: $L[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d] = \left\{ \sum_{i=1}^{d} x_i \boldsymbol{b}_i : x_i \in \mathbb{Z} \right\}$. A lattice basis is usually not unique, but all the bases have the same number of elements, called the *dimension* or *rank* of the lattice. In dimension higher than

one, there are infinitely many bases, but some are more interesting than others: they are called *reduced*. Roughly speaking, a reduced basis is a basis made of reasonably short vectors which are almost orthogonal. Finding good reduced bases has proved invaluable in many fields of computer science and mathematics, particularly in cryptology (see for instance the survey [18]); and the computational complexity of lattice problems has attracted considerable attention in the past few years (see for instance the book [16]), following Ajtai's discovery [1] of a connection between the worst-case and average-case hardness of certain lattice problems.

There exist many different notions of reduction, such as those of Hermite [8], Minkowski [17], Korkine-Zolotarev (KZ) [9, 11], Venkov [19], Lenstra-Lenstra-Lovász [13], *etc.* Among these, the most intuitive one is perhaps Minkowski's reduction; and up to dimension four, it is arguably optimal compared to all other known reductions, because it reaches all the so-called successive minima. However, finding a Minkowski-reduced basis or a KZ-reduced basis is NP-hard under randomized reductions as the dimension increases, because such bases contain a shortest lattice vector, and the shortest vector problem is NP-hard under randomized reductions [2, 15]. In order to better understand lattice reduction, it is tempting to study the low-dimensional case. Improvements in low-dimensional lattice reduction may lead to significant running-time improvements in high-dimensional lattice reduction, as the best lattice reduction algorithms known in theory and in practice for high-dimensional lattices, namely Schnorr's blockwise KZ-reduction [20] and its heuristic variants [21, 22], are based on a repeated use of low-dimensional KZ-reduction.

The classical Gaussian algorithm [5] computes in quadratic time (without fast integer arithmetic [23]) a Minkowski-reduced basis of any two-dimensional lattice. This algorithm was extended to dimension three by Vallée [27] in 1986 and Semaev [24] in 2001: Semaev's algorithm is quadratic without fast integer arithmetic, whereas Vallée's algorithm has cubic complexity. More generally, Helfrich [7] showed in 1986 by means of the LLL algorithm [13] how to compute in cubic time a Minkowski-reduced basis of any lattice of fixed (arbitrary) dimension, but the hidden complexity constant grows very fast with the dimension.

In this paper, we generalize the Gaussian algorithm to arbitrary dimension. Although the obtained greedy algorithm is arguably the simplest lattice basis reduction algorithm known, its analysis becomes remarkably more and more complex as the dimension increases. Semaev [24] was the first to prove that the algorithm was still polynomial-time in dimension three, but the polynomial-time complexity remained open for higher dimension. We show that up to dimension four, the greedy algorithm computes a Minkowski-reduced basis in quadratic time without fast arithmetic. This implies that a shortest vector and a KZ-reduced basis can be computed in quadratic time up to dimension four. Independently of the running time improvement, we hope our analysis will help to design new lattice reduction algorithms. The main novelty of our approach compared to previous work is that we use geometrical properties of low-dimensional lattices. In dimension two, the method is very close to the argument given by

Semaev in [24], which is itself very different from previous analyses of the Gaussian algorithm [12, 28, 10]. In dimension three, Semaev's analysis [24] is based on a rather exhaustive analysis of all the possible behaviors of the algorithm, which involves quite a few computations and makes it difficult to extend to higher dimension. We replace the main technical arguments by geometrical considerations on two-dimensional lattices. This makes it possible to extend the analysis to dimension four, by carefully studying geometrical properties of three-dimensional lattices, although a few additional difficulties appear. However, it is still unknown whether or not the greedy algorithm remains polynomial-time beyond dimension four. Besides, we show that the output basis may not even reach the shortest vector beyond dimension four.

The paper is organized as follows. In Section 2, we recall useful facts about lattices. In Section 3, we recall Gauss' algorithm and describe its natural greedy generalization. In Section 4, we analyze Gauss' algorithm and extend the analysis to dimensions three and four, using geometrical results. We explain why our analysis breaks down in dimension five. In Section 5, we prove geometrical results on low-dimensional lattices which are useful to prove the so-called gap lemmata, an essential ingredient of the complexity analysis of Section 4.

Important remark: Due to lack of space, this extended abstract contains few proofs, and we only show that the algorithm is polynomial-time. The proof of the quadratic complexity will appear in the full version of the paper.
Notations: $\|.\|$ and $\langle .,. \rangle$ denote respectively the Euclidean norm and inner product of $\mathbb{R}^n$; variables in bold are vectors; whenever the notation $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ is used, we have $||\boldsymbol{b}_1|| \leq \ldots \leq ||\boldsymbol{b}_n||$ and in that case we say that the $\boldsymbol{b}_i$'s are *ordered*. Besides, the complexity model we use is the RAM model, and the computational cost is measured in elementary operations on bits. In any complexity statement, we assume that the underlying lattice L is integral ($L \subseteq \mathbb{Z}^n$). If $x \in \mathbb{R}$, then $\lfloor x \rceil$ denotes a nearest integer to x.

2 Preliminaries

We assume the reader is familiar with geometry of numbers (see [4, 6, 14, 25]).
Gram-Schmidt orthogonalization. Let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ be vectors. The Gram matrix $G(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ of $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ is the $d \times d$ matrix $(\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle)_{1 \leq i,j \leq d}$ formed by all the inner products. $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ are linearly independent if and only if the determinant of $G(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ is not zero. The volume $\mathrm{vol}(L)$ of a lattice L is the square root of the determinant of the Gram matrix of any basis of L. The orthogonality-defect of a basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ of L is defined as $(\prod_{i=1}^{d} \|\boldsymbol{b}_i\|)/\mathrm{vol}(L)$: it is always greater than 1, with equality if and only if the basis is orthogonal. Let $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ be linearly independent vectors. The *Gram-Schmidt orthogonalization* $(\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_d^*)$ is defined as follows: $\boldsymbol{b}_i^*$ is the component of $\boldsymbol{b}_i$ orthogonal to the subspace spanned by $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1}$.
Successive minima and Minkowski reduction. Let L be a d-dimensional lattice in $\mathbb{R}^n$. For $1 \leq i \leq d$, the *i-th minimum* $\lambda_i(L)$ is the radius of the smallest

closed ball centered at the origin containing at least i linearly independent lattice vectors. The most famous lattice problem is the *shortest vector problem* (SVP): given a basis of a lattice L, find a lattice vector of norm $\lambda_1(L)$. There always exist linearly independent lattice vectors $\boldsymbol{v}_i$'s such that $\|\boldsymbol{v}_i\| = \lambda_i(L)$ for all i. Surprisingly, as soon as $d \geq 4$, such vectors do not necessarily form a lattice basis, and when $d \geq 5$, there may not even exist a lattice basis reaching all the minima. A basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ of L is *Minkowski-reduced* if for all $1 \leq i \leq d$, $\boldsymbol{b}_i$ has minimal norm among all lattice vectors $\boldsymbol{b}_i$ such that $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_i]$ can be extended to a basis of L. Surprisingly, up to dimension six, one can easily decide if a given basis is Minkowski-reduced or not, by checking a small number of explicit norm inequalities, known as Minkowski's conditions. A basis reaching all the minima must be Minkowski-reduced, but a Minkowski-reduced basis may not reach all the minima, except the first four ones (see [29]): if $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ is a Minkowski-reduced basis of L, then for all $1 \leq i \leq \min(d, 4)$, $\|\boldsymbol{b}_i\| = \lambda_i(L)$. Thus, a Minkowski-reduced basis is optimal in a natural sense up to dimension four. A classical result (see [29]) states that the orthogonality-defect of a Minkowski-reduced basis can be upper-bounded by a constant which only depends on the lattice dimension.

Voronoï cell and Voronoï vectors. The *Voronoï cell* [30] of $L = L[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$, denoted by $\mathrm{Vor}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$, is the set of vectors $\boldsymbol{x}$ in the linear span of L which are closer to 0 than to any other lattice vector: for all $\boldsymbol{v} \in L$, $\|\boldsymbol{x} - \boldsymbol{v}\| \geq \|\boldsymbol{x}\|$, that is $\|\boldsymbol{v}\|^2 \geq 2|\langle \boldsymbol{v}, \boldsymbol{x} \rangle|$. The Voronoï cell is a finite polytope which tiles the linear span of L by translations by lattice vectors. We extend the notation $\mathrm{Vor}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ to the case where the first vectors may be zero (the remaining vectors being linearly independent): $\mathrm{Vor}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ denotes the Voronoï cell of the lattice spanned by the non-zero $\boldsymbol{b}_i$'s. A lattice vector $\boldsymbol{v} \in L$ is called a *Voronoï vector* if $\boldsymbol{v}/2$ belongs to the Voronoï cell (in which case $\boldsymbol{v}/2$ will be on the boundary of the Voronoï cell). $\boldsymbol{v} \in L$ is a *strict Voronoï vector* if $\boldsymbol{v}/2$ is contained in the interior of a $(d-1)$-dimensional face of the Voronoï cell. A classical result states that Voronoï vectors correspond to the minima of the cosets of $L/2L$. We say that $(x_1, \ldots, x_d) \in \mathbb{Z}^d$ is a *possible Voronoï coord* if there exists a Minkowski-reduced basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ such that $x_1\boldsymbol{b}_1 + \ldots + x_d\boldsymbol{b}_d$ is a Voronoï vector. In some parts of the article, we will deal with Voronoï coordinates with respect to other types of reduced bases: the kind of reduction considered will be clear from the context. The *covering radius* $\rho(L)$ of a lattice L is half of the diameter of the Voronoï cell. The *closest vector problem* (CVP) is a non-homogeneous version of SVP: given a basis of a lattice and an arbitrary vector $\boldsymbol{x}$ of $\mathbb{R}^n$, find a lattice vector $\boldsymbol{v}$ minimizing the distance $\|\boldsymbol{v} - \boldsymbol{x}\|$; in other words, if $\boldsymbol{y}$ denotes the orthogonal projection of $\boldsymbol{x}$ over the linear span of L, find $\boldsymbol{v} \in L$ such that $\boldsymbol{v} - \boldsymbol{y}$ belongs to the Voronoï cell of L.

3 A Greedy Generalization of Gauss' Algorithm

In dimension two, there is a simple and efficient lattice basis reduction algorithm due to Gauss. We view Gauss' algorithm as a greedy algorithm based on

the one-dimensional CVP, which suggests a natural generalization to arbitrary dimension that we call the greedy reduction algorithm. We study properties of the bases output by the greedy algorithm by defining a new type of reduction and comparing it to Minkowski reduction.

3.1 Gauss' Algorithm

Gauss' algorithm – described in Figure 1 – can be seen as a two-dimensional

Input: A basis $[\boldsymbol{u}, \boldsymbol{v}]$ with its Gram matrix.
Output: A reduced basis of $L[\boldsymbol{u}, \boldsymbol{v}]$, together with its Gram matrix.
1. Repeat
2. $\quad \boldsymbol{r} := \boldsymbol{v} - x\boldsymbol{u}$ where $x := \left\lfloor \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\|^2} \right\rceil$,
3. $\quad \boldsymbol{v} := \boldsymbol{u}$,
4. $\quad \boldsymbol{u} := \boldsymbol{r}$,
5. $\quad$ Update the Gram matrix of $(\boldsymbol{u}, \boldsymbol{v})$,
6. Until $\|\boldsymbol{u}\| \geq \|\boldsymbol{v}\|$.
7. Return $[\boldsymbol{v}, \boldsymbol{u}]$ and its Gram matrix.

Fig. 1. Gauss' algorithm.

generalization of the centered Euclidean algorithm [28]. At Step 2 of each loop iteration, $\boldsymbol{u}$ is shorter than $\boldsymbol{v}$, and one would like to shorten $\boldsymbol{v}$ rather than $\boldsymbol{u}$, while preserving the fact that $[\boldsymbol{u}, \boldsymbol{v}]$ is a basis of L. This can be achieved by subtracting to $\boldsymbol{v}$ a multiple $x\boldsymbol{u}$ of $\boldsymbol{u}$, because such a transformation is unimodular. The optimal choice is when $x\boldsymbol{u}$ is the closest vector to $\boldsymbol{v}$, in the one-dimensional lattice spanned by $\boldsymbol{u}$, which gives rise to $x := \left\lfloor \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\|^2} \right\rceil$. The values $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$ and $\|\boldsymbol{u}\|^2$ are extracted from $G(\boldsymbol{u}, \boldsymbol{v})$, which is updated at Step 5 of each loop iteration. The complexity of Gauss' algorithm is given by the following classical result:

Theorem 1. *Given as input a basis $[\boldsymbol{u}, \boldsymbol{v}]$ of a lattice L, Gauss' algorithm outputs a Minkowski-reduced basis of L in time $O(\log \|\boldsymbol{v}\| \cdot [1 + \log \|\boldsymbol{v}\| - \log \lambda_1(L)])$.*

Note that this result is not trivial to prove. It is not even clear *a priori* why Gauss' algorithm outputs a Minkowski-reduced basis.

3.2 The Greedy Reduction Algorithm

Gauss' algorithm suggests the general greedy algorithm described in Figure 2, which uses reduction and closest vectors in dimension $d - 1$, to reduce bases in dimension d. We make a few remarks on the description of the algorithm. If the Gram matrix is not given, we may compute it in time $O(\log^2 ||\boldsymbol{b}_d||)$ for fixed d. The algorithm updates the Gram matrix each time the basis changes.

Name: Greedy($\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$).
Input: A basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ together with its Gram matrix.
Output: An ordered greedy-reduced basis of $L[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ with its Gram matrix.
1. If $d = 1$, return $\boldsymbol{b}_1$.
2. Repeat
3. Order $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ by increasing lengths and update the Gram matrix,
4. $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}] :=$ Greedy$(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$,
5. Compute a vector $\boldsymbol{c}$ closest to $\boldsymbol{b}_d$, in $L[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$,
6. $\boldsymbol{b}_d := \boldsymbol{b}_d - \boldsymbol{c}$ and update the Gram matrix,
7. Until $||\boldsymbol{b}_d|| \geq ||\boldsymbol{b}_{d-1}||$.
8. Return $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ and its Gram matrix.

Fig. 2. The greedy lattice basis reduction algorithm in dimension d.

Step 3 is easy: if this is the first iteration of the loop, the basis is already ordered; otherwise, $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$ is already ordered, and only $\boldsymbol{b}_d$ has to be inserted among $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}$. At Step 4, the greedy algorithm calls itself recursively in dimension $d-1$: $G(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$ does not need to be computed before calling the algorithm, since $G(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ is already known. At this point, we do not explain how Step 5 (the computation of closest vectors) is performed: this issue is postponed to subsection 3.4. Note that for $d = 2$, the greedy algorithm is exactly Gauss' algorithm. From a geometrical point of view, the goal of Steps 5 and 6 is to make sure that the orthogonal projection of $\boldsymbol{b}_d$ over the lattice spanned by $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$ lies in the Voronoï cell of that lattice.

An easy proof by induction on d shows that the algorithm terminates. Indeed, the new vector $\boldsymbol{b}_d$ of Step 6 is strictly shorter than $\boldsymbol{b}_{d-1}$ if the loop does not end at Step 7. Thus the product of the norms of the $\boldsymbol{b}_i$'s decreases strictly at each iteration of the loop which is not the last one. But for all B, the number of lattice vectors of norm less than B is finite, which completes the proof.

Although the description of the greedy algorithm is fairly simple, analyzing its bit complexity seems very difficult. Even the two-dimensional case of the Gaussian algorithm is not trivial.

3.3 Greedy Reduction

In this subsection, we study properties of the bases output by the greedy algorithm. As previously mentioned, it is not clear why Gauss' algorithm outputs a Minkowski-reduced basis. But it is obvious that the output basis $[\boldsymbol{u}, \boldsymbol{v}]$ satisfies: $\|\boldsymbol{u}\| \leq \|\boldsymbol{v}\| \leq \|\boldsymbol{v} - x\boldsymbol{u}\|$ for all $x \in \mathbb{Z}$. This suggests the following definition:

Definition 2. *An ordered basis* $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ *is* greedy-reduced *if for all* $2 \leq i \leq d$ *and for all* $x_1, \ldots, x_{i-1} \in \mathbb{Z}$: $\|\boldsymbol{b}_i\| \leq \|\boldsymbol{b}_i + x_1\boldsymbol{b}_1 + \ldots + x_{i-1}\boldsymbol{b}_{i-1}\|$.

In other words, we have the following recursive definition: a one-dimensional basis is always greedy-reduced, and an ordered basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ is greedy-reduced if

and only if $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$ is greedy-reduced and the projection of $\boldsymbol{b}_d$ over the linear span of $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}$ lies in the Voronoï cell $\mathrm{Vor}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$. The greedy algorithm outputs a greedy-reduced basis, and if the input basis is greedy-reduced, the output basis will be equal to the input basis.

The fact that Gauss' algorithm outputs Minkowski-reduced bases is a particular case of the following result, which compares greedy-reduction with Minkowski reduction:

Lemma 3. *The following statements hold:*
1. Any Minkowski-reduced basis is greedy-reduced.
2. A basis of $d \leq 4$ vectors is Minkowski-reduced if and only if it is greedy-reduced.
3. If $d \geq 5$, there exists a basis of d vectors which is greedy-reduced, but not Minkowski-reduced.

As a consequence, the greedy algorithm outputs a Minkowski-reduced basis up to dimension four, thus reaching all the successive minima of the lattice; but beyond dimension four, the greedy algorithm outputs a greedy-reduced basis which may not be Minkowski-reduced. The following lemma shows that greedy-reduced bases may considerably differ from Minkowski-reduced bases beyond dimension four:

Lemma 4. *Let $d \geq 5$. For all $\varepsilon > 0$, there exists a lattice L and a greedy-reduced basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ of L such that: $\lambda_1(L)/\|\boldsymbol{b}_1\| \leq \varepsilon$ and $\mathrm{vol}(L)/\prod_{i=1}^{d} \|\boldsymbol{b}_i\| \leq \varepsilon$.*

Such properties do not hold for Minkowski-reduced bases. The first phenomenon shows that greedy-reduced bases may be arbitrarily far from the first minimum, while the second one shows that a greedy-reduced basis may be far from being orthogonal.

3.4 Computing Closest Vectors From Minkowski-Reduced Bases

We now explain how Step 5 of the greedy algorithm can be implemented efficiently up to $d = 5$. Step 5 is trivial only when $d \leq 2$. Otherwise, note that after Step 4, the $(d-1)$-dimensional basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$ is greedy-reduced, and therefore Minkowski-reduced as long as $d \leq 5$. And we know the Gram matrix of $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}, \boldsymbol{b}_d]$.

Theorem 5. *Let $d \geq 1$ be an integer. There exists an algorithm which, given as input a Minkowski-reduced basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$, a target vector $\boldsymbol{t}$ longer than all the $\boldsymbol{b}_i$'s, and the Gram matrix of $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}, \boldsymbol{t}]$, outputs a closest lattice vector $\boldsymbol{c}$ to $\boldsymbol{t}$ (in the lattice spanned by the $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$), and the Gram matrix of $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}, \boldsymbol{t}-\boldsymbol{c})$, in time $O(\log ||\boldsymbol{t}|| \cdot [1 + \log ||\boldsymbol{t}|| - \log ||\boldsymbol{b}_\alpha||])$, where $1 \leq \alpha \leq d$ is any integer such that $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{\alpha-1}, \boldsymbol{t}]$ is Minkowski-reduced.*

Intuitively, the algorithm works as follows: an approximation of the coordinates (with respect to the $\boldsymbol{b}_i$'s) of the closest vector is computed using linear algebra, and the approximation is then corrected by a suitable exhaustive search.

Let $\boldsymbol{h}$ be the orthogonal projection of $\boldsymbol{t}$ over the linear span of $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}$. There exist $y_1, \ldots, y_{d-1} \in \mathbb{R}$ such that $\boldsymbol{h} = \sum_{i=1}^{d-1} y_i \boldsymbol{b}_i$. If $c = \sum_{i=1}^{d-1} x_i \boldsymbol{b}_i$ is a closest vector to $\boldsymbol{t}$, then $\boldsymbol{h} - \boldsymbol{c}$ belongs to $\mathrm{Vor}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$. However, for any $C > 0$, the coordinates (with respect to any basis of orthogonality-defect $\leq C$) of any point inside the Voronoï cell can be bounded independently from the lattice (see [26]). It follows that if we know an approximation of the y_i's with sufficient precision, then $\boldsymbol{c}$ can be derived from a $O(1)$ exhaustive search, since the coordinates $y_i - x_i$ of $\boldsymbol{h} - \boldsymbol{c}$ are bounded, and so is the orthogonality-defect of a Minkowski-reduced basis.

To approximate the y_i's, we use linear algebra. Let $G = G(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$ and $H = \left(\frac{\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle}{||\boldsymbol{b}_i||^2}\right)_{1 \leq i,j \leq d-1}$. We have:

$$G \cdot \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = - \begin{bmatrix} \langle \boldsymbol{b}_1, \boldsymbol{t} \rangle \\ \vdots \\ \langle \boldsymbol{b}_n, \boldsymbol{t} \rangle \end{bmatrix} \text{ therefore } \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = -H^{-1} \cdot \begin{bmatrix} \frac{\langle \boldsymbol{b}_1, \boldsymbol{t} \rangle}{||\boldsymbol{b}_1||^2} \\ \vdots \\ \frac{\langle \boldsymbol{b}_n, \boldsymbol{t} \rangle}{||\boldsymbol{b}_n||^2} \end{bmatrix}.$$

We use the latter formula to compute the y_i's with a one-bit accuracy, in the expected time. Let $r = \max_i \lceil \log \frac{\langle \boldsymbol{b}_i, \boldsymbol{t} \rangle}{||\boldsymbol{b}_i||^2} \rceil$. Notice that $r = O(1 + \log ||\boldsymbol{t}|| - \log ||\boldsymbol{b}_{\alpha-1}||)$ by bounding $\langle \boldsymbol{b}_i, \boldsymbol{t} \rangle$ depending on whether $i \geq \alpha$. Notice also that the entries of H are all ≤ 1 in absolute value (because $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$ is Minkowski-reduced), and $\det(H)$ is lower bounded by some universal constant (because the orthogonality-defect of $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1}]$ is bounded). It follows that one can compute the entries of H^{-1} with a $\Omega(r)$-bit accuracy, in $O(r^2)$ binary operations. One eventually derives the y_i's with a one-bit accuracy,

4 Complexity Analysis of the Greedy Algorithm

4.1 A Geometric Analysis of Gauss' Algorithm

We provide yet another proof of the classical result that Gauss' algorithm has quadratic complexity. Compared to other proofs, our method closely resembles the recent one of Semaev [24], itself relatively different from [12, 28, 10, 3]. The analysis is not optimal (as opposed to [28]), but its basic strategy can be extended up to dimension four. Consider the value of x at Step 2:

- If $x = 0$, this must be the last iteration of the loop.
- If $|x| = 1$, there are two cases:
  - If $\|\boldsymbol{v} - x\boldsymbol{u}\| \geq \|\boldsymbol{u}\|$, then this is the last loop iteration.
  - Otherwise, the inequality can be rewritten as $\|\boldsymbol{u} - x\boldsymbol{v}\| < \|\boldsymbol{u}\|$, which means that $\boldsymbol{u}$ can be shortened with the help of $\boldsymbol{v}$, which can only happen if this is the first loop iteration, because of the greedy strategy.
- Otherwise, $|x| \geq 2$, which implies that $x\boldsymbol{u}$ is not a Voronoï vector of the lattice spanned by $\boldsymbol{u}$. Intuitively, this means that $x\boldsymbol{u}$ is far away from $\mathrm{Vor}(\boldsymbol{u})$, so that $\boldsymbol{v} - x\boldsymbol{u}$ is considerably shorter than $\boldsymbol{v}$. More precisely, one can show that $\|\boldsymbol{v}\|^2 \geq \|\boldsymbol{v} - x\boldsymbol{u}\|^2 + 2\|\boldsymbol{u}\|^2$, which is therefore $> 3\|\boldsymbol{v} - x\boldsymbol{u}\|^2$ if this is not the last loop iteration.

This shows that the product of the basis vectors norms decreases by a factor at least $\sqrt{3}$ every loop iteration except possibly the first and last ones. Thus, the number τ of loop iterations is bounded by: $\tau \leq 2 + \log_{\sqrt{3}} \|\boldsymbol{v}\| - \log_{\sqrt{3}} \lambda_1(L)$.

It remains to estimate the cost of each Step 2, which is the cost of computing x. Because $\frac{|\langle \boldsymbol{u}, \boldsymbol{v} \rangle|}{\|\boldsymbol{u}\|^2} \leq \frac{\|\boldsymbol{v}\|}{\|\boldsymbol{u}\|}$, one can see that the bit complexity of Step 2 is $O(\log \|\boldsymbol{v}\| \cdot [1 + \log \|\boldsymbol{v}\| - \log \|\boldsymbol{u}\|])$. If we denote by $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ the values of $\boldsymbol{u}$ and $\boldsymbol{v}$ at the i-th iteration, then $\boldsymbol{v}_{i+1} = \boldsymbol{u}_i$ and we obtain that the bit complexity of Gauss' algorithm is bounded by:

$$\begin{aligned} O\left(\textstyle\sum_{i=1}^{\tau} \log \|\boldsymbol{v}_i\| \cdot [1 + \log \|\boldsymbol{v}_i\| - \log \|\boldsymbol{u}_i\|]\right) \\ &= O\left(\log \|\boldsymbol{v}\| \cdot \textstyle\sum_{i=1}^{\tau} [1 + \log \|\boldsymbol{v}_i\| - \log \|\boldsymbol{v}_{i+1}\|]\right) \\ &= O\left(\log \|\boldsymbol{v}\| \cdot [\tau + \log \|\boldsymbol{v}\| - \log \lambda_1(L)]\right). \end{aligned}$$

This completes the proof of Theorem 1.

4.2 A Geometric Analysis Up To Dimension Four

The main result of the paper is the following:

Theorem 6. *Let $1 \leq d \leq 4$. Given as input an ordered basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$, the greedy algorithm of Figure 2 based on the algorithm of Theorem 5 outputs a Minkowski-reduced basis of $L[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$, using a number of bit operations bounded by $O(\log \|\boldsymbol{b}_d\| \cdot [1 + \log \|\boldsymbol{b}_d\| - \log \lambda_1(L)])$.*

However, due to lack of space, we only prove the following weaker result in this extended abstract:

Theorem 7. *Let $1 \leq d \leq 4$. Given as input an ordered basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$, the greedy algorithm of Figure 2 based on the algorithm of Theorem 5 outputs a Minkowski-reduced basis of $L[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$, using a number of bit operations bounded by a polynomial in $\log \|\boldsymbol{b}_d\|$.*

The result asserts the polynomial-time complexity of the greedy algorithm, which is by far the hardest part of Theorem 6. Both theorems are proved iteratively: the case $d = 4$ is based on the case $d = 3$, which is itself based on the case $d = 2$. The analysis of Gauss' algorithm (Section 4.1) was based on the fact that if $|x| \geq 2$, $x\boldsymbol{u}$ is far away from the Voronoï cell of the lattice spanned by $\boldsymbol{u}$. The proof of Theorem 7 relies on a similar phenomenon in dimensions two and three. However, the situation is considerably more complex, as the following basic remarks hint:

- For $d = 2$, we considered the value of x, but if $d \geq 3$, there will be several coefficients instead of a single x, and it is not clear which coefficient will be useful in the analysis.
- For $d = 2$, Step 4 cannot change the basis, as there are only two bases in dimension one. If $d \geq 3$, Step 4 may completely change the vectors, and it could be hard to keep track of what is going on.

In order to prove Theorem 7, we introduce a few notations. Consider the i-th loop iteration. Let $[\boldsymbol{a}_1^i, \ldots, \boldsymbol{a}_d^i]$ denote the basis $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ at the beginning

of the i-th loop iteration. The basis $[\boldsymbol{a}_1^i, \ldots, \boldsymbol{a}_d^i]$ becomes $[\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_{d-1}^i, \boldsymbol{a}_d^i]$ with $||\boldsymbol{b}_1^i|| \leq \ldots \leq ||\boldsymbol{b}_{d-1}^i||$ after Step 4, and $(\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_d^i)$ after Step 6, where $\boldsymbol{b}_d^i = \boldsymbol{a}_d^i - \boldsymbol{c}^i$ and $\boldsymbol{c}^i$ is the closest vector $\boldsymbol{c}$ found at Step 5. Let p_i be the number of integers $1 \leq j \leq d$ such that $\|\boldsymbol{b}_j^i\| \leq \|\boldsymbol{b}_d^i\|$. Let π_i be the rank of $\boldsymbol{b}_d^i$ once $(\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_d^i)$ is sorted by length: for example, $\pi_i = 1$ if $\|\boldsymbol{b}_d^i\| < \|\boldsymbol{b}_1^i\|$. Clearly, $1 \leq \pi_i \leq p_i \leq d$, if $p_i = d$ then the loop terminates, and otherwise $||\boldsymbol{a}_{\pi_i}^{i+1}|| = ||\boldsymbol{a}_{p_i}^{i+1}||$. Note that π_i may not be equal to p_i because there may be several choices when sorting the vectors by length in case of equalities.

Now consider the $(i+1)$-th loop iteration for some $i \geq 1$. Recall that by definition of π_i, we have $\boldsymbol{a}_{\pi_i}^{i+1} = \boldsymbol{b}_d^i = \boldsymbol{a}_d^i - \boldsymbol{c}^i$, while $[\boldsymbol{a}_j^{i+1}]_{j \neq \pi_i} = [\boldsymbol{b}_j^i]_{1 \leq j \leq d-1}$. The closest vector $\boldsymbol{c}^{i+1}$ belongs to $L[\boldsymbol{b}_1^{i+1}, \ldots, \boldsymbol{b}_{d-1}^{i+1}] = L[\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1}]$: there exist integers $x_1^{i+1}, \ldots, x_{d-1}^{i+1}$ such that $\boldsymbol{c}^{i+1} = \sum_{j=1}^{d-1} x_j^{i+1} \boldsymbol{a}_j^{i+1}$.

Suppose we know that Theorem 7 is correct in dimension $d-1$ with $2 \leq d \leq 4$; we are to prove that it is still valid in dimension d. Because of this induction hypothesis and of Theorem 5, the number of bit operations performed in the i-th loop iteration is bounded by a polynomial in $\log ||\boldsymbol{a}_d^i||$. Since $||\boldsymbol{a}_d^i|| \leq ||\boldsymbol{a}_d^1||$ for any i, it is sufficient to prove that the number of loop iterations is bounded by a polynomial in $\log ||\boldsymbol{a}_d^1||$. Indeed, we show that there exist a universal constant $C_d > 1$ such that for any execution of the d-dimensional greedy algorithm, in any d consecutive iterations of the loop, the product of the lengths of the current vectors decreases by some factor higher than C_d:

$$\frac{||\boldsymbol{a}_1^i|| \ldots ||\boldsymbol{a}_d^i||}{||\boldsymbol{a}_1^{i+d}|| \ldots ||\boldsymbol{a}_d^{i+d}||} \geq C_d. \tag{1}$$

This automatically ensures that the number of loop iterations is at most proportional to $\log ||\boldsymbol{a}_d^1||$, and that the total number of bit operations is bounded by a polynomial in $\log ||\boldsymbol{a}_d^1||$.

We deal with the first difficulty mentioned: which coefficient will be useful? The trick is to consider the value of $x_{\pi_i}^{i+1}$, that is, the coefficient of $\boldsymbol{a}_{\pi_i}^{i+1} = \boldsymbol{a}_d^i - \boldsymbol{c}^i$ in $\boldsymbol{c}^{i+1}$, and to use the greedy properties of the algorithm.

Lemma 8. *Among d consecutive iterations of the loop of the greedy algorithm of Figure 2, there is at least one iteration of index $i+1$ such that $p_{i+1} \leq p_i$. Moreover, for such a loop iteration, we have $|x_{\pi_i}^{i+1}| \geq 2$.*

Proof. The first statement is obvious. Consider one such loop iteration $i+1$. Suppose we have a small $|x_{\pi_i}^{i+1}|$, that is $|x_{\pi_i}^{i+1}| = 0$ or $|x_{\pi_i}^{i+1}| = 1$.

- If $x_{\pi_i}^{i+1} = 0$, $\boldsymbol{c}^{i+1} \in L[\boldsymbol{a}_j^{i+1}]_{j \neq \pi_i, j \leq d-1} = L[\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_{d-2}^i]$. We claim that the $(i+1)$-th iteration must be the last one. Because the i-th loop iteration was not terminal, we have $\boldsymbol{a}_d^{i+1} = \boldsymbol{b}_{d-1}^i$. Moreover, $[\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_{d-1}^i]$ is greedy-reduced because of Step 4 of the i-th loop iteration. These two facts imply that $\boldsymbol{c}^{i+1}$ must be zero, and the $(i+1)$-th loop iteration is the last one.
- If $|x_{\pi_i}^{i+1}| = 1$, we claim that $p_{i+1} > p_i$. We have $\boldsymbol{c}^{i+1} = \sum_{j=1}^{d-1} x_j^{i+1} \boldsymbol{a}_j^{i+1}$ where $\boldsymbol{a}_{\pi_i}^{i+1} = \boldsymbol{a}_d^i - \boldsymbol{c}^i$ and $[\boldsymbol{a}_j^{i+1}]_{j \neq \pi_i} = [\boldsymbol{b}_j^i]_{1 \leq j \leq d-1}$. Thus, $\boldsymbol{c}^{i+1}$ can be

written as $\boldsymbol{c}^{i+1} = \pm(\boldsymbol{a}_d^i - \boldsymbol{c}^i) + \boldsymbol{e}$ where $\boldsymbol{e} \in L[\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_{d-2}^i]$. Therefore $\boldsymbol{a}_d^{i+1} - \boldsymbol{c}^{i+1} = \boldsymbol{b}_{d-1}^i \pm (\boldsymbol{a}_d^i - \boldsymbol{c}^i) + \boldsymbol{e}$. In other words, $\|\boldsymbol{a}_d^{i+1} - \boldsymbol{c}^{i+1}\| = \|\boldsymbol{a}_d^i - \boldsymbol{f}\|$ for some $\boldsymbol{f} \in L[\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_{d-1}^i]$. It follows that $p_{i+1} \geq 1 + p_i$, which achieves the claim.

□

We will see that in dimension three, any such loop iteration $i+1$ implies that at least one of the basis vectors significantly decreases in the $(i+1)$-th loop iteration, or had significantly decreased in the i-th loop iteration. This is only "almost" true in dimension four: fortunately, we will be able to isolate the bad cases, and to show that when a bad case occurs, the number of remaining loop iterations can be bounded by some universal constant.

We now deal with the second difficulty. Recall that $\boldsymbol{c}^{i+1} = \sum_{j=1}^{d-1} x_j^{i+1} \boldsymbol{a}_j^{i+1}$ but the basis $[\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1}]$ is not necessarily greedy-reduced. We distinguish two cases:

- 1) The basis $[\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1}]$ is somehow far from being greedy-reduced. Then $\boldsymbol{b}_d^i$ was significantly shorter than $\boldsymbol{a}_d^i$. Note that this length decrease concerns the i-th loop iteration and not the $(i+1)$-th.
- 2) Otherwise, the basis $[\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1}]$ is almost greedy-reduced. The fact that $|x_{\pi_i}^{i+1}| \geq 2$ roughly implies that $\boldsymbol{c}^{i+1}$ is somewhat far away from the Voronoï cell $\mathrm{Vor}(\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1})$: this phenomenon will be precisely captured by the so-called Gap Lemma. When this is the case, the new vector $\boldsymbol{b}_d^{i+1}$ will be significantly shorter than $\boldsymbol{a}_d^{i+1}$.

To capture the property that a set of vectors is almost greedy-reduced, we introduce the so-called *ε-reduction* where $\varepsilon \geq 0$, which is defined as follows:

Definition 9. *A single vector $\boldsymbol{b}_1$ is always ε-reduced; for $d \geq 2$, a d-tuple $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ is ε-reduced if $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$ is ε-reduced, $\|\boldsymbol{b}_{d-1}\| \leq \|\boldsymbol{b}_d\|$, and the orthogonal projection of $\boldsymbol{b}_d$ over the linear span of $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$ belongs to $(1+\varepsilon)\,Vor(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d-1})$.*

With this definition, a greedy-reduced basis is ε-reduced for any $\varepsilon \geq 0$. In the definition of ε-reduction, we did not assume that the $\boldsymbol{b}_i$'s were nonzero nor linearly independent. This is because the Gap Lemma is essentially based on compactness properties: the set of ε-reduced d-tuples needs to be closed (from a topological point of view), while a limit of bases may not be a basis.

We can now give the precise statements of the two cases described above. Lemma 10 corresponds to case 1), and Lemma 11 to case 2).

Lemma 10. *Let $2 \leq d \leq 4$. There exists a constant $\varepsilon_1 > 0$ such that for any $\varepsilon \leq \varepsilon_1$ there exists $C_\varepsilon > 1$ such that the following statement holds. Consider the $(i+1)$-th loop iteration of an execution of the d-dimensional greedy algorithm. If $[\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1}]$ is not ε-reduced, then $||\boldsymbol{a}_d^i|| \geq C_\varepsilon ||\boldsymbol{b}_d^i||$.*

Lemma 11. *Let $2 \leq d \leq 4$. There exist two constants $\varepsilon_2 > 0$ and $D > 0$ such that the following statement holds. Consider the $(i+1)$-th loop iteration of an execution of the d-dimensional greedy algorithm. Suppose that $[\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1}]$ is ε_2-reduced, and that $||\boldsymbol{a}_k^{i+1}|| \geq (1-\varepsilon_2)||\boldsymbol{a}_d^{i+1}||$ for some $1 \leq k \leq d-1$. Then, if $|x_k| \geq 2$ and if we are not in the 211-case, we have:*

$$||\boldsymbol{b}_d^{i+1}||^2 + D||\boldsymbol{b}_k^{i+1}||^2 \leq ||\boldsymbol{a}_d^{i+1}||^2,$$

where the 211-case is: $d = 4$, $|x_k| = 2$ and the other $|x_j|$'s are all equal to 1.

This last lemma is a direct consequence of Pythagore and the Gap Lemma (which is crucial to our analysis, and to which the next section is devoted):

Theorem 12 (Gap Lemma). *Let $2 \leq d \leq 4$. There exist two constants $\varepsilon_2 > 0$ and $D > 0$ such that the following statement holds. Let $[\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{d-1}]$ be ε-reduced vectors, $\boldsymbol{u}$ be a vector of $Vor(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{d-1})$ and $x_1, \ldots, x_{d-1}$ be integers. If $||\boldsymbol{a}_k|| \geq (1-\varepsilon)||\boldsymbol{a}_{d-1}||$ for some $k \leq d-2$, then:*

$$||\boldsymbol{u}||^2 + D||\boldsymbol{b}_k||^2 \leq ||\boldsymbol{u} + \sum_{j=1}^{d-1} x_j \boldsymbol{b}_j||^2,$$

where $|x_k| \geq 2$, and if $d = 4$ the two other $|x_j|$'s are not all equal to 1.

This completes the overall description of the proof of Theorem 7. Indeed, choose three constants $\varepsilon, D > 0$ and $C > 1$ such that we can apply Lemmata 10 and 11. We prove that Equation (1) holds for $C_d = \min(C, \sqrt{1+D}, \frac{1}{1-\varepsilon}) > 1$. Consider a loop iteration $i+1$ such that $p_{i+1} \leq p_i$. Recall that among any d consecutive iterations of the loop, there is at least one such iteration. For such an iteration, we have $|x_{\pi_i}^{i+1}| \geq 2$. We distinguish four cases:

- $[\boldsymbol{a}_1^{i+1}, \ldots, \boldsymbol{a}_{d-1}^{i+1}]$ is not ε-reduced: then Lemma 10 gives the result through the i-th loop iteration.
- $||\boldsymbol{a}_{\pi_i}^{i+1}|| < (1-\varepsilon)||\boldsymbol{a}_d^{i+1}||$: because $p_{i+1} \leq p_i$, we have the inequalities $||\boldsymbol{b}_d^{i+1}|| < ||\boldsymbol{a}_{p_i}^{i+1}|| = ||\boldsymbol{a}_{\pi_i}^{i+1}|| < (1-\varepsilon)||\boldsymbol{a}_d^{i+1}||$.
- We are in the 211-case, i.e. $d = 4$ with $|x_{\pi_i}| = 2$ and the other $|x_j|$'s are all equal to 1, we refer to the detail analysis of subsection 4.3.
- Otherwise, we apply Lemma 11, which gives the expected result through the $(i+1)$-th loop iteration.

We described our strategy to prove that the greedy algorithm is polynomial-time up to dimension four. One can further prove that the bit complexity is in fact quadratic, by carefully assessing the costs of each loop iteration and combining them, but the proof is much more technical than in the two-dimensional case.

4.3 Concluding in Dimension Four

In the previous subsections, we showed that the greedy algorithm is polynomial-time in dimensions two and three, but we noticed that a new difficulty arose in dimension four: the Gap Lemma is useless in the so-called 211-case. This is because there are three-dimensional Minkowski-reduced bases $[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3]$ for which $2\boldsymbol{b}_i + s_1\boldsymbol{b}_j + s_2\boldsymbol{b}_k$ – with $\{i, j, k\} = \{1, 2, 3\}$ and $|s_1| = |s_2| = 1$ – is a Voronoï vector. Indeed consider the lattice spanned by the columns $\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3$ of the following matrix:

$$M = \begin{bmatrix} 1 & 1 & -1 \\ 1 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

This basis is Minkowski-reduced and $||\boldsymbol{b}_1 + \boldsymbol{b}_2 + 2\boldsymbol{b}_3|| = ||\boldsymbol{b}_1 + \boldsymbol{b}_2|| \leq ||(2k_1 + 1)\boldsymbol{b}_1 + (2k_2 + 1)\boldsymbol{b}_1 + 2k_3\boldsymbol{b}_3||$ for any $k_1, k_2, k_3 \in \mathbb{Z}$. Therefore, a vector in the Voronoï cell centered in $\boldsymbol{b}_1 + \boldsymbol{b}_2 + 2\boldsymbol{b}_3$ can avoid being significantly shortened when translated inside the Voronoï cell centered in 0.

The Gap Lemma cannot tackle this problem. However, we note that $(1, 1, 2)$ is rarely a Voronoï coordinate (with respect to a Minkowski-reduced basis), and when it is, it cannot be a strict Voronoï coord: we can prove that if $(1, 1, 2)$ is a Voronoï coord, then $||\boldsymbol{b}_1+\boldsymbol{b}_2|| = ||\boldsymbol{b}_1+\boldsymbol{b}_2+2\boldsymbol{b}_3||$, which tells us that $\boldsymbol{b}_1+\boldsymbol{b}_2+2\boldsymbol{b}_3$ is not the only vector in its coset of $L/2L$ reaching the minimum. It turns out that the lattice spanned by the columns of M is essentially the only one for which $(1, 1, 2)$ – modulo any change of sign and permutation of coordinates – can be a Voronoï coord. More precisely, if $(1, 1, 2)$ – modulo any change of sign and permutation of coordinates – is a Voronoï coord for a lattice basis, then the basis matrix can be written as rUM where r is any non-zero real number and U is any orthogonal matrix. Since a basis can be arbitrarily close to one of these without being one of them, we need to consider a small compact set of normalized bases around the annoying ones. More precisely, this subset is:

$$\{[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3] \text{ } \varepsilon\text{-reduced} \,/\, \exists \sigma \in \mathcal{S}_3, ||\frac{1}{||\boldsymbol{b}_3||^2}|G(\boldsymbol{b}_{\sigma(1)}, \boldsymbol{b}_{\sigma(2)}, \boldsymbol{b}_{\sigma(3)})| - |M^t M|\,||_\infty \leq \varepsilon\},$$

for some sufficiently small $\varepsilon > 0$, where $||M||_\infty$ is the maximum of the absolute values of the matrix M and $|M|$ is the matrix of the absolute values.

Now, consider we are in the 211-case at some loop iteration $i+1$. We distinguish three cases:

- $[\boldsymbol{a}_1^{i+1}, \boldsymbol{a}_2^{i+1}, \boldsymbol{a}_3^{i+1}]$ is outside the compact. In this case, a variant of the Gap Lemma (Lemma 29) proved in Section 5 is valid, and can be used to show that $\boldsymbol{b}_4^{i+1}$ is significantly shorter than $\boldsymbol{a}_4^{i+1}$.
- $[\boldsymbol{a}_1^{i+1}, \boldsymbol{a}_2^{i+1}, \boldsymbol{a}_3^{i+1}]$ is inside the compact, but $\boldsymbol{a}_4^{i+1}$ is far from the Voronoï cell $\mathrm{Vor}(\boldsymbol{a}_1^{i+1}, \boldsymbol{a}_2^{i+1}, \boldsymbol{a}_3^{i+1})$. In this case, $\boldsymbol{b}_4^{i+1}$ is significantly shorter than $\boldsymbol{a}_4^{i+1}$.
- Otherwise the overall geometry of $[\boldsymbol{a}_1^{i+1}, \boldsymbol{a}_2^{i+1}, \boldsymbol{a}_3^{i+1}, \boldsymbol{a}_4^{i+1}]$ is very precisely known, and we can show that there remain at most $O(1)$ loop iterations.

More precisely, by using Lemma 29, we show that:

Lemma 13. *There exist three constants $C, \varepsilon > 0$, and $c \in \mathbb{Z}$ such that the following holds. Consider an execution of the four-dimensional greedy algorithm, and a loop iteration $i+1$ for which $p_{i+1} \leq p_i$, $[\boldsymbol{a}_1^{i+1}, \boldsymbol{a}_2^{i+1}, \boldsymbol{a}_3^{i+1}]$ is ε-reduced, $||\boldsymbol{a}_{p_i}^{i+1}|| \geq (1-\varepsilon)||\boldsymbol{a}_4^{i+1}||$, and $(|x_{\sigma(1)}|, |x_{\sigma(2)}|, |x_{\sigma(3)}|) = (1,1,2)$ for some permutation σ of $\{1,2,3\}$. Then either $||\boldsymbol{a}_4^{i+1}|| \geq (1+C)||\boldsymbol{b}_4^{i+1}||$ or:*

$$||\frac{1}{||\boldsymbol{a}_4^{i+1}||^2}|G(\boldsymbol{a}_{\sigma(1)}^{i+1}, \boldsymbol{a}_{\sigma(2)}^{i+1}, \boldsymbol{a}_{\sigma(3)}^{i+1}, \boldsymbol{a}_4^{i+1})| - A||_\infty \leq \varepsilon, \textit{with} : A = \begin{bmatrix} 1 & 0 & \frac{1}{2} & 0 \\ 0 & 1 & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 1 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 \end{bmatrix}.$$

In order to prove this result, we restrict more and more the possible geometry of $[\boldsymbol{a}_1^i, \boldsymbol{a}_2^i, \boldsymbol{a}_3^i, \boldsymbol{a}_4^i]$. Note that this critical geometry corresponds to the root lattice D_4. We treat this last case by applying the following lemma, which roughly says that if the Gram matrix of a basis is sufficiently close to some invertible matrix, then the number of short vectors generated by the basis remains bounded.

Lemma 14. *Let A be an invertible $d \times d$ matrix, and $B > 0$. Then there exists $\varepsilon, N > 0$ such that for any basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$, if $||G(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) - A||_\infty \leq \varepsilon$, then:*

$$|\{(x_1, \ldots, x_d) \ / \ ||x_1\boldsymbol{b}_1 + \ldots + x_d\boldsymbol{b}_d|| \leq B\}| \ \leq \ N.$$

4.4 Failure in Dimension Five

In this subsection, we explain why the analysis of the greedy algorithm breaks down in dimension five. First of all, the basis returned by the algorithm is not necessarily Minkowski reduced, since greedy and Minkowski reductions differ in dimension five. Consider the lattice spanned by the columns of the following matrix:

$$\begin{bmatrix} 2 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & \varepsilon \end{bmatrix},$$

where $0 < \varepsilon < 1$. This basis is clearly greedy reduced, but the vector $(0,0,0,0,\varepsilon)^t$ belongs to the lattice. Moreover, for a small ε, this shows that a greedy reduced basis can be arbitrarily far from the first minimum, and can have an arbitrarily large orthogonality defect: $||\boldsymbol{b}_5^*||$ is very small towards $||\boldsymbol{b}_5||$. For ε close to 1, this basis shows that the length decrease factor through one loop iteration of the five-dimensional greedy algorithm can be arbitrarily close to 1.

Nevertheless, the greedy algorithm is well-defined in dimension five (the four-dimensional greedy algorithm can be used for Step 4, and since it returns a Minkowski reduced basis, the CVP algorithm of Theorem 5 can be used in Step 5). Despite the fact that the algorithm does not return a Minkowski reduced basis, one may wonder if the analysis remains valid, and if the number of loop iterations of the 5-dimensional greedy algorithm is linear in $\log ||\boldsymbol{b}_5||$.

The analysis in dimensions two, three and four essentially relies on the fact that if one of the x_j's found at Step 5 has absolute value higher than 2, then $||\boldsymbol{b}_d^i||$ is significantly shorter than $||\boldsymbol{a}_d^i||$. This fact is derived from the so-called Gap Lemma. In dimension four, this was only partly true, but the exception (the 211-case) happened in very few cases and could be dealt by considering the very specific shape of the lattices for which it could go wrong. Things worsen in dimension five. Indeed, for Minkowski-reduced bases, $(1,1,1,2)$ and $(1,1,2,2)$ – modulo any change of sign and permutation of coordinates – are possible Voronoï coords. Here is an example of a lattice where $(1,1,2,2)$ is a Voronoï coord:

$$\begin{bmatrix} 1 & -1 & 0 & 0 \\ 1 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The lattice basis given by the columns is Minkowski-reduced, but:

$$||\boldsymbol{b}_1+\boldsymbol{b}_2+2\boldsymbol{b}_3+2\boldsymbol{b}_4|| = 2 = ||\boldsymbol{b}_1+\boldsymbol{b}_2|| \leq ||(2k_1+1)\boldsymbol{b}_1+(2k_2+1)\boldsymbol{b}_2+2k_3\boldsymbol{b}_3+2k_4\boldsymbol{b}_4||,$$

for any $k_1, k_2, k_3, k_4 \in \mathbb{Z}$. Note that $(1,1,2,2)$ cannot be a strict Voronoï coord: if $\boldsymbol{b}_1+\boldsymbol{b}_2+2\boldsymbol{b}_3+2\boldsymbol{b}_4$ reaches the length minimum of its coset of $L/2L$, then so does $\boldsymbol{b}_1+\boldsymbol{b}_2$. Thus it might be possible to work around the difficulty coming from $(1,1,2,2)$ like in the previous subsection. However, the case $(1,1,1,2)$ would still remain, and this possible Voronoï coordinate can be strict.

5 The Geometry of Low-Dimensional Lattices

In this section, we give some results about Voronoï cells in dimensions two and three, which are crucial to our complexity analysis of the greedy algorithm described in Section 3. More precisely, the analysis is based on the Gap Lemma (subsection 5.3), which is derived from the study of Voronoï cells in the case of ε-reduced vectors (subsection 5.2), itself derived from the study of Voronoï cells for Minkowski-reduced bases (subsection 5.1).

5.1 Voronoï Cells in the Case of Minkowski-Reduced Bases

We give simple bounds on the diameter of the Voronoï cell and on the Gram-Schmidt orthogonalization of a Minkowski-reduced basis:

Lemma 15. *Let $d \geq 1$. Let $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ be a basis of a lattice L. Then $\rho(L) \leq \frac{\sqrt{d}}{2}||\boldsymbol{b}_d||$. As a consequence, if $d \leq 4$ and if $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ is a Minkowski-reduced basis, then $||\boldsymbol{b}_d^*|| \geq \frac{\sqrt{5-d}}{2}||\boldsymbol{b}_d||$.*

The following lemma provides the possible Voronoï vectors of a two-dimensional lattice given by a Minkowski-reduced basis. Such a basis confines the coordinates of Voronoï vectors:

Lemma 16. *In dimension two, the possible Voronoï coords are* $(1,0)$ *and* $(1,1)$*, modulo any change of signs and permutation of coordinates, i.e. any nonzero* $(\varepsilon_1, \varepsilon_2)$ *where* $|\varepsilon_1|, |\varepsilon_2| \leq 1$*.*

The proof relies on a detailed study of the expression $||(2x_1+\varepsilon_1)\cdot \boldsymbol{b}_1+(2x_2+\varepsilon_2)\cdot \boldsymbol{b}_2||^2 - ||\varepsilon_1\boldsymbol{b}_1 + \varepsilon_2\boldsymbol{b}_2||^2$, where $[\boldsymbol{b}_1, \boldsymbol{b}_2]$ is Minkowski-reduced, $\varepsilon_1, \varepsilon_2 \in \{0,1\}$ and $x_1, x_2 \in \mathbb{Z}$. Indeed, since Voronoï coords of a lattice L are given by the minima of the non-zero cosets of $L/2L$, it suffices to show that if $x_1x_2 \neq 0$, then this expression is strictly positive. We do this by a rather technical study.

We generalize this analysis to the three-dimensional case. The underlying ideas of the proof are the same, but because of the increasing number of variables, the analysis becomes more tedious.

Lemma 17. *In dimension three, the possible Voronoï coordinates are* $(1,0,0)$*,* $(1,1,0)$*,* $(1,1,1)$ *and* $(2,1,1)$*, modulo any change of signs and permutation of coordinates.*

The possible Voronoï coord $(2,1,1)$ creates difficulties when analyzing the greedy algorithm in dimension four, because it contains a two, which cannot be handled with the greedy argument used for the ones. We tackle this problem as follows: we show that when $(2,1,1)$ happens to be a Voronoï coord, the lattice has a very specific shape, for which the behavior of the algorithm is well-understood.

Lemma 18. *Suppose* $[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3]$ *is a Minkowski-reduced basis.*
1. If any of $(s_1, s_2, 2)$ *is a Voronoï coord with* $s_i = \pm 1$ *for* $i \in \{1,2\}$*, then* $||\boldsymbol{b}_1|| = ||\boldsymbol{b}_2|| = ||\boldsymbol{b}_3||$*,* $\langle \boldsymbol{b}_1, \boldsymbol{b}_2\rangle = 0$ *and* $\langle \boldsymbol{b}_i, \boldsymbol{b}_3\rangle = -s_i||\boldsymbol{b}_1||^2/2$ *for* $i = 1,2$*.*
2. If any of $(s_1, 2, s_3)$ *is a Voronoï coord with* $s_i = \pm 1$ *for* $i \in \{1,3\}$*, then* $||\boldsymbol{b}_1|| = ||\boldsymbol{b}_2||$*. Moreover, if* $||\boldsymbol{b}_1|| = ||\boldsymbol{b}_2|| = ||\boldsymbol{b}_3||$*, then* $\langle \boldsymbol{b}_1, \boldsymbol{b}_3\rangle = 0$ *and* $\langle \boldsymbol{b}_i, \boldsymbol{b}_2\rangle = -s_i||\boldsymbol{b}_1||^2/2$ *for* $i = 1,3$*.*
3. If any of $(2, s_2, s_3)$ *is a Voronoï coord with* $s_i = \pm 1$ *for* $i \in \{2,3\}$ *and* $||\boldsymbol{b}_1|| = ||\boldsymbol{b}_2|| = ||\boldsymbol{b}_3||$*, then* $\langle \boldsymbol{b}_2, \boldsymbol{b}_3\rangle = 0$ *and* $\langle \boldsymbol{b}_i, \boldsymbol{b}_1\rangle = -s_i||\boldsymbol{b}_1||^2/2$ *for* $i = 2,3$*.*

5.2 Voronoï Cells in the Case of ε-Reduced Vectors

We extend the results of the previous subsection to the case of ε-reduced vectors. The idea is that if we compact the set of Minkowski-reduced bases and slightly enlarge it, the possible Voronoï coords remain the same. Unfortunately, by doing so, some of the vectors we consider may be zero, and this creates an infinity of possible Voronoï coords: for example, if $\boldsymbol{b}_1 = 0$, any pair $(x_1, 0)$ is a Voronoï coord of $[\boldsymbol{b}_1, \boldsymbol{b}_2]$. To tackle this problem, we restrict to $\boldsymbol{b}_i$ with "similar" lengths. More precisely, we use the so-called Topological Lemma: if we can guarantee that the possible Voronoï coords of the enlargement of the initial compact set of bases are bounded, then for a sufficiently small enlargement, the possible Voronoï coords remain the same. We first give rather simple results on ε-reduced vectors and their Gram-Schmidt orthogonalization, then we introduce the Topological Lemma (Lemma 21), from which we finally derive the relaxed versions of Lemmata 16, 17 and 18.

Lemma 19. *There exists a constant $c > 0$ such that for any sufficiently small $\varepsilon > 0$, if $[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3]$ are ε-reduced, then the following inequalities hold:*

$$|\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle| \leq \frac{1+c\varepsilon}{2} ||\boldsymbol{b}_i||^2 \qquad \text{for any } i < j,$$
$$|\langle \boldsymbol{b}_3, \varepsilon_1 \boldsymbol{b}_1 + \varepsilon_2 \boldsymbol{b}_2 \rangle| \leq \frac{1+c\varepsilon}{2} ||\varepsilon_1 \boldsymbol{b}_1 + \varepsilon_2 \boldsymbol{b}_2||^2 \quad \text{for any } \varepsilon_1, \varepsilon_2 \in \{-1, 1\}.$$

This result implies that if $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ are ε-reduced, the only case for which the $\boldsymbol{b}_i$'s can be linearly dependent is when some of them are zero, but this case cannot be avoided since we need compacting the set of Minkowski-reduced bases. The following lemma generalizes Lemma 15. It shows that even with ε-reduced vectors, if the dimension is below four, then the Gram-Schmidt orthogonalization process cannot arbitrarily decrease the lengths of the initial vectors.

Lemma 20. *There exists $C > 0$ such that for any $1 \leq d \leq 4$ and any sufficiently small $\varepsilon > 0$, if $[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ are ε-reduced vectors, then we have $||\boldsymbol{b}_d^*|| \geq C ||\boldsymbol{b}_d||$.*

The Topological Lemma is the key argument when extending the results on possible Voronoï coords from Minkowski-reduced bases to ε-reduced vectors. When applying it, X_0 will correspond to the x_i's, K_0 to the $\boldsymbol{b}_i$'s, X to the possible Voronoï coordinates, and f to the continuous function of real variables $f : (y_i)_i, (\boldsymbol{b}_i)_i \longrightarrow ||y_1 \boldsymbol{b}_1 + \ldots + y_d \boldsymbol{b}_d||$.

Lemma 21 (Topological Lemma). *Let $n, m \geq 1$. Let X_0 and K_0 be compact sets of $\mathbb{R}^n$ and $\mathbb{R}^m$. Let f be a continuous function from $K_0 \times X_0$ to $\mathbb{R}$. For any $a \in K_0$ we define $M_a = \{x \in X_0 \cap \mathbb{Z}^n \ / \ f(a,x) = min_{x' \in X_0 \cap \mathbb{Z}^n}(f(a,x'))\}$. Let $K \subset K_0$ be a compact and $X = \cup_{a \in K} M_a \subset X_0 \cap \mathbb{Z}^n$. With these notations, there exists $\varepsilon > 0$ such that if $b \in K_0$ satisfies dist$(b, K) \leq \varepsilon$, we have $M_b \subset X$.*

In order to apply the Topological Lemma, we need to map the relaxed bases into a compact set. For any $\varepsilon \geq 0$ and any $\alpha \in [0, 1]$, we define:

$$K_2(\varepsilon, \alpha) = \{(\boldsymbol{b}_1, \boldsymbol{b}_2)/\boldsymbol{b}_1, \boldsymbol{b}_2 \ \varepsilon\text{-reduced}, \ \alpha \leq ||\boldsymbol{b}_1|| \leq ||\boldsymbol{b}_2|| = 1\}$$
$$K_3(\varepsilon, \alpha) = \{(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3)/\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3 \ \varepsilon\text{-reduced}, \ \alpha \leq ||\boldsymbol{b}_1|| \leq ||\boldsymbol{b}_2|| \leq ||\boldsymbol{b}_3|| = 1\}.$$

Lemma 22. *If $\varepsilon \geq 0$ and $\alpha \in [0, 1]$, $K_2(\varepsilon, \alpha)$ and $K_3(\varepsilon, \alpha)$ are compact sets.*

The following lemma is the relaxed version of Lemma 16. It can also be viewed as a reciprocal to Lemma 19.

Lemma 23. *For any $\alpha \in \ ]0, 1]$ and any sufficiently small $\varepsilon > 0$, the possible Voronoï coords of $[\boldsymbol{b}_1, \boldsymbol{b}_2] \in K_2(\varepsilon, \alpha)$ are the same as for Minkowski-reduced bases, i.e. $(1, 0)$ and $(1, 1)$, modulo any change of signs and permutation of coordinates.*

We now relax Lemma 17 in the same manner.

Lemma 24. *For any $\alpha \in]0,1]$ and any sufficiently small $\varepsilon > 0$, the possible Voronoï coords of $[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3] \in K_3(\varepsilon, \alpha)$ are the same as for Minkowski-reduced bases.*

The following result generalizes Lemma 18 about the possible Voronoï coord $(1,1,2)$. As opposed to the two previous results, there is no need to use the Topological Lemma in this case, because only a finite number of (x_1, x_2, x_3)'s is considered.

Lemma 25. *There exists a constant $c' > 0$ such that for any sufficiently small $\varepsilon > 0$, if $[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3]$ are ε-reduced and $||\boldsymbol{b}_3|| = 1$, then:*
1. If any of $(s_1, s_2, 2)$ is a Voronoï coord with $s_i = \pm 1$ for $i \in \{1,2\}$, then: $||\boldsymbol{b}_1|| \geq 1 - c'\varepsilon$, $|\langle \boldsymbol{b}_1, \boldsymbol{b}_2 \rangle| \leq c'\varepsilon$ and $|\langle \boldsymbol{b}_i, \boldsymbol{b}_3 \rangle + s_i \frac{||\boldsymbol{b}_1||^2}{2}| \leq c'\varepsilon$ for $i = 1,2$.
2. If any of $(s_1, 2, s_3)$ is a Voronoï coord with $s_i = \pm 1$ for $i \in \{1,3\}$, then $(1 - c'\varepsilon)||\boldsymbol{b}_2|| \leq ||\boldsymbol{b}_1|| \leq ||\boldsymbol{b}_2||$. Moreover, if $||\boldsymbol{b}_1|| \geq 1 - \varepsilon$, then: $|\langle \boldsymbol{b}_1, \boldsymbol{b}_3 \rangle| \leq c'\varepsilon$ and $|\langle \boldsymbol{b}_i, \boldsymbol{b}_2 \rangle + s_i \frac{||\boldsymbol{b}_1||^2}{2}| \leq c'\varepsilon$ for $i = 1,3$.
3. If any of $(2, s_2, s_3)$ is a Voronoï coord with $s_i = \pm 1$ for $i \in \{2,3\}$ and if $||\boldsymbol{b}_1|| \geq 1 - \varepsilon$, then: $|\langle \boldsymbol{b}_2, \boldsymbol{b}_3 \rangle| \leq c'\varepsilon$ and $|\langle \boldsymbol{b}_i, \boldsymbol{b}_1 \rangle + s_i \frac{||\boldsymbol{b}_1||^2}{2}| \leq c'\varepsilon$ for $i = 2,3$.

5.3 The Gap Lemma

The goal of this subsection is to prove that even with relaxed bases, if one adds a lattice vector with not too small coordinates to a vector of the Voronoï cell, this vector becomes significantly longer. This result will be used the other way round: if the x_i's found at Step 5 of the greedy algorithm are not too small, then $\boldsymbol{b}_d$ is significantly shorter than $\boldsymbol{a}_d$. We first need to generalize the compact sets K_2 and K_3. For any $\varepsilon \geq 0$ and any $\alpha \in [0,1]$, we define:

$$K_2'(\varepsilon, \alpha) = \{(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{u})/(\boldsymbol{b}_1, \boldsymbol{b}_2) \in K_2(\varepsilon, \alpha),\ \boldsymbol{u} \in \mathrm{Vor}(\boldsymbol{b}_1, \boldsymbol{b}_2)\}$$
$$K_3'(\varepsilon, \alpha) = \{(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3, \boldsymbol{u})/(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3) \in K_3(\varepsilon, \alpha),\ \boldsymbol{u} \in \mathrm{Vor}(\boldsymbol{b}_1, \boldsymbol{b}_2)\}.$$

Lemma 26. *If $\varepsilon > 0$ and $\alpha \in [0,1]$, $K_2'(\varepsilon, \alpha)$ and $K_3'(\varepsilon, \alpha)$ are compact sets.*

The next result is the two-dimensional version of the Gap Lemma.

Lemma 27. *There exist two constants $\varepsilon, C > 0$ such that for any ε-reduced vectors $[\boldsymbol{b}_1, \boldsymbol{b}_2]$ and any $\boldsymbol{u} \in Vor(\boldsymbol{b}_1, \boldsymbol{b}_2)$, if at least one of the following conditions holds, then: $\|\boldsymbol{u} + x_1\boldsymbol{b}_1 + x_2\boldsymbol{b}_2\|^2 \geq \|\boldsymbol{u}\|^2 + C\|\boldsymbol{b}_2\|^2$.*

- (1) $|x_2| \geq 2$,
- (2) $|x_1| \geq 2$ *and* $\|\boldsymbol{b}_1\|^2 \geq ||\boldsymbol{b}_2||^2/2$.

We now give the three-dimensional Gap Lemma, on which relies the analysis of the four-dimensional greedy algorithm.

Lemma 28. *There exist two constants $\varepsilon, C > 0$ such that for any ε-reduced vectors $[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3]$ and any $\boldsymbol{u} \in Vor(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3)$, if at least one of the following conditions holds, then:*

$$\|\boldsymbol{u} + x_1\boldsymbol{b}_1 + x_2\boldsymbol{b}_2 + x_3\boldsymbol{b}_3\|^2 \geq \|\boldsymbol{u}\|^2 + C\|\boldsymbol{b}_3\|^2.$$

- (1) $|x_3| \geq 3$, *or* $|x_3| = 2$ *and* $(|x_1|, |x_2|) \neq (1, 1)$;
- (2) $||\boldsymbol{b}_2|| \geq ||\boldsymbol{b}_3||/2$ *and:* $|x_2| \geq 3$, *or* $|x_2| = 2$ *with* $(|x_1|, |x_3|) \neq (1, 1)$;
- (3) $||\boldsymbol{b}_1|| \geq ||\boldsymbol{b}_3||/2$ *and:* $|x_1| \geq 3$, *or* $|x_1| = 2$ *with* $(|x_2|, |x_3|) \neq (1, 1)$;

Like in the previous subsections, we now consider the case of the possible Voronoï coords $(\pm 1, \pm 1, \pm 2)$ modulo any permutation of coordinates.

Lemma 29. *There exist two constants* $\varepsilon, C > 0$ *such that for any* ε*-reduced vectors* $[\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3]$ *and any* $\boldsymbol{u} \in Vor(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3)$*, if at least one of the following conditions holds, then:*

$$||\boldsymbol{u} + x_1\boldsymbol{b}_1 + x_2\boldsymbol{b}_2 + x_3\boldsymbol{b}_3||^2 \geq ||\boldsymbol{u}||^2 + C||\boldsymbol{b}_3||^2.$$

- *1-* $(x_1, x_2, x_3) = (s_1, s_2, 2)$, $|s_i| = 1$ *for* $i \in \{1, 2\}$ *and at least one of the following conditions holds:*
  $||\boldsymbol{b}_1|| \leq (1-\varepsilon)||\boldsymbol{b}_3||$, *or* $||\boldsymbol{b}_2|| \leq (1-\varepsilon)||\boldsymbol{b}_3||$, *or* $|\langle \boldsymbol{b}_1, \boldsymbol{b}_2 \rangle| \geq \varepsilon||\boldsymbol{b}_3||^2$, *or* $|\langle \boldsymbol{b}_1, \boldsymbol{b}_3 \rangle + s_1\frac{||\boldsymbol{b}_1||^2}{2}| \geq \varepsilon||\boldsymbol{b}_3||^2$, *or* $|\langle \boldsymbol{b}_2, \boldsymbol{b}_3 \rangle + s_2\frac{||\boldsymbol{b}_1||^2}{2}| \geq \varepsilon||\boldsymbol{b}_3||^2$.
- *2a-* $(x_1, x_2, x_3) = (s_1, 2, s_3)$, $|s_i| = 1$ *for* $i \in \{1, 3\}$ *and* $||\boldsymbol{b}_1|| \leq (1-\varepsilon)||\boldsymbol{b}_3||$.
- *2b-* $(x_1, x_2, x_3) = (s_1, 2, s_3)$, $|s_i| = 1$ *for* $i \in \{1, 3\}$, $||\boldsymbol{b}_1|| \geq (1-\varepsilon)||\boldsymbol{b}_3||$ *and at least one of the following conditions holds:* $|\langle \boldsymbol{b}_1, \boldsymbol{b}_3 \rangle| \geq \varepsilon||\boldsymbol{b}_3||^2$, *or* $|\langle \boldsymbol{b}_1, \boldsymbol{b}_2 \rangle + s_1\frac{||\boldsymbol{b}_1||^2}{2}| \geq \varepsilon||\boldsymbol{b}_3||^2$, *or* $|\langle \boldsymbol{b}_3, \boldsymbol{b}_2 \rangle + s_3\frac{||\boldsymbol{b}_1||^2}{2}| \geq \varepsilon||\boldsymbol{b}_3||^2$.
- *3-*$(x_1, x_2, x_3) = (2, s_2, s_3)$, $|s_i| = 1$ *for* $i \in \{2, 3\}$, $||\boldsymbol{b}_1|| \geq (1-\varepsilon)||\boldsymbol{b}_3||$ *and at least one of the following conditions holds:* $|\langle \boldsymbol{b}_2, \boldsymbol{b}_3 \rangle| \geq \varepsilon||\boldsymbol{b}_3||^2$, *or* $|\langle \boldsymbol{b}_2, \boldsymbol{b}_1 \rangle + s_2\frac{||\boldsymbol{b}_1||^2}{2}| \geq \varepsilon||\boldsymbol{b}_3||^2$, *or* $|\langle \boldsymbol{b}_3, \boldsymbol{b}_1 \rangle + s_3\frac{||\boldsymbol{b}_1||^2}{2}| \geq \varepsilon||\boldsymbol{b}_3||^2$.

Acknowledgements.

We thank Ali Akhavi, Florian Heß, Igor Semaev and Jacques Stern for helpful discussions and comments.

References

1. M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proc. of the 28th Symposium on the Theory of Computing*, pages 99–108. ACM Press, 1996.
2. M. Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract). In *Proc. of the 30th Symposium on the Theory of Computing*, pages 10–19. ACM Press, 1998.
3. A. Akhavi and C. Moreira dos Santos. Another view of the Gaussian algorithm. In *Proc. of LATIN '04*, Lecture Notes in Computer Science. Springer-Verlag, 2004.
4. J.W.S. Cassels. *An Introduction to the Geometry of Numbers*. Springer-Verlag, Berlin, 1959.
5. C.F. Gauss. *Disquisitiones Arithmeticæ*. Leipzig, 1801.
6. M. Gruber and C. G. Lekkerkerker. *Geometry of Numbers*. North-Holland, Amsterdam, 1987.
7. B. Helfrich. Algorithms to construct Minkowski reduced and Hermite reduced lattice bases. *Th. Computer Science*, 41:125–139, 1985.

8. C. Hermite. Extraits de lettres de M. Hermite à M. Jacobi sur différents objets de la théorie des nombres, deuxième lettre. *J. Reine Angew. Math.*, 40:279–290, 1850.
9. C. Hermite. *Œuvres*. Gauthier-Villars, Paris, 1905.
10. M. Kaib and C. P. Schnorr. The generalized Gauss reduction algorithm. *J. of Algorithms*, 21(3):565–578, 1996.
11. A. Korkine and G. Zolotareff. Sur les formes quadratiques. *Math. Ann.*, 6:336–389, 1873.
12. J. C. Lagarias. Worst-case complexity bounds for algorithms in the theory of integral quadratic forms. *J. of Algorithms*, 1:142–186, 1980.
13. A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261:513–534, 1982.
14. J. Martinet. *Perfect Lattices in Euclidean Spaces*. Springer-Verlag, Heidelberg, 2002.
15. D. Micciancio. The shortest vector problem is NP-hard to approximate to within some constant. In *Proc. of the 39th Symposium on the Foundations of Computer Science*, pages 92–98. IEEE, 1998.
16. D. Micciancio and S. Goldwasser. *Complexity of lattice problems: A cryptographic perspective*. Kluwer Academic Publishers, Boston, 2002.
17. H. Minkowski. *Geometrie der Zahlen*. Teubner-Verlag, Leipzig, 1896.
18. P. Q. Nguyen and J. Stern. The two faces of lattices in cryptology. In *Proc. of CALC '01*, volume 2146 of *Lecture Notes in Computer Science*, pages 146–180. Springer-Verlag, 2001.
19. S. S. Ryskov. On Hermite, Minkowski and Venkov reduction of positive quadratic forms in n variables. *Soviet Math. Doklady*, 13:1676–1679, 1972.
20. C. P. Schnorr. A hierarchy of polynomial lattice basis reduction algorithms. *Th. Computer Science*, 53:201–224, 1987.
21. C. P. Schnorr and M. Euchner. Lattice basis reduction: improved practical algorithms and solving subset sum problems. *Math. Programming*, 66:181–199, 1994.
22. C. P. Schnorr and H. H. Hörner. Attacking the Chor-Rivest cryptosystem by improved lattice reduction. In *Proc. of Eurocrypt '95*, volume 921 of *Lecture Notes in Computer Science*, pages 1–12. Springer-Verlag, 1995.
23. A. Schönhage and V. Strassen. Schnelle Multiplikation grosser Zahlen. *Computing*, 7:281–292, 1971.
24. I. Semaev. A 3-dimensional lattice reduction algorithm. In *Proc. of CALC '01*, volume 2146 of *Lecture Notes in Computer Science*, pages 181–193. Springer-Verlag, 2001.
25. C. L. Siegel. *Lectures on the Geometry of Numbers*. Springer-Verlag, Berlin, Heidelberg, 1989.
26. M. I. Stogrin. *Regular Dirichlet-Voronoï partitions for the second triclinic group*. American Mathematical Society, 1975. English translation of the Proceedings of the Steklov Institute of Mathematics, Number 123 (1973).
27. B. Vallée. *Une Approche Géométrique de la Réduction de Réseaux en Petite Dimension*. PhD thesis, Université de Caen, 1986.
28. B. Vallée. Gauss' algorithm revisited. *J. of Algorithms*, 12(4):556–572, 1991.
29. B. L. van der Waerden. Die Reduktionstheorie der positiven quadratischen Formen. *Acta Mathematica*, 96:265–309, 1956.
30. G. Voronoï. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. *J. Reine Angew. Math.*, 134:198–287, 1908.