EPFL

Thèse n° 9934

Geometric Considerations in Lattice Programming

Présentée le 24 février 2023

Faculté des sciences de base
Chaire d'optimisation discrète
Programme doctoral en mathématiques

pour l'obtention du grade de Docteur ès Sciences

par

Moritz Andreas VENZIN

Acceptée sur proposition du jury

Prof. T. Mountford, président du jury
Prof. F. Eisenbrand, directeur de thèse
Prof. D. Dadush, rapporteur
Prof. N. Stephens-Davidowitz, rapporteur
Prof. M. Kapralov, rapporteur

École
polytechnique
fédérale
de Lausanne

2023

Acknowledgements

This thesis would not have been possible without the support of many people. I would like to thank everyone who contributed, be it either through interesting (mathematical) discussions, encouragement or for non thesis-related activities.
First of all, I would like thank Fritz Eisenbrand. It is still a mystery to me why he accepted me as a PhD student four years ago; his optimism and constant encouragement were invaluable. He also introduced me to the interplay of lattices and convex bodies, the main theme of this thesis. I would like to thank my co-authors Márton Naszódi and Thomas Rothvoss (and Fritz) for the fruitful collaboration. My gratitude goes to my colleagues (and friends) from the DisOpt lab. In no particular order, these are Manuel, Igor, Christoph, Jana, Jonas, Martina, Matthieu, Eleonore, Lukas, Kim, Georg, Matthias, Adam, Alex, Fritz, Pauline and Jocelyne. They all contributed to a nice and stimulating work environment here at EPFL and went out of their way to help me in any way possible. They made sure I had enough fun on (and off) the job so that I would not finish this thesis too early. I would also like to thank all my friends from before my journey at EPFL and those I have made along the way. Boardgame nights, late night poker matches, long hikes, climbing trips, incredible ski touring, or fun evenings with great food, lots of laughs and interesting discussions; I have always had a fantastic time! Finally, a huge thank you goes to my parents and my sisters. They have always been there for me, for this I am very grateful.

Abstract

In this thesis we consider the shortest and the closest vector problem in general norms $\|\cdot\|_K$. For lattices of rank n, we show that both of these problems admit an $O(1)$-approximation in $O(2^{0.802n})$ time. This contrasts recent lower bounds showing that these problems cannot be approximated within a factor $1+\varepsilon_0$ for some constant $\varepsilon_0 > 0$ in better-than-2^n time. Our approach is based on the following geometric consideration: How many scaled ellipsoids are needed to cover K, the unit norm ball of $\|\cdot\|_K$, but, conversely, each such ellipsoid can be covered by few translates of K. This provides a geometric framework allowing us to convert algorithms for the shortest and closest vector problem in general norms to their counterparts in the Euclidean norm.

Keywords: shortest vector problem, closest vector problem, lattice, algorithm, fine-grained reduction, convex bodies, translative covering numbers

Zusammenfassung

In dieser Arbeit betrachten wir das Problem des nächsten und des kürzesten Vektors in beliebigen Normen $\|\cdot\|_K$. Für Gitter mit Rang n stellen wir einen $O(1)$-Approximationsalgorithmus mit einer Laufzeit von $2^{0.802n}$ vor. Das ergänzt untere Laufzeitschranken, die beweisen, dass es keinen $(1+\varepsilon_0)$-Approximationsalgorithmus für ein konstantes $\varepsilon_0 > 0$ mit einer Laufzeit schneller als 2^n für diese Probleme geben kann. Unser Ansatz basiert auf der folgenden geometrischen Aufgabe: Wie viele Ellipsoide braucht es um K, die Einheitskugel bezüglich der Norm $\|\cdot\|_K$, zu überdecken, wobei jedes dieser Ellipsoide von möglichst wenigen Translaten von K überdeckt werden kann. Diese geometrische Überlegung erlaubt es uns, die Probleme des kürzesten und nächsten Vektors in beliebigen Normen auf deren Gegenstück in der Euklidischen Norm zu überführen.

Keywords: Problem des kürzesten Vektors, Problem des nächsten Vektors, Gitter, Algorithmus, Reduktion, konvexe Körper, Überdeckungszahlen

Contents

Introduction

We consider the following two *lattice problems* on integer points and convex bodies. For some centrally symmetric convex body $K \subseteq \mathbb{R}^n$ centered at $\mathbf{0}$, determine whether it contains a nonzero integer point. If K is not centered at $\mathbf{0}$, determine whether it contains an integer point at all. These problems are illustrated in Figure 1.

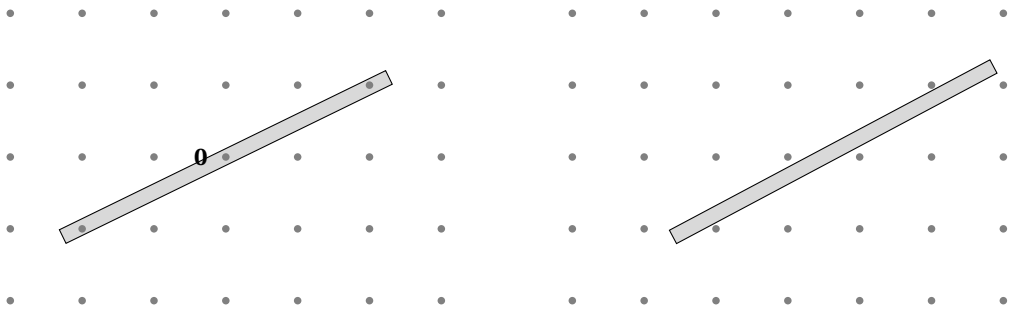

Figure 1 – The shortest and closest vector problem.

These two simple-to-state problems go by the name of *shortest and closest vector problem* and go back to the likes of Gauss, Hermite, Lagrange and Minkowski and are central in the geometry of numbers (Min10). In more recent years, the shortest and closest vector problem have also gained prominence in many areas at the intersection of mathematics and theoretical computer science such as discrete optimization, integer programming, cryptanalysis and cryptography to name a few. Indeed, many theoretical or practical problems can be naturally formulated in this geometric setting. For instance if K is given by the intersection of halfspaces, the closest vector problem is equivalent to the integer programming problem and figures among Karp's classic NP-complete problems (Kar72).

Much work has been devoted to the complexity of the shortest and closest vector problem. The currently best algorithms for the shortest and closest vector problem have a time complexity of $2^{O(n)}$ and $n^{O(n)}$ respectively (AKS01; Kan87). Only when K is an ellipsoid, the time complexity of the closest vector problem has been brought down to singly-exponential in n, 2^n in fact; it now exactly matches the time complexity of the currently best known algorithm for the shortest vector problem (MV10a; ADRS15). It is a major open question whether the running time of the closest vector problem for general convex bodies can be improved to $2^{O(n)}$, matching the complexity of the shortest vector problem for general convex bodies. Such an exponential running time is needed even for small approximation factors, e.g. we are allowed to slightly *scale* the convex body K. In this setting, the currently fastest algorithms for both problems

still take $2^{O(n)}$ time, or 2^n time for ellipsoids, and there is compelling evidence that no better-than-2^n-time algorithm for the shortest and closest vector problem should exist for general convex bodies as simple as parallelepipeds. These lower bounds even hold for some small, but constant, approximation factor (BGS17). On the other hand, if the convex body is an ellipsoid, one can compute a constant approximation in $2^{0.802n}$ time, and it is not known how to extend these strong lower bounds to this setting.

In this thesis, we investigate the relationship between convex bodies and the complexity of lattice problems. Exploring the connection between high-dimensional convex geometry and lattices, we obtain the currently fastest algorithms for the approximate shortest and closest vector problem. In particular, we obtain a $2^{0.802n}$ time algorithm to compute a constant approximation to the shortest and closest vector problem for any convex body, contrasting the corresponding 2^n time complexity-theoretic lower bounds. The underlying geometric ideas related to translative covering numbers can be generalized to a reduction from the shortest or closest vector problem to lattice problems in the Euclidean norm, i.e. the convex body K being an ellipsoid, or any other symmetric convex body. These results indicate that already for constant approximation factors, the specific choice of the convex body K for the shortest and closest vector problem does not seem to determine their respective complexity.

Overview and summary of results

The first chapter of this thesis introduces the necessary background and we review in more detail the complexity and algorithmic landscape of lattice problems. The technical details in this part are treated informally; a certain familiarity with complexity theory, linear algebra and probability theory is assumed.

The second chapter provides reductions among lattice problems in various norms. This is achieved through geometric covering techniques such as translative covering numbers tailored to our setting. This is based on joint works with Friedrich Eisenbrand and Thomas Rothvoss, respectively (EV22; RV22).

The third chapter provides approximation algorithms for the shortest and closest vector problem in the near-exact setting and for large approximation factors. This is achieved through the use of geometric covering techniques as used in the second part, combined with certain technical properties of the currently fastest algorithms for lattice algorithms in the Euclidean norm, e.g. when the convex body is an ellipsoid. This is based on joint works with Thomas Rothvoss and Márton Naszódi respectively (RV22; NV22).

The fourth and final chapter presents the geometric constructions as used in the previous two chapters. This is based on joint work with Thomas Rothvoss (RV22).

1 Introduction to Lattice Problems

This introductory chapter is divided into four parts. Section 1.1 defines lattices, convex bodies and two central computational problems on lattices and convex bodies treated in this thesis. Section 1.2 presents the computational model. Section 1.3 provides an overview on the algorithmic landscape of lattice problems. Section 1.4 reviews the complexity of lattice problems.

1.1 Lattices and Convex Bodies

Given a matrix $\mathbf{B} \in \mathbb{R}^{d \times n}$ of full column rank, the *lattice* $\mathscr{L}(\mathbf{B})$ spanned by $\mathbf{B}$ is defined as

$$\mathscr{L}(\mathbf{B}) := \left\{ \mathbf{B} \cdot z \,|\, z \in \mathbb{Z}^n \right\}.$$

The *rank* of the lattice is n, its (ambient) *dimension* equals d.

Equivalently, a lattice $\mathscr{L}$ is a discrete and additive sub-group of $\mathbb{R}^d$. For any such lattice $\mathscr{L}$, there exists a suitable choice of $\mathbf{B} \in \mathbb{R}^{d \times n}$ such that $\mathscr{L} = \mathscr{L}(\mathbf{B})$. This matrix B is refered to as the *basis* of the lattice, its columns $\{\mathbf{b}_1, \cdots, \mathbf{b}_n\} \subseteq \mathbb{R}^d$ are the *generators* of the basis and are said to *span* the lattice. Such a lattice basis is not unique: whenever there exists a unimodular matrix $U \in \mathbb{Z}^{n \times n}$ such that $\mathbf{B}_1 = \mathbf{B}_2 \cdot U$, the columns of $\mathbf{B}_1$ and $\mathbf{B}_2$ span the same lattice, see Figure 1.1. It should be noted that $d \geq n$ follows from the requirement that $\mathbf{B}$ is of full column rank, i.e. $\det(\mathbf{B}^T \cdot \mathbf{B}) \neq 0$. Equivalently, this also follows from the requirement that $\mathscr{L}$ is discrete. Whenever the lattice is *rational*, i.e. $\mathscr{L}(\mathbf{B}) \subseteq \mathbb{Q}^d$, the requirement that $\mathbf{B} \in \mathbb{Q}^{d \times n}$ is of full column rank only guarantees that the rank of the lattice equals n (and is not strictly smaller than n) and is redundant otherwise. Throughout this thesis, we will assume that lattices are given by their basis. Whenever clear, we omit the reference to the basis and simply write $\mathscr{L}$. Finally, we will use bold-faced letters to denote lattice vectors and matrices involving lattice vectors to help distinguish from arbitrary vectors and matrices. Consequently, the all zero vector will be denoted by $\mathbf{0}$.

Given some lattice, we will study important parameters such as the length of the shortest nonzero vector or the distance from a given target vector to the closest lattice point. Here,

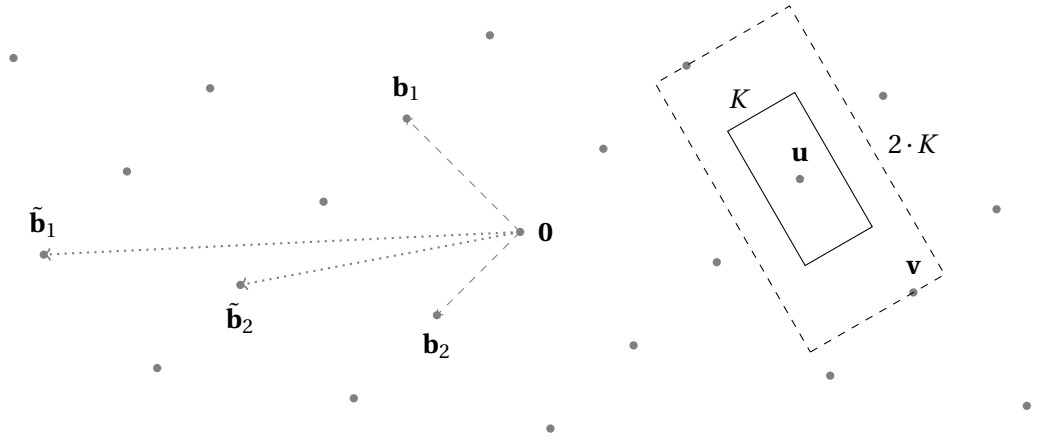

Figure 1.1 – A lattice spanned by $(\mathbf{b}_1, \mathbf{b}_2)$ or, equivalently, $(\tilde{\mathbf{b}}_1, \tilde{\mathbf{b}}_2)$. The lattice vectors $\mathbf{u}$ and $\mathbf{v}$ have distance 2 with respect to $\|\cdot\|_K$.

"short" and "close" will be measured with respect to a given norm. It will be convenient to define a norm through the use of *(symmetric) convex bodies*. A set $K \subseteq \mathbb{R}^d$ is said to be a convex body, if it is compact and if for any $x, y \in K$ and $\lambda \in [0,1]$, $\lambda \cdot x + (1-\lambda) \cdot y$ belongs to K as well. Furthermore, K is symmetric if $-K = K$. The norm $\|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ associated with the convex body K is then defined as

$$\|x\|_K := \min\{s \mid x \in s \cdot K\}.$$

See Figure 1.1 for an illustration.

Conversely, any norm $\|\cdot\| : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$, i.e. a function verifying the triangle inequality ($\|x+y\| \leq \|x\| + \|y\|$), homogeneity ($\|r \cdot x\| = |r| \cdot \|x\|$) and positive-definiteness ($\|x\| = 0$ iff $x = \mathbf{0}$), is induced by the symmetric and convex body $K := \{x \in \mathbb{R}^d \mid \|x\| \leq 1\}$. For this reason and to keep the discussion as geometric as possible, we always denote norms by $\|\cdot\|_K$, where K is the convex body inducing the norm. For the important cases of ℓ_p norms defined by $\|x\|_{\ell_p} := (\sum_{i=1}^d |x_i|^p)^{1/p}$, we simply write $\|\cdot\|_p$.

In this thesis we will mostly consider the following two related computational problems on lattices. The first is the *shortest vector problem*, abbreviated by SVP.

Shortest Vector Problem (SVP)
INPUT: $\mathbf{B} \in \mathbb{Q}^{d \times n}$ and a convex body $K \subseteq \mathbb{R}^d$.
OUTPUT: A shortest, nonzero lattice vector of $\mathbf{v} \in \mathscr{L}(\mathbf{B}) \setminus \{\mathbf{0}\}$ with respect to $\|\cdot\|_K$.

The *closest vector problem* (CVP) is then defined as an inhomogeneous version of the shortest vector problem. Note that unlike for the shortest vector problem, we do not insist on a nonzero lattice vector.

Closest Vector Problem (CVP)
INPUT: $\mathbf{B} \in \mathbb{Q}^{d\times n}$, a target $t \in \mathbb{Q}^d$ and a convex body $K \subseteq \mathbb{R}^d$.
OUTPUT: A lattice vector $\mathbf{v} \in \mathscr{L}(\mathbf{B})$ with $\|t-\mathbf{v}\|_K = \min_{\mathbf{w}\in\mathscr{L}(\mathbf{B})} \|t-\mathbf{w}\|_K$.

Whenever we emphasize the norm $\|\cdot\|_K$ under consideration, we denote these problems by SVP_K and CVP_K, respectively. See Figure 1.2 for an illustration.

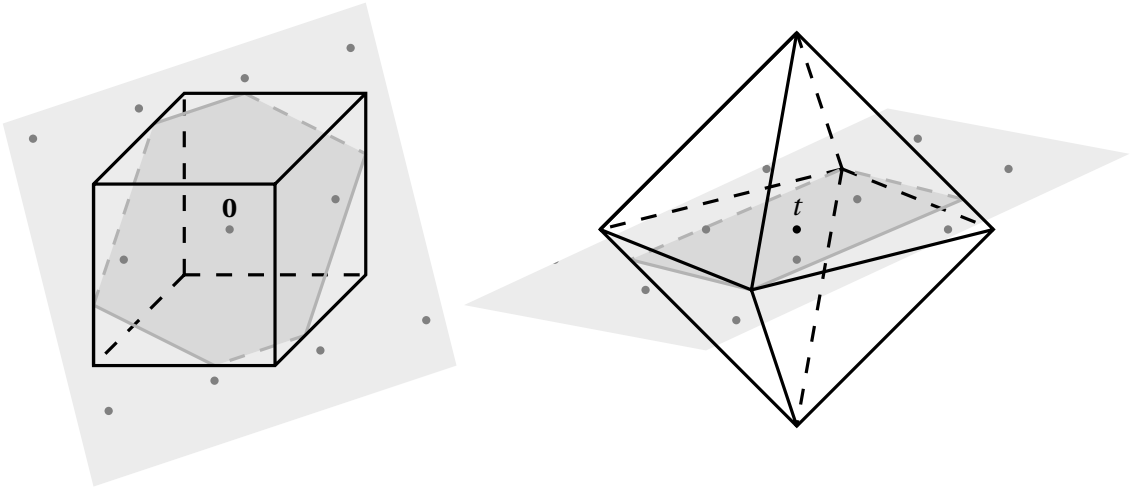

Figure 1.2 – SVP_∞ and CVP_1. The respective lattices are 3-dimensional and their rank equals 2.

We often consider *approximations* to the shortest and closest vector problem.

We denote by γ-SVP_K a γ-approximation to the shortest vector problem, where we want to find $\mathbf{v} \in \mathscr{L}$ with $\|\mathbf{v}\|_K \leq \gamma \cdot \lambda_1^{(K)}(\mathscr{L})$, where we define $\lambda_1^{(K)}(\mathscr{L}) := \min_{\mathbf{w}\in\mathscr{L}\setminus\{\mathbf{0}\}} \|\mathbf{w}\|_K$. Similarly for γ-CVP_K, where we care to find a lattice vector $\mathbf{v} \in \mathscr{L}$ with $\|t-\mathbf{v}\| = \gamma \cdot \text{dist}_K(\mathscr{L}, t)$, where $\text{dist}_K(\mathscr{L}, t) := \min_{\mathbf{w}\in\mathscr{L}(B)} \|t-\mathbf{w}\|_K$.

1.2 Computational Model

Throughout this thesis, we will assume that the lattice $\mathscr{L}(\mathbf{B})$ is rational, i.e. $\mathbf{B} \in \mathbb{Q}^{d\times n}$. The *encoding size* of $\mathbf{B}$, denoted by size($\mathbf{B}$), is defined as the number of bits required to store the matrix $\mathbf{B}$. For a single rational number represented as $\frac{p}{q}$, $p, q \in \mathbb{Z}$, the encoding size equals $1 + \log_2(|p|+1) + \log_2(|q|+1)$. For the matrix $\mathbf{B}$, we can upper bound size($\mathbf{B}$) by $d \cdot n$ times the largest encoding size of its entries. It is clear that any algorithm $\mathscr{A}$ taking $\mathbf{B}$ as input will take time proportional to the size of $\mathbf{B}$ before outputting an answer. In all lattice algorithms that we will encounter in this thesis, the running times will depend *polynomially* on the input size of the respective basis. For this reason, we will omit the encoding size in all statements of algorithms and in all proofs: whenever we say that algorithm $\mathscr{A}$ runs in time T, what we really mean is that algorithm $\mathscr{A}$ runs in time T times some polynomial in the input size of $\mathscr{A}$. When stating running times and space requirements, we omit all lower order terms.

As discussed in the first section, every norm $\|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ is induced by some convex body $K = -K \subseteq \mathbb{R}^d$ and vice versa. Whenever K is given explicitly, e.g., by a set of inequalities, as an intersection of some ℓ_p norm ball with some hyperplane, etc., we can directly calculate the norm $\|\cdot\|_K$. Specifically, in time polynomial in the encoding size of these inequalities and $\log(1/\varepsilon)$, we can calculate the norm up to an additive error of ε. To avoid these numerical technicalities altogether and to ensure that our algorithms work in the most general setting possible, we will assume that we are given *oracle access* to K. For any $x \in \mathbb{R}^d$ and any $\varepsilon > 0$, we are allowed to make queries to the oracle of the form: "is $x \in K$?". In the *weak membership oracle* model, the oracle returns

$$\begin{cases} \text{YES} & \text{if } x \in K \setminus \varepsilon \cdot B_2^d := \{x \in \mathbb{R}^d \mid x + \varepsilon \cdot B_2^n \subseteq K\} \\ \text{NO}, & \text{if } x \notin K + \varepsilon \cdot B_2^d \\ \text{YES or NO}, & \text{else.} \end{cases}$$

In the *weak separation oracle* model, in addition to a NO output (when $x \notin K + \varepsilon \cdot B_2^d$), the oracle returns a *separating hyperplane*. This is an inequality of the form $a^T \cdot y \leq \gamma$ such that $a^T \cdot z \leq \gamma$ for all $z \in K$ and $a^T \cdot x > \gamma$. In this model, we treat the encoding size of the separating hyperplane as being of unit cost. This model is stronger than a weak membership oracle, but, whenever $s + r \cdot B_2^d \subseteq K \subseteq R \cdot B_2^d$ for some radii $r, R \in \mathbb{R}_{>0}$ and $s \in \mathbb{R}^d$, one call to a weak separation oracle for K can be evaluated using a polynomial (in d, $\log(R)$, $\log(r)$ and $\log(1/\varepsilon)$) number of calls to a weak membership oracle for K, meaning these two models are polynomially equivalent, see (GLS88). For these reasons, we will assume that any convex body K inducing the norm $\|\cdot\|_K$ is given by a weak separation oracle. This simplifies matters as it avoids having to consider the input size of K and will be sufficient for all algorithmic applications of convex bodies and norms in the subsequent chapters.

Finally, we discuss the role of *randomness*. Indeed, most algorithms and reductions presented in this thesis rely on it. For instance, we need to sample uniformly random integers within some range, or sample points distributed uniformly within some convex body $C \subseteq \mathbb{R}^d$, (DFK91). These tasks can be accomplished in polynomial time, up to some error. The error is measured by some *statistical distance*. If the distribution is $p(x)$ for $x \in X$ and we are able to sample according to some other distribution $\tilde{p}(\cdot)$, one such statistical distance is given by the *total variation*, $d_{TV}(p, \tilde{p}) := \sum_{x \in X} |p(x) - \tilde{p}(x)|$. For our purposes, the support of X is finite. Hence, when we say that we sample a point uniformly within a convex body $C \subseteq \mathbb{R}^n$, what we really mean is that for some $\delta > 0$, we sample a point distributed uniformly within $C \cap \delta \cdot \mathbb{Z}^n$. This also takes care of any issues with the encoding size of the sampled point. If the total variation is small enough, the resulting distribution can be thought of as follows: we flip a (biased) coin, if it lands heads, the distribution $\tilde{p}(\cdot)$ is distributed exactly as $p(\cdot)$. Otherwise, if it lands tails, we cannot conclude anything. The probability that the coin lands heads, i.e. the distribution $\tilde{p}(\cdot)$ being distributed according to $p(x)$, increases as the total variation decreases. To justify this intuition, we can write the distribution $\tilde{p}(\cdot)$ as a *convex combination* of the distribution

$p(\cdot)$ and some other probability distribution $p^{\text{error}}(\cdot)$. For some $\bar{\lambda} \in [0,1]$, we write

$$\tilde{p}(\cdot) = \bar{\lambda} \cdot p(\cdot) + (1-\bar{\lambda}) \cdot \underbrace{\frac{\tilde{p}(\cdot) - \bar{\lambda} \cdot p(\cdot)}{1-\bar{\lambda}}}_{:= p^{\text{error}}(\cdot)}.$$

Provided $\bar{\lambda} \in [0,1]$ is such that $p^{\text{error}}(x)$ is positive for each $x \in X$, this decomposition is well defined. In particular, for X finite,

$$\bar{\lambda} = \max\left\{0, 1 - \frac{d_{TV}(\tilde{p}, p)}{\min_{x \in X,\, p(x) \neq 0} p(x)}\right\} \in [0,1]$$

is a valid choice. Here, $\bar{\lambda}$ corresponds to the probability that the coin lands heads, i.e. that the distribution $\tilde{p}(\cdot)$ is distributed exactly as $p(\cdot)$. For the tasks considered in this thesis, such as sampling points distributed uniformly within some convex body or sampling uniformly random integers within some range, the time to obtain a sample $x \sim \tilde{X}$ that is distributed according to $\tilde{p}(\cdot)$ that is *close* to the uniform distribution $p(\cdot)$, i.e. $d_{TV}(\tilde{p}, p) < \varepsilon$, depends on a polynomial of the relevant parameters and of the *logarithm* of $1/\varepsilon$. The logarithmic dependence on $1/\varepsilon$ proves to be crucial in the subsequent chapters, where $p(\cdot)$ is the uniform distribution on $\{0, \ldots, 2^{\Omega(n)}\}$ or on $C \cap \delta \cdot \mathbb{Z}^n$, for some convex body $C \subseteq \mathbb{R}^n$ and $\delta = O(1/\operatorname{poly}(n))$. In these cases, $\min_{x \in X,\, p(x) \neq 0} p(x)$ is of the order $2^{-\operatorname{poly}(n)}$. Nonetheless, in time polynomial in n, i.e. the logarithm of $1/2^{-\operatorname{poly}(n)}$, one can sample from a distribution that is arbitrarily close to the actual (uniform) distribution. Hence, for small enough statistical distance, we can use the respective random variables X and $\tilde{X}$ interchangeably. In particular, probabilistic arguments such as the *Union Bound* or *Markov's or Chebychev's Inequality* valid for X, see (MU05), also hold for $\tilde{X}$. For this reason, in the remainder of this thesis, we leave out all technicalities involving randomness and just assume that we have access to samples that are distributed exactly according to the respective distributions. For a thorough overview on the source of randomness and its application in algorithms and complexity theory, we refer to (AB09). For a more hands-on approach, see (ABB+55).

1.3 Algorithms for Lattice Problems

Both SVP and CVP and their respective approximations have found considerable applications, both in theory and practice. These include integer programming (Len83; Kan87), factoring polynomials over the rationals (LLL82) and cryptanalysis (Odl90). Since the seminal works of Lenstra-Lenstra-Lovász and Lenstra, a wide range of algorithmic techniques have been developed for lattice problems. Informally, these approaches can be loosely grouped into two categories. *Polynomial time algorithms* based on *basis-reduction* that achieve slightly sub-exponential approximations to the closest and shortest vector problem, and *exponential time algorithms* based on *sieving* or *enumeration* that yield exact or near-exact solutions to the closest and shortest vector problem.

1.3.1 Approximations in Polynomial Time

The shortest vector problem is a classic problem in mathematics, with the likes of Gauss, Lagrange and Hermite working on the topic. Motivated by applications in number theory such as quadratic reciprocity and the four-square-theorem, their works gave efficient algorithms for computing reasonably short lattice vectors assuming the rank of the lattice is fixed, (Eis10). Its importance was further accentuated, when in the late nineteen hundreds, Minkowski created the *geometry of numbers*, (Min10). However, it was only in 1982 when in their seminal work, Lenstra, Lenstra and Lovász (LLL) gave the first polynomial time algorithm that computes an approximation to the shortest vector in a lattice.

Theorem 1.3.1 (Lenstra-Lenstra-Lovász, (LLL82))**.** *Given is some lattice $\mathscr{L}$ with basis $\mathbf{B} \in \mathbb{Q}^{d\times n}$. In polynomial time, one can find a basis $\tilde{\mathbf{B}} \in \mathbb{Q}^{d\times n}$ such that*

$$\|\tilde{b}^*_{i+k}\|_2 \geq 2^{-k}\|\tilde{b}^*_i\|_2 \quad i+k \leq n,\, i,k \geq 0,$$

where $\tilde{b}^*_i$ is the Gram-Schmidt orthogonalization of $\tilde{\mathbf{b}}_i$ (with respect to the order of $\tilde{\mathbf{b}}_1, \cdots, \tilde{\mathbf{b}}_n$). Consequently, $\|\tilde{\mathbf{b}}_1\|_2$ is at most $2^{n/2}\cdot\lambda^{(2)}_1(\mathscr{L})$, where $\lambda^{(2)}_1(\mathscr{L}) := \min_{\mathbf{w}\in\mathscr{L}\setminus\{\mathbf{0}\}}\|\mathbf{w}\|_2$.

Since its appearance, there has been tremendous work to find new, polynomial time algorithms à la LLL with an improved approximation to the shortest vector. The currently best approximation guarantees achievable in polynomial time are of the order $2^{O(n\log\log(n)/\log(n))}$, see (Sch87; GN08).

These algorithms do also yield a $2^{O(n\log\log(n)/\log(n))}$-approximation to the shortest vector problem with respect to any norm $\|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ and for any rank n lattice. To do so, we will invoke John's theorem, that asserts that up to a linear transformation and a multiplicative factor of $d+1$, the convex body K can be approximated by an Euclidean ball.

Theorem 1.3.2 (John's Ellipsoid, (Joh48))**.** *Let $K = -K \subseteq \mathbb{R}^d$ be compact. There exists some linear transformation $A \subseteq \mathbb{R}^{d\times d}$ such that*

$$A\cdot B^n_2 \subseteq K \subseteq \sqrt{d}\cdot A\cdot B^n_2.$$

Given a (weak) separation oracle for K and in time polynomial in d and $\log(R/r+1)$ (where $r\cdot B^n_2 \subseteq K \subseteq R\cdot B^n_2$), (GLS88), one can compute a linear transformation $\tilde{A} \in \mathbb{Q}^{d\times d}$ such that

$$\tilde{A}\cdot B^n_2 \subseteq K \subseteq (d+1)\cdot\tilde{A}\cdot B^n_2.$$

For the shortest vector problem, we can always restrict the convex body $K \subseteq \mathbb{R}^d$ defining the norm to be n-dimensional by considering $K \cap \operatorname{span}(\mathbf{b}_1, \cdots, \mathbf{b}_n)$ instead of K. Formally, one can rotate $\mathscr{L}$ and K onto $\mathbb{R}^n \times \{0\}^{d-n}$ and then project onto the first n coordinates. We then compute the linear transformation $\tilde{A}$ guaranteed by Theorem 1.3.2. We then consider the lattice $\tilde{\mathscr{L}} := \mathscr{L}(\tilde{A}^{-1}\cdot\mathbf{B})$, compute an α-approximation $\tilde{\mathbf{v}} \in \tilde{\mathscr{L}}$ to the shortest vector with

respect to the ℓ_2 norm and return $\mathbf{v} := \tilde{A} \cdot \tilde{\mathbf{v}}$. We claim that this is a $(n+1) \cdot \alpha$-approximation to the shortest vector in $\mathscr{L}$ with respect to $\|\cdot\|_K$.

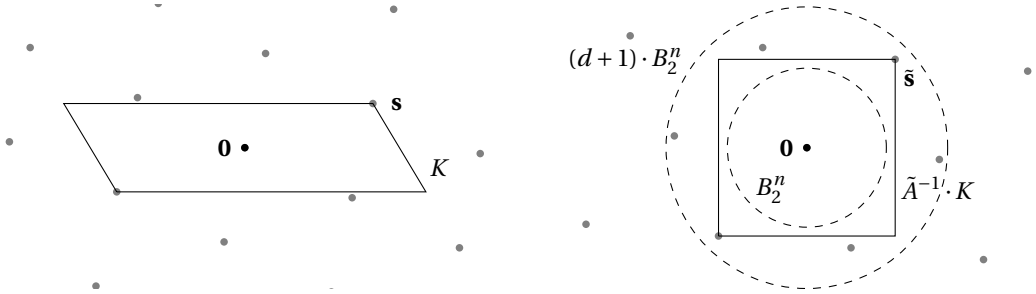

Figure 1.3 – Left: An instance of the shortest vector problem with lattice $\mathscr{L}(\mathbf{B})$. Right: Applying $\tilde{A}^{-1}$ to K and $\mathscr{L}(\mathbf{B})$. The resulting norm is Euclidean up to a multiplicative factor of $d+1$.

It is easy to see that for any $\mathbf{v} \in \mathscr{L}(\mathbf{B})$ and any (invertible) linear transformation $\tilde{A}$:

$$\mathbf{v} \in \mathscr{L}(\mathbf{B}),\ \|\mathbf{v}\|_K = \beta \text{ if and only if } A^{-1} \cdot \mathbf{v} =: \tilde{\mathbf{v}} \in \mathscr{L}(\tilde{A}^{-1} \cdot \mathbf{B}),\ \|\tilde{\mathbf{v}}\|_{\tilde{A}^{-1} \cdot K} = \beta.$$

Hence, since $B_2^n \subseteq \tilde{A}^{-1} \cdot K \subseteq (n+1) \cdot B_2^n$, meaning $\frac{1}{n+1} \cdot \|\cdot\|_2 \leq \|\cdot\|_{\tilde{A}^{-1} \cdot K} \leq \|\cdot\|_2$, any α-approximation with respect to the ℓ_2 norm is automatically a $(n+1) \cdot \alpha$-approximation with respect to $\|\cdot\|_{\tilde{A}^{-1} \cdot K}$. See Figure 1.3 for an illustration.

These approximation algorithms for the shortest vector problem also yield an approximation to the closest vector problem.

Theorem 1.3.3 (Babai's Nearest Plane Algorithm, (Bab86))**.** *Given a lattice $\mathscr{L}$ with basis $\mathbf{B} \in \mathbb{Q}^{d \times n}$, a target $t \in \mathbb{Q}^d$ and in polynomial time (in n and d), one can find an α-approximation to the closest vector problem (with respect to ℓ_2), where*

$$\alpha \leq n \cdot \max_{i \geq j} \frac{\|b_j^*\|_2}{\|b_i^*\|_2},$$

where $(b_1^*, \ldots, b_n^*)$ is the Gram-Schmidt orthogonalisation of $(\mathbf{b}_1, \ldots, \mathbf{b}_n) := \mathbf{B}$.

When instantiated with the block reduced basis from Schnorr, (Sch87), Babai's Nearest Plane Algorithm achieves an approximation guarantee of $\alpha \leq 2^{O(n \log\log(n)/\log(n))}$.

Using John's Theorem as outlined just above, this directly yields a $(d+1) \cdot 2^{O(n \log\log(n)/\log(n))}$ approximation to the closest vector problem. This can be improved to $O(n) \cdot 2^{O(n \log\log(n)/\log(n))} = \Omega(2^{O(n \log\log(n)/\log(n))})$. We first project the target t onto the span of the lattice (with respect to $\|\cdot\|_K$) and consider the resulting instance where the norm is induced by $K \cap \text{span}(\mathscr{L})$ and where we have restricted to $d = n$ by eliminating $d - n$ zero coordinates as for the shortest vector problem. This argument is sketched in the next section in Lemma 3.1.3.

1.3.2 Exponential Time Algorithms

Using John's theorem, tools from the geometry of numbers and the polynomial-time LLL basis-reduction algorithm, Lenstra gave the first algorithm for CVP_∞, or equivalently, the integer programming problem, (Len83). His algorithm ran in time 2^{n^3}. This was improved to $n^{O(n)}$ by Kannan (Kan87). He presented a different type of basis reduction that, in $n^{O(n)}$ time and polynomial space, finds a shortest vector. Using techniques similar to those of Lenstra, his approach then yields $n^{O(n)}$ time algorithms for the closest and shortest vector problem in any norm. In terms of time and space complexity, the algorithm of Kannan essentially remains the state-of-the-art for the shortest and closest vector problem. For the shortest vector problem, it remains a wide open question whether there is a $2^{O(n)}$-time algorithm, that, using polynomial space only, computes a polynomial approximation to the shortest vector. On the other hand, it is a major open problem whether there is an $n^{o(n)}$-time algorithm for the (exact) closest vector problem, even when allowing exponential space and preprocessing of the lattice, (KPV12; DB15; HRS20).

The first single-exponential time algorithm for the shortest vector problem was given by Ajtai, Kumar and Sivakumar, (AKS01) and it computes a shortest vector in $2^{O(n)}$ time and space. Their approach is based on *randomized sieving* where an exponential number of (random) lattice vectors are generated and subtracted from each other to yield shorter and shorter lattice vectors. They later extended this approach to $(1+\varepsilon)$-CVP with a time and space complexity of $2^{O(1+1/\varepsilon)n}$, (AKS02). In a sequence of works, this approach was then extended to arbitrary norms with a time complexity of $O(1+1/\varepsilon)^n$, (BN09; EHN11; Dad12b).

Since the randomized sieving approach, two new singly-exponential time algorithms for lattice problems in the ℓ_2 have emerged.

The first such algorithm is from Micciancio and Voulgaris (MV10a) and is based on the Voronoi cell of the lattice. Their algorithm walks along the segment $[\mathbf{0}, t]$, always keeping track in which lattice vector's Voronoi cell it lies in, see Figure 1.4. Their algorithm is *deterministic* and computes an exact closest vector in the ℓ_2 norm in 2^{2n} time and 2^n space. Their algorithm was also the first to solve SVP_2 in deterministic, singly-exponential time (as there is a efficient reduction from SVP to CVP, (GMSS99)). Their algorithm has also been instrumental to give deterministic algorithms for SVP and $(1+\varepsilon)$-CVP in general norms, (DPV11; DK16).

The second such algorithm is based on Discrete Gaussian Sampling and solves SVP_2 and (exact) CVP_2 in time and space 2^n, (ADRS15; ADS15; AS18b). Their algorithm is randomized and can be seen as a version of randomized sieving: instead of taking pairwise differences between $\mathbf{v}_1$ and $\mathbf{v}_2$ when the resulting lattice vector $\mathbf{v}_1 - \mathbf{v}_2$ becomes shorter, they take the average $\frac{\mathbf{v}_1+\mathbf{v}_2}{2}$ whenever this lies in the lattice (which happens whenever $\mathbf{v}_1$ and $\mathbf{v}_2$ lie in the same coset modulo $2\cdot\mathscr{L}$). *In expectation*, the resulting lattice vector decreases by a factor of $\sqrt{2}$. This approach yields the currently fastest algorithm for SVP_2 and CVP_2 and even gives a $2^{n/2}$ algorithm for approximate *decisional* SVP_2 (we need to decide whether $\lambda_1(\mathscr{L}) \le \lambda$, or $\lambda_1(\mathscr{L}) > \alpha\cdot\lambda$ for some approximation factor $\alpha \ge 1$.)

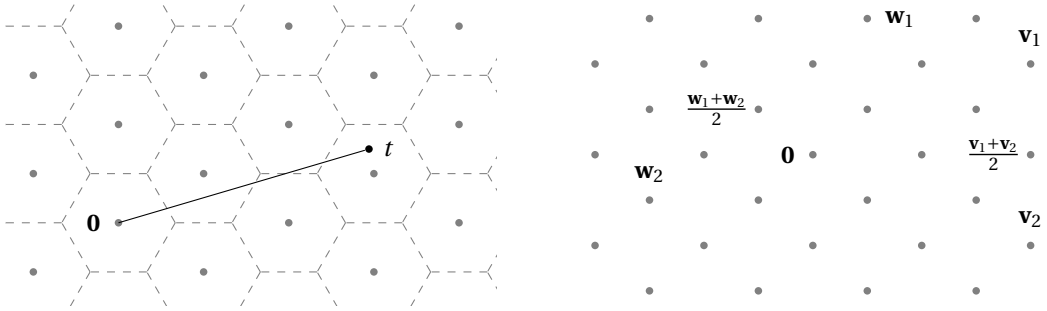

Figure 1.4 – Walking to t along the Voronoi cells and Discrete Gaussian Sampling

Both of these approaches are tailored to the Euclidean norm and do not carry over to any other norm. For general norms, only the randomized sieving approach or the enumeration technique (essentially resorting to the ℓ_2 norm) due to Dadush et al. (DPV11; DK16) seem available. For both of these approaches, the constants in the exponents of the running times are rather large. Even for some (large) constant approximation factors, these running times are nowhere close to 2^n, (AM18; Muk21). The only exception is again the ℓ_2 norm. Here, a careful analysis of the randomized sieving approach reveals that a constant factor approximation to the shortest vector problem in the ℓ_2 norm can be achieved in $2^{0.802n}$ time, (MV10b; PS09; LWXZ11). This result will be central as a subroutine for our improved algorithms in general norms, and we will review this approach in more detail in Chapter 3.

1.4 Complexity of Lattice Problems

The study of the complexity of SVP and CVP was initiated by van Emde Boas, (vEB81). He showed that CVP_p and SVP_∞ are NP-hard and raised the question if the same could be said about SVP_2. While the hardness of the shortest vector problem remained open for quite some time, subsequent work proved hardness of approximation for the closest vector problem up to almost-polynomial factors assuming P $\neq$ NP, (Aro95; DKRS03). Parallel to these results, it was shown that certain cryptographic schemes can be based on the worst-case hardness of (approximations of) certain lattice problems related to the shortest and closest vector problem, (Ajt96; Reg09; Gen09), renewing the interest in the complexity of lattice problems. Finally, SVP_2 was shown to be NP-hard which was soon thereafter extended to show hardness of approximation of SVP in any norm and for almost polynomial factors assuming RQP $\neq$ NP, (Ajt98; Mic01; Kho05; RR06; HR07).

Despite these impressive results on the hardness of approximations of lattice problems, the currently best polynomial-time algorithms only achieve exponential approximation factors (LLL82; Bab86; Sch87). This huge gap is further highlighted by the fact that these problems are in CO-NP and CO-AM for approximation factors of $\sqrt{n}$ and $\sqrt{n/\log(n)}$, respectively (GG00; AR05; Pei08), and are not believed to be NP-hard to approximate within small polynomial factors. On the other hand, the only algorithms for SVP and CVP that achieve

reasonable approximation factors do run in time $2^{\Omega(n)}$, (Kan87; AKS01; MV10a; ADRS15), and it is not clear whether there are fundamentally faster algorithms (say, of the order $2^{\sqrt{n}}$ or even $2^{0.1n}$) or if this is the end of the road. Clearly, the assumption P $\neq$ NP is too coarse as to give evidence to such questions.

Addressing this issue, recent work has focused on the *quantitative hardness* of exact and approximation for SVP and CVP, (BGS17; AS18a; ABGS21). This more *fine-grained* viewpoint on the hardness of lattice problems is based on the *(strong) exponential time hypothesis* (SETH), (IP01). Informally, the (strong) exponential time hypothesis states that for any $\varepsilon > 0$ there exists some $k(\varepsilon) \in \mathbb{N}$ such that there is no algorithm that solves $k(\varepsilon)$-SAT in time $2^{(1-\varepsilon)n}$. Assuming the SETH, they have shown that exact CVP_p, for $p \neq 0 \pmod 2$, and SVP_∞ cannot be solved in time $2^{(1-\varepsilon)n}$ for lattices of rank n.

Assuming the stronger GAP-SETH hypothesis asserting that *approximately* deciding the validity of a $k(\varepsilon)$-SAT formula takes $2^{(1-\varepsilon)n}$ time (we are allowed to violate a small fraction of clauses), see (Din16; MR17), these lower bounds also hold for the shortest vector and even for some small, but constant, approximation factors. In the setting of arbitrary norms, these results can be restated as follows. For any $\varepsilon > 0$, there exists some constant $\gamma_\varepsilon > 1$ only depending on ε, some norm $\|\cdot\|_K$ and some lattice $\mathscr{L} \subseteq \mathbb{Q}^n$ of rank *and* dimension n such that there is no $2^{(1-\varepsilon)n}$ time algorithm that solves γ_ε-CVP_K or γ_ε-SVP_K on $\mathscr{L}$.

While this strong lower bound holds for a wide range of norms, we emphasize that these hardness results do not hold for any norm. For the important case of the ℓ_2 norm, incidentally the *only* norm for which a (large) *constant* approximation to the shortest vector problem was known in better-than-2^n time, these strong lower bounds are not known to hold. Assuming the GAP-SETH, we can only rule out $2^{o(n)}$ time algorithms for c-approximate SVP_2 and CVP_2, for $c > 1$ some small constant.

2 Reductions across various Norms

This chapter presents *reductions* among the shortest and closest vector problem in various norms. Intuitively, a reduction from Problem A to Problem B is a procedure that provides a method to *solve* Problem A by making queries to an *oracle* for Problem B. This is a central theme in computational complexity. It gives a clean way to relate the hardness of different computational problems. For instance, if the number of queries to the oracle is restricted to a polynomial and Problem A is NP-complete, then this allows us to conclude that Problem B is no easier than Problem A, meaning Problem B is at least NP-complete, possibly even harder. However, such *polynomial-time reductions* say very little in the context of *exponential-time* problems. Indeed, if there is, say, a reduction from k-SAT on n variables to CVP on a lattice of rank n^3, assuming the SETH, this only allows to rule out $2^{o(n^{1/3})}$-time algorithms for CVP. Considering that the fastest approximation algorithms for CVP run in time $2^{O(n)}$ (or even $2^{O(d)}$ in the near exact setting, for any norm other than the ℓ_2 norm), such a hardness result is unsatisfactory. Hence, for fine-grained reductions involving lattice problems, the rank n and the ambient dimension d of the resulting lattices needs to be controlled. Ideally, the rank and dimension are preserved during the reduction (or only increase by some additive constant). This is a big restriction, but in this setting, it makes sense to allow for the reduction to take exponential time. Indeed, for lattice problems requiring $2^{\Omega(n)}$ time under various assumptions, it certainly makes sense to allow for an extra factor of the order, say, $2^{0.01n}$. In this chapter, we present three such exponential time reductions.

Section 2.2 provides a reduction from $O_\varepsilon(\alpha)$-SVP_Q to $2^{\varepsilon d}$ calls to an oracle for α-SVP_K, assuming Q and K follow a certain covering property, see Theorem 2.2.1. This reductions preserves the rank and the dimension of the lattice. As a consequence, this yields a reduction from approximate SVP_q to approximate SVP_p, for $p \le q$. For the case of ℓ_p norms, the underlying geometric ideas related to coverings were first introduced in an algorithm in (EV22), and were subsequently turned into a randomized reduction in (ACK+21).

Section 2.3 provides a reduction from $O_\varepsilon(\alpha)$-CVP_Q to $2^{\varepsilon d}$ calls to an oracle for α-approximate CVP_K, assuming Q and K follow the covering property from the previous reduction *in reverse*, see Theorem 2.3.1. The reduction is randomized and preserves the rank and the dimension of the lattice. In particular, this yields a randomized reduction from approximate CVP_p to CVP_q,

for $p \leq q$. The latter first appeared in (EV22) (somewhat implicitly for $q > 2$), and was later turned into a *deterministic* reduction in (ACK+21).

Section 2.4 provides a reduction from $O_\varepsilon(\alpha)$-SVP$_Q$ to $2^{\varepsilon n}$ calls to an oracle for α-CVP$_K$. This reduction works for *any* pair of norms Q and K and consequently, the rank of the lattice is preserved and the dimension d can be taken equal to the rank n, see Theorem 2.4.2. Using the currently fastest algorithm for approximate CVP with respect to the ℓ_2 norm, see Theorem 3.1.5, this reduction yields a $2^{0.802n}$ time algorithm for approximate SVP in any norm. This reduction is from (RV22) and is based on the following covering argument. For any pair of convex bodies, $Q, K \subseteq \mathbb{R}^n$, one can compute a linear transformation $T : \mathbb{R}^n \to \mathbb{R}^n$, such that Q can be covered by fewer than $2^{\varepsilon n}$ translates of $T(K)$ and vice versa. The existence of such a transformation follows from (regular) M-ellipsoids, a key concept in high-dimensional convex geometry. The construction of such a linear transformation is deferred to Chapter 4.

The first and third reduction rely on *lattice sparsification* and are randomized. Given a lattice $\mathscr{L}$ and some prime number p as a parameter, lattice sparsification deletes in an *almost uniform* manner a $1 - 1/p$ fraction of the lattice. This technique was first developed by Khot to show (quasi-) NP-hardness of approximating the shortest vector problem to within almost polynomial factors, (Kho05), and has since become an indispensable tool in the study of lattice problems. Section 2.1 provides a high-level overview on lattice sparsification.

2.1 Lattice Sparsification

Let $\mathscr{L}(\mathbf{B})$, $\mathbf{B} \in \mathbb{Q}^{d \times n}$, be the lattice under consideration and fix some prime number p. We sample a nonzero vector $a \sim \{0, 1, \ldots, p-1\}^n$ uniformly at random and we consider the sublattice $\mathscr{L}' \subseteq \mathscr{L}$ given by

$$\mathscr{L}' := \left\{ \mathbf{B} \cdot z \mid z \in \mathbb{Z}^n \text{ and } a^T \cdot z = 0 \pmod{p} \right\}.$$

This is illustrated in Figure 2.1, with $a \in \mathbb{Z}_p^n$ selected deterministically for aesthetic reasons.

Figure 2.1 – Left: $\mathscr{L}$ spanned by $\binom{3}{1}$ and $\binom{0}{3}$. Right: $\mathscr{L}$ sparsified with $a = \binom{1}{2}$ and $p = 5$.

Since $p \cdot \mathscr{L} \subseteq \mathscr{L}'$, the sparsified lattice $\mathscr{L}'$ is of full rank n. Its basis can be computed in polynomial time. Up to switching the roles of a_1 with that of a nonzero a_i, $i \in \{2, \ldots, n\}$, we

define

$$S := \begin{bmatrix} p & -a_2 & -a_3 & \cdots & -a_n \\ & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \end{bmatrix}.$$

The basis of $\mathscr{L}'$ is then given by $\mathbf{B} \cdot S$. Indeed, it easily checked that any $\bar{z} \in \mathbb{Z}^n$ with $a^T \cdot \bar{z} = 0 \pmod{p}$ belongs to the lattice spanned by S, i.e. $\bar{z} = S \cdot y$, $y \in \mathbb{Z}^n$.

We can now discuss the effect of sparsification on $\mathscr{L}$. For this, let $\mathbf{v}_1, \mathbf{v}_2$ be any two lattice vectors with $\lambda_1, \lambda_2 \in \mathbb{Z}^n$ as their respective coefficients, i.e. $\mathbf{v}_1 = \mathbf{B} \cdot \lambda_1$ and $\mathbf{v}_2 = \mathbf{B} \cdot \lambda_2$. Now, unless $\mathbf{v}_1 \in p \cdot \mathscr{L}$ (or $\lambda_1 \in p \cdot \mathbb{Z}^n$), the probability that $\mathbf{v}_1$ belongs to the sparsified sublattice $\mathscr{L}'$ equals exactly $\frac{1}{p}$. Here, the probability is over the randomness of $a \sim \{0, 1, \ldots, p-1\}^n$. Whenever λ_2 is a multiple of λ_1 (i.e. there is a line passing through $\mathbf{0}$, $\mathbf{v}_1$ and $\mathbf{v}_2$) or $\lambda_2 - \lambda_1 \in p \cdot \mathbb{Z}^n$, then $\mathbf{v}_1 \in \mathscr{L}'$ implies that $\mathbf{v}_2 \in \mathscr{L}'$ and vice versa. For all other cases, the event that $\mathbf{v}_1$ belongs to the sparsified lattice is independent of $\mathbf{v}_2$ belonging to the sparsified lattice. This is formalized in the following theorem.

Theorem 2.1.1 ((Kho05; Ste16))**.** *Let p be any prime, $\mathscr{L}$ any lattice and fix any lattice vectors $\mathbf{w}, \mathbf{v}_1, \ldots, \mathbf{v}_N \in \mathscr{L} \setminus \{\mathbf{0}\}$ such that $\mathbf{v}_i \notin \mathbf{w} + p \cdot \mathscr{L}$ and $\mathbf{v}_i \neq \alpha \cdot \mathbf{w}$, $\alpha \in \mathbb{R}$ for all $i \in \{1, \ldots, N\}$. Then, in polynomial time, one can sample a random sublattice $p \cdot \mathscr{L} \subseteq \mathscr{L}' \subseteq \mathscr{L}$ such that*

$$\Pr[\mathbf{w} \in \mathscr{L}' \text{ and } \mathbf{v}_1, \ldots, \mathbf{v}_N \notin \mathscr{L}'] \geq \frac{1}{p} - \frac{N}{p^2}.$$

The vector $\mathbf{0}$ always belongs to the sparsified lattice $\mathscr{L}'$. This is problematic when working with an oracle for CVP which then might, on any (sparsified) sublattice, have the option to return $\mathbf{0}$. Indeed, this issue arises in Section 2.4 from this chapter. This can be remedied by sampling a *shifted* sublattice. This shifted sublattice will be of the form $\mathbf{u} + \mathscr{L}'$, where $\mathbf{u} \in \mathscr{L}$ and $\mathscr{L}' \subseteq \mathscr{L}$. We proceed similarly to the preceding sparsification. For some prime number p, we sample some nonzero $a \sim \{0, 1, \ldots, p-1\}^n$ *and* $c \sim \{0, 1, \ldots, p-1\}$. We then consider the following shifted lattice,

$$\mathscr{L}' := \left\{ \mathbf{B} \cdot z \mid z \in \mathbb{Z}^n \text{ and } a^T \cdot z = c \pmod{p} \right\}.$$

This is illustrated in Figure 2.2. To see that this is indeed a shifted sublattice of $\mathscr{L}$, we can calculate $\bar{z} \in \mathbb{Z}_p^n$ with $a^T \bar{z} = -c \pmod{p}$. Since p is prime, this equation has a solution and it can be found with the Extended Euclidean Algorithm. For $\mathbf{u} := \mathbf{B} \cdot \bar{z}$, the shifted sublattice is then given by

$$\mathbf{u} + \left\{ \mathbf{B} \cdot z \mid z \in \mathbb{Z}^n \text{ and } a^T \cdot z = 0 \pmod{p} \right\}.$$

Theorem 2.1.2 ((Ste16))**.** *Let p be any prime, $\mathscr{L}$ be any lattice of rank n and fix any lattice vectors $\mathbf{w}, \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N \in \mathscr{L}$ with $\mathbf{v}_i \notin \mathbf{w} + p \cdot \mathscr{L} \setminus \{\mathbf{0}\}$. Then, in polynomial time, one can sample*

Figure 2.2 – Left: $\mathscr{L}$ spanned by $\binom{3}{1}$ and $\binom{0}{3}$. Right: $\mathscr{L}$ sparsified with $a = \binom{1}{2}$, $c = 1$ and $p = 5$.

a shifted sublattice of the form $\mathbf{u} + \mathscr{L}'$ *where* $p \cdot \mathscr{L} \subseteq \mathscr{L}' \subseteq \mathscr{L}$ *and* $\mathbf{u} \in \mathscr{L}$ *such that:*

$$\Pr\left[\mathbf{w} \in \mathbf{u} + \mathscr{L}' \text{ and } \mathbf{v}_1, \ldots, \mathbf{v}_N \notin \mathbf{u} + \mathscr{L}'\right] \geq \frac{1}{p} - \frac{N}{p^2} - \frac{N}{p^{n-1}}.$$

2.2 Self-Reduction of the Shortest Vector Problem in various Norms

We now outline our first geometric idea in the setting of an algorithm for the shortest vector problem with respect to ℓ_∞. Let us denote by $\mathbf{s}$ a shortest lattice vector and assume the full-rank lattice $\mathscr{L} \subseteq \mathbb{R}^n$ is scaled such that $\|\mathbf{s}\|_\infty = 1$. The Euclidean norm of $\mathbf{s}$, $\|\mathbf{s}\|_2$, is then bounded by $\sqrt{n}$. Imagine now there is a procedure (for instance Theorem 3.1.1) that allows us to sample *distinct* lattice vectors $\mathbf{v}_1, \ldots, \mathbf{v}_N \in \mathscr{L}$ that are short with respect to ℓ_2, e.g. $\|\mathbf{v}_i\|_2 \leq \alpha \cdot \sqrt{n}$ for all $i \in \{1, \ldots, N\}$.

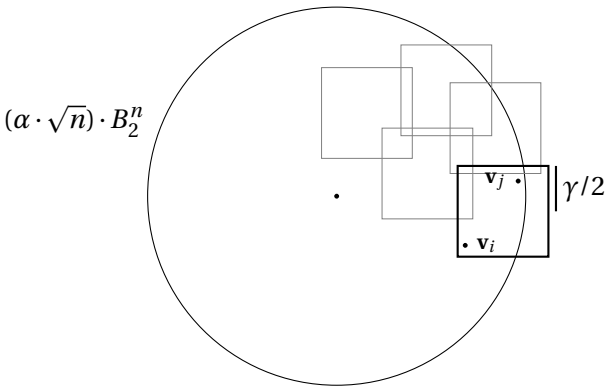

Figure 2.3 – The difference $\mathbf{v}_j - \mathbf{v}_i$ is a γ-approximation to SVP_∞.

How many distinct lattice vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ are needed to guarantee that there are two of them, $\mathbf{v}_i \neq \mathbf{v}_j$, with

$$\|\mathbf{v}_j - \mathbf{v}_i\|_\infty \leq \gamma?$$

Here, γ is the approximation guarantee we want to achieve, see Figure 2.3.

Suppose that N is strictly larger than the number of boxes of side length γ required to cover $\alpha\sqrt{n} \cdot B_2^n$. Then, by the pigeon-hole principle, there must exist two lattice vectors $\mathbf{v}_i, \mathbf{v}_j$ lying in

the same box. Hence, their difference is a γ-approximation to the shortest vector (with respect to ℓ_∞).

Thus, we are interested in the *translative covering number* $N(K,L)$, which is the least number of translates of L that are required to cover K. Formally,

$$N(K,L) := \min\left|\left\{S \subseteq \mathbb{R}^n \mid K \subseteq \bigcup_{t\in S} t+L\right\}\right|.$$

In the setting as described above, whenever the number of distinct lattice vectors exceeds $N(\alpha\sqrt{n}\cdot B_2^n, \gamma/2\cdot B_\infty^n)$, we are guaranteed that two of them must be closer than γ with respect to ℓ_∞, hence their difference is a γ-approximation to the shortest vector. For γ_ε some constant (only depending on ε), $N(\alpha\sqrt{n}\cdot B_2^n, (\alpha\cdot\gamma_\varepsilon/2)\cdot B_\infty^n) \le 2^{\varepsilon n}$. Hence, generating $2^{\varepsilon n}+1$ distinct lattice vectors that are shorter than $\alpha\sqrt{n}\cdot B_2^n$ in the ℓ_2 norm, by taking pairwise differences, we are guaranteed to find an $(\alpha\cdot\gamma_\varepsilon)$-approximation to the shortest vector with respect to ℓ_∞.

This can be generalized to arbitrary pairs of norms. An unfortunate limitation of this technique is the necessity to distinguish between the rank n and dimension d of the lattice. We discuss this at the end of this chapter.

Theorem 2.2.1. *Let $K, Q \subseteq R^d$ two origin-symmetric convex bodies such that $Q \subseteq K$ and $N(K, \gamma_\varepsilon\cdot Q) \le 2^{\varepsilon d}$ for some $\gamma_\varepsilon > 0$. Then, there is a reduction from $(2\cdot\gamma_\varepsilon\cdot\alpha)$-approximate SVP_Q on any rank n lattice to $2^{\varepsilon d}$ calls to an oracle for α-approximate SVP_K on rank n lattices. The reduction is randomized and requires polynomial space.*

Before we begin with the proof of the theorem, we need a small geometric lemma.

Lemma 2.2.2. *Let $Q \subseteq K \subseteq \mathbb{R}^n$ be two origin-symmetric convex bodies such that $N(K,Q) \le M$. Then $\|\cdot\|_K \le \|\cdot\|_Q \le M\cdot\|\cdot\|_K$.*

Proof. Since $Q \subseteq K$, the first inequality is clear. For the second inequality we will show that $K \subseteq M\cdot Q$: Assume by contradiction that there exists $x, -x \in K$ such that $x, -x \notin M\cdot Q$. Let $t \in \mathbb{R}^n$ be such that the length of $(t+Q)\cap[-x,x]$ is maximal. By symmetry, this equals the length of $(-t+Q)\cap[-x,x]$, and thus, by convexity, we can assume that $t = 0$. Hence, the maximal length that a translate of Q covers of the segment of $[-x,x]$ equals $|Q\cap[-x,x]|$. Since $x, -x \notin M\cdot Q$, it is impossible to cover the segment $[-x,x]$ by fewer than $M+1$ translates of Q, contradicting the fact that $N(K,Q) \le M$. □

Proof of Theorem 2.2.1. Let us denote by $\mathbf{s}$ a shortest lattice vector with respect to $\|\cdot\|_Q$. We will use the sparsification procedure from Theorem 2.1.1. Recall that given any lattice vectors $\mathbf{w}, \mathbf{v}_1, \ldots, \mathbf{v}_N \in \mathscr{L}$, this procedure returns a sublattice $p\cdot\mathscr{L} \subseteq \mathscr{L}' \subseteq \mathscr{L}$ such that

$$\Pr[\mathbf{w} \in \mathscr{L}' \text{ and } \mathbf{v}_1, \ldots, \mathbf{v}_N \notin \mathscr{L}'] \ge \frac{1}{p} - \frac{N}{p^2},$$

unless $\mathbf{v}_i \in \mathbf{w} + p \cdot \mathscr{L}$, $\mathbf{v}_i \in p \cdot \mathscr{L}$, or $\mathbf{w} = \alpha \cdot \mathbf{v}_i$, for some $i \in \{1, \dots, N\}$ and $\alpha \in \mathbb{R}$.

The full reduction is described in Figure 2.4. For simplicity, we prove correctness for a slightly modified reduction requiring $2^{\varepsilon d}$ space. We fix a prime p with $2^{2\varepsilon d} \leq p \leq 2 \cdot 2^{2\varepsilon d}$ and compute a list L of lattice vectors in the following way: Using the sparsification procedure from Theorem 2.1.1 and p, we sample a sublattice $\mathscr{L}' \subseteq \mathscr{L}$ and call the oracle for α-approximate SVP_K on $\mathscr{L}'$. We store the resulting lattice vector in the list L. We repeat, but we only ever store the $2^{\varepsilon d} + 1$ shortest, pairwise distinct lattice vectors. Let us see how many iterations are necessary as to guarantee that with high probability, our final list contains at least $2^{\varepsilon d} + 1$ distinct lattice vectors shorter than $\alpha \cdot \|\mathbf{s}\|_K$ (or one vector being a multiple of $\mathbf{s}$.). At time 0, i.e. when the list is empty, the probability that $\mathbf{s}$ is contained in $\mathscr{L}' \subseteq \mathscr{L}$ equals $2^{-\varepsilon d}$. Hence, conditional on $\mathbf{s} \in \mathscr{L}'$ and since $Q \subseteq K$, the oracle must return a lattice vector $\mathbf{v}$ shorter than $\alpha \cdot \|\mathbf{s}\|_K$. Let us now assume that our list consists of distinct lattice vectors $\{\mathbf{v}_1, \dots, \mathbf{v}_m\}$ shorter than $\alpha \cdot \|\mathbf{s}\|_K$. When we sparsify and call the SVP_K oracle, by Theorem 2.1.1, the probability that $\mathbf{s} \in \mathscr{L}'$ and $\{\mathbf{v}_1, \dots, \mathbf{v}_m\} \notin \mathscr{L}'$ is at least $\frac{1}{p} - \frac{m}{p^2}$, unless $\mathbf{v}_i \in \mathbf{s} + p \cdot \mathscr{L}$, $\mathbf{v}_i \in p \cdot \mathscr{L}$ or $\mathbf{v}_i = \alpha \cdot \mathbf{s}$ ($\alpha \in \mathbb{Z}$) for some $i \in \{1, \dots, m\}$. Hence, to add a new vector to our list, we need to exclude these three possibilities.

Clearly, $\mathbf{v} \notin p \cdot \mathscr{L}$ since otherwise $\|\mathbf{v}\|_K \geq p \cdot \|\mathbf{v}/p\|_K \geq p \cdot \lambda_1^{(K)}(\mathscr{L}) \gg \alpha \cdot \lambda_1^{(K)}(\mathscr{L})$. Here, we are assuming $\alpha = 2^{o(d)} = o(p)$. This is without loss of generality. Whenever $\alpha = 2^{\Omega(d \log\log(d)/\log(d))}$, we can use the LLL-algorithm and its variants running in polynomial time.
Whenever $\mathbf{v}$ is a multiple of $\mathbf{s}$, we are done anyways: if $\mathbf{v} = \beta \cdot \tilde{\mathbf{v}}$, $\beta \in \mathbb{N}$, consider $\tilde{\mathbf{v}}$ instead of $\mathbf{v}$. The scalar β can be found with the Extended Euclidean algorithm.
Finally, by Lemma 2.2.2, we can also exclude the possibility that $\mathbf{v} \in \mathbf{s} + p \cdot \mathscr{L} \setminus \{\mathbf{0}\}$. Indeed, by the triangle inequality, if $\mathbf{v} := p \cdot \mathbf{u} + \mathbf{s}$, $\mathbf{u} \in \mathscr{L} \setminus \{\mathbf{0}\}$,

$$\begin{aligned} \|\mathbf{v}\|_K &\geq \|p \cdot \mathbf{u}\|_K - \|\mathbf{s}\|_K \\ &\geq (\tfrac{p}{\gamma_\varepsilon^2 \cdot M} - 1) \cdot \|\mathbf{s}\|_K. \end{aligned}$$

In the last line, we have used that $\frac{1}{\gamma_\varepsilon} \cdot \|\cdot\|_K \leq \|\cdot\|_Q \leq (M \cdot \gamma_\varepsilon) \cdot \|\cdot\|_K$, this follows from Lemma 2.2.2. Hence, since $p \approx 2^{2\varepsilon d}$, $M \leq 2^{\varepsilon d}$ and γ_ε some constant, assuming that $\alpha = 2^{o(n)}$, such a vector is too large and cannot be returned by an oracle for α-approximate SVP_K.

It follows that at any given time, i.e. when the list L contains lattice vectors $\mathbf{v}_1, \dots, \mathbf{v}_m$, the probability that we can add a new and distinct (from vectors in L) lattice vector to the list L (or find $\mathbf{s}$) is lower bounded by $2^{-\varepsilon d} - m \cdot 2^{-2\varepsilon d}$. Hence, with probability $1 - 2^{-d}$ and after $2^{3\varepsilon d}$ iterations, the list L contains $2^{\varepsilon d} + 1$ distinct lattice vectors of $\|\cdot\|_K$-norm at most $\alpha \cdot \|\mathbf{s}\|_K$ or contains (a multiple of) $\mathbf{s}$. In the latter case we are done, so suppose not. Since

$$N(K, \gamma_\varepsilon \cdot Q) = N((\alpha \cdot \|\mathbf{s}\|_K) \cdot K, (\alpha \cdot \gamma_\varepsilon \cdot \|\mathbf{s}\|_K) \cdot Q) \leq 2^{\varepsilon d},$$

there must be two lattice vectors $\mathbf{v}_i, \mathbf{v}_j \in \mathrm{L}$ such that

$$\mathbf{v}_i - \mathbf{v}_j \in (2 \cdot \gamma_\varepsilon \cdot \alpha \cdot \|\mathbf{s}\|_K) \cdot Q \subseteq (2 \cdot \gamma_\varepsilon \cdot \alpha \cdot \|\mathbf{s}\|_Q) \cdot Q.$$

This means that $\mathbf{v} := \mathbf{v}_i - \mathbf{v}_j$ is a $(2 \cdot \gamma_\varepsilon \cdot \alpha)$-approximation to $\mathbf{s}$ with respect to $\|\cdot\|_Q$.

As described, this reduction takes time and space $2^{3\varepsilon d}$. To bring the latter down to a polynomial, we sample two random numbers $i, j \in \{1, 2, \ldots, 2^{3\varepsilon d}\}$ and only store the lattice vectors returned in the i^{th} and j^{th} iteration. The numbers i and j are not disclosed to the oracle. Hence, with probability at least $2^{-6\varepsilon d}$, we pick the right pair (or the right index if a multiple of $\mathbf{s}$ is sampled) and this modified procedure succeeds. The overall probability of success can then be boosted to $1 - 2^{-d}$ by repeating $2^{7\varepsilon d}$ times. □

Input: $\mathscr{L} \subseteq \mathbb{R}^d$, $\varepsilon > 0$, $\|\cdot\|_Q, \|\cdot\|_K : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$
Initialize $\mathbf{v}$ as any nonzero lattice vector;
Compute a prime $p \in \mathbb{N}$ with $2^{2\varepsilon d} \leq p \leq 2 \cdot 2^{2\varepsilon d}$;
for $\ell \in \{1, \ldots, 2^{7\varepsilon d}\}$ **do**
  for $k \in \{1, \ldots, 2^{3\varepsilon d}\}$ **do**
    Sample $i, j \in \{1, 2, \ldots, 2^{\varepsilon d}\}$;
    Sample sparsified lattice $\mathscr{L}' \subseteq \mathscr{L}$ by Theorem 2.1.1 and p ;
    Set $\mathbf{u}$ the lattice vector returned by oracle for γ-SVP$_K$ on $\mathscr{L}'$;
    if $k = i$ *or* $k = j$ **then**
      $\mathbf{v}_k \leftarrow \mathbf{u}$;
    else
      Delete $\mathbf{u}$;
    end
  end
  $\mathbf{v} \leftarrow \arg\min_{k_1, k_2 \in \mathbb{Z} : \mathbf{v}_1/k_1, \mathbf{v}_2/k_2 \in \mathscr{L}} \{\|\mathbf{v}\|_Q, \|\mathbf{v}_i/k_1\|_Q, \|\mathbf{v}_j/k_2\|_Q, \|\mathbf{v}_i - \mathbf{v}_j\|_Q\}$;
end
Output: $\mathbf{v}$

Figure 2.4 – Reducing approximate SVP$_Q$ to approximate SVP$_K$.

The following corollary of Theorem 2.4 first appeared in (ACK+21) and combined the geometric covering ideas from (EV22) together with lattice sparsification.

Corollary 2.2.3 ((ACK+21))**.** *For any $p \leq q$, any lattice $\mathscr{L} \subseteq \mathbb{R}^d$ of rank n and any $\varepsilon > 0$, there is a randomized, $2^{\varepsilon d}$ time reduction from $(\alpha \cdot \gamma_\varepsilon)$-SVP$_q$ to α-SVP$_p$. γ_ε is a constant depending only on ε.*

Proof. We use Theorem 2.4.2 with $Q := B_q^d$ and $K := d^{1/p-1/q} \cdot B_p^d$. We need to check two things:

$$B_q^d \subseteq d^{1/p-1/q} \cdot B_p^d,$$

and

$$N(d^{1/p-1/q} \cdot B_p^d, \tfrac{\gamma_\varepsilon}{2} \cdot B_q^d) \leq 2^{\varepsilon d},$$

for some $\gamma_\varepsilon > 0$ only depending on ε.

The first inclusion follows from Hölder's inequality. For $1 \le p \le q \le +\infty$, we have that

$$\left(\sum_{k=1}^{d} |x_k|^p\right)^{1/p} \le \left(\sum_{k=1}^{d} |x_k|^q\right)^{1/q} \cdot d^{1/p-1/q}.$$

Whenever $q = \infty$, $1/q$ is defined as 0 and $\left(\sum_{k=1}^{d} |x_k|^q\right)^{1/q} := \max_{k=1,\dots,d} |x_k|$. It follows that

$$\forall x \in \mathbb{R}^d : \|x\|_p \le \|x\|_q \cdot d^{1/p-1/q}.$$

Hence, for any point $x \in B_q^d$, we have that $\|x\|_p \le d^{1/p-1/q}$ and (2.2) follows.

For the estimate on the covering number, we simplify matters by only considering B_1^∞ and B_∞^n. Indeed, by Hölder's inequality,

$$d^{-1/q} \cdot B_\infty^d \subseteq B_q^d \quad \text{and} \quad d^{1/p-1/q} \cdot B_p^d \subseteq d^{1-1/q} \cdot B_1^d. \tag{2.1}$$

Hence,

$$N(d^{1/p-1/q} \cdot B_p^d, \gamma \cdot B_q^d) \le N(d^{1-1/p} \cdot B_1^d, \gamma \cdot d^{-1/q} \cdot B_\infty^d) = N(d \cdot B_1^d, \gamma \cdot B_\infty^d),$$

and it is sufficient to find some $\gamma_\varepsilon > 0$ such that

$$N(d \cdot B_1^d, \tfrac{\gamma_\varepsilon}{2} \cdot B_\infty^d) \le 2^{\varepsilon d}. \tag{2.2}$$

There are several ways to show this. In (EV22), this was proven through volume estimates, passing through the ℓ_2 using *intrinsic volumes*. For any $\varepsilon > 0$, there exists some $\tilde{\gamma}_{\varepsilon/2}$ such that

$$N(d \cdot B_1^d, \tfrac{\tilde{\gamma}_{\varepsilon/2}}{2} \cdot \sqrt{d} \cdot B_2^d),\ N(\sqrt{d} B_2^n, \tfrac{\tilde{\gamma}_{\varepsilon/2}}{2} \cdot B_\infty^d) \le 2^{(\varepsilon/2)\, d}.$$

This then directly yields (2.2) with $\gamma_\varepsilon := \tilde{\gamma}_{\varepsilon/2}^2/2$, by observing that for any convex bodies A, B, C,

$$N(A, C) \le N(A, B) \cdot N(B, C).$$

This was improved in (ACK+21) using *lattice point counting*. Their approach yields a much better dependency of the approximation guarantee $\gamma_{\alpha,\varepsilon}$ with respect to ε, they show that γ_ε can be taken of the order of $\widetilde{O}(\varepsilon^{-1/p})$ for the reduction from $(\alpha \cdot \gamma_\varepsilon)$-SVP$_q$ to α-SVP$_p$, $p \le q$, as well as for the reduction from $(\alpha \cdot \gamma_\varepsilon)$-CVP$_p$ to α-CVP$_q$, $p \le q$, in the next section.

For completeness and nostalgic reasons, we include an elementary proof of (2.2) using volume estimates. It can be found in Lemma 4.1.2 in Chapter 4. □

In view of the next reductions in this chapter, we would like to note two limitations of Theorem 2.2.1:

First, the reduction is only efficient (with respect to $2^{O(n)}$ time algorithms) when $d = O(n)$.

This is since the translative covering number of two d-dimensional convex bodies typically depends exponentially on the dimension d. When we replace K and Q by $K \cap \mathrm{span}(\mathscr{L})$ and $Q \cap \mathrm{span}(\mathscr{L})$ respectively, it is not clear whether this can be made to depend on n instead of d. Even for ℓ_p norm balls, we do not know whether, say, inequality (2.2) implies $N(d \cdot B_1^d \cap \mathrm{span}(\mathscr{L}), \alpha_\varepsilon \cdot B_\infty^d \cap \mathrm{span}(\mathscr{L})) \leq 2^{\varepsilon n}$. Hence, already for ℓ_p norms and d of the order $\Omega(n \cdot \log(n))$, this approach breaks down.

Second, the approximation guarantee of the reduction depends on the number γ_ε depending on $\varepsilon > 0$ for which $N(K, \gamma_\varepsilon \cdot Q) \leq 2^{\varepsilon d}$ *and* $Q \subseteq K$. For some pair of norms, γ_ε depends on d and cannot be considered a constant for some fixed $\varepsilon > 0$. This is the case for $Q := B_1^d$ and $K = B_2^d$, i.e. we want to reduce approximate SVP_1 to (approximate) SVP_2 - Corollary 2.2.3 in reverse. Since $\mathrm{Vol}(B_2^d) \geq 2^{d \cdot \log(d)/2 + O(n)} \cdot \mathrm{Vol}(B_1^d)$, for any constant $\varepsilon \in \mathbb{R}_{>0}$, γ_ε will have to be of order $\Omega(\sqrt{d})$, (Joh48). Incidentally, in this case, we can reduce approximate CVP_1 to approximate CVP_2 in $2^{\varepsilon d}$ time.

2.3 Self-Reduction of the Closest Vector Problem in various Norms

Let us illustrate our reduction in the setting of the closest vector problem with respect to ℓ_1. We denote by t the target and by $\mathbf{c} \in \mathscr{L}$ the closest lattice vector to t. Finally, suppose that the full-rank lattice $\mathscr{L} \subseteq \mathbb{R}^n$ is scaled such that $\|t - \mathbf{c}\|_1 = 1$.

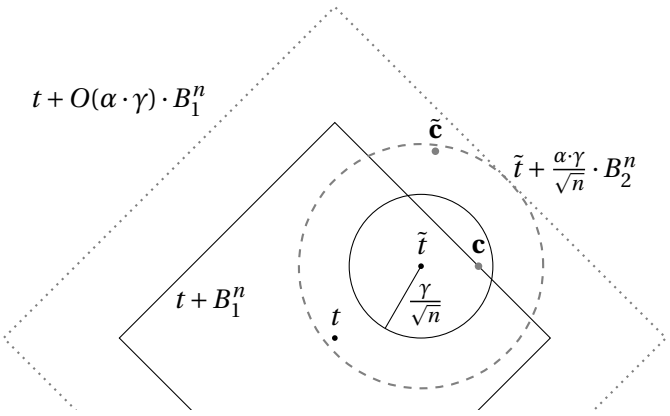

Figure 2.5 – Approximate CVP_1 through approximate CVP_2.

Suppose we have an algorithm for α-approximate CVP_2 at hand and we are so lucky as to know some point $\tilde{t} \in \mathbb{R}^n$ such that $\|\tilde{t} - \mathbf{c}\|_2 \leq \gamma/\sqrt{n}$. We claim that in this case, we can find an approximation to $\mathbf{c}$ using the algorithm for approximate CVP_2. Indeed, if we run the α-CVP_2 algorithm with target $\tilde{t}$, this will return some vector $\tilde{\mathbf{c}}$ with

$$\|\tilde{\mathbf{c}} - \tilde{t}\|_2 \leq \alpha \cdot \mathrm{dist}_{\ell_2}(\tilde{t}, \mathscr{L}) \leq \alpha \cdot \gamma/\sqrt{n}.$$

Since $\frac{1}{\sqrt{n}} \cdot B_2^n \subseteq B_1^n$, or, equivalently, $\|\cdot\|_1 \leq \sqrt{n} \cdot \|\cdot\|_2$, it follows that

$$\|\tilde{\mathbf{c}} - \tilde{t}\|_1 \leq \alpha \cdot \gamma \quad \text{and} \quad \|\tilde{t} - \mathbf{c}\|_1 \leq \gamma.$$

We can now invoke the triangle inequality to infer that $\tilde{\mathbf{c}}$ is a good to the closest vector.

$$\|t - \tilde{\mathbf{c}}\| \leq \|t - \mathbf{c}\|_1 + \|\mathbf{c} - \tilde{t}\|_1 + \|\tilde{t} - \tilde{\mathbf{c}}\|_1 \leq (1 + \gamma + \alpha \cdot \gamma) = O(\alpha \cdot \gamma).$$

This is illustrated in Figure 2.5.

In the situation as just described, it turns out that we do not need much luck, only $2^{\varepsilon n}$ time. For any $\varepsilon > 0$, there exists some $\gamma_\varepsilon > 0$ such that

$$\mathrm{Vol}(B_1^n + \tfrac{\gamma_\varepsilon}{\sqrt{n}} \cdot B_2^n) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(\tfrac{\gamma_\varepsilon}{\sqrt{n}} \cdot B_2^n). \tag{2.3}$$

We prove this below. Hence, for a *uniformly random* $\tilde{t} \sim B_1^n + \frac{\gamma_\varepsilon}{\sqrt{n}} \cdot B_2^n$, the probability that $\tilde{t}$ is such that $\mathbf{c} \in \tilde{t} + \frac{\gamma_\varepsilon}{\sqrt{n}} \cdot B_2^n$ is exactly the following ratio

$$\frac{\mathrm{Vol}(\frac{\gamma_\varepsilon}{\sqrt{n}} \cdot B_2^n)}{\mathrm{Vol}(B_1^n + \frac{\gamma_\varepsilon}{\sqrt{n}} \cdot B_2^n)}.$$

By the volume inequality (2.3), this is at least $2^{-\varepsilon n}$. Hence, if we repeat the above procedure for $2^{O(\varepsilon)n}$ randomly selected targets $\tilde{t} \sim t + B_1^n + \frac{\gamma_\varepsilon}{\sqrt{n}} \cdot B_2^n$, we will obtain an approximation to $\mathbf{c}$, the closest vector to t, with overwhelming probability.

This idea is generalised in the following theorem.

Theorem 2.3.1. *Let $Q, K \subseteq \mathbb{R}^d$ two symmetric convex bodies such that $K \subseteq Q$ and $\mathrm{Vol}(Q + \gamma_\varepsilon \cdot K) \leq 2^{\varepsilon d} \cdot \mathrm{Vol}(\gamma_\varepsilon \cdot K)$ for some $\gamma_\varepsilon > 0$. Then, for any $\varepsilon > 0$ and any lattice $\mathscr{L} \subseteq \mathbb{R}^d$, there is a reduction from $O(\alpha \cdot \gamma_\varepsilon)$-approximate CVP_Q on $\mathscr{L}$ to $2^{\varepsilon d}$ calls to an oracle for α-approximate CVP_K on $\mathscr{L}$. The reduction is randomized and requires polynomial space.*

Proof. We assume that the lattice is scaled so that $\|t - \mathbf{c}\|_Q$, the distance of the target to its closest lattice vector with respect to $\|\cdot\|_Q$, is between $1 - 1/n$ and 1. Up to estimating this distance by Babai's Nearest Plane Algorithm and binary search, the number of possibilities for this distance can be restricted to a polynomial and we can try out all of them.

We now describe an iteration of the reduction that succeeds with probability at least $2^{-\varepsilon d}$.

We begin by sampling a uniformly random point $\tilde{t} \sim t + Q + \gamma_\varepsilon \cdot K$. With probability at least $2^{-\varepsilon d}$, the point $\tilde{t}$ is such that $\mathbf{c} \in \tilde{t} + \gamma_\varepsilon \cdot K$. To see this, we observe that this holds whenever $\tilde{t} \in \mathbf{c} + \gamma_\varepsilon \cdot K$. Since this point is distributed uniformly in $t + Q + \gamma_\varepsilon \cdot K$ and since $\mathbf{c} + \gamma_\varepsilon \cdot K \subseteq t + Q + \gamma_\varepsilon \cdot K$, we have that

$$\Pr_{\tilde{t} \sim t + Q + \gamma_\varepsilon \cdot K}[\tilde{t} \in \mathbf{c} + \gamma_\varepsilon \cdot K] = \frac{\mathrm{Vol}(\gamma_\varepsilon \cdot K)}{\mathrm{Vol}(Q + \gamma_\varepsilon \cdot K)} \geq 2^{-\varepsilon d}.$$

Input: $\mathscr{L} \subseteq \mathbb{Q}^n$, $\varepsilon > 0$, $\gamma_\varepsilon > 0$, $\|\cdot\|_Q, \|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
Guess (enumerate) k such that $(1+1/n)^{k-1} \leq \|t-\mathbf{c}\|_Q < (1+1/n)^k$;
Initialize $\mathbf{w}$ to $\mathbf{0}$;
for $\ell \in \{1, \ldots, d \cdot 2^{2\varepsilon d}\}$ **do**
  Sample $\tilde{t} \sim t + (1+1/n)^k \cdot Q + \gamma_\varepsilon \cdot (1+1/n)^k \cdot K$ uniformly;
  Set $\mathbf{v}$ the lattice vector returned by the oracle for α-CVP$_K$ on $\tilde{t}$ and $\mathscr{L}$;
  if $\|t-\mathbf{v}\|_Q < \|t-\mathbf{w}\|_Q$ **then**
    $\mathbf{w} \leftarrow \mathbf{v}$;
  else
    Delete $\mathbf{v}$;
  end
end
Output: $\mathbf{w}$

Figure 2.6 – Reducing approximate CVP to approximate CVP.

We condition on this happening, i.e. $\tilde{t}$ being such that $\mathbf{c} \in \tilde{t} + \gamma_\varepsilon \cdot K$.

We now run the oracle for α-approximate CVP$_K$ with target $\tilde{t}$ and $\mathscr{L}$. Denote by $\tilde{\mathbf{c}}$ its output. By the triangle inequality,

$$\begin{aligned} \|t-\tilde{\mathbf{c}}\|_Q &\leq \|t-\mathbf{c}\|_Q + \|\mathbf{c}-\tilde{t}\|_Q + \|\tilde{t}-\tilde{\mathbf{c}}\|_Q \\ &\overset{(*)}{\leq} 1 + \|\mathbf{c}-\tilde{t}\|_K + \|\tilde{t}-\tilde{\mathbf{c}}\|_K \\ &\overset{(**)}{\leq} 1 + \gamma_\varepsilon + \alpha \cdot \gamma_\varepsilon. \end{aligned}$$

In $(*)$ we have used that $\|\cdot\|_Q \leq \|\cdot\|_K$ (since $K \subseteq Q$, by assumption). In $(**)$, we have used that $\tilde{t} \in \mathbf{c} + \gamma_\varepsilon \cdot K$ and hence, the α-approximate oracle for CVP$_K$ on target $\tilde{t}$ returns a lattice vector of distance no larger than $\alpha \cdot \gamma_\varepsilon$.

Since $\|t-\mathbf{c}\|_Q \geq 1 - 1/n$, this vector $\tilde{\mathbf{c}}$ is a $3 \cdot (1-1/n)^{-1} \cdot (\alpha \cdot \gamma)$-approximation to the closest vector.

One such iteration succeeds whenever $\tilde{t} \in \mathbf{c} + \gamma_\varepsilon \cdot K$. This happens with probability at least $2^{-\varepsilon d}$. To boost the overall probability of success to $1 - 2^{-d}$, we repeat this procedure $d \cdot 2^{\varepsilon d}$ times from where we sampled the random point $\tilde{t} + Q + \gamma_\varepsilon \cdot K$. The space requirement is polynomial since we only need to store the lattice vector that is currently the closest to the target t. □

The following corollary is very similar to Corollary 2.2.3, but the role of p and q are reversed.

Corollary 2.3.2 ((EV22; ACK+21)). *For any $p \leq q$, any lattice $\mathscr{L} \subseteq \mathbb{R}^d$ of rank n and any $\varepsilon > 0$, there is a randomized reduction from $(\alpha \cdot \gamma_\varepsilon)$-CVP$_p$ on $\mathscr{L}$ to $2^{\varepsilon d}$ calls to an oracle for α-CVP$_q$ on $\mathscr{L}$. The number γ_ε only depends on ε.*

Proof. For the correctness, we use Theorem 2.3.1 with $Q := B_p^d$ and $K := d^{-1/p+1/q} \cdot B_q^d$. By Hölder's inequality, see Inequality 2.1 from the previous section, it follows that $K \subseteq Q$.

To sample the target $\tilde{t}$, we proceed slightly different (though we still assume that $(1-1/n) \leq \|t-\mathbf{c}\|_Q \leq 1$). We set $\gamma_\varepsilon = \Omega(\varepsilon^{-2})$ from Lemma 4.1.2. We sample a random point $\tilde{t} \sim d^{1-1/p} \cdot B_1^d + \gamma_\varepsilon \cdot d^{-1/p} \cdot B_\infty^d$. We claim that with probability at least $2^{-\varepsilon d}$, this point $\tilde{t}$ is such that $\mathbf{c} \in \tilde{t}+K$. Indeed, by Hölder's inequality, we have the following inclusions,

$$B_p^d \subseteq d^{1-1/p} \cdot B_1^d \quad \text{and} \quad d^{-1/p} \cdot B_\infty^d \subseteq d^{-1/p+1/q} \cdot B_q^d.$$

Hence, if the vector $\tilde{t} \sim t + d^{1-1/p} \cdot B_1^n + \gamma_\varepsilon \cdot d^{-1/p} \cdot B_\infty^d$ lands inside $\mathbf{c} + \gamma_\varepsilon \cdot d^{-1/p} \cdot B_\infty$, then $\tilde{t} \in \mathbf{c} + \gamma_\varepsilon \cdot d^{-1/p+1/q} \cdot B_q^d$ as well. By Lemma 4.1.2,

$$\mathrm{Vol}(d^{1-1/p} \cdot B_1^d + \gamma_\varepsilon \cdot d^{-1/p} \cdot B_\infty^d) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(\gamma_\varepsilon \cdot d^{-1/p} \cdot B_\infty^d),$$

and the claim follows. □

2.4 Approximate Shortest Vectors in Any Norm Reduces to the Closest Vector Problem

We now show how for any pair of norms $\|\cdot\|_Q$ and $\|\cdot\|_K$, we can unconditionally reduce approximate SVP_Q to approximate CVP_K in $2^{\varepsilon n}$ time on any rank n lattice. This improves on the previous self-reductions as it removes the exponential dependence on d, the dimension of the lattice, and applies to any norm, not just certain pairs of ℓ_p norms. The drawback is that we reduce from the shortest vector problem to the closest vector problem, a problem that is potentially substantially harder than the shortest vector problem.

Our main idea will again involve coverings. As we will show, for any two convex bodies $Q, K \subseteq \mathbb{R}^n$ and any $\varepsilon > 0$, we can compute a linear transformation $T_\varepsilon : \mathbb{R}^n \to \mathbb{R}^n$ such that Q can be covered by fewer than $2^{\varepsilon n}$ translates of $c_\varepsilon \cdot T(K)$, and, conversely, $T_\varepsilon(K)$ can be covered by fewer than $2^{\varepsilon n}$ translates of $c_\varepsilon \cdot Q$. Here, c_ε is a constant only depending on ε but not on n. The respective coverings of Q by translates of $c_\varepsilon \cdot T(K)$ and of $T(K)$ by translates of $c_\varepsilon \cdot Q$ are constructive and can be computed with high probability in $2^{\varepsilon n}$ time. The exact properties of these coverings are made precise in Theorem 2.4.1 below. We first illustrate how we use this construction to reduce approximate SVP_Q to $2^{\varepsilon n}$ calls to an oracle for approximate CVP_2.

We assume that the lattice is scaled such that $\mathbf{s}$, the shortest lattice vector with respect to $\|\cdot\|_Q$, is of norm 1. For a given $\varepsilon > 0$, we compute the linear transformation $T_\varepsilon(\cdot)$ with the properties as described above, i.e. with $K = B_2^n$. Up to applying the linear transformation $T_\varepsilon^{-1}(\cdot)$ to Q and $\mathscr{L}$, we might as well assume that $T_\varepsilon(\cdot) = \mathbb{I}_n$, the identity. Now, compute a covering of Q by $2^{\varepsilon n}$ translates of $c_\varepsilon \cdot B_2^n$ with the properties as above. One of the $2^{\varepsilon n}$ translates of $c_\varepsilon \cdot B_2^n$ covering Q must hold $\mathbf{s}$. Hence, assume that $\mathbf{s} \in t + c_\varepsilon \cdot B_2^n$, or, in other words, $\|t-\mathbf{s}\|_2 \leq c_\varepsilon$.

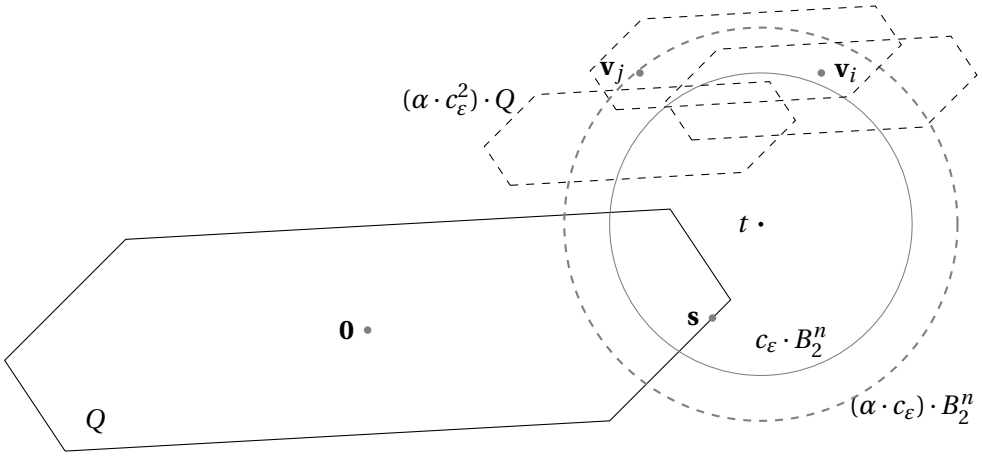

Figure 2.7 – The difference $\mathbf{v}_j - \mathbf{v}_i$ is a $(\alpha \cdot 2 \cdot c_\varepsilon^2)$-approximation to the shortest vector.

We can now use the oracle for α-approximate CVP_K with target t in combination with lattice sparsification to generate distinct vectors $\mathbf{v}_1, \ldots, \mathbf{v}_N \in \mathscr{L}$ such that $\|t - \mathbf{v}_i\|_2 \leq \alpha \cdot \|t - \mathbf{s}\|_2$. Specifically, when we already have generated (distinct) lattice vectors $\mathbf{v}_1, \ldots, \mathbf{v}_m$, we sparsify the lattice using the procedure from Theorem 2.1.2 with $p \approx 2^{\varepsilon n}$ and run the oracle for approximate CVP_2 on the resulting shifted sublattice. The reason for not applying the simpler sparsification procedure from Theorem 2.1.1 is that, if $\mathbf{0} \in t + \alpha \cdot c_\varepsilon \cdot B_2^n$, the oracle can always return $\mathbf{0}$. But in our setting and with probability roughly $2^{-\Omega(\varepsilon)n}$, $\mathbf{s} \in \mathscr{L}'$ but $\mathbf{v}_1, \ldots, \mathbf{v}_m \notin \mathscr{L}'$. In this case, the oracle *must* return a lattice vector inside $t + \alpha \cdot c_\varepsilon \cdot B_2^n$ that is different from $\mathbf{v}_1, \ldots, \mathbf{v}_m$. Repeating sufficiently many times, we can generate $N = 2^{\varepsilon n} + 1$ distinct lattice vectors inside $t + \alpha \cdot c_\varepsilon \cdot B_2^n$ (unless we find $\mathbf{s}$). Since $N(\alpha \cdot c_\varepsilon \cdot B_2^n, (\alpha \cdot c_\varepsilon^2) \cdot K) \leq 2^{\varepsilon n}$, there must be two distinct lattice vectors, $\mathbf{v}_i$ and $\mathbf{v}_j$ say, such that their pairwise difference must lie in some translate of $(\alpha \cdot c_\varepsilon^2) \cdot K$. Hence their pairwise difference $\mathbf{v}_j - \mathbf{v}_i$ is a $(\alpha \cdot 2 \cdot c_\varepsilon^2)$-approximation to the shortest vector with respect to $\|\cdot\|_K$. See Figure 2.7 for an illustration.

We now state our main geometric construction.

Theorem 2.4.1. *For any $\varepsilon > 0$, and for any two symmetric convex bodies $Q, K \subseteq \mathbb{R}^n$, there exists an invertible linear transformation $T_\varepsilon(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ and some constant $c_\varepsilon > 0$ only depending on ε such that*

$$\mathrm{Vol}(Q + c_\varepsilon \cdot T_\varepsilon(K)) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(c_\varepsilon \cdot T_\varepsilon(K)) \tag{2.4}$$

and

$$\mathrm{Vol}(T_\varepsilon(K) + c_\varepsilon \cdot Q) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(c_\varepsilon \cdot Q). \tag{2.5}$$

In particular, $N(Q, c_\varepsilon \cdot T_\varepsilon(K)), N(T_\varepsilon(K), c_\varepsilon \cdot Q) \leq 2^{\varepsilon n}$.
The linear transformation $T_\varepsilon(\cdot)$ can be computed in $n^{O(\log(n))}$ time.

We defer the proof of Theorem 2.4.1 to Chapter 4.

We note that the volume estimate $\mathrm{Vol}(Q + c_\varepsilon T_\varepsilon(K)) \le 2^{\varepsilon n} \cdot \mathrm{Vol}(c_\varepsilon \cdot T(K))$ in (2.4) implies that $N(Q, c_\varepsilon \cdot T_\varepsilon(K)) \le 2^{\varepsilon n}$. It makes the covering of $T_\varepsilon(K)$ by translates of B_2^n constructive. Indeed, any point inside Q will be covered with probability at least $2^{-\varepsilon n}$ if we sample a random point within $Q + c_\varepsilon \cdot T_\varepsilon(K)$ and place a copy of $c_\varepsilon \cdot T_\varepsilon(K)$ around it. Repeating this for $O(n^2 \cdot 2^{\varepsilon n})$ iterations yields, with high probability, a full covering of Q by translates of $c_\varepsilon \cdot T_\varepsilon(K)$, see (Nas14). Similarly for the volume inequality (2.5). For convenience, we prove this in Lemma 4.1.1 in the last chapter.

We can now leverage Theorem 2.4.1 to show that for any two norms $\|\cdot\|_K, \|\cdot\|_Q$ and any rank n lattice, approximate SVP_Q reduces to an oracle for approximate CVP_K.

Theorem 2.4.2. *Let $\|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\ge 0}, \|\cdot\|_Q : \mathbb{R}^n \to \mathbb{R}_{\ge 0}$ be any two norms. For any $\varepsilon > 0$, there is a constant γ_ε, such that there is a reduction from $(\alpha \cdot \gamma_\varepsilon)$-approximate SVP_K on any lattice of rank n to $2^{\varepsilon n}$ calls to an oracle for α-approximate CVP_Q on lattices of rank and dimension n. The reduction is randomized and requires polynomial space.*

The reduction is formally described in Figure 2.8.

Input: $\mathscr{L} \subseteq \mathbb{Q}^n$, $\varepsilon > 0$, $\|\cdot\|_Q, \|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\ge 0}$
Guess (enumerate) k such that $(1+1/n)^{k-1} \le \lambda_1^{(Q)}(\mathscr{L}) < (1+1/n)^k$;
Compute $T_\varepsilon(\cdot)$ from Theorem 2.4.1;
Initialize $\mathbf{w}$ as any nonzero lattice vector;
Compute a prime $p \in \mathbb{N}$ with $2^{2\varepsilon n} \le p \le 2 \cdot 2^{2\varepsilon n}$;
for $\ell \in \{1, \ldots, 2^{8\varepsilon n}\}$ **do**
  Sample $t \sim (1+1/n)^k \cdot (Q + c_\varepsilon \cdot T_\varepsilon(K))$ uniformly;
  Sample $i, j \in \{1, 2, \ldots, 2^{3\varepsilon n}\}$;
  for $k \in \{1, \ldots, 2^{3\varepsilon n}\}$ **do**
    Sample a shifted sublattice $\mathbf{u} + \mathscr{L}' \subseteq \mathscr{L}$ by Theorem 2.1.2 and p;
    Set $\mathbf{v}$ the lattice vector returned by oracle for α-CVP_K on $T_\varepsilon^{-1}(\mathscr{L}')$ and $T_\varepsilon^{-1}(t - \mathbf{u})$;
    if $k = i$ *or* $k = j$ **then**
      $\mathbf{v}_k \leftarrow T_\varepsilon(\mathbf{v}) + \mathbf{u}$;
    else
      Delete $\mathbf{v}$;
    end
  end
  $\mathbf{w} \leftarrow \arg\min\{\|\mathbf{w}\|_Q, \|\mathbf{v}_i - \mathbf{v}_j\|_Q, \|\mathbf{v}_i\|_Q, \|\mathbf{v}_j\|_Q\}$;
  Delete $\mathbf{v}_i$ and $\mathbf{v}_j$;
end
Output: $\mathbf{w}$

Figure 2.8 – Reducing approximate SVP_Q to approximate CVP_K.

Before embarking on the proof, we need two simple lemmas.

Lemma 2.4.3. *Let $T : \mathbb{R}^n \to \mathbb{R}^n$ any (invertible) linear transformation, $\mathscr{L}$ some lattice and $x \in \mathbb{R}^n$ some shift. Any oracle for α-approximate CVP_K yields an oracle for α-approximate $\mathrm{CVP}_{T(K)}$ on $x + \mathscr{L}$.*

Proof. To solve an instance of the α-approximate $\mathrm{CVP}_{T(K)}$ on the shifted lattice $x + \mathscr{L}$ with target t, it is sufficient to solve α-approximate $\mathrm{CVP}_{T(K)}$ on the shifted lattice $\mathscr{L}$. Indeed, $\mathbf{v} + x$ is a solution to α-approximate $\mathrm{CVP}_{T(K)}$ on the shifted lattice $x + \mathscr{L}$, where $\mathbf{v} \in \mathscr{L}$ is a solution to α-approximate $\mathrm{CVP}_{T(K)}$ on $\mathscr{L}$ with target $t - x$.

To solve α-approximate $\mathrm{CVP}_{T(K)}$ with target t and lattice $\mathscr{L}$ using an oracle for α-approximate CVP_K , we proceed as follows. We run the oracle for α-CVP_K with target $T^{-1}(t)$ and lattice $T^{-1}(\mathscr{L})$ and obtain the lattice vector $\mathbf{v}$. We then output $T(\mathbf{v})$. We claim that this is a α-approximation to the closest lattice vector $\mathbf{c}$ (with respect to $\|\cdot\|_{T(K)}$) to t. Indeed, by linearity of $T(\cdot)$,

$$\forall \mathbf{w} \in \mathscr{L} : \quad \|t - \mathbf{w}\|_{T(K)} = \|T^{-1}(t) - T^{-1}(\mathbf{w})\|_K.$$

Hence, for any vector $\mathbf{v} \in T^{-1}(\mathscr{L})$ with $\|T^{-1}(t) - \mathbf{v}\|_K \le \alpha \cdot \|T^{-1}(t) - T^{-1}(\mathbf{c})\|_K$, we must have that $\|t - T(\mathbf{v})\|_{T(K)} \le \alpha \cdot \|t - \mathbf{c}\|_{T(K)}$. Hence $T(\mathbf{v})$ is an α-approximation to $\mathrm{CVP}_{T(K)}$. □

The following is a slight generalization of Lemma 2.2.2.

Lemma 2.4.4. *Let $L, T \subseteq \mathbb{R}^n$ symmetric and convex bodies such that $N(L, \beta \cdot T), N(T, \beta \cdot L) \le M$. Then,*

$$\tfrac{1}{M \cdot \beta} \cdot \|\cdot\|_T \le \|\cdot\|_L \le (M \cdot \beta) \cdot \|\cdot\|_T.$$

Proof. We will show that $\frac{1}{M \cdot \beta} \cdot T \subseteq L \subseteq (M \cdot \beta) \cdot T$, this implies the claim. To show that $L \subseteq (M \cdot \beta) \cdot T$ we proceed by contradiction: Assume there exists $x, -x \in L$ such that $x, -x \notin (M \cdot \beta) \cdot T$. Let $t \in \mathbb{R}^n$ be such that the length of $(t + T) \cap [-x, x]$ is maximal. By symmetry, this equals the length of $(-t + T) \cap [-x, x]$, and thus, by convexity, we can assume that $t = 0$. Hence, the maximal length that a translate of T covers of the segment of $[-x, x]$ equals $|T \cap [-x, x]|$. Since $x, -x \notin (M \cdot \beta) \cdot T$, it is impossible to cover the segment $[-x, x]$ by fewer than $M + 1$ translates of $\beta \cdot T$, contradicting the fact that $N(L, \beta \cdot T) \le M$. By exchanging the role of L and T in the argument above, it follows that $T \subseteq (M \cdot \beta) \cdot L$ and we are done. □

Proof of Theorem 2.4.2. For convenience, we may assume that the lattice is preprocessed such that the lattice is of full rank (for instance by restricting to $\mathrm{span}(\mathscr{L})$ and rotating) and such that the length of the shortest vector with respect to $\|\cdot\|_Q$, $\mathbf{s}$, has norm between $(1 + 1/n)^{-1}$ and 1. We now prove the correctness of the reduction as described in Figure 2.7.

Given $\varepsilon > 0$, we find a prime $p \in [2^{2\varepsilon n}, 2 \cdot 2^{2\varepsilon n}]$ and compute the invertible linear transformations $T_\varepsilon(\cdot)$ guaranteed by Theorem 2.4.1. We now describe one iteration of the reduction that will succeed with probability $2^{-O(\varepsilon)n}$.

We first sample a point $t \sim Q + c_\varepsilon \cdot T_\varepsilon(K)$ uniformly at random. Since $\mathrm{Vol}(Q + c_\varepsilon \cdot T_\varepsilon(K)) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(c_\varepsilon \cdot T_\varepsilon(K))$, with probability at least $2^{-\varepsilon n}$, t is such that $\mathbf{s} \in t + c_\varepsilon \cdot T_\varepsilon(K)$. We condition on this event. We are now going to "collect" different lattice vectors from $t + \alpha \cdot c_\varepsilon \cdot T_\varepsilon(K)$ using an α-approximate oracle for $\mathrm{CVP}_{T_\varepsilon(K)}$ on shifted sublattices of $\mathscr{L}$. Such an oracle can be constructed from the oracle for α-approximate CVP_K , see Lemma 2.4.3. To do so, we will use the sparsification procedure from Theorem 2.1.2 instantiated with the lattice $\mathscr{L}$ and the prime number p. Recall that given any lattice vectors $\mathbf{w}, \mathbf{v}_1, \ldots, \mathbf{v}_N \in \mathscr{L}$, this procedure returns a shifted sublattice $\mathbf{u} + \mathscr{L}'$, $\mathbf{u}, \mathscr{L}' \subseteq \mathscr{L}$ such that

$$\Pr[\mathbf{w} \in \mathbf{u} + \mathscr{L}' \text{ and } \mathbf{v}_1, \ldots, \mathbf{v}_N \notin \mathbf{u} + \mathscr{L}'] \geq \frac{1}{p} - \frac{N}{p^2} - \frac{N}{p^{n-1}},$$

unless $\mathbf{w} \in \mathbf{v}_i + p \cdot \mathscr{L}$ for some $i \in \{1, \ldots, N\}$. Using this procedure with $\mathbf{w} := \mathbf{s}$ and by Lemma 2.4.4, any such lattice vector $\mathbf{v}_i$ is very far from t. Indeed, by the triangle inequality:

$$\begin{aligned}
\| t - \mathbf{v}_i \|_{T_\varepsilon(K)} &= \| \mathbf{s} - \mathbf{v}_i - (\mathbf{s} - t) \|_{T_\varepsilon(K)} \\
&\geq \| \mathbf{s} - \mathbf{v}_i \|_{T_\varepsilon(K)} - \| \mathbf{s} - t \|_{T_\varepsilon(K)} \\
&\geq p \cdot \| \frac{\mathbf{s} - \mathbf{v}_i}{p} \|_{T_\varepsilon(K)} - c_\varepsilon \\
&\overset{(*)}{\geq} p \cdot \lambda_1^{(T_\varepsilon(K))}(\mathscr{L}) - c_\varepsilon \\
&\overset{(**)}{\geq} p \cdot \frac{(1 + 1/n)^{-1}}{2^{\varepsilon n} \cdot c_\varepsilon} - c_\varepsilon.
\end{aligned}$$

In $(*)$ we have used that if $\mathbf{s} \in \mathbf{v}_i + p \cdot \mathscr{L}$, $\frac{\mathbf{s} - \mathbf{v}_i}{p} \in \mathscr{L}$ and is of $\| \cdot \|_K$-norm at least 1.
In $(**)$ we use Lemma 2.4.4 to conclude that $\lambda_1^{(T_\varepsilon(K))}(\mathscr{L}) \geq \lambda_1^{(Q)}(\mathscr{L}) / (2^{\varepsilon n} \cdot c_\varepsilon)$. Since we scaled the lattice such that $(1 + 1/n)^{-1} \leq \| \mathbf{s} \|_Q \leq 1$, the inequality follows.

Hence, by our choice of p and since c_ε is some constant, any lattice vector $\mathbf{v}$ with $\mathbf{v} \in \mathbf{s} + p \cdot \mathscr{L}$ has $\| \cdot \|_{T_\varepsilon(K)}$-norm at least $O(2^{\varepsilon n})$. It follows that whenever $\mathbf{s} \in \mathbf{u} + \mathscr{L}'$, where $\mathbf{u} + \mathscr{L}'$ is the sparsified sublattice from Theorem 2.1.2, the α-approximate oracle for $\mathrm{CVP}_{T_\varepsilon(K)}$ cannot return a lattice vector $\mathbf{v} \in \mathbf{s} + \mathscr{L}'$. Here, we are assuming that $\alpha = 2^{o(n)}$, otherwise, we could have used the (polynomial-time) Nearest Plane Algorithm due to Babai in the first place. Thus, whenever we have a list $\{\mathbf{v}_1, \ldots, \mathbf{v}_m\}$ of distinct lattice vectors inside $t + c_\varepsilon \cdot T_\varepsilon(K)$, $m \leq 2^{\varepsilon n} + 1$, and sparsify the lattice to obtain $\mathscr{L}'$, with probability at least $\Omega(2^{-2\varepsilon n})$, $\mathbf{s} \in \mathscr{L}'$ and $\mathbf{v}_1, \ldots, \mathbf{v}_m \notin \mathscr{L}'$. Hence, running the α-approximate oracle for $\mathrm{CVP}_{T_\varepsilon(K)}$ on $\mathbf{u} + \mathscr{L}'$ yields a lattice vector $\mathbf{v}_{m+1}$ distinct from $\mathbf{v}_1, \ldots, \mathbf{v}_m$ with $\| t - \mathbf{v}_{m+1} \|_{T_\varepsilon(K)} \leq \alpha \cdot c_\varepsilon$. By Chebychev's inequality and with probability at least $1/2$, after $2^{3\varepsilon n}$ iterations, the final list contains at least $2^{\varepsilon n} + 1$ distinct lattice vectors (or $\mathbf{s}$) inside $t + c_\varepsilon \cdot T_\varepsilon(K)$. We condition on this event as well.

If the final list contains $\mathbf{s}$ we are done. Assume not. Since $N(c_\varepsilon \cdot T_\varepsilon(K), c_\varepsilon^2 \cdot Q) \leq 2^{\varepsilon n}$, there are two lattice vectors in the list, $\mathbf{v}_i$ and $\mathbf{v}_j$ say, such that they lie in the same translated copy of $\alpha \cdot c_\varepsilon^2 \cdot Q$. Hence

$$\mathbf{v}_i - \mathbf{v}_j \in 2 \cdot \alpha \cdot c_\varepsilon^2 \cdot Q.$$

Since $\|\mathbf{s}\|_Q \geq (1+1/n)^{-1}$ and setting $\gamma_\varepsilon = (1+1/n)\cdot 2\cdot c_\varepsilon^2$, the lattice vector $\mathbf{v}_1 - \mathbf{v}_2$ is an $(\alpha\cdot\gamma_\varepsilon)$-approximation to the shortest vector with respect to $\|\cdot\|_Q$.

As described, one such an iteration requires space $2^{\varepsilon n}+1$ and succeeds with constant probability (assuming t is such that $\mathbf{s}\in t + c_\varepsilon\cdot T_\varepsilon(K)$). To bring this down to polynomial space, we guess the indices $i,j\in\{1,\ldots,2^{3\varepsilon n}\}$ for which $\mathbf{v}_i-\mathbf{v}_j\in 2\cdot\alpha\cdot c_\varepsilon^2\cdot Q$ (or for which $\mathbf{v}_i$ or $\mathbf{v}_j$ equals $\mathbf{s}$) and only store the two lattice vectors that we obtain at iterations i and j, respectively. This succeeds with probability $\Omega(2^{-6\varepsilon n})$. Hence, the resulting reduction, i.e. one iteration of the outer for loop in Figure 2.7, succeeds with probability at least $\Omega(2^{-7\varepsilon n})$. Repeating $2^{8\varepsilon n}$ times boosts the probability of success to $1-2^{-n}$. □

3 Algorithms in any Norm

This chapter presents approximation algorithms for the shortest and closest vector problem in any norm. These algorithms combine geometric properties of convex bodies with the original *randomized sieving* approach due to Ajtai, Kumar and Sivakumar (AKS01; AKS02) and with the lattice sparsification and enumeration approach due to Dadush and Kun (DK16), respectively.

Section 3.1 provides a high-level overview on the randomized sieving approach for the shortest and closest vector problem due to Ajtai et al. (AKS01; AKS02). Despite the emergence of newer algorithmic techniques such as Discrete Gaussian Sampling (which can be seen as a version of sieving) and enumeration through the Voronoi Cell that result in faster algorithms for SVP and CVP in the ℓ_2 norm, the randomized sieving approach has remained very relevant. Indeed, for cryptanalytic purposes, it is often sufficient to compute a (small) polynomial approximation to the shortest or closest vector problem. In this setting, randomized sieving is the only approach that provably achieves a constant factor approximation in $2^{0.802n}$ time. Any other algorithm requires at least 2^n time. Using block-wise reduction, randomized sieving can be leveraged to an n^c-approximation in $2^{0.802/(c+1)\,n}$ time, (GN08; ALNSD20; ALS21). Furthermore, randomized sieving is believed to perform much better. Under a believable (geometric) conjecture, certain heuristics (e.g. unproven assumptions) on the behaviour of the algorithm for *random* instances and the use of quantum computers, the running time of randomized sieving is estimated to be of the order of $2^{0.26n}$, (LMvdP15). This optimism is shared by cryptographers (in their case, pessimism, rather), and the key-sizes of proposed post-quantum cryptographic protocols are set in such a way as to achieve the desired security against lattice attacks based on randomized sieving, see for instance (ABD$^+$; DKL$^+$). The main technical properties of the randomized sieving procedure are resumed in Theorem 3.1.1. Following (EV22), Theorem 3.1.5 illustrates how this yields a constant factor approximation to the closest vector problem in the ℓ_2 norm in $2^{0.802n}$ time.

Section 3.2 is based on (RV22) and provides a partial converse to the reduction from approximate SVP_Q to approximate CVP_K. In Theorem 3.2.1 it is shown how $2^{\varepsilon n}$ calls to a sieving algorithm for the shortest vector problem with respect to $\|\cdot\|_K$ yields a constant factor approximation to the closest vector problem in $\|\cdot\|_K$, for any pair of norms $\|\cdot\|_Q$ and $\|\cdot\|_K$. This is achieved by combining the geometric considerations from the previous chapter with a

somewhat technical property of the randomized sieving property as outlined in Theorem 3.1.1. When instantiated with $K = B_2^n$, this yields a $2^{0.802n}$ time algorithm to compute a constant factor approximation to the shortest and closest vector problem in any norm.

Section 3.3 is concerned with the near-exact closest vector problem in any norm. Currently, the state-of-the-art algorithm for this problem from (DK16) yields a $(1+\varepsilon)$-approximation to the closest vector problem in any norm in $O(1+1/\varepsilon)^n$ time and 2^n space. Their algorithm is deterministic and is based on lattice sparsification combined with geometric considerations quite similar to those of this thesis. But for various norms, this can be improved. It was first realized by Eisenbrand, Hähnle and Niemeier (EHN11) that $O(\log(2+1/\varepsilon))^d$ calls to an oracle for 2-approximate CVP_∞ yields a $(1+\varepsilon)$-approximation to CVP_∞. This was achieved by decomposing the cube B_∞^d into $O(\log(2+1/\varepsilon))^d$ parallelepipeds, each of which, when scaled by a factor 2 around their respective centers, is contained inside $(1+\varepsilon)\cdot K$. This section is based on (NV22), where this technique was extended to *smooth* norms, in particular, (sections of) ℓ_p norms. For $p \geq 2$, this yields an algorithm for $(1+\varepsilon)$-CVP_p with a running time of $O(1+1/\varepsilon)^{n/2}$. The main geometric observation could also be directly incorporated in the algorithm of Dadush and Kun - essentially only changing a single parameter. This tweaked algorithm is described in Theorem 3.3.6.

3.1 Sieving for Shortest and Closest Vectors

Let us describe the randomized list-sieving approach of (MV10b) for the case of the shortest vector problem to a level that is necessary to understand our main result, Theorem 3.1.1. The sampling procedure described in this theorem is implicit in all sieving algorithms for the shortest vector problem. Our exposition follows closely the one given in (PS09).

Suppose we are given an instance of SVP_K with lattice $\mathscr{L}(\mathbf{B})$ with $d = n$. Let us denote by $\mathbf{s} \in \mathscr{L}$ some *special* lattice vector. Typically, $\mathbf{s}$ is a shortest nonzero lattice vector of $\mathscr{L}$ with respect to $\|\cdot\|_K$, i.e. $\|\mathbf{s}\|_K = \lambda_1^{(K)}(\mathscr{L})$, but it does not need to be. Assume that $d = n$ (which can be achieved by intersecting K with $\mathrm{span}(\mathscr{L})$) and that the lattice is scaled such that

$$1 - 1/n \leq \|\mathbf{s}\|_K \leq 1.$$

This assumes we have a bound on the length of $\mathbf{s}$. When $\mathbf{s}$ is the shortest or closest vector, by the LLL-algorithm or Babai's Nearest Plane Algorithm, this is without loss of generality since one can enumerate over a polynomial number of possibilities.

The list-sieve algorithm has two stages. In the *first stage*, the algorithm constructs a list L of random lattice vectors. This list is then passed to the *second stage* of the algorithm.

The second stage of the algorithm proceeds by sampling points $y_1, \ldots, y_N$ independently from

$$O(1) \cdot K.$$

It then transforms these points via the algorithm $\texttt{ListRed}_L$ into lattice points

$$\texttt{ListRed}_L(y_1), \ldots, \texttt{ListRed}_L(y_N) \in \mathscr{L}(\mathbf{B}).$$

The algorithm $\texttt{ListRed}_L$ is deterministic and straightforward. It subtracts lattice vectors in L from the y_i to obtain shorter and shorter vectors.

Input: A list $\mathrm{L} \subseteq \mathscr{L}$, $y \in \mathbb{R}^n$ and a parameter $\varepsilon > 0$
$u' \leftarrow y \pmod{\mathbf{B}}$, $\mathbf{u} = u' - y$;
while $\exists \mathbf{w} \in \mathrm{L}$ *such that* $\|u' - \mathbf{w}\|_K < (1-\varepsilon) \cdot \|u'\|_K$ **do**
  $(\mathbf{u}, u') \leftarrow (\mathbf{u} - \mathbf{w}, u' - \mathbf{w})$
end
Output: u

Figure 3.1 – The $\texttt{ListRed}_L$ procedure.

Two central questions are the following. First, how to construct a *small* list L of lattice vectors so that $\texttt{ListRed}_L$ produces (approximately) short vectors. Second, how to set the distribution so that this naïve procedure produces *lattice* vectors that are *short* and *nonzero*.

The first problem is of geometric nature. To address it, we first clarify how the procedure $\texttt{ListRed}_L$ works, see Figure 3.1. Given $y \in \mathbb{R}^n$, the procedure converts y into a tuple $(u', \mathbf{u})$, where $u' \in \mathbb{R}^n$ and $\mathbf{u} \in \mathscr{L}$. The way u' and $\mathbf{u}$ are created, ensures that $\|u' - \mathbf{u}\|_K \leq \|y\|_K$. As we will precise later, the vector y will be drawn uniformly from $O(1) \cdot K$, hence u' and $\mathbf{u}$ are close. However, crucially, we will base our decision on whether we subtract some lattice vector $\mathbf{v} \in \mathrm{L}$ from $\mathbf{u}$ according to whether $\|u' - \mathbf{v}\|_K$ decreases u' in the $\|\cdot\|_K$ norm.

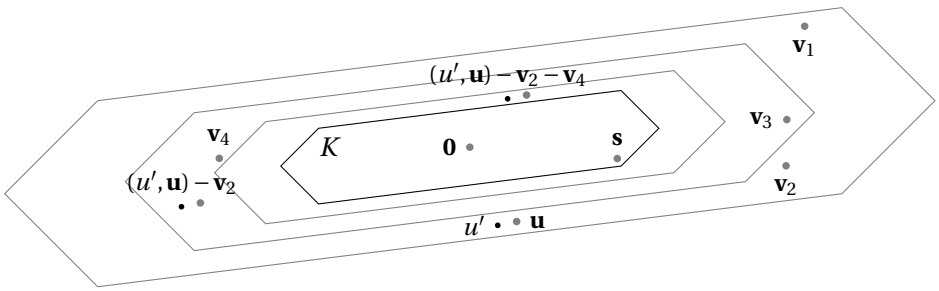

Figure 3.2 – The pair $(u', \mathbf{u})$ is being reduced by $\texttt{ListRed}_L$ with the list $\mathrm{L} := \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4\}$: First, we subtract the lattice vector $\mathbf{v}_2$, then $\mathbf{v}_4$.

For now, let us pretend that we are *given* a list L of lattice vectors such that all vectors in L inside a layer of the form $\{x \in \mathbb{R}^n \mid (1+\varepsilon)^k < \|x\|_K \leq (1+\varepsilon)^{k+1}\}$ form a *maximum packing* with translates of $(1+\varepsilon)^k \cdot \frac{1}{2} \cdot K$. In other words, the list is such that for the (k^{th}) layer of the form

$\{x \in \mathbb{R}^n \mid (1+\varepsilon)^k < \|x\|_K \leq (1+\varepsilon)^{k+1}\}$ and for any two distinct vectors $\mathbf{v}_1, \mathbf{v}_2 \in \mathrm{L}$ in this layer, we have that

$$\mathbf{v}_1 + (1+\varepsilon)^k \cdot \tfrac{1}{2} \cdot K \cap \mathbf{v}_2 + (1+\varepsilon)^k \cdot \tfrac{1}{2} \cdot K = \emptyset,$$

but, for any $x \notin \mathrm{L}$ in the k^{th} layer, $x + (1+\varepsilon)^k \cdot \frac{1}{2} \cdot K$ intersects some $\mathbf{v} + (1+\varepsilon)^k \cdot \frac{1}{2} \cdot K$ for $\mathbf{v} \in \mathrm{L}$ in the k^{th} layer. Whenever the list L forms such a maximum packing, any vector u' falling into the k^{th} layer has distance at most $(1+\varepsilon)^k$ to at least one vector in L in the same layer and its length can be reduced further.

The number of vectors required in each layer to form a maximum packing can be related to (a variant of) the *kissing number* of K. We define $\tilde{k}(K,\gamma)$ to be the maximum number N of non-overlapping translates of $(\frac{1-\gamma}{2}) \cdot K$ with centers in $K \setminus (1-\gamma) \cdot \mathrm{int}(K)$. Equivalently, it is the maximum number of points $x_1, \ldots, x_N \in K \setminus (1-\gamma) \cdot K$ so that $\|x_i - x_j\|_K \geq 1-\gamma$ for all $i \neq j \in \{1, \ldots, N\}$. This generalizes the kissing number for K, which is defined as $\tilde{k}(K,0)$. We note that for $K = B_2^n$ and γ small enough, this number can be upper bounded by $2^{0.401n}$, but for arbitrary K such as B_∞^n, this is as high as 3^n. From this definition and for a maximum packing of vectors as described above, it follows that the maximum number of vectors in each layer cannot exceed $\tilde{k}(K,\varepsilon)$. Since we only need to consider a polynomial number of layers (we have that $\mathbf{v}_i \in 2^{O(n)} \cdot K$), the total number of vectors in the list L is upper bounded by $\mathrm{poly}(n) \cdot \tilde{k}(K,\gamma)$. The running time of $\texttt{ListRed}_L$ is of order $\mathrm{poly}(n) \cdot \tilde{k}(K,\varepsilon)^2$ and the space requirement is of order $\mathrm{poly}(n) \cdot \tilde{k}(K,\varepsilon)$.

For the algorithm to work, it will not be necessary to construct a list consisting of a maximum packing of lattice vectors in each layer. We can construct a list L in $2^{\varepsilon n} \cdot \tilde{k}(K,\varepsilon)^2$ time on the fly, for which $\texttt{ListRed}_L(\cdot)$ works with high probability. Before discussing this further, we discuss the second point, how to generate the y's to be reduced by $\texttt{ListRed}_L$.

For $c_\varepsilon = O(1/\varepsilon)$, we sample $y \sim c_\varepsilon \cdot K$ uniformly at random and compute $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$ with

$$y = \sum_{i=1}^{n} \lambda_i \cdot \mathbf{b}_i.$$

We then compute the *remainder* (modulo the lattice basis)

$$y \pmod{\mathbf{B}} := \sum_{i=1}^{n} \{\lambda_i\} \cdot \mathbf{b}_i,$$

where $\{\lambda_i\} \in [0,1[$ is the fractional part of λ_i, i.e. $\lambda_i = \lfloor \lambda_i \rfloor + \{\lambda_i\}$, $\lfloor \lambda_i \rfloor \in \mathbb{Z}$. Consequently,

$$y \pmod{\mathbf{B}} - y = \sum_{i=1}^{n} -\lfloor \lambda_i \rfloor \cdot \mathbf{b}_i \in \mathscr{L}.$$

The algorithm $\texttt{ListRed}_L$ takes as input y, computes $y \pmod{\mathbf{B}}$ and then subtracts lattice vectors $\mathbf{v}_1, \ldots, \mathbf{v}_k \in \mathrm{L}$ from $y \pmod{\mathbf{B}}$ to decrease its length and obtain

$$y \pmod{\mathbf{B}} - \mathbf{v}_1 - \ldots - \mathbf{v}_k.$$

The final output of $\texttt{ListRed}_L(y)$ is

$$y \pmod{\mathbf{B}} - y - \mathbf{v}_1 - \ldots - \mathbf{v}_k \in \mathscr{L}.$$

Since $y \pmod{\mathbf{B}} - y$ is a lattice vector, so is the final output. When $y \pmod{\mathbf{B}} - \mathbf{v}_1 - \ldots - \mathbf{v}_k$ has norm ℓ, $y \pmod{\mathbf{B}} - \mathbf{v}_1 - \ldots - \mathbf{v}_k - y$ has norm (with respect to $\|\cdot\|_K$) at most $\ell + c_\varepsilon$.

It is crucial that we base our decision to subtract vectors from L on $y \pmod{\mathbf{B}}$ and not on y directly. Indeed, since

$$y \pmod{\mathbf{B}} = y - \mathbf{s} \pmod{\mathbf{B}},$$

the algorithm acts the same way for $y \pmod{\mathbf{B}}$ as it would for $y - \mathbf{s} \pmod{\mathbf{B}}$. This is why we can imagine that we defer the decision whether we initially started with y or $y - \mathbf{s}$ until the very end of the algorithm. If y is being reduced to $\mathbf{0}$ by the list L, then, $y - \mathbf{s}$ would be reduced to $\mathbf{s}$.

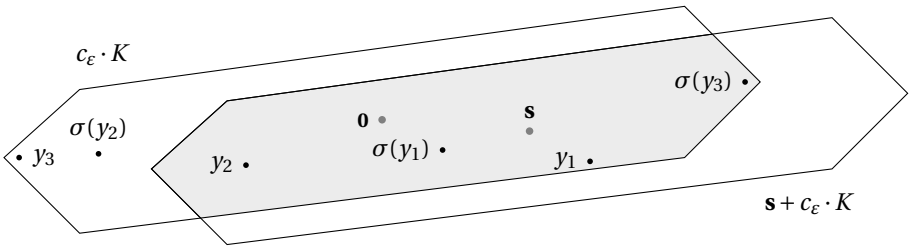

Figure 3.3 – The set $c_\varepsilon \cdot K \cap (\mathbf{s} + c_\varepsilon \cdot K)$ and the bijection $\sigma(\cdot)$.

It remains to argue why sampling $y \sim c_\varepsilon \cdot K$ such that $y - \mathbf{s} \in c_\varepsilon \cdot K$ is about as likely as sampling $y \sim c_\varepsilon \cdot K$. To see this, we consider the region $c_\varepsilon \cdot K \cap (\mathbf{s} + c_\varepsilon \cdot K)$ depicted in Figure 3.3. This is commonly referred to as the *lens* and was introduced by Regev in a conceptual simplification of the original sieving algorithm, (Reg04). After sampling $y \sim c_\varepsilon \cdot K$, we apply the following bijection $\sigma : c_\varepsilon \cdot K \to c_\varepsilon \cdot K$ and return $\sigma(y)$ instead,

$$\sigma(y) = \begin{cases} y - \mathbf{s} & \text{if } y \in c_\varepsilon \cdot K \cap (\mathbf{s} + c_\varepsilon \cdot K), \\ -y & \text{else.} \end{cases}$$

The bijection $\sigma(\cdot)$ is measure preserving. In other words, if y is distributed uniformly within $c_\varepsilon \cdot K$, so is $\sigma(y)$. So the uniform distribution on $c_\varepsilon \cdot K$ is equivalent to first sampling $y \sim c_\varepsilon \cdot K$ uniformly, then, with probability $1/2$, applying $\sigma(\cdot)$ to y. This is also illustrated in Figure 3.3. The resulting distribution is uniform and we see that with probability (at least)

$$\tfrac{1}{2} \Pr_{x \sim c_\varepsilon \cdot K}[x \in c_\varepsilon \cdot K \cap (\mathbf{s} + c_\varepsilon \cdot K)] = \frac{\mathrm{Vol}(c_\varepsilon \cdot K \cap (\mathbf{s} + c_\varepsilon \cdot K))}{2 \cdot \mathrm{Vol}(c_\varepsilon \cdot K)},$$

y is being replaced by $y - \mathbf{s}$. By the choice of c_ε, this probability is at least $2^{-\varepsilon n}$.

We rephrase these properties in the following theorem.

Theorem 3.1.1 ((EV22; RV22)). *Given $\varepsilon > 0$, $R > 0$, $N \in \mathbb{N}$, a norm $\|\cdot\|_K : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ and a lattice $\mathscr{L} \subseteq \mathbb{R}^n$, there is a randomized procedure that produces independent and identically distributed samples $\mathbf{u}_1, \ldots, \mathbf{u}_N \sim \mathscr{D}$, where the distribution $\mathscr{D}$ satisfies the following two properties:*

1. *Every sample $\mathbf{u} \sim \mathscr{D}$ has $\mathbf{u} \in \mathscr{L}$ and, with probability at least $1/2$, $\|\mathbf{u}\|_K \leq a_\varepsilon \cdot R$. Here, a_ε is a constant only depending on ε.*
2. *For any $\mathbf{s} \in \mathscr{L}$ with $\|\mathbf{s}\|_K \leq R$, there are distributions $\mathscr{D}_0^{\mathbf{s}}$ and $\mathscr{D}_1^{\mathbf{s}}$ and some parameter $\rho_{\mathbf{s}}$ with $2^{-\varepsilon n} \leq \rho_{\mathbf{s}} \leq 1$ such that the distribution $\mathscr{D}$ is equivalent to the following process:*
   (a) *With probability $\rho_{\mathbf{s}}$, sample $\mathbf{u} \sim \mathscr{D}_0^{\mathbf{s}}$. Then, flip a fair coin and with probability $1/2$, return $\mathbf{u}$, otherwise return $\mathbf{u} + \mathbf{s}$.*
   (b) *With probability $1 - \rho_{\mathbf{s}}$, sample $\mathbf{u} \sim \mathscr{D}_1^{\mathbf{s}}$.*

This procedure takes time $2^{\varepsilon n} \cdot \tilde{k}(K,\varepsilon)^2 + N \cdot \tilde{k}(K,\varepsilon)$, requires $N + 2^{\varepsilon n} \cdot \tilde{k}(K,\varepsilon)$ space and succeeds with probability at least $1/2$.

The time and space requirements of $2^{\varepsilon n} \cdot \tilde{k}(K,\varepsilon)^2$ and $2^{\varepsilon n} \cdot \tilde{k}(K,\varepsilon)$ stem from the construction of the list, the first phase of the algorithm. The algorithm is given in Figure 3.4.

Input: $\|\cdot\|_K : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, $\mathscr{L}(B) \subseteq \mathbb{R}^n$ with $(1 - 1/n) \leq \|\mathbf{s}\|_K \leq 1$ and $\varepsilon > 0, N \in \mathbb{N}$
Pick $M \sim \{1, \ldots, \text{poly}(n) \cdot 2^{\varepsilon n} \cdot \tilde{k}(K, 2 \cdot \varepsilon)\}$;
$\mathrm{L} \leftarrow \emptyset$;
for *i in* 1 *to* M **do**
  Sample $y_i \sim c_\varepsilon \cdot K$;
  $\mathbf{v}_i \leftarrow \texttt{ListRed}_L(y_i)$;
  if $\|\mathbf{v}_i\|_K \geq c_\varepsilon^2$ **then**
    $\mathrm{L} \leftarrow \mathrm{L} \cup \{\mathbf{v}_i\}$
  end
end
for *j in* 1 *to* N **do**
  Sample $y_j \sim c_\varepsilon \cdot K$;
  $\mathbf{u}_j \leftarrow \texttt{ListRed}_L(y_j)$
end
Output: $\mathbf{u}_1, \ldots, \mathbf{u}_N$

Figure 3.4 – The `ListSieve` algorithm. With probability $1/2$, it outputs N lattice vectors that are distributed according to Property 1 and 2 of Theorem 3.1.1.

We now give a quick overview on the construction of such a list. The list is created by adding one lattice vector to it at a time. Suppose we have a partial list $\mathrm{L} = \{\mathbf{v}_1, \ldots, \mathbf{v}_m\}$. We sample some vector $y \sim c_\varepsilon \cdot K$ uniformly and reduce the length of $y \pmod{\mathbf{B}}$ by subtracting vectors $\mathbf{v}_1, \ldots, \mathbf{v}_k$ from the partial list as in the second phase of the algorithm. We stop when the vector $y \pmod{\mathbf{B}} - \mathbf{v}_1 - \ldots - \mathbf{v}_k$ cannot be reduced any further and add $y \pmod{\mathbf{B}} - \mathbf{v}_1 - \ldots - \mathbf{v}_k - y$

to the list and repeat. Throughout the procedure, we *throw out* all lattice vectors of norm smaller than $c_\varepsilon^2 = O(1/\varepsilon^2)$, i.e. do not add them to the list. This guarantees that we can use the packing bound to upper bound the number of vectors in each layer. In particular, this small technicality ensures that the list never holds more than $n_L \cdot \tilde{k}(K, 2\cdot\varepsilon)$ lattice vectors in total, where $n_L = O(\text{poly}(n))$ is the total number of layers we need to consider. Here, the factor 2 in $\tilde{k}(K,\cdot)$ appears since we subtract the respective y from the reduced $y \pmod{\mathbf{B}}$ right before adding it to the list. To obtain the final list, we repeat this step $M := 3 \cdot n_L \cdot 2^{\varepsilon n} \cdot \tilde{k}(K, 2\cdot\varepsilon)$ times, obtain the list $\{\mathbf{v}_1, \ldots, \mathbf{v}_M\}$, sample a random integer N uniformly in the interval $[0, M]$ and pass the *truncated* list $\mathrm{L} := \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ to the second stage. It is important to note that the list is unchanged in the second stage, hence the lattice vectors that are outputted by the algorithm `ListSieve` are independent and identically distributed.

We now argue that this list is good for the second stage with probability at least $1/2$, i.e. verifies Properties 1 and 2 in Theorem 3.1.1. Observe that out of the $3 \cdot n_L \cdot 2^{\varepsilon n} \cdot \tilde{k}(K, 2\cdot\varepsilon)$ samples uniformly distributed within $c_\varepsilon \cdot K$, by Chebychev's inequality and with high probability, at least $2 \cdot n_L \cdot \tilde{k}(K, 2\cdot\varepsilon)$ samples come from the lens. During the creation of the list, *half* of these samples must have been thrown out (since the list can only hold $n_L \cdot \tilde{k}(K, 2\cdot\varepsilon)$ points in total), hence they must have been mapped to some point of $\|\cdot\|_K$ norm smaller than c_ε^2. It follows that for a uniform index $N \sim \{1, \ldots, M\}$, the partial list $\{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ reduces a sample y coming from the lens to some lattice vector of $\|\cdot\|_K$ norm at most $c_\varepsilon^2 = O(1/\varepsilon^2)$, possibly $\mathbf{0}$, with probability at least $1/2$. In fact, this also holds for samples y not coming for the lens and shows Property 1. To show Property 2, we use the bijection $\sigma(\cdot)$. The list L is such that $\texttt{ListRed}_L$ reduces some y sampled uniformly from the lens to a lattice vector of norm at most c_ε^2. For such a y, the list L reduces $y - \mathbf{s} = \sigma(y)$ in the exact same way as y. Hence, if $\texttt{ListRed}_L$ maps y to some lattice vector $\mathbf{w}$, then $y - \mathbf{s}$ would be mapped to $\mathbf{w} + \mathbf{s}$. To see that this implies Property 2, we reinterpret the uniform distribution as follows. We first sample $z \sim c_\varepsilon \cdot K$, and, with probability $1/2$, set $y = \sigma(z)$ and $y = z$ otherwise. Since $\sigma(\cdot)$ is measure preserving, y is distributed exactly as z, meaning that the respective distributions $\texttt{ListRed}_L(y)$ and $\texttt{ListRed}_L(z)$ are identical. Provided z came from the lens, the decision of applying $\sigma(\cdot)$ can be deferred until the very end of the algorithm, at which point we can flip a coin to decide the outcome. Since the probability of sampling from the lens is at least $2^{-\varepsilon n}$, Property 2 follows.

We note that the formal proofs of these statements can be found in (PS09). While they are stated for the ℓ_2 norm, it is straightforward to adapt it to the setting of general norms. In the next two subsections, we show how to use the `ListSieve` procedure as described in the previous section to obtain an approximation to the shortest and closest vector problem. Even though there is an efficient reduction from the shortest vector problem to the closest vector problem that preserves the approximation guarantee, see (GMSS99), we state the two algorithms separately.

3.1.1 Sieving for the Shortest Vector Problem

We note that the following theorem is implicit in previous works on the shortest vector problem in the ℓ_2 norm, (MV10b; PS09; LWXZ11).

Theorem 3.1.2. *Given any instance of the shortest vector problem* SVP_K *on a rank n lattice and for any* $\varepsilon > 0$*, one can compute an* α_ε*-approximation to the shortest vector in (randomized) time* $2^{\varepsilon n} \cdot \tilde{k}(K,\varepsilon)^2$ *and* $\tilde{k}(K,\varepsilon)$ *space.*

Proof. Without loss of generality, we may assume that $d = n$. Let us denote by $\mathbf{s}$ a shortest vector with respect to $\|\cdot\|_K$. By the LLL-algorithm combined with John's theorem (see Theorem 1.3.1 and Theorem 1.3.2), we can estimate $\|\mathbf{s}\|_K$ up to a factor of $n \cdot 2^n$. Then, guessing (i.e. enumerating over the polynomial number of possibilities) the length of $\mathbf{s}$ up to a factor $(1+1/n)$ and up to scaling the lattice accordingly, we may assume $1 - 1/n \leq \|\mathbf{s}\|_K \leq 1$.

We now use the procedure from Theorem 3.1.1, initialized with $R = 1$ and with $\varepsilon, \|\cdot\|_K$ and $\mathscr{L}$ from this theorem, to generate $N = n \cdot 2^{\varepsilon n}$ lattice vectors. This procedure succeeds with probability $1/2$, meaning the distribution $\mathscr{D}$ satisfies Property 1 and Property 2.

Since the samples $\mathbf{v}_1, \ldots, \mathbf{v}_N$ are independent and identically distributed, the number of samples from the distribution $\mathscr{D}_0^{\mathbf{s}}$ obeys a binomial distribution with parameter $\rho_{\mathbf{s}}$. By Chebychev's inequality and with probability at least $1/2$, one of the $\mathbf{v}_i$ is drawn according to $\mathscr{D}_0^{\mathbf{s}}$. Hence, the following event has probability at least $1/4$:

$$E := \{\exists \mathbf{v}_i \in \{\mathbf{v}_1, \ldots, \mathbf{v}_N\} : \mathbf{v}_i \sim \mathscr{D}_0^{\mathbf{s}} \text{ and } \|\mathbf{v}_i\|_K \leq a_\varepsilon\}.$$

We now use Property 2. We split the event E into two disjoint events, E^{head} and E^{tail}, where we distinguish whether the fair coin lands head (return $\mathbf{v}_i$) or tail (return $\mathbf{v}_i + \mathbf{s}$) when sampling $\mathbf{v}_i \sim \mathscr{D}_0^{\mathbf{s}}$. The probability of the events E^{head} and E^{tail} are both $1/8$. Since at least one of the events E^{head} or E^{tail} returns a nonzero lattice vector, we can conclude that with probability at least $1/8$, the list $\{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ contains a nonzero lattice vector $\mathbf{v}$ of length at most a_ε. Since $\|\mathbf{s}\|_K \geq 1 - 1/n$, setting $\alpha_\varepsilon := (1-1/n)^{-1} \cdot a_\varepsilon$, $\mathbf{v}$ is an α_ε-approximation to $\mathbf{s}$.

Thus, the above procedure yields an α_ε-approximation to $\mathbf{s}$ with probability at least $1/8$. This can be boosted to probability $1 - 2^{-n}$ by repeating the procedure, starting where we sample $N = n \cdot 2^{\varepsilon n}$ lattice vectors from the randomized procedure given by Theorem 3.1.1, a polynomial number of times. □

3.1.2 Sieving for the Closest Vector Problem

The case of the closest vector problem is slightly more involved and there are two small issues we would like to get out of the way.

First, unlike for the shortest vector problem, whenever t lies outside of the span of the lattice, the norm on $\mathbb{R}^d$ induced by K (centered at t) does (generally) not induce a norm when restricted to $\mathrm{span}(\mathscr{L})$. This can be seen by lifting t and the crosspolytope with it in Figure 1.2. To invoke the procedure from Theorem 3.1.1 nonetheless, we show how to restrict to the case $d = n$, albeit at a small loss in the approximation guarantee.

Lemma 3.1.3 ((RV22))**.** *Consider an instance of the closest vector problem,* $\mathrm{CVP}_K(\mathscr{L}, t)$, $\mathscr{L} \subseteq \mathbb{Q}^d$ *of rank* n*. In polynomial time, one can find a lattice* $\tilde{\mathscr{L}} \subseteq \mathbb{Q}^n$ *of rank and dimension* n*, target* $\tilde{t} \in \mathbb{Q}^n$ *and norm* $\|\cdot\|_{\tilde{K}}$ *so that an* α*-approximation to* $\mathrm{CVP}_{\tilde{K}}$ *on* $\tilde{\mathscr{L}}$ *with target* $\tilde{t}$ *can be efficiently transformed in a* $(2\cdot\alpha+1)$*-approximation to* $\mathrm{CVP}_K(\mathscr{L}, t)$*.*

Proof. Let us define $t' := \mathrm{argmin}\{\|t - x\|_K,\, x \in \mathrm{span}(\mathscr{L})\}$. Such a point may not be unique, for instance for $K = B_1^n$ or $K = B_\infty^n$, but it suffices to consider any point t' realizing this minimum or an approximation thereof. Given a (weak) separation oracle for K, this can be computed in polynomial time. We now show that an α-approximation to $\mathrm{CVP}_K(\mathscr{L}, t')$ yields a $(2\cdot\alpha+1)$-approximation to $\mathrm{CVP}_K(\mathscr{L}, t)$. Indeed, since $\|t - t'\|_K$ is smaller than $\mathrm{dist}_K(t, \mathscr{L})$, the distance of t to its closest lattice vector, we have that

$$\mathrm{dist}_K(t', \mathscr{L}) \leq 2 \cdot \mathrm{dist}_K(t, \mathscr{L}).$$

Denote by $\mathbf{c}_\alpha \in \mathscr{L}$ an α-approximation to the closest lattice vector to t'. By the triangle inequality,

$$\|t - \mathbf{c}_\alpha\|_K \leq \|t - t'\|_K + \|t' - \mathbf{c}_\alpha\|_K \leq \mathrm{dist}(t, \mathscr{L}) + \alpha \cdot \mathrm{dist}(t', \mathscr{L}) \leq (2\cdot\alpha+1) \cdot \mathrm{dist}(t, \mathscr{L}).$$

This means that an α-approximation of the closest vector to t' is a $(2\cdot\alpha+1)$-approximation to the closest vector to t. Whenever $K = B_2^n$ (and, obviously, whenever $t \in \mathrm{span}(\mathscr{L})$), by orthogonality, this holds exactly, i.e. with $2\cdot\alpha+1$ replaced by α.

For such a $t' \in \mathrm{span}(\mathscr{L})$, we can restrict to the case $d = n$: Let $\mathscr{O}_n : \mathbb{R}^d \to \mathbb{R}^n$ be a linear transformation that first applies a rotation sending $\mathrm{span}(\mathscr{L})$ to $\mathbb{R}^n \times \{0\}^{d-n}$ and then restricts onto its first n coordinates. The transformation $\mathscr{O}_n : \mathrm{span}(\mathscr{L}) \to \mathbb{R}^n$ is invertible. The n-dimensional instance of the closest vector problem is then obtained by setting $\tilde{\mathscr{L}} \leftarrow \mathscr{O}_n(\mathscr{L})$, $\tilde{t} \leftarrow \mathscr{O}_n(t')$ and $\|\cdot\|_{\tilde{K}}$ where $\tilde{K} \leftarrow \mathscr{O}_n(K)$. Whenever $\tilde{\mathbf{c}}_\alpha \in \mathscr{L}'$ is an α-approximation to $\mathrm{CVP}_{\tilde{K}}(\tilde{\mathscr{L}}, \tilde{t})$, the vector $\mathscr{O}_n^{-1}(\mathbf{c}_\alpha) \in \mathscr{L}$ is a $(2\cdot\alpha+1)$-approximation to $\mathrm{CVP}_K(\mathscr{L}, t)$ (or an α-approximation to $\mathrm{CVP}_K(\mathscr{L}, t)$, if $t \in \mathrm{span}(\mathscr{L})$ or $K = B_2^n$). □

Second, it will be convenient to view the closest vector problem on a lattice in $\mathbb{R}^n$ as a special case of a shortest vector problem on a lattice in $\mathbb{R}^{n+1}$, i.e. one dimension higher. This idea was introduced by Kannan (Kan87) and goes by the name of *Kannan's embedding technique*. The norm under consideration must be adapted to account for this, for instance, it might only be defined on $\mathbb{R}^n$. For ℓ_p norms, this proves to be straightforward, simply pick the ℓ_p norm on $\mathbb{R}^{n+1}$. For general convex bodies, it is less clear how to extend the norm $\|\cdot\|_K : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$

to $\mathbb{R}^{n+1}$ in a meaningful way. Before proposing an extension, let us first outline Kannan's embedding technique. Informally, we are going to slightly *lift* the target t and consider an instance of the shortest vector problem on a lattice of one dimension higher by adding t to the basis. Specifically, given an instance of the closest vector problem with basis $\mathbf{B}$ and for some (small) constant $1 > \mu > 0$, we define a new lattice $\mathscr{L}' \subseteq \mathbb{R}^{n+1}$ of rank $n+1$ with the following basis:

$$\tilde{\mathbf{B}} = \begin{pmatrix} \mathbf{B} & t \\ 0 & \mu \end{pmatrix} \in \mathbb{Q}^{(n+1)\times(n+1)}.$$

If $\mathbf{c}$ is the *closest* lattice vector to t in $\mathscr{L}$, then the vector $\mathbf{s} := \begin{pmatrix} t-\mathbf{c} \\ \mu \end{pmatrix}$ is a shortest lattice vector in $\mathscr{L}'$ restricted to $\mathscr{L}' \cap \{x \in \mathbb{R}^{n+1} \mid x_{n+1} = \mu\}$. Here, shortest is with respect to the point $\begin{pmatrix} \mathbf{0} \\ \mu \end{pmatrix}$, $\mathbf{0} \in \mathscr{L}$, which does not belong to the lattice unless $t \in \mathscr{L}$. This is depicted in Figure 3.5.

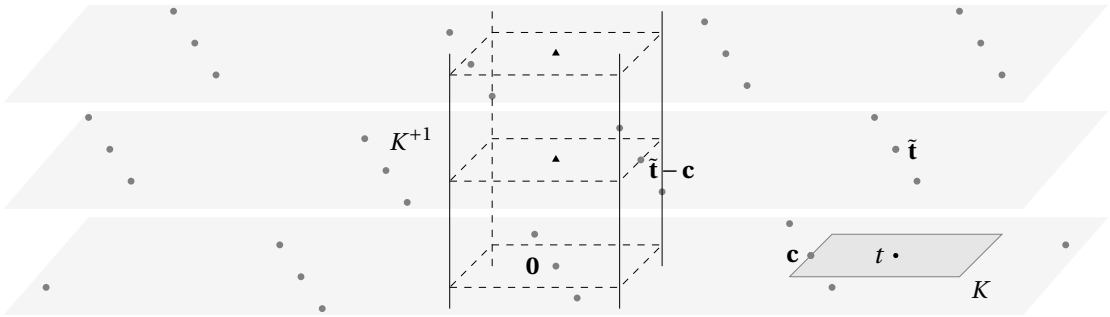

Figure 3.5 – The $n+1$ dimensional lattice after lifting t. $\mathbf{c}$ is the closest lattice vector to t. After lifting t and adding it to the basis, the lattice vector $\tilde{\mathbf{t}} - \begin{pmatrix} \mathbf{c} \\ 0 \end{pmatrix} = \begin{pmatrix} t-\mathbf{c} \\ \mu \end{pmatrix}$ is the shortest vector on the first layer.

This also holds for approximations to the closest vector problem. An α-approximate closest lattice vector to t in $\mathscr{L}$ corresponds to an α-approximate shortest lattice vector restricted to $\mathscr{L}' \cap \{x \in \mathbb{R}^{n+1} \mid x_{n+1} = \mu\}$ and vice versa.

Going back to the issue at hand, we propose to extend $\|\cdot\|_K$ to $\mathbb{R}^{n+1}$ by considering $K^{+1} \subseteq \mathbb{R}^{n+1}$ instead of K.

$$K^{+1} := \left\{(x, x_{n+1}) \in \mathbb{R}^n \times \mathbb{R} \mid x \in K, x_{n+1} \in [-1,1]\right\}. \tag{3.1}$$

The convex body K^{+1} is the $n+1$ dimensional cylinder obtained by taking K as its base. The norm $\|\cdot\|_{K^{+1}}$ restricted to $\{x \in \mathbb{R}^{n+1} \mid x_{n+1} = \mu\}$ is equivalent to $\|\cdot\|_K$ (centered at $\begin{pmatrix} \mathbf{0} \\ \mu \end{pmatrix}$).

With the procedure from Theorem 3.1.1 in mind, it remains to check that the respective numbers $\tilde{k}(K,\cdot)$ and $\tilde{k}(K^{+1},\cdot)$ remain comparable.

Lemma 3.1.4. *Let $Q \subseteq \mathbb{R}^n$ be an origin-symmetric convex body. Define $Q^{+1} \subseteq \mathbb{R}^{n+1}$ as in (3.1). Then*

$$\tilde{k}(Q^{+1}, \gamma) \leq O(\tfrac{1}{(1-\gamma)} \cdot \tilde{k}(Q, \gamma)).$$

Proof. Consider the maximum number of disjoint translates $x_i + \frac{1-\gamma}{2} \cdot Q^{+1}$, $i \in \{1, \ldots, \tilde{k}(Q^{+1}, \gamma)\}$, where the x_i lie in $Q^{+1} \setminus (1-\gamma) \cdot Q^{+1}$. Each such translate intersects at least one slice of the form $Q^{+1} \cap \{x \in \mathbb{R}^{n+1} \mid x_{n+1} = k \cdot (1-\gamma)\}$, for some $k \in \{-\lceil (1-\gamma)^{-1} \rceil, \ldots, \lceil (1-\gamma)^{-1} \rceil\}$. Since the translates are mutually disjoint, each such slice can intersect at most $\tilde{k}(Q, \gamma)$ translates of $\frac{1-\gamma}{2} \cdot Q^{+1}$. The bound follows. □

We can now use the sampling procedure from Theorem 3.1.1 to obtain a constant factor approximation to the closest vector problem.

Theorem 3.1.5 ((EV22))**.** *Given any instance of the closest vector problem* CVP_K *on a rank n lattice and any* $\varepsilon > 0$*, one can compute an* α_ε*-approximation to the closest lattice vector in (randomized) time* $2^{\varepsilon n} \cdot \tilde{k}(K, \varepsilon)^2$ *and* $2^{\varepsilon n} \cdot \tilde{k}(K, \varepsilon)$ *space.*

The currently best bounds for $\tilde{k}(K, \varepsilon)^2$ are known for $K = B_2^n$. For some constant $\varepsilon_0 > 0$, $\tilde{k}(K, \varepsilon_0)$ is of the order $2^{0.401n}$, see (KL78). The constant in the exponent is in fact (slightly) smaller than 0.401, hence Theorem 3.1.5 implies a $2^{0.802n}$ time, $2^{0.401n}$ space algorithm for constant approximate closest vector problem in the ℓ_2 norm.

Proof of Theorem 3.1.5. By Lemma 3.1.3, up to a constant factor loss in the final approximation guarantee, we may assume that the lattice is full dimensional, i.e. $d = n$. Furthermore, we assume the lattice and target t are scaled so that the distance of the closest lattice vector $\mathbf{c} \in \mathscr{L}$ to t is close to 1, i.e. $1 - 1/n \leq \|t - \mathbf{c}\|_K \leq 1$. Indeed, by Babai's Nearest Plane Algorithm, see Lemma 1.3.3, we can estimate $\|t - \mathbf{c}\|_K$ up to a factor 2^n. We can then guess (enumerate over all polynomial number of possibilities) the correct power of $(1 + 1/n)$ of $\|t - \mathbf{c}\|_K$ and scale the lattice and t accordingly.

We now use Kannan's embedding technique. We define the lattice $\mathscr{L}' \subseteq \mathbb{R}^{n+1}$ with the following basis:

$$\tilde{\mathbf{B}} = \begin{pmatrix} \mathbf{B} & t \\ 0 & 1/n \end{pmatrix} \in \mathbb{Q}^{(n+1)\times(n+1)}.$$

On $\mathscr{L}'$ we consider the norm $\|\cdot\|_{K^{+1}}$ induced by $K^{+1} \subseteq \mathbb{R}^{n+1}$, see Definition 3.1. We set $\mathbf{s} := \begin{pmatrix} t - \mathbf{c} \\ 1/n \end{pmatrix} \in \mathbb{R}^{n+1}$. Observe that $1 - 1/n \leq \|\mathbf{s}\|_{K^{+1}} \leq 1$.

We now use the procedure from Theorem 3.1.1 with $R := 1$, norm $\|\cdot\|_{K^+}$ and ε from this theorem, and sample $N := 2$ lattice vectors of length at most a_ε. With probability at least $\frac{1}{2} \cdot 2^{-2\varepsilon(n+1)}$, this succeeds and, in particular, both resulting lattice vectors, $\mathbf{v}_1$ and $\mathbf{v}_2$, are generated according to the distribution $\mathscr{D}_0^{\mathbf{s}}$, see Property (2a). We condition on this event but defer the decision of flipping the coins for $\mathbf{v}_1$ and $\mathbf{v}_2$. Since $\|\mathbf{v}_1\|_{K^{+1}}, \|\mathbf{v}_2\|_{K^{+1}} \leq a_\varepsilon$, they must both lie on one of the $2 \cdot \lfloor a_\varepsilon \rfloor \cdot n + 1$ layers $\mathscr{L}' \cap \{x \in \mathbb{R}^{n+1} \mid x_{n+1} = k/n\}$ for $k \in \{-\lfloor a_\varepsilon \rfloor \cdot n, \ldots, \lfloor a_\varepsilon \rfloor \cdot n\}$. Hence, since $\mathbf{v}_1$ and $\mathbf{v}_2$ are identically and independently distributed, with probability at least $1/(2 \cdot \lfloor a_\varepsilon \rfloor \cdot n + 1)$, $\mathbf{v}_1$ and $\mathbf{v}_2$ both land on the same layer. We again condition on this event. It follows that

$$\mathbf{v}_1 - \mathbf{v}_2 \in (2 \cdot a_\varepsilon) \cdot K \times \{0\}$$

holds with probability at least $\frac{1}{(4\cdot\lfloor a_\varepsilon\rfloor\cdot n+1)}\cdot 2^{-2\varepsilon(n+1)}$. We can now use property (2a). With probability at least $1/4$, instead of returning $\mathbf{v}_1$ and $\mathbf{v}_2$, the procedure returns $\mathbf{v}_1+\mathbf{s}$ and $\mathbf{v}_2$. Their difference is then

$$(\mathbf{v}_1+\mathbf{s})-\mathbf{v}_2 \in (2\cdot a_\varepsilon)\cdot K\times\{0\}+\mathbf{s}.$$

We can rewrite this as

$$(\mathbf{v}_1+\mathbf{s})-\mathbf{v}_2=\begin{pmatrix}\mathbf{u}\\0\end{pmatrix}+\mathbf{s}=\begin{pmatrix}\mathbf{u}-\mathbf{c}\\0\end{pmatrix}+\begin{pmatrix}\tilde{t}\\1/n\end{pmatrix},$$

for some $\mathbf{u}\in\mathscr{L}$. Crucially,

$$\|\mathbf{u}\|_K=\left\|\begin{pmatrix}\mathbf{u}\\0\end{pmatrix}\right\|_{K^{+1}}\leq 2\cdot a_\varepsilon.$$

The lattice vector $\mathbf{c}-\mathbf{u}$ is then the desired approximation. Indeed, by the triangle inequality,

$$\|t-(\mathbf{c}-\mathbf{u})\|_K\leq\|t-\mathbf{c}\|_K+\|\mathbf{u}\|_K\leq 1+2\cdot a_\varepsilon:=\beta_\varepsilon.$$

We set $\alpha_\varepsilon:=(1-1/n)^{-1}\cdot(2\cdot\beta_\varepsilon+1)$ as we have used Lemma 3.1.3 and have scaled the lattice such that $\|t-\mathbf{c}\|_K\geq 1-1/n$. The lattice vector $\mathbf{c}-\mathbf{u}$ is then an α_ε-approximation to the closest lattice vector to t.

As described, this procedure succeeds with probability $2^{-\Omega(\varepsilon)n}$. To boost the probability of success to $1-2^{-n}$, we repeat this procedure $2^{\Omega(\varepsilon)n}$ times starting from where we sampled the two lattice vectors $\mathbf{v}_1$ and $\mathbf{v}_2$. Using Lemma 3.1.4, the time and space requirements follow from those of the procedure from Theorem 3.1.1. This concludes the proof. □

3.2 Approximating the Closest Vector by Sieving in any Norm

We now show how for any pair of norms, $\|\cdot\|_Q,\|\cdot\|_K$, we can solve constant factor approximate CVP_Q using a sieving algorithm with respect to $\|\cdot\|_K$. We will make use of the geometric construction from the previous section, Theorem 2.4.1, and are going to combine this with the randomized procedure from Theorem 3.1.1.

Theorem 3.2.1. *Let $\varepsilon>0$ and $K=-K\subseteq\mathbb{R}^n$ be such that $\tilde{k}(K,\varepsilon)\leq 2^{\beta n}$. Then, for any norm $\|\cdot\|_Q$ and in time $2^{(2\beta+\varepsilon)n}$ and space $2^{(\beta+\varepsilon)n}$, one can compute an α_ε-approximation to CVP_Q on any lattice of rank n. In particular, a constant factor approximation to CVP in any norm can be computed in $2^{0.802n}$ time.*

The pseudocode is given in Figure 3.7.

Proof. We invoke Lemma 3.1.3 to restrict to a lattice of rank and dimension n. This increases the final approximation factor by at most a factor 3. Using Babai's algorithm, Theorem 1.3.3, we scale the lattice such that the distance of the closest lattice lattice vector $\mathbf{c}\in\mathscr{L}$ to t is is between $(1+1/n)^{-1}$ and 1, i.e. $(1+1/n)^{-1}\leq\|t-\mathbf{c}\|_Q\leq 1$. We then compute the linear

transformation $T_\varepsilon(\cdot)$ guaranteed by Theorem 2.4.1 with Q, K and ε from this theorem. We now describe one iteration of the algorithm that succeeds with probability $2^{-\Omega(\varepsilon)n}$.

We sample a point $\tilde{t}$ within $t + Q + c_\varepsilon \cdot T_\varepsilon(K)$ uniformly at random. By Lemma 2.4.1 and with probability at least $2^{-\varepsilon n}$, $\tilde{t}$ is such that $\mathbf{c} \in \tilde{t} + c_\varepsilon \cdot T_\varepsilon(K)$. We condition on this event. For this vector $\tilde{t}$, we define the lattice $\mathscr{L}' \subseteq \mathbb{R}^{n+1}$ of rank $n+1$ with the following basis:

$$\tilde{\mathbf{B}} = \begin{pmatrix} \mathbf{B} & \tilde{t} \\ 0 & 1/n \end{pmatrix} \in \mathbb{Q}^{(n+1)\times(n+1)}.$$

We define the following two vectors in $\mathscr{L}'$:

$$\tilde{\mathbf{t}} := \begin{pmatrix} \tilde{t} \\ 1/n \end{pmatrix} \text{ and } \mathbf{s} := \begin{pmatrix} \tilde{t} - \mathbf{c} \\ 1/n \end{pmatrix}.$$

We consider the norm $\|\cdot\|_{T_\varepsilon(K)^{+1}}$, see Definition 3.1. Since $\mathbf{c} \in \tilde{t} + c_\varepsilon \cdot T_\varepsilon(K)$, it follows that $\|\mathbf{s}\|_{T_\varepsilon(K)^{+1}} \leq c_\varepsilon$.

We now use the procedure from Theorem 3.1.1 with norm $\|\cdot\|_{T_\varepsilon(K)^{+1}}$, radius $R := c_\varepsilon$ and ε from this theorem and sample $N := 2$ lattice vectors of length at most $a_\varepsilon \cdot c_\varepsilon$. With probability at least $\frac{1}{2} \cdot 2^{-2\varepsilon n}$ this succeeds and both lattice vectors are generated according to the distribution $\mathscr{D}_0^{\mathbf{s}}$, see (2a). We condition on this event and denote by $\mathbf{v}_1, \mathbf{v}_2$ the resulting lattice vectors.

Both $\mathbf{v}_1$ and $\mathbf{v}_2$ must both land in a layer of $T_\varepsilon(K)^{+1}$ of the form

$$\mathscr{L}' \cap \{(x, x_{n+1}) \in \mathbb{R}^n \times \mathbb{R} \mid x_{n+1} = \tfrac{k}{n}\} \cap T_\varepsilon(K)^{+1},$$

for some $k \in \{-\lceil a_\varepsilon \cdot c_\varepsilon / n \rceil, \ldots, \lceil a_\varepsilon \cdot c_\varepsilon / n \rceil\}$. Since $N(T_\varepsilon(K), c_\varepsilon \cdot Q) \leq 2^{\varepsilon n}$, each such layer can be covered by at most $2^{\varepsilon n}$ translates of $(a_\varepsilon \cdot c_\varepsilon^2) \cdot Q \times \{0\}$. Hence, $\mathbf{v}_1$ and $\mathbf{v}_2$ must land in one of at most $\tilde{N} := (2 \cdot n \cdot \lceil a_\varepsilon \cdot c_\varepsilon \rceil + 1) \cdot 2^{\varepsilon n}$ translates of $(a_\varepsilon \cdot c_\varepsilon^2) \times Q \times \{0\}$. Since $\mathbf{v}_1, \mathbf{v}_2$ are independently and identically distributed, with probability at least $\tilde{N}^{-2}$, there is a translate of $(a_\varepsilon \cdot c_\varepsilon^2) \cdot Q \times \{0\}$ that holds both $\mathbf{v}_1$ and $\mathbf{v}_2$. In this case, it follows that

$$\mathbf{v}_1 - \mathbf{v}_2 \in (2 \cdot c_\varepsilon^2 \cdot a_\varepsilon) \cdot Q \times \{0\}.$$

We can now use Property (2a) of Theorem 3.1.1. With probability at least 1/4, instead of returning $\mathbf{v}_1$ and $\mathbf{v}_2$, the procedure returns $\mathbf{v}_1 + \mathbf{s}$ instead of $\mathbf{v}_1$ and leaves $\mathbf{v}_2$ unchanged. We condition on this event. Their difference is then

$$(\mathbf{v}_1 + \mathbf{s}) - \mathbf{v}_2 \in (2 \cdot a_\varepsilon \cdot c_\varepsilon^2) \cdot Q \times \{0\} + \mathbf{s}.$$

This is depicted in Figure 3.6.

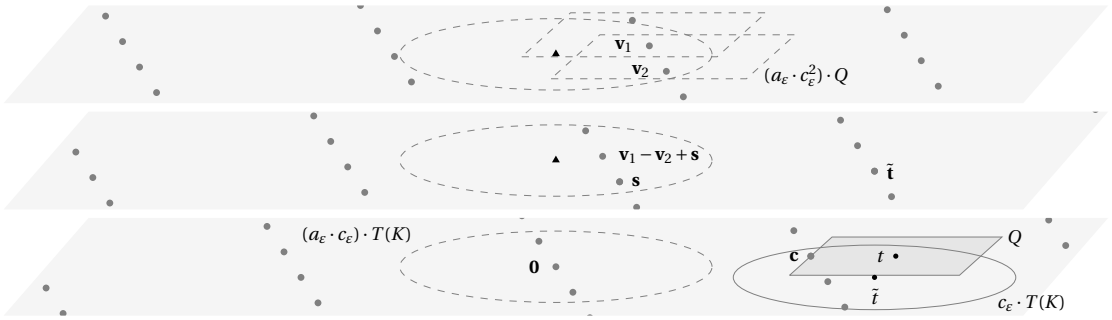

Figure 3.6 – One translate of $(a_\varepsilon \cdot c_\varepsilon^2) \cdot Q \times \{0\}$ holds $\mathbf{v}_1$ and $\mathbf{v}_2$. Sampling $\mathbf{v}_1 + \mathbf{s}$ instead of $\mathbf{v}_1$ results in a good approximation to the vector $\mathbf{s}$.

We can rewrite this as

$$(\mathbf{v}_1 + \mathbf{s}) - \mathbf{v}_2 = \begin{pmatrix} \mathbf{u} \\ 0 \end{pmatrix} + \mathbf{s},$$

for some $\mathbf{u} \in \mathscr{L}$. It follows that

$$\|\mathbf{u}\|_Q = \left\| \begin{pmatrix} \mathbf{u} \\ 0 \end{pmatrix} \right\|_{Q^{+1}} \leq (2 \cdot c_\varepsilon^2 \cdot a_\varepsilon).$$

Rewriting this again, we have that

$$(\mathbf{v}_1 + \mathbf{s}) - \mathbf{v}_2 = \begin{pmatrix} \mathbf{u} - \mathbf{c} \\ 0 \end{pmatrix} + \begin{pmatrix} \tilde{t} \\ 1/n \end{pmatrix}.$$

The lattice vector $\mathbf{c} - \mathbf{u}$ is then the desired approximation. Indeed, by the triangle inequality,

$$\|t - (\mathbf{c} - \mathbf{u})\|_Q \leq \|t - \mathbf{c}\|_Q + \|\mathbf{u}\|_Q \leq 1 + (2 \cdot c_\varepsilon^2 \cdot a_\varepsilon) := \beta_\varepsilon.$$

We set $\alpha_\varepsilon := (1 + 1/n)^{-1} \cdot (2 \cdot \beta_\varepsilon + 1)$ as we have used Lemma 3.1.3 and scaled the lattice so that $\|t - \mathbf{c}\|_K \geq (1 + 1/n)^{-1}$. The lattice vector $\mathbf{c} - \mathbf{u}$ is then an α_ε-approximation to the closest lattice vector to t.

All in all, one such iteration succeeds with probability $2^{-\Omega(\varepsilon)n}$. Hence, repeating this procedure $2^{\Omega(\varepsilon)n}$ times starting from where we sampled $\tilde{t} \in t + Q + c_\varepsilon \cdot T_\varepsilon(K)$ boosts the overall probability of success to $1 - 2^{-n}$. It remains to argue that each iteration of the procedure of Theorem 3.1.1 takes time $O(\tilde{k}(K,\varepsilon)^2) \cdot 2^{\varepsilon n}$ and takes space $O(\tilde{k}(K,\varepsilon)) \cdot 2^{\varepsilon n}$. By Lemma 3.1.4, we have that $\tilde{k}(T_\varepsilon(K)^{+1},\varepsilon) \leq O(\frac{1}{1-\gamma} \cdot \tilde{k}(T_\varepsilon(K),\varepsilon))$. It is then easy to see that $\tilde{k}(T(K),\varepsilon) = \tilde{k}(K,\varepsilon)$ for any linear transformation T, whence the bounds on the running time and space requirement are proven. □

Theorem 3.2.1 suggests two ways to improve the running time for constant factor approximate CVP in any norm.

Input: $\mathscr{L} \subseteq \mathbb{Q}^d$ of rank n, $t \in \mathbb{Q}^d$, $\varepsilon > 0$, $\|\cdot\|_Q, \|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
Use Lemma 3.1.3 to restrict to $d = n$;
Guess $k \in \mathbb{Z}$ so that $(1+1/n)^k \leq \mathrm{dist}_Q(\mathscr{L}, t) \leq (1+1/n)^{k+1}$;
Compute $T_\varepsilon(\cdot)$ from Theorem 2.4.1;
Initialize $\mathbf{v}$ as any nonzero lattice vector;
for $\ell \in \{1, \ldots, 2^{6\varepsilon n}\}$ **do**
  Sample $\tilde{t} \sim t + Q + c_\varepsilon \cdot T(K)$ uniformly;
  Set $\mathscr{L}' := \mathscr{L}(\mathbf{B}')$, where $\mathbf{B}'$ is as in (3.2);
  Sample $\mathbf{v}_1, \mathbf{v}_2 \in \mathscr{L}'$ with Theorem 3.1.1 with $\|\cdot\|_{T(K)}$, $\mathscr{L}'$, ε, $R := c_\varepsilon \cdot (1+1/n)^{k+1}$;
  if $(\mathbf{v}_1 - \mathbf{v}_2)_{n+1} = 1/n$ **then**
    $\mathbf{u} := -(\mathbf{v}_1 - \mathbf{v}_2)_{\leq n} + \tilde{t} \in \mathscr{L}$;
    $\mathbf{v} \leftarrow \arg\min\{\|\mathbf{v}\|_Q, \|\mathbf{u}\|_Q\}$;
  end
end
Output: $\mathbf{v}$

Figure 3.7 – Approximate CVP_Q by sieving with respect to $\|\cdot\|_K$.

The first is to look for convex bodies K such that $\tilde{k}(K, \varepsilon)$ is (provably) smaller than $2^{0.401n}$, the currently best *upper bound* on $\tilde{k}(B_2^n, \varepsilon)$, (KL78). This would immediately imply faster algorithms for constant factor approximate CVP in any norm. The best *lower bound* on $\tilde{k}(B_2^n, \varepsilon)$ is of the order $2^{0.21n}$, which is believed to be tight. Pushing this further, one might even argue that there *could* be a convex body $K = -K$ for which $\tilde{k}(K, \varepsilon) < \tilde{k}(B_2^n, \varepsilon)$. While this is a very optimistic conjecture, quite possibly too optimistic, we note that in this setting there are also lower bounds. In (Tal98), it is shown that for any $\varepsilon \geq 0$ and any convex body K, there is a universal constant $c(\varepsilon) > 0$ only depending on ε such that $\tilde{k}(K, \varepsilon) \geq 2^{c(\varepsilon)n}$. Hence, it would be interesting to explore the connection of the constant $c(\varepsilon)$ for general convex bodies with respect to that of the Euclidean Ball.

Second, we note that in Procedure 3.1.1, we did not use any efficient data-structures to find a close pair of lattice vectors. For any new vector we reduce, we go over *all* vectors in the list, resulting in an overall running time that is *quadratic* in the number of elements in the list. Indeed, for the ℓ_2 norm, it is not known how to devise a data-structure that takes less than quadratic time in this setting. However, for the ℓ_∞ norm, it is possible to devise a data-structure that, in $O(\mathrm{poly}(n))$ time, finds an approximately closest lattice vector, (AM18), resulting in a running time for approximate SVP_∞ that is *linear* in the number of elements in the list. Unfortunately, $\tilde{k}(B_\infty^n, \varepsilon) = 3^n - 1$, so this does not yield any improvement over our approach. But it does raise the question whether there exists a convex body $K = -K$ and a sub-quadratic data-structure to find close pairs with respect to $\|\cdot\|_K$ so that the procedure from Theorem 3.1.1 can be made to work in less than $2^{0.802n}$ time. Combined with our geometric covering approach, this would imply faster algorithms for constant factor approximate CVP in any norm.

3.3 Lattice Sparsification and the Closest Vector Problem

In this section, we are going to describe a simple connection between lattice sparsifiers, as used by Dadush and Kun (DK16) for their $(1+\varepsilon)$-approximation algorithm for CVP in general norms, and the modulus of smoothness of the norm $\|\cdot\|_K$. Using this observation, we will be able to improve the running time of $O(1+1/\varepsilon)^n$ for $(1+\varepsilon)-\mathrm{CVP}_p$ to $O(1+1/\varepsilon)^{n/2}$ for $p \geq 2$ and $O(1+1/\varepsilon)^{n/p}$ for $p \in [1,2[$.

We first sketch Dadush and Kun's approach. Let us denote by $\mathscr{L} \subseteq \mathbb{R}^n$, $t \in \mathbb{R}^n$ and $K = -K$ the lattice, target and norm under consideration. Without loss of generality, $d_K(\mathscr{L}, t) := \min_{\mathbf{v}\in\mathscr{L}} \|\mathbf{v} - t\|_K$ is between $1 - 1/n$ and 1. Their algorithm consists of two subprocedures, `Lattice-Enumerator` and `Lattice-Sparsifier`, see Theorems 3.3.1 and 3.3.3. The algorithm `Lattice-Enumerator` first computes a covering of K using $2^{O(n)}$ M-ellipsoids. Each such M-ellipsoid has the property that it can be covered using $2^{O(n)}$ translates of K. For each M-ellipsoid from this covering, `Lattice-Enumerator` enumerates over all lattice vectors contained in it. Here, the issue is that the number of lattice vectors in such an ellipsoid can be *exponential in the encoding size*, in particular much larger than, say, $n^{O(n)}$. Hence, enumeration over *all* lattice vectors in such an ellipsoid is too costly. The solution is sparsification. For any $\varepsilon > 0$, there is a sparsified sublattice $\mathscr{L}' \subseteq \mathscr{L}$ such that the distance of t to $\mathscr{L}'$ (compared to the distance t to $\mathscr{L}$) only increases by a factor of $1+\varepsilon$, and, any translate of K only contains $2^{O(n)}(1+1/\varepsilon)^n$ lattice vectors of $\mathscr{L}'$. Since every M-ellipsoid covering K can be covered by at most $2^{O(n)}$ translates of K, this also implies that any M-ellipsoid contains at most $2^{O(n)}(1+1/\varepsilon)^n$ lattice vectors of $\mathscr{L}'$. Remarkably, the algorithm `Lattice-Sparsifier` finds such a sublattice in $2^{O(n)}$ time and 2^n space. For the final algorithm, Dadush and Kun apply `Lattice-Sparsifier` to $\mathscr{L}$ and pass the resulting sublattice to `Lattice-Enumerator`. Among the lattice vectors returned by `Lattice-Enumerator`, they output the closest lattice vector to t. By the first property of `Lattice-Sparsifier`, this will be a $(1+\varepsilon)$-approximation to the closest lattice vector in the original lattice. The running time is dominated by the enumeration over all lattice vectors contained in one of the $2^{O(n)}$ M-ellipsoids covering K. By the second property of `Lattice-Sparsifier`, each such ellipsoid contains at most $O(1+1/\varepsilon)^n$ lattice vectors of the sparsified sublattice. The overall run time is thus of order $2^{O(n)}(1+1/\varepsilon)^n$ and the space requirement of order 2^n.

Theorem 3.3.1 (`Lattice-Enumerator`$(K, t, \mathscr{L})$, (DPV11; Dad12a))**.** *Let $\mathscr{L}(A)$ be a lattice and $K \subseteq \mathbb{R}^n$ a convex body. There is a deterministic algorithm that outputs, one by one, all lattice vectors contained inside $t+K$. This takes $2^{O(n)} \cdot G(K, \mathscr{L})$ time and 2^n space.*

Here, $G(K, \mathscr{L})$ denotes the maximal number of lattice vector any translate of K can contain,

$$G(K, \mathscr{L}) := \max_{x\in\mathbb{R}^n} |(K + x) \cap \mathscr{L}|.$$

Definition 3.3.2 (Lattice sparsifier for origin-symmetric K, (DK16))**.** Let $K \subseteq \mathbb{R}^n$ be an origin-symmetric convex body, $\mathscr{L}$ be a n-dimensional lattice and $\delta \geq 0$. A (K, δ) sparsifier for $\mathscr{L}$ is a sublattice $\mathscr{L}' \subseteq \mathscr{L}$ satisfying

1. $\forall x \in \mathbb{R}^n$, $d_K(\mathscr{L}', x) \leq d_K(\mathscr{L}, x) + \delta$,
2. $G(K, \mathscr{L}') \leq O(1 + \frac{1}{\delta})^n$.

Theorem 3.3.3 (`Lattice-Sparsifier`$(\mathscr{L}(A), K, \delta)$, (DK16))**.** *For any lattice $\mathscr{L}$ of rank n, any convex $K = -K \subseteq \mathbb{R}^n$ and any $\delta \geq 0$, we can compute (a basis of) a (K, δ)-sparsifier for $\mathscr{L}$ in (deterministic) $2^{O(n)}$ time and 2^n space.*

We now combine these two algorithms, `Lattice-Enumerator` and `Lattice-Sparsifier`, with the notion of modulus of smoothness to yield improved running times for the approximate closest vector problem.

Definition 3.3.4. The modulus of smoothness of an origin-symmetric convex body K, $\rho_K(\tau) : (0,1) \to (0,1)$, is defined by

$$\rho_K(\tau) := \frac{1}{2} \sup_{\|x\|_K = 1, \|y\|_K = \tau} (\|x+y\|_K + \|x-y\|_K - 2).$$

To see why the modulus of smoothness is relevant for the closest vector problem, denote by $\mathbf{c} \in \mathscr{L}$ a closest vector to the target t and assume the lattice $\mathscr{L}$ is scaled such that $\|t - \mathbf{c}\|_K = 1$. Denote by $\mathscr{L}' \subseteq \mathscr{L}$ the lattice returned by $\texttt{Lattice} - \texttt{Sparsifier}(\mathscr{L}, K, \varepsilon)$. By the first property of Definition 3.3.2, there is some vector $\mathbf{u} \in \mathscr{L}'$ with $\|\mathbf{u} - \mathbf{c}\|_K \leq \varepsilon$. Since $\mathbf{c}$ is the closest lattice vector to t:

$$\|t - \mathbf{c} - (\mathbf{u} - \mathbf{c})\|_K, \|t - \mathbf{c} + (\mathbf{u} - \mathbf{c})\|_K \geq 1.$$

It follows that

$$\begin{aligned} \|t - \mathbf{c} - (\mathbf{u} - \mathbf{c})\|_K &\leq 2 - \|t - \mathbf{c} + (\mathbf{u} - \mathbf{c})\|_K + 2 \cdot \rho_K(\|\mathbf{u} - \mathbf{c}\|_K) \\ &\leq 1 + 2 \cdot \rho_K(\varepsilon). \end{aligned}$$

By subadditivity of the norm, $\rho_K(\tau)$ is at most linear in τ. Hence, $\|t - \mathbf{u}\|_K \leq 1 + 2 \cdot \varepsilon$.

However, whenever the (exponent of) the modulus of smoothness is larger than 1, this bound is much stronger than subadditivity suggests. Hence, we can sparsify more aggressively (hence enumerating over fewer lattice vectors) while retaining the metric information. These properties are illustrated in Figure 3.8 and resumed in the following lemma.

Lemma 3.3.5 ((NV22))**.** *Let $K = -K$ be a convex body with $\rho_K(\tau) \leq C \cdot \tau^q$, $q \geq 1$ and assume that $t + K$ contains at least one lattice vector but $t + (1 - 1/n) \cdot K$ is lattice-point-free. Then $\mathscr{L}' := \texttt{Lattice} - \texttt{Sparsifier}(\mathscr{L}, K, \varepsilon) \subseteq \mathscr{L}$ is a $2 \cdot (1 + 1/n) \cdot C \cdot \varepsilon^q$ sparsifier and consequently*

$$d_K(\mathscr{L}', t) \leq (1 + 2 \cdot (1 - 1/n)^{-1} \cdot C \cdot \varepsilon^q) \cdot d_K(\mathscr{L}, t).$$

The proof is a straight forward adaption of the argumentation given above. This then readily yields the following theorem.

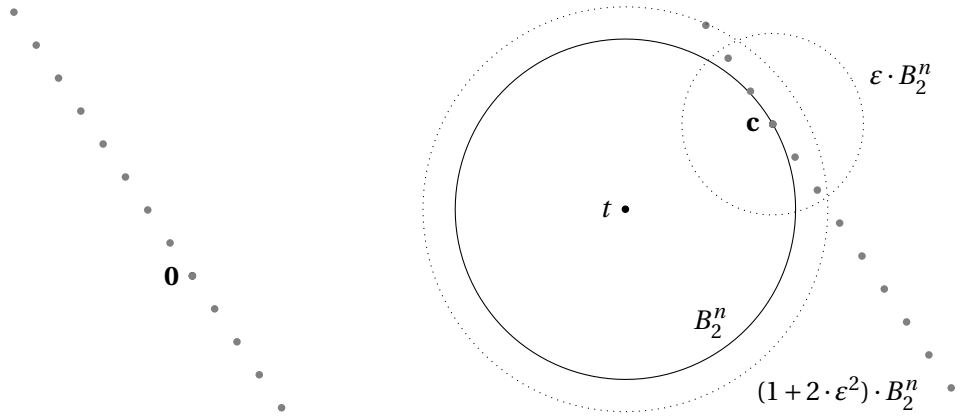

Figure 3.8 – The modulus of smoothness for the ℓ_2 norm, $\rho_{B_2^n}(\tau)$, is bounded by τ^2. $\texttt{Lattice}-\texttt{Sparsifier}(\mathscr{L}, B_2^n, \varepsilon)$ yields a $(1+2\cdot\varepsilon^2)$-approximation to the closest vector problem.

Theorem 3.3.6 ((NV22))**.** *For any norm $\|\cdot\|_K : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ with modulus of smoothness $\rho_K(\tau) \leq C\cdot\tau^q$, any lattice $\mathscr{L}$ with $t \in span(\mathscr{L})$ and any $\varepsilon > 0$, one can compute a $(1+\varepsilon)$-approximation to the closest vector problem in $2^{O(n)}(1+C/\varepsilon)^{n/q}$ time and 2^n space.*

Proof. Let $\tilde{K} := K \cap \mathrm{span}(\mathscr{L})$. Since $\tilde{K} \subseteq K$, $\rho_{\tilde{K}}(\tau) \leq \rho_K(\tau)$. Also, it is trivial to check that the modulus of smoothness does not change under linear transformations. Thus, without loss of generality, we can assume that $K \subseteq \mathbb{R}^n$ and $B_2^n \subseteq K \subseteq (n+1)\cdot B_2^n$. Thus, by Babai's Nearest Plane Algorithm, Theorem 1.3.3, we can approximate the distance of t to $\mathscr{L}$ up to a factor of 2^n. Hence, up to enumerating over a polynomial number of possibilities, we can also assume that $1 - 1/n \leq d_K(\mathscr{L}, t) \leq 1$.

We set $\mathscr{L}' := \texttt{Lattice}-\texttt{Sparsifier}(\mathscr{L}, K, (\varepsilon/(2\cdot(1-1/n)^{-1}C))^{1/q})$, this takes time $2^{O(n)}$ and 2^n space. By Lemma 3.3.5,

$$d_K(\mathscr{L}', t) \leq (1+\varepsilon)\cdot d_K(\mathscr{L}, t).$$

We then call $\texttt{Lattice}-\texttt{Enumerator}(K, t, \mathscr{L})$ and return the closest lattice vector to t. This takes $2^{O(n)}(1+C/\varepsilon)^{n/q}$ time and 2^n space. □

For the ℓ_p norm, concrete estimates on the modulus of smoothness are available.

Theorem 3.3.7 (Modulus of smoothness for ℓ_p spaces, (Lin63))**.** *We have*

$$\rho_{\ell_p}(\tau) = \begin{cases} (((1+\tau)^p + (1-\tau)^p)/2)^{1/p} - 1, & \text{if } 2 \leq p < \infty. \\ (1+\tau^p)^{1/p} - 1, & \text{if } 1 \leq p \leq 2. \end{cases}$$

These expressions are bounded by $2^p\cdot\tau^2$ and τ^p/p respectively. We thus obtain the following corollary.

Corollary 3.3.8 ((NV22))**.** *Let $\mathscr{L} \subseteq \mathbb{R}^d$ a lattice of rank n and $t \in span(\mathscr{L})$. One can compute a $(1+\varepsilon)$-approximation to CVP_p in $2^{O(2^p \cdot n)} \cdot (1+1/\varepsilon)^{n/p}$ time and 2^n space if $p \geq 2$, and in $2^{O(n)} \cdot (1+1/\varepsilon)^{n/p}$ time and 2^n space if $p \in [1,2[$.*

4 Covering Numbers and Ellipsoids

This final chapter gives an overview and proofs on the geometric results used in the two previous chapters.

Section 4.1 presents a simple proof on the relationship of volume estimates and covering numbers, see Lemma 4.1.1. The proof follows (Nas14), we include it here for completeness. Combined with elementary polyhedral techniques, this is then used in Lemma 4.1.2 to derive upper bounds on $N(n^{1/p-1/q} \cdot B_p^d, t \cdot B_q^d)$ for varying t, as used in Chapter 2.

Section 4.2 outlines the construction of the linear transformation from Theorem 2.4.1. This linear transformation follows from the existence of certain ellipsoids associated to convex bodies. For any convex body $L = -L$ and any $\varepsilon > 0$, there exists an *ellipsoid* $\mathscr{E}$ such that L can be covered by fewer than $2^{\varepsilon n}$ translates of L and, conversely, $\mathscr{E}$ can be covered by fewer than $2^{\varepsilon n}$ translates of L. The linear transformation from Theorem 2.4.1 can be obtained by first constructing such ellipsoids for K and Q and then composing the respective linear transformations sending these ellipsoids to B_2^n, the unit norm ball. The construction of these ellipsoids and that of the linear transformation is described in Theorem 4.2.1. To put this into perspective, we note that these ellipsoids are a variation of Milman's M-ellipsoids and Pisier's α-regular ellipsoids. Milman's M-ellipsoids do guarantee a weaker covering property, $N(L, \mathscr{E}_M), N(\mathscr{E}_M, L) \leq 2^{O(n)}$, where the constant in the exponent is universal, (Mil88). The construction of these ellipsoids were subsequently made algorithmic by Dadush, Peikert and Vempala (DPV11). On the other hand, Pisier showed that for any $L = -L$, there exists an ellipsoid $\mathscr{E}_\alpha$ such that, slightly oversimplified, $N(L, t \cdot \mathscr{E}_\alpha), N(\mathscr{E}_\alpha, t \cdot L) \leq 2^{\Omega(t^{-1})n}$, (Pis89). Hence, up to scaling, this ellipsoid fits the premise from Theorem 2.4.1 for *any* $\varepsilon > 0$.

4.1 Volume Estimates and Coverings

Lemma 4.1.1 ((Nas14))**.** *Let $K, Q \subseteq \mathbb{R}^n$ be convex and origin-symmetric, and assume that*

$$\mathrm{Vol}(K + Q) \leq M \cdot \mathrm{Vol}(Q).$$

Then, $N(K,Q) \leq \text{poly}(n) \cdot M \cdot \ln(M)$. *Such a covering can be constructed in* $\text{poly}(n) \cdot M \cdot \ln(M)$ *time with high probability.*

Proof. To obtain the covering of K by translates of K we will proceed by random sampling. Specifically, it will be sufficient to sample $N := \text{poly}(n) \cdot M \cdot \ln(M)$ points $x_1, \ldots, x_N \sim K + Q$ uniformly at random and place translates of Q around them. With high probability, this will cover any point inside K.

Already, we observe that the volume inequality is invariant under linear transformations, i.e. $\text{Vol}(T(K) + T(Q)) \leq \text{Vol}(T(Q))$. Hence, without loss of generality, we may assume that $B_2^n \subseteq K \subseteq \sqrt{n} \cdot B_2^n$. It also follows from the volume inequality that $K \subseteq M \cdot Q$. Indeed, otherwise, we could pack at least M disjoint translates of Q inside $K + Q$.

We now discretize K by a fine mesh of side length $(n^{3/2} \cdot M)^{-1}$. We define the point set $\mathscr{P}$ by

$$\mathscr{P} := \left\{ p \in (n^{3/2} \cdot M)^{-1} \cdot \mathbb{Z}^n \mid p \in K \right\}.$$

We will show that $|\mathscr{P}| = O(\text{poly}(n) \cdot M)^n$ by a volume argument. Since

$$\mathscr{P} + (2 \cdot n^{3/2} \cdot M)^{-1} \cdot B_\infty^n \subseteq K + n^{-1} \cdot B_2^n \subseteq (1 + 1/n) \cdot K,$$

the volume of $\mathscr{P} + (2 \cdot n^{3/2} \cdot M)^{-1} \cdot B_\infty^n$ is at most $e \cdot \text{Vol}(K)$. On the other hand, for any $p_1 \neq p_2 \in \mathscr{P}$,

$$\text{Vol}(p_1 + (2 \cdot n^{3/2} \cdot M)^{-1} \cdot B_\infty^n \cap p_2 + (2 \cdot n^{3/2} \cdot M)^{-1} \cdot B_\infty^n) = 0,$$

since these two sets can only intersect on the boundary. Hence

$$|\mathscr{P}| \leq \frac{\text{Vol}(\mathscr{P} + (2 \cdot n^{3/2} \cdot M)^{-1} \cdot B_\infty^n)}{\text{Vol}((2 \cdot n^{3/2} \cdot M)^{-1} \cdot B_\infty^n)} \leq \frac{e \cdot \text{Vol}(K)}{(2 \cdot n^2 \cdot M)^{-n} \cdot \text{Vol}(K)} = O(n^2 \cdot M)^n,$$

where in the last inequality we have used that $(1/\sqrt{n}) \cdot B_\infty \subseteq B_2^n \subseteq K$.

Now, for any point $p \in \mathscr{P}$,

$$\Pr_{t \sim K+Q}[p \in t + (1 - 1/n) \cdot Q] = \Pr_{t \sim K+Q}[t \in p + (1 - 1/n) \cdot Q] = \frac{\text{Vol}((1 - 1/n)^{-1} \cdot Q)}{\text{Vol}(K + Q)} \geq \Omega(M^{-1}).$$

On the other hand, every point $x \in K$ has at most distance $(n^{3/2} \cdot M)^{-1}$ from $\mathscr{P}$ (with respect to $\|\cdot\|_K$). Since $\frac{1}{M} \cdot K \subseteq Q$, whenever we cover all points of $\mathscr{P}$ by N translates of $(1 - 1/n) \cdot Q$, then enlarging all such translates by an additive factor of $\frac{1}{n} \cdot Q$ yields a full covering of K.

Hence, we have the classic SETCOVER Problem. The ground set consists of all points in $\mathscr{P}$ and, for every $t \in K + Q$, the sets are of the form $\mathscr{P} \cap (t + (1 - 1/n) \cdot Q)$. For any point $p \subseteq \mathscr{P}$ is covered with probability $\Omega(M^{-1})$ by a random set of the form $t + (1 - 1/n) \cdot Q$ (where the randomness is with respect to $t \sim K + Q$). Hence, by the Union Bound, picking $n \cdot \ln(|\mathscr{P}|) \cdot M = \text{poly}(n) \cdot M \cdot \ln(M)$ many such sets covers all points of $\mathscr{P}$ with probability at least $1 - 2^{-n}$. □

Lemma 4.1.2. *The following inequality holds*

$$\mathrm{Vol}(d \cdot B_1^d + t \cdot B_\infty^d) \leq 2^{O(t^{-1/2} d)} \cdot \mathrm{Vol}(t \cdot B_\infty^d).$$

Consequently, for any $p \leq q$ and for any $\varepsilon > 0$, there exists some $\gamma_\varepsilon = O(\varepsilon^{-2})$ such that

$$N(d^{1/p-1/q} B_p^d, \gamma_\varepsilon \cdot B_q^d) \leq 2^{\varepsilon d}.$$

Proof. For the estimate on the covering number, we simplify matters by only considering B_1^d and B_∞^d. Indeed, by Hölder's inequality,

$$d^{-1/q} \cdot B_\infty^d \subseteq B_q^d \quad \text{and} \quad d^{1/p-1/q} \cdot B_p^d \subseteq d^{1-1/q} \cdot B_1^d.$$

Hence, for any $\gamma > 0$,

$$N(d^{1/p-1/q} \cdot B_p^d, \gamma \cdot B_q^d) \leq N(d^{1-1/q} \cdot B_1^d, \gamma \cdot d^{-1/q} \cdot B_\infty^d) = N(d \cdot B_1^d, \gamma \cdot B_\infty^d).$$

By the preceding lemma, Lemma 4.1.1, it is sufficient to show that for $\gamma_\varepsilon = O(\varepsilon^{-2})$ we have that

$$\mathrm{Vol}(d \cdot B_1^d + \gamma_\varepsilon \cdot B_\infty^d) \leq 2^{\varepsilon d} \cdot \mathrm{Vol}(\gamma_\varepsilon \cdot B_\infty^d). \tag{4.1}$$

We now derive a closed formula for the volume of $d \cdot B_1^d + t \cdot B_\infty^d$, for some parameter $t \in \mathbb{R}_{\geq 0}$. This method was pointed out to us by Matthias Schymura. We first *decompose* the boundary of $d \cdot B_1^d + t \cdot B_\infty^d$ into ($(d-1)$-dimensional) *facets* $F_i^{(d-1)}$. This decomposition of $d \cdot B_1^d + t \cdot B_\infty^d$ overlaps only on the boundaries of these cones (on a set of measure 0), meaning

$$\mathrm{Vol}_d(d \cdot B_1^d + t \cdot B_\infty^d) = \sum_i \mathrm{Vol}_d(\mathrm{conv}(\mathbf{0}, F_i^{d-1})), \tag{4.2}$$

where the right-hand side runs over all $(d-1)$-dimensional facets of $d \cdot B_1^d + t \cdot B_\infty^d$, see Figure 4.1.

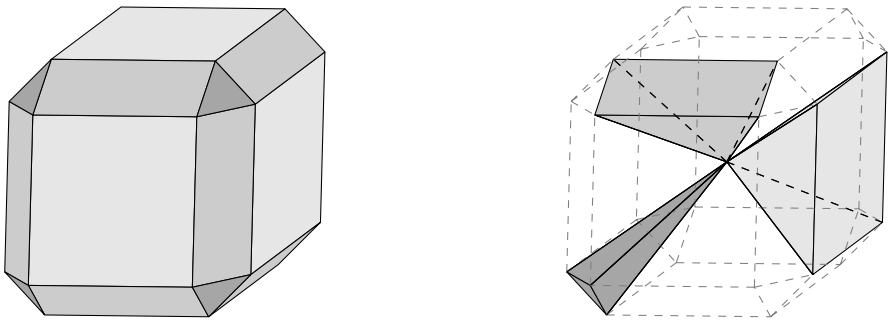

Figure 4.1 – The Minkowski sum $t \cdot B_\infty^d + B_1^d$ and part of its decomposition according to Equation 4.2.

We recall the definition of a facet, and, more generally, a face of a polytope. For a (full-dimensional) polytope $P \subseteq \mathbb{R}^d$, the face (of P) with respect to the objective $w \in \mathbb{R}^d \setminus \{\mathbf{0}\}$ is

defined as $F_w(P) := \{y \in P \mid w^T y \geq w^T x \text{ for all } x \in P\}$. A facet is an inclusion-wise maximal face. By definition, P is not a face. Hence, a facet is $(d-1)$-dimensional.

Let us now identify the facets of $d \cdot B_1^d + t \cdot B_\infty^d$. For any $w = (w_1, \ldots, w_d) \in \mathbb{R}^d$, we define $\operatorname{sgn}(w) := (\operatorname{sgn}(w_1), \ldots, \operatorname{sgn}(w_d)) \in \{-1,0,1\}^d$. It is easily checked that for any $w \in \mathbb{R}^d \setminus \{\mathbf{0}\}$:

$$F_w(d \cdot B_1^d + t \cdot B_\infty^d) \subseteq F_{\operatorname{sgn}(w)}(d \cdot B_1^d + t \cdot B_\infty^d).$$

Since facets are maximal faces, every facet can be written of the form $F_w(d \cdot B_1^d + t \cdot B_\infty^d)$ for some $w \in \{-1,0,1\}^d \setminus \{\mathbf{0}\}$. This simplifies (4.2) to

$$\operatorname{Vol}_d(d \cdot B_1^d + t \cdot B_\infty^d) = \sum_{w \in \{-1,0,1\}^d \setminus \{\mathbf{0}\}} \operatorname{Vol}_d(\operatorname{conv}(\mathbf{0}, F_w(d \cdot B_1^d + t \cdot B_\infty^d))). \tag{4.3}$$

To simplify the terms on the right-hand side, we use that

$$F_w(d \cdot B_1^d + t \cdot B_\infty^d) = F_w(d \cdot B_1^d) + F_w(t \cdot B_\infty^d).$$

This holds for arbitrary polytopes P and Q, see (BS18). Using this with $w_i^T := (\underbrace{1, \ldots, 1}_{i \text{ times}}, 0, \ldots, 0)$:

$$\begin{aligned} F_{w_i}(d \cdot B_1^d) &= d \cdot \operatorname{conv}(e_1, \ldots, e_i), \\ F_{w_i}(t \cdot B_\infty^d) &= (\underbrace{t, \ldots, t}_{i \text{ times}}, 0, \ldots, 0) + [-t \cdot e_{i+1}, t \cdot e_{i+1}] + \ldots + [-t \cdot e_d, t \cdot e_d]. \end{aligned}$$

Clearly, $F_{w_i}(d \cdot B_1^d + t \cdot B_\infty^d)$ is $(d-1)$-dimensional, hence it is a facet. Since $F_{w_i}(d \cdot B_1^d)$ and $F_{w_i}(t \cdot B_\infty^d)$ are orthogonal to each other, we obtain

$$\begin{aligned} \operatorname{Vol}_{d-1}(F_{w_i}(d \cdot B_1^d + t \cdot B_\infty^d)) &= \operatorname{Vol}_{i-1}(d \cdot B_1^d) \cdot \operatorname{Vol}_{d-i}([-t \cdot e_{i+1}, t \cdot e_{i+1}] + \ldots + [-t \cdot e_d, t \cdot e_d]) \\ &= \tfrac{d^i}{i!} \cdot (2 \cdot t)^{d-i}. \end{aligned}$$

We now compute the ℓ_2 distance of $\mathbf{0}$ to the affine hull of $F_{w_i}(d \cdot B_1^d) + F_{w_i}(t \cdot B_\infty^d)$. The latter belongs to the hyperplane $\{x \in \mathbb{R}^d \mid w_i^T \cdot x = t \cdot i + d\}$. Since $\|w_i\|_2 = \sqrt{i}$, the height formed by the pyramid with base $F_{w_i}(d \cdot B_1^d) + F_{w_i}(t \cdot B_\infty^d)$ and apex $\mathbf{0}$ equals $t \cdot \sqrt{i} + d/\sqrt{i}$. Since the d-dimensional pyramid with ($(d-1)$-dimensional) A and height h has volume $A \cdot h/d$, we obtain

$$\operatorname{Vol}_d(\operatorname{conv}(\mathbf{0}, F_{w_i}(d \cdot B_1^d + t \cdot B_\infty^d))) = \frac{d^i \cdot (2 \cdot t)^{d-i}}{i!} \cdot \frac{t \cdot \sqrt{i} + d/\sqrt{i}}{d}.$$

Note that any $w \in \{-1,0,1\}^d$ with exactly i 1's or -1's defines a combinatorially equivalent facet identical to the one above, with identical volumes. Hence, we can rewrite the right-hand side of (4.3) as

$$\operatorname{Vol}_d(d \cdot B_1^d + t \cdot B_\infty^d) = \sum_{i \geq 1}^{d} 2^i \cdot \binom{d}{i} \cdot \frac{d^i \cdot (2 \cdot t)^{d-i}}{i!} \cdot \frac{t \cdot \sqrt{i} + d/\sqrt{i}}{d}.$$

To show (4.1), we are only interested in upper bounding the ratio between $\mathrm{Vol}_d(d \cdot B_1^d + t \cdot B_\infty^d)$ and $\mathrm{Vol}_d(t \cdot B_\infty^d) = (2t)^d$. Since t can be taken smaller than d, we see that

$$\frac{\mathrm{Vol}_d(d \cdot B_1^d + t \cdot B_\infty^d)}{\mathrm{Vol}_d(t \cdot B_\infty^d)} = O(\mathrm{poly}(d)) \cdot \sum_{i \geq 1}^{d} \binom{d}{i} \cdot \frac{d^i}{i! \cdot t^i}.$$

By Stirling's approximation, $n! \approx_{\mathrm{poly}(n)} (\frac{n}{e})^n$, and the estimate $\binom{n}{k} \leq (\frac{n \cdot e}{k})^k$, we obtain

$$\frac{\mathrm{Vol}_d(d \cdot B_1^d + t \cdot B_\infty^d)}{\mathrm{Vol}_d(t \cdot B_\infty^d)} \quad \leq \quad O(\mathrm{poly}(d)) \cdot \sum_{i \geq 1}^{d} \left(\frac{e \cdot d}{i \cdot \sqrt{t}} \right)^{2i}.$$

Taking the derivative with respect to i for some fixed t, we see that the function inside the sum is maximized when $i = \frac{d}{\sqrt{t}}$. Plugging this back in, we obtain

$$\frac{\mathrm{Vol}_d(d \cdot B_1^d + t \cdot B_\infty^d)}{\mathrm{Vol}_d(t \cdot B_\infty^n)} \quad \leq \quad O(\mathrm{poly}(d)) \cdot 2^{\Theta(t^{-1/2} \cdot d)}.$$

Hence, setting $t = \gamma_\varepsilon = O(\varepsilon^{-2})$, (4.1) follows. □

4.2 Computing the Linear Transformation

In this section we discuss the construction of the linear transformation $T_\varepsilon(\cdot)$ from Theorem 2.4.1. To prove it, we make a detour through the Euclidean norm ball. We will show that for any given symmetric convex body $K \subseteq \mathbb{R}^n$ and any fixed $\varepsilon > 0$, one can construct an ellipsoid $\mathscr{E}$ such that K can be covered by fewer than $2^{\varepsilon n}$ translates of $\mathscr{E}$, and, conversely, $\mathscr{E}$ can be covered by fewer than $2^{\varepsilon n}$ translates of K. Up to a linear transformation sending $\mathscr{E}$ to B_2^n, we resume these properties in the following theorem.

Theorem 4.2.1. *For any symmetric convex body $K \subseteq \mathbb{R}^n$ and for any $\varepsilon > 0$, there exists an (invertible) linear transformation $T_\varepsilon : \mathbb{R}^n \to \mathbb{R}^n$ and some constant $c_\varepsilon = \widetilde{O}(\varepsilon^{-3})$ such that*

1. $\mathrm{Vol}(T_\varepsilon(K) + B_2^n) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(B_2^n)$, *and,*
2. $\mathrm{Vol}(B_2^n + c_\varepsilon \cdot T_\varepsilon(K)) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(c_\varepsilon \cdot T_\varepsilon(K))$.

In particular, $N(T_\varepsilon(K), B_2^n), N(B_2^n, c_\varepsilon \cdot T_\varepsilon(K)) \leq 2^{\varepsilon n}$.
Provided K is given by a weak separation oracle, the linear transformation $T_\varepsilon(\cdot)$ can be computed in (randomized) $n^{O(\log(n))}$ time.

We note that that the ellipsoid $T_\varepsilon^{-1}(B_2^n)$ is a special case of an M-ellipsoid for K, a key concept from convex geometry. In particular, this already yields Theorem 2.4.1 when $K = B_2^n$. To obtain the full theorem for an arbitrary pair of conves bodies K and Q, we will use Theorem 4.2.1 twice, once with K and B_2^n and once with Q and B_2^n.

Proof of Theorem 2.4.1 using Theorem 4.2.1. Let $T^Q_{\varepsilon/2}(\cdot), T^K_{\varepsilon/2}(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ be the linear transformations guaranteed by Theorem 4.2.1 and $c_{\varepsilon/2}$ the corresponding constant. We will show that we can set $T_\varepsilon(\cdot) := (T^Q_{\varepsilon/2})^{-1} \circ T^K_{\varepsilon/2}(\cdot)$ and, slightly abusing notation, $c_\varepsilon := c_{\varepsilon/2}$. To see this, set $\tilde{Q} := T^Q_{\varepsilon/2}(Q)$ and $\tilde{K} := T^K_{\varepsilon/2}(K)$. We use the properties provided by Theorem 4.2.1 to bound

$$\begin{aligned}\mathrm{Vol}(\tilde{Q} + c_{\varepsilon/2}\cdot\tilde{K}) &\leq N(\tilde{Q} + c_{\varepsilon/2}\cdot\tilde{K}, B_2^n + c_{\varepsilon/2}\cdot\tilde{K})\cdot\mathrm{Vol}(B_2^n + c_{\varepsilon/2}\cdot\tilde{K}) \\ &\overset{(*)}{\leq} \underbrace{N(\tilde{Q}, B_2^n)}_{\leq 2^{\varepsilon/2n}} \cdot \underbrace{\mathrm{Vol}(B_2^n + c_{\varepsilon/2}\cdot\tilde{K})}_{\leq 2^{\varepsilon/2n}\cdot\mathrm{Vol}(c_{\varepsilon/2}\cdot\tilde{K})} \leq 2^{\varepsilon n}\cdot\mathrm{Vol}(c_{\varepsilon/2}\cdot\tilde{K}).\end{aligned}$$

In $(*)$ we use the fact that for any symmetric convex bodies A, B, C one has $N(A+C, B+C) \leq N(A,B)$ as any covering of A with translates of B implies a covering of $A+C$ with translates of $B+C$.

Applying $(T^Q_{\varepsilon/2})^{-1}$ to both $\tilde{Q} + c_{\varepsilon/2}\cdot\tilde{K}$ and $c_{\varepsilon/2}\cdot\tilde{K}$, we obtain:

$$\begin{aligned}\mathrm{Vol}(Q + c_{\varepsilon/2}\cdot T(K)) &= \det(T^Q_{\varepsilon/2})^{-1}\cdot\mathrm{Vol}(\tilde{Q} + c_{\varepsilon/2}\cdot\tilde{K}) \\ &\leq \det(T^Q_{\varepsilon/2})^{-1}\cdot 2^{\varepsilon n}\cdot\mathrm{Vol}(c_{\varepsilon/2}\cdot\tilde{K}) \\ &= 2^{\varepsilon n}\cdot\mathrm{Vol}(c_{\varepsilon/2}\cdot T(K)).\end{aligned}$$

Slightly abusing notation by setting $c_\varepsilon \leftarrow c_{\varepsilon/2}$, we obtain the first volume inequality. The other volume inequality, $\mathrm{Vol}(T(K) + c_{\varepsilon/2}\cdot Q) \leq 2^{\varepsilon n}\cdot\mathrm{Vol}(c_{\varepsilon/2}\cdot Q)$, is analogous, simply exchange $\tilde{Q}$ and $\tilde{K}$ in the inequalities above. The translative covering inequalities are then a simple consequence of these two volume inequalities. By Lemma 4.1.1 in the previous section, it follows that $N(T_\varepsilon(K), B_2^n), N(B_2^n, c_\varepsilon\cdot T_\varepsilon(K)) \leq \mathrm{poly}(n)\cdot 2^{\varepsilon n} = 2^{\varepsilon n + o(n)}$. □

We now discuss the construction of the ellipsoid / linear transformation from Theorem 4.2.1. This construction is based on an iterative procedure called *isomorphic symmetrization*. It is taken almost verbatim from the proof of existence of M-ellipsoids due to Milman (Mil88). In particular, this technique has been made algorithmic by Dadush and Vempala (DV13) to obtain (deterministic) algorithms for volume computation. Our contribution is largely the observation that this procedure can be stopped earlier to yield the desired properties. For a detailed overview on this procedure and on convex geometry in general, we refer to (AAGM15), see also the wonderful lecture notes of Thomas Rothvoss (Rot21).

Proof of Theorem 4.2.1. We first introduce the *polar* (or *dual*) K° of K. Given $K \subseteq \mathbb{R}^n$, we define

$$K^\circ := \left\{x \in \mathbb{R}^n \mid x^T y \leq 1,\ \forall y \in K\right\}.$$

Whenever $K = -K \subseteq \mathbb{R}^n$ is full dimensional and with $\mathbf{0}$ in its interior, so is K°. Note that applying a linear invertible transformation A to K transforms K° by A^{-1}, i.e. $(A\cdot K)^\circ = A^{-1}\cdot K^\circ$. We can now define the *M-values* $M(K)$ and $M(K^\circ)$ of K and K° respectively.

$$M(K) = \mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|x\|_K] \text{ and } M(K^\circ) = \mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|x\|_{K^\circ}].$$

Here $\mathbb{S}^{n-1} := \{x \in \mathbb{R}^n \mid \|x\|_2 = 1\}$ is the sphere. These quantities can be estimated to arbitrary precision using samples from K and K° respectively.

We can now state the celebrated MM° *estimate* which follows from combining results of Pisier (Pis80) and of Figiel and Tomczak-Jaegermann (FTJ79). For any symmetric convex body $K \subseteq \mathbb{R}^n$, there is a linear transformation $T : \mathbb{R}^n \to \mathbb{R}^n$ and a universal constant C such that

$$M(T(K)) \cdot M((T(K))^\circ) \leq C \cdot \log(1 + d(K, B_2^n)). \tag{4.4}$$

Here, $d(K, B_2^n)$ is the *Banach-Mazur distance* of K to the Euclidean ball. Specifically, it is the smallest number s such that $B_2^n \subseteq A \cdot K \subseteq s \cdot B_2^n$ for some affine map $A : \mathbb{R}^n \to \mathbb{R}^n$. It is known that for any symmetric convex body $K \subseteq \mathbb{R}^n$, one always has $d(K, B_2^n) \leq \sqrt{n}$ (Joh48).

The linear transformation $T : \mathbb{R}^n \to \mathbb{R}^n$ in (4.4) is obtained by solving the following *convex* program:

$$\begin{gathered} \max \det(A) \\ \mathbb{E}_{x \in \gamma_n}\left[\|A(x)\|_K^2\right]^{1/2} \leq 1 \\ A \in \mathbb{R}^{n \times n} \text{ symmetric positive-definite.} \end{gathered} \tag{4.5}$$

Here, γ_n denotes the *Gaussian distribution* on $\mathbb{R}^n$ with density function given by $\frac{1}{(2\pi)^{n/2}} \cdot e^{-\|x\|_2^2/2}$. This program can be solved in (randomized) polynomial time within arbitrary precision. We justify this after the description of the procedure. The linear transformation $T(\cdot)$ is then the *inverse* of $\bar{A}$, $\bar{A}^{-1}$, an (approximately) optimal solution to the convex program. This is justified in detail in Lemma 3.2 of (DV13), we give a brief overview on their proof below.

We can now define an iterative procedure. To initialize it, we set $K_0 \leftarrow K$ and find the linear transformation T_0 (guaranteed by the MM° estimate) such that

$$M(T_0(K_0)) \cdot M((T_0(K_0))^\circ) \leq C \cdot \log(1 + d(K_0, B_2^n)).$$

We now set $\alpha_0 := \max\{d(K_0, B_2^n)^{1/4}, \varepsilon^{-1/2}\}$. It is sufficient to have a 2-approximation for α_0 (and for $\alpha_1, \ldots$, for later iterations). Such an approximation for α_0 and the subsequent α_i can be guessed or enumerated, we briefly comment on this after the description of the procedure.

We now set $R_{\text{out}} := (M((T_0(K_0))^\circ) \cdot \alpha_0$, $R_{\text{in}} := (M(T_0(K_0)) \cdot \alpha_0)^{-1}$, and define the next convex body

$$K_1 = \text{conv}\Big(\big(T_0(K_0) \cap R_{\text{out}} \cdot B_2^n\big) \cup R_{\text{in}} \cdot B_2^n\Big).$$

This new body K_1 is contained in the ball of radius R_{out} and contains the ball of radius R_{in}, see Figure 4.2. In particular, its Banach-Mazur distance to the Euclidean ball has dropped to (at most) $R_{\text{out}}/R_{\text{in}}$. Taking into account that we only know a 2-approximation for α_0 and slightly

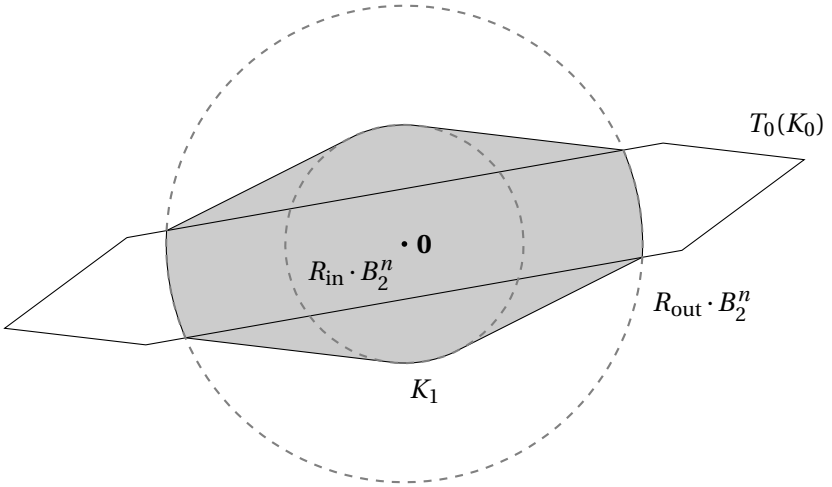

Figure 4.2 – One iteration of the isomorphic symmetrization. The Banach-Mazur distance of K_1 to the Euclidean ball is at most $R_{\text{out}}/R_{\text{in}}$.

abusing notation, this can be bounded in terms of the Banach-Mazur distance $d(K_0, B_2^n)$.

$$\begin{aligned} R_{\text{out}}/R_{\text{in}} = M(T_0(K_0)) \cdot M(T_0(K_0^\circ)) \cdot (2\cdot\alpha_0)^2 &\leq C \cdot \log(1+\alpha_0^4) \cdot (2\cdot\alpha_0)^2 \\ &= O(\log(1+d(K_0,B_2^n)) \cdot d(K_0,B_2^n)^{1/2}). \end{aligned}$$

Crucially, we have the following volume estimate. For any symmetric convex body $P \subseteq \mathbb{R}^n$,

$$2^{-Cn/\alpha_0^2} \cdot \text{Vol}(K_1 + P) \overset{(*)}{\leq} \text{Vol}(T_0(K_0) + P) \overset{(**)}{\leq} 2^{Cn/\alpha_0^2} \cdot \text{Vol}(K_1 + P). \tag{4.6}$$

We briefly outline the proof for these volume estimates. For any symmetric convex bodies $L, T \subseteq \mathbb{R}^n$ and any $r > 0$ one has

$$\text{Vol}(\text{conv}(L \cup r\cdot B_2^n) + T) \cdot (6\cdot\beta\cdot n \cdot N(r\cdot B_2^n, L))^{-1} \leq \text{Vol}(L+T),$$

where $\beta \geq 1$ is such that $r\cdot B_2^n \subseteq \beta \cdot L$. We can now use the dual Sudakov inequality. For $t > 0$, we have

$$N(B_2^n, t\cdot L) \leq e^{C\cdot n\cdot(\frac{M(L)}{t})^2}.$$

Setting $L := T_0(K_0)$, $T := P$ and $r := R_{\text{in}}$, we observe that

$$\text{Vol}(\underbrace{\text{conv}(T_0(K_0) \cup R_{\text{in}} \cdot B_2^n)}_{\supseteq K_1} + P) \cdot (6\cdot\beta\cdot n \cdot e^{C\cdot n\cdot\alpha_0^{-2}})^{-1} \leq \text{Vol}(T_0(K_0)+P).$$

Since $\frac{1}{M(K_0)\cdot\sqrt{n}} \cdot B_2^n \subseteq K_0$ (this is implied by the definition of the M-value), we can set $\beta = O(\sqrt{n})$ and the left-hand-side $(*)$ follows. For the right-hand side, we use the fact that that for any

symmetric convex bodies $L, T \subseteq \mathbb{R}^n$ and $r > 0$, we have

$$\mathrm{Vol}(L+T) \leq N(L, r \cdot B_2^n) \cdot \mathrm{Vol}((L \cap r \cdot B_2^n) + T).$$

By the Sudakov inequality we have that

$$N(L, t \cdot B_2^n) \leq e^{O(n) \cdot (\frac{M(K^\circ)}{t})^2}.$$

Setting $r := R_{\mathrm{out}}$, $L = T_0(K_0)$, $T = P$ and observing that $K_1 \subseteq T_0(K_0) \cap R_{\mathrm{out}} \cdot B_2^n$, $(**)$ follows. We note that the constants in the exponents of the primal and dual Sudakov inequality are universal, so we just denote them by C as well. For a proof, we refer to Chapter 8 of (AAGM15).

We now define K_j, for $j \geq 2$. Given K_{j-1}, we first compute the linear transformation T_{j-1} given by the MM° estimate (4.4), and, analogously to before with $T_0(K_0)$ replaced by $T_{j-1}(K_{j-1})$, $\alpha_{j-1} = \max\{d(K_{j-1}, B_2^n)^{1/4}, \varepsilon^{-1/2}\}$, $R_{\mathrm{out}}^{j-1} := (M((T_{j-1}(K_{j-1}))^\circ) \cdot \alpha_{j-1}$ and $R_{\mathrm{in}}^{j-1} := (M(T_{j-1}(K_{j-1})) \cdot \alpha_{j-1})^{-1}$, we define

$$K_j = \mathrm{conv}\Big(\big(T_{j-1}(K_{j-1}) \cap R_{\mathrm{out}}^{j-1}) \cdot B_2^n\big) \cup R_{\mathrm{in}}^{j-1} \cdot B_2^n\Big).$$

The volume estimate for K_j and $T_{j-1}(K_{j-1})$ are as follows. For any symmetric convex body $P \subseteq \mathbb{R}^n$,

$$2^{-Cn/\alpha_{j-1}^2} \cdot \mathrm{Vol}(K_j + P) \leq \mathrm{Vol}(T_{j-1}(K_{j-1}) + P) \leq 2^{Cn/\alpha_{j-1}^2} \cdot \mathrm{Vol}(K_j + P) \tag{4.7}$$

In each iteration $K_{j-1} \to K_j$, the respective Banach Mazur distance to the Euclidean ball is dropping. Specifically, if $d(K_{j-1}, B_2^n) \geq \varepsilon^{-1/2}$ (we may assume ε is small enough), we have

$$d(K_j, B_2^n) \leq R_{\mathrm{out}}^{j-1} / R_{\mathrm{in}}^{j-1} \leq O(C) \cdot d(K_{j-1}, B_2^n)^{1/2} \cdot \log(1 + d(K_{j-1}, B_2^n)) \leq \frac{1}{2} d(K_{j-1}, B_2^n).$$

Since $B_2^n \subseteq K \subseteq \sqrt{n} \cdot B_2^n$, after at most $\log(n)$ iterations, $\alpha_{t-1} = \varepsilon^{-1/2}$. Up to scaling the resulting convex body K_t by $(R_{\mathrm{in}}^{t-1})^{-1}$, we have the following inclusions,

$$\varepsilon^{1/2} \cdot B_2^n \subseteq K_t \subseteq C \cdot \varepsilon^{-1/2} \cdot \log(1 + \varepsilon^{-2}) \cdot B_2^n. \tag{4.8}$$

For the right-hand side, we have used the MM° estimate (4.4) and that $d(K_{t-1}, B_2^n) \leq \varepsilon^{-2}$. Indeed, otherwise, α_{t-1} would be set to $d(K_{t-1}, B_2^n)^{1/4} > \varepsilon^{-1/2}$, contradicting the choice of t.

Since $\alpha_{j-1} \leq 2 \cdot \alpha_j$ for $j \leq t-1$ and $\alpha_{t-1} = \varepsilon^{-1/2}$, we have that $\frac{1}{\alpha_0^2} + \ldots + \frac{1}{\alpha_{t-1}^2} \leq \varepsilon \cdot O(1)$. Hence, we can iteratively combine the volume estimates in (4.7) to arrive at

$$2^{-O(\varepsilon)n} \cdot \mathrm{Vol}(K_t + P) \leq \mathrm{Vol}(T_{t-1} \circ \ldots \circ T_0(K) + P) \leq 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(K_t + P). \tag{4.9}$$

We can now give bounds on $N(T_{t-1} \circ \ldots \circ T_0(K), \rho_1 \cdot B_2^n)$ and $N(\rho_2 \cdot B_2^n, T_{t-1} \circ \ldots \circ T_0(K))$.

For $\rho_1 := (C \cdot \varepsilon^{-3/2} \cdot \log(1+\varepsilon^{-2}))$, we obtain that,

$$\begin{aligned} \mathrm{Vol}(T_{t-1} \circ \ldots \circ T_0(K) + \rho_1 \cdot B_2^n) &\overset{(*)}{\leq} 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(K_t + \rho_1 \cdot B_2^n) \\ &\overset{(**)}{\leq} 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(\rho_1 \cdot B_2^n). \end{aligned} \tag{4.10}$$

For $(*)$, we have used the rightmost inequality in (4.9), valid for any convex body $-P = P$, with $P = \rho_1 \cdot B_2^n$. For $(**)$, we use that $K_t \subseteq (\varepsilon \cdot \rho_1) \cdot B_2^n$, given by the inclusion on the right-hand side in (4.8), combined with $\mathrm{Vol}((1+\varepsilon) \cdot \rho_1 \cdot B_2^n) \leq 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(\rho_1 \cdot B_2^n)$.

For $\rho_2 := \varepsilon^{3/2}$, we proceed analogously. We first use the rightmost inequality in (4.9), followed by the inclusion on the left-hand side in (4.8) combined with $\mathrm{Vol}((1+\varepsilon) \cdot K_t) \leq 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(K_t)$, and finally, the leftmost inequality in (4.9) with $P = 0$, to arrive at

$$\begin{aligned} \mathrm{Vol}(T_{t-1} \circ \ldots \circ T_0(K) + \rho_2 \cdot B_2^n) &\leq 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(K_t + \rho_2 \cdot B_2^n) \\ &\leq 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(K_t) \\ &\leq 2^{O(\varepsilon)n} \cdot \mathrm{Vol}(T_{t-1} \circ \ldots \circ T_0(K)). \end{aligned} \tag{4.11}$$

We set $T_\varepsilon(\cdot) := \rho_1^{-1} \cdot T_{t-1} \circ \ldots \circ T_0(\cdot)$ and $c_\varepsilon := \rho_2/\rho_1 = O(\varepsilon^{-3} \cdot \log(1/\varepsilon))$. By (4.10) and (4.11), the volume inequalities

$$\mathrm{Vol}(T_\varepsilon(K) + B_2^n) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(B_2^n)$$

and

$$\mathrm{Vol}(B_2^n + c_\varepsilon \cdot T_\varepsilon(K)) \leq 2^{\varepsilon n} \cdot \mathrm{Vol}(c_\varepsilon \cdot T_\varepsilon(K))$$

immediately follow by replacing ε by a fraction of itself in the beginning. This concludes the description of the procedure.

We now discuss several implementation details.

We have only assumed the existence of a (weak) separation oracle for the original body $K_0 := K$. This is sufficient to construct a separation oracle for all intermediate bodies K_j appearing in the description of the procedure above and to compute their respective linear transformations guaranteed by the $M M^\circ$ estimate in quasi-polynomial time. Each such body is of the form $K_j = \mathrm{conv}((T_{j-1}(K_{j-1}) \cap R_{\mathrm{out}}^{j-1} \cdot B_2^n) \cup R_{\mathrm{in}}^{j-1} \cdot B_2^n)$. Indeed, for any convex bodies $L, T \subseteq \mathbb{R}^n$ given by a separation oracle, we can construct a weak separation oracle for L°, $\mathrm{conv}(L \cup T)$ and $L \cap T$. One call to such a weak separation oracle can be evaluated using a polynomial number (in n and $\log(1/\varepsilon)$, the additive error) of calls to a weak separation oracle for L and T. This follows from the ellipsoid method and the equivalence of optimization and separation, see (GLS88). Hence, given a weak membership oracle for K_{j-1}, we can solve the separation problem for K_j in polynomial time. Since there are at most $\log(n)$ iterations to consider, each call to a separation oracle for K_j can be evaluated by $n^{O(\log(n))}$ calls to the separation oracle for K_0 and results in an overall running time of $n^{O(\log(n))}$.

We do rely on (approximately) computing the M-values $E_{x \sim \mathbb{S}^{n-1}}[\|x\|_L]$, $E_{x \sim \mathbb{S}^{n-1}}[\|x\|_{L^\circ}]$ and

$\mathbb{E}_{x\in\gamma_n}\left[\|T(x)\|_L^2\right]^{1/2}$. If the convex body is presented by a weak separation oracle, this can be achieved in polynomial time to any desired accuracy. Let us illustrate this for computing $M(L)$. We first apply a linear transformation so that $B_2^n \subseteq L \subseteq (n+1)\cdot B_2^n$. We sample random points $x_1,\dots,x_m \sim B_2^n$, and compute

$$\bar{X} := \frac{1}{m}\sum_{i=1}^{m} \|\tfrac{x}{\|x\|_2}\|_L.$$

If $m = 1$, i.e. we only consider one such sample, this equals the M-value of L, in expectation. Since $L \subseteq (n+1)\cdot B_2^n$, this is lower bounded by $\frac{1}{n+1}$. Since $B_2^n \subseteq L$, the variance in turn is bounded by 1. Hence, by Chebychev's inequality, for $m = O(n^3)$, the probability that

$$|\bar{X} - M(L)| > \tfrac{1}{n}$$

is bounded by $1/n$. This can be boosted by taking larger m or by applying the median trick. For the M-value of its dual, we can proceed similarly, using the fact that a weak separation oracle for L implies the existence of a weak separation oracle for L°, see (GLS88). Finally, for $\mathbb{E}_{x\in\gamma_n}\left[\|T(x)\|_L^2\right]^{1/2}$, we observe that this equals $\sqrt{n}\cdot E_{x\sim\mathbb{S}^{n-1}}[\|T(x)\|_L^2]^{1/2}$. This can be seen by polar integration. Hence, this can be estimated similar to $M(L)$ and $M(L^\circ)$.

In each iteration $K_j \to K_{j+1}$ we rely on guessing $d(K_i, B_2^n)$ up to a factor of 2 and seem to know the right iteration t at which to stop. This is without loss of generality. Indeed, since $1 \le d(K_i, B_2^n) \le \sqrt{n}$ and $d(K_i, B_2^n) \le \frac{1}{2}\cdot d(K_i, B_2^n)$, there are at most $\log(n)^{\log(n)}$ possibilities in total. We can either guess and succeed with probability $\log(n)^{-\log(n)}$, or we return $\log(n)^{\log(n)}$ different convex bodies, one of which verifies the desired properties. Both of these options are fine for our purpose.

Finally, let us outline how the convex program (4.5) can be (approximatively) solved using the ellipsoid method and why such an approximation still implies the $M M^\circ$-inequality (4.4).

To approximately solve the convex program, we work directly in $\mathbb{R}^{n\cdot(n+1)/2}$ which can be identified with the space of symmetric $n\times n$ matrices $A = (A_{ij})$ in the obvious way. We define $\mathscr{A} \subseteq \mathbb{R}^{n\cdot(n+1)/2}$ as the set of all symmetric and positive-definite matrices that are feasible for the convex program (4.5). Clearly, $\mathscr{A}$ is convex. We now build a weak separation oracle for $\mathscr{A}$. We recall that a weak membership oracle for $\mathscr{A}$ can be turned into a weak separation oracle for $\mathscr{A}$, provided there exist $r, R > 0$ and $s \in \mathbb{R}^{n\times n}$ such that $s + r\cdot B_2^n \subseteq \mathscr{A} \subseteq R\cdot B_2^n$. This can be achieved in polynomial time in n, $\log(r)$ and $\log(R)$. It will be easier to work with a weak membership oracle. Indeed, for any symmetric linear transformation $A \in \mathscr{A}$, we can estimate $\mathbb{E}_{x\in\gamma_n}\left[\|A(x)\|_K^2\right]^{1/2}$ to any desired accuracy using a polynomial number of calls to a weak separation oracle. Hence, a weak membership oracle is trivial to implement. It remains to exhibit some $s \in \mathbb{R}^{n\cdot(n+1)/2}$ and $r, R \in \mathbb{R}_{>0}$ so that $s + r\cdot B_2^{n\cdot(n+1)/2} \subseteq \mathscr{A}$ and $\mathscr{A} \subseteq R\cdot B_2^{n\cdot(n+1)/2}$, to turn this into a weak separation oracle. Note that we can assume that K is in John's position, i.e. $B_2^n \subseteq K \subseteq (n+1)\cdot B_2^n$. Using polar coordinates, we rewrite $\mathbb{E}_{x\in\gamma_n}\left[\|A(x)\|_K^2\right]^{1/2} = \sqrt{n}\cdot\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|A(x)\|_K^2]^{1/2}$. Since $B_2^n \subseteq K$, $\|\cdot\|_K \le \|\cdot\|_2$, and the latter is bounded by $\sqrt{n}\cdot\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|A(x)\|_2^2]^{1/2}$. Thus, for $A^{(s)} = (2\cdot\sqrt{n})^{-1}\cdot I_n$, we have that

$\mathbb{E}_{x\in\gamma_n}\left[\|A^{(s)}(x)\|_K^2\right]^{1/2} \leq 1/2$. Hence, we can set $s := A^{(s)}$. By Gershgorin's circle theorem, the eigenvalues of $A^{(s)}$ only change slightly when we slightly perturb its entries, i.e. when we consider $A^{(s)} + \tilde{A}$, for $\|\tilde{A}\|_\infty$ small. Furthermore, by Minkowski's inequality, this changes the constraint only very slightly:

$$\begin{aligned}\mathbb{E}_{x\sim\gamma_n}[\|(A^{(s)}+\tilde{A})(x)\|_K^2]^{1/2} &\leq \mathbb{E}_{x\sim\gamma_n}[\|A^{(s)}(x)\|_K^2]^{1/2} + \mathbb{E}_{x\sim\gamma_n}[\|\tilde{A}(x)\|_K^2]^{1/2}\\ &\overset{(*)}{\leq} 1/2 + \sqrt{n}\cdot\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|\tilde{A}(x)\|_2^2]^{1/2}\\ &\overset{(**)}{\leq} 1/2 + \sqrt{n}\cdot\|\tilde{A}\|_F.\end{aligned}$$

For $(*)$ we use polar integration combined with $B_2^n \subseteq K$, i.e. $\|\cdot\|_K \leq \|\cdot\|_2$ as done earlier. For $(**)$, we note that $\|\tilde{A}\|_F$ is the *Frobenius norm* of $\tilde{A}$, i.e. the ℓ_2 norm of all coefficients of the matrix $\tilde{A}$. Equivalently, the Frobenius norm of $\tilde{A}$ is the ℓ_2 norm of all eigenvalues of $\tilde{A}$. Hence, it is clear that this is an upper bound on $\|\tilde{A}(x)\|_2^2$, $x \in \mathbb{S}^{n-1}$, meaning that for $r \approx n^{-2}$, $A^{(s)} + r\cdot B_2^{n\cdot(n+1)/2} \subseteq \mathscr{A}$. On the other hand, we observe that any matrix in $\mathscr{A}$ cannot have an eigenvector of eigenvalue larger than n^3. Indeed, if such an eigenvector $v \in \mathbb{S}^{n-1}$ were to exist, we can lower bound the constraint as follows

$$\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|A(x)\|_K^2]^{1/2} > \tfrac{1}{n+1}\Pr[x\in\mathbb{S}^{n-1} \mid v^T\cdot x \geq 1/n]\cdot[\|T(\tfrac{v}{n})\|_2] \geq \Omega(n).$$

Here, we have used the fact that probability of $\Pr[x \in \mathbb{S}^{n-1} \mid v^T\cdot x \geq 1/n]$ is at least a constant and that $K \subseteq (n+1)\cdot B_2^n$. Hence, the Frobenius norm of any matrix in $\mathscr{A}$ is bounded by n^4, and we can set $R = n^4$. Hence, in time polynomial in $n\cdot(n+1)/2$ (the dimension of the problem), $\log(1/r)$, $\log(R)$ and $\log(1/\varepsilon)$, we can turn a weak membership oracle for $\mathscr{A}$ into a weak separation oracle for $\mathscr{A}$. Here, $\varepsilon > 0$ is the additive error of the weak membership and weak separation oracle. Hence, one call to a weak separation oracle for $\mathscr{A}$ can be evaluated in time polynomial in n. Since the convex program (4.5) can be approximated to arbitrary precision using a polynomial number of calls to a weak separation oracle for $\mathscr{A}$, this implies an overall running time polynomial in n and $\log(1/\varepsilon)$ to find a solution within a factor of $(1-\varepsilon)$ of the optimum.

It remains to argue why approximate solutions to the convex program (4.5) are sufficient for our purpose, i.e. why their inverse satisfy the MM° estimate (4.4). To do so, it will be helpful to work with the ℓ-norm, which is defined as $\ell_K(A) = \mathbb{E}_{x\in\gamma_n}[\|A(x)\|_K^2]^{1/2}$. Here, K is supposed to be fixed, hence this is a matrix norm. We note however that this norm is related to $M(A(K))$, i.e. $\Theta(\sqrt{n})\cdot M(A(K)) = \ell_K(A^{-1})$, we justify this below. Since $\ell_K(\cdot)$ is a matrix norm, one can define a dual norm $\ell_K^* : \mathbb{R}^{n\times n} \to \mathbb{R}_{\geq 0}$ by

$$\ell_K^*(A) = \sup(\operatorname{tr}(A\cdot S) \mid S \in \mathbb{R}^{n\times n}, \ell_K(S) \leq 1)$$

Using a variational argument, Lewis (Lew79) showed that there exists a linear transformation $\bar{A} \in \mathbb{R}^{n\times n}$ such that

$$\ell_K(\bar{A}) \cdot \ell_K^*(\bar{A}^{-1}) \leq n. \tag{4.12}$$

In fact, this holds for any matrix norm $\alpha : \mathbb{R}^{n\times n} \to \mathbb{R}_{\geq 0}$ (exchange $\ell_K(\cdot)$ by $\alpha(\cdot)$ in the definitions above) and such a linear transformation $\bar{A}$ is given by

$$\begin{gathered} \arg\max \det(A) \\ \alpha(A) \leq 1 \\ A \in \mathbb{R}^{n\times n} \text{ invertible.} \end{gathered}$$

For the special case of $\ell_K(\cdot)$, by rotational symmetry of the Gaussian, one can restrict to linear transformations $A \in \mathbb{R}^{n\times n}$ that are symmetric positive-definite. In particular, we recover (4.5). For a near optimal optimizer $\widetilde{A}$ to this convex program, (e.g. $\det(\widetilde{A}) \geq (1-\varepsilon)\cdot\det(\bar{A})$), the estimate 4.12 holds up to a small multiplicative factor,

$$\ell_K(\widetilde{A}) \cdot \ell_K^*(\widetilde{A}^{-1}) \leq n \cdot (1 + 6 \cdot \mathrm{poly}(n) \cdot \sqrt{\varepsilon}). \tag{4.13}$$

This was shown in detail by Dadush and Vempala in (DV13), in Lemma 3.2. Their proof is stated for general matrix norms $\alpha(\cdot)$, under the assumption that $\alpha(\cdot)$ is such that $\alpha(A) \leq \|A\|_F \leq \mathrm{poly}(n)\cdot\alpha(A)$ (which holds in our case). Hence, for a sufficiently small ε, the right-hand-side of (4.13) can be taken to be equal to $2\cdot n$. This then implies the MM° estimate. Indeed, results of Figiel, Tomczak-Jaegermann and Pisier (FTJ79; Pis80) show that $\ell_K^*(\cdot) \leq O(\ell_{K^\circ}(\cdot)\cdot\log(1+d(K,B_2^n))$. Hence, for such an $\widetilde{A}$, one has

$$\ell_K(\widetilde{A}) \cdot \ell_{K^\circ}(\widetilde{A}^{-1}) \leq O(n) \cdot \log(1 + d(K, B_2^n)) \tag{4.14}$$

To relate this to the MM° estimate, we observe that

$$\ell_K(\widetilde{A}) \overset{(*)}{=} \sqrt{n}\cdot\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|\widetilde{A}(x)\|_K^2]^{1/2} \overset{(**)}{=} \Theta(\sqrt{n})\cdot\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|\widetilde{A}(x)\|_K] \overset{(***)}{=} \Theta(\sqrt{n})\cdot M(\widetilde{A}^{-1}(K)),$$

where in $(*)$ we have used partial integration, in $(**)$ have used the Khintchine-Kahane inequality, and in $(***)$ we have used that $\|\widetilde{A}(x)\|_K = \|x\|_{\widetilde{A}^{-1}(K)}$. Similarly, we see that

$$\ell_{K^\circ}(\widetilde{A}^{-1}) = \sqrt{n}\cdot\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|\widetilde{A}^{-1}(x)\|_{K^\circ}^2]^{1/2} = \Theta(\sqrt{n})\cdot\mathbb{E}_{x\sim\mathbb{S}^{n-1}}[\|\widetilde{A}^{-1}(x)\|_{K^\circ}] \overset{(*)}{=} \Theta(\sqrt{n})\cdot M((\widetilde{A}^{-1}(K))^\circ),$$

where in $(*)$ we have used that $\|\widetilde{A}^{-1}(x)\|_K = \|x\|_{\widetilde{A}(K^\circ)}$ and $\widetilde{A}(K^\circ) = (\widetilde{A}^{-1}(K))^\circ$. Hence, inequality (4.14) yields the MM° estimate (4.4) by setting $T(\cdot) = \widetilde{A}^{-1}$. This concludes the proof.

□

Bibliography

[AAGM15] Shiri Artstein-Avidan, Apostolos Giannopoulos, and Vitali Milman, *Asymptotic geometric analysis, part i*, AMS, 2015.

[AB09] Sanjeev Arora and Boaz Barak, *Computational complexity: a modern approach*, Cambridge University Press, 2009.

[ABB+55] Paul Armer, E.C. Bower, Bernice Brown, G.W. Brown, Walter Frantz, J.J. Goodpasture, W.F. Gunning, Cecil Hastings, Olaf Helmer, M.L Juncosa, J.D. Madden, A.M. Mood, R.T. Nash, and J.D. Williams, *A million random digits with 100,000 normal deviates*, RAND Corporation, 1955.

[ABD+] Erdem Alkim, Joppe W. Bos, Léo Ducas, Patrick Longa, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Chris Peikert, Ananth Raghunathan, and Douglas Stebila, *Frodokem - learning with errors key encapsulation (algorithm specifications and supporting documentation)*.

[ABGS21] Divesh Aggarwal, Huck Bennett, Alexander Golovnev, and Noah Stephens-Davidowitz, *Fine-grained hardness of CVP(P)– Everything that we can prove (and nothing else)*, SODA, 2021.

[ACK+21] Divesh Aggarwal, Yanlin Chen, Rajendra Kumar, Li, and Noah Stephens-Davidowitz, *Dimension-preserving reductions between SVP and CVP in different p-norms*, SODA, 2021.

[ADRS15] Divesh Aggarwal, Daniel Dadush, Oded Regev, and Noah Stephens-Davidowitz, *Solving the shortest vector problem in* 2^n *time using discrete gaussian sampling*, STOC, 2015.

[ADS15] Divesh Aggarwal, Daniel Dadush, and Noah Stephens-Davidowitz, *Solving the closest vector problem in* 2^n *time – the discrete gaussian strikes again!*, FOCS, 2015.

[Ajt96] Miklós Ajtai, *Generating hard instances of lattice problems (extended abstract)*, STOC, 1996.

[Ajt98] Miklós Ajtai, *The shortest vector problem in* ℓ_2 *is NP-hard for randomized reductions (extended abstract)*, STOC, 1998.

[AKS01] Miklós Ajtai, Ravi Kumar, and D. Sivakumar, *A sieve algorithm for the shortest lattice vector problem*, STOC, 2001.

[AKS02] Miklós Ajtai, Ravi Kumar, and D. Sivakumar, *Sampling short lattice vectors and the closest lattice vector problem*, CCC, 2002.

[ALNSD20] Divesh Aggarwal, Jianwei Li, Phong Q. Nguyen, and Noah Stephens-Davidowitz, *Slide reduction, revisited – filling the gaps in svp approximation*, CRYPTO, 2020.

[ALS21] Divesh Aggarwal, Zeyong Li, and Noah Stephens-Davidowitz, *A $2^{n/2}$-time algorithm for $\sqrt{n}$-SVP and $\sqrt{n}$-hermite SVP, and an improved time-approximation tradeoff for (H)SVP*, EUROCRYPT, 2021.

[AM18] Divesh Aggarwal and Priyanka Mukhopadhyay, *Improved algorithms for the shortest vector problem and the closest vector problem in the infinity norm*, ISAAC, 2018.

[AR05] Dorit Aharonov and Oded Regev, *Lattice problems in NP ∩ coNP*, J. ACM 52.5 (2005).

[Aro95] Sanjeev Arora, *Probabilistic checking of proofs and hardness of approximation problems*, Ph.D. thesis, University of California at Berkeley, 1995.

[AS18a] Divesh Aggarwal and Noah Stephens-Davidowitz, *(Gap/S)ETH hardness of SVP*, STOC, 2018.

[AS18b] Divesh Aggarwal and Noah Stephens-Davidowitz, *Just take the average! an embarrassingly simple 2^n-time algorithm for SVP (and CVP)*, SOSA, 2018.

[Bab86] László Babai, *On lovász' lattice reduction and the nearest lattice point problem*, Combinatorica 6 (1986).

[BGS17] Huck Bennett, Alexander Golovnev, and Noah Stephens-Davidowitz, *On the quantitative hardness of CVP*, FOCS, 2017.

[BN09] Johannes Blömer and Stefanie Naewe, *Sampling methods for shortest vectors, closest vectors and successive minima*, Theor. Comput. Sci. 410.18 (2009).

[BS18] Matthias Beck and Raman Sanyan, *Combinatorial reciprocity theorems*, AMS, 2018.

[Dad12a] Daniel Dadush, *Integer programming, lattice algorithms, and deterministic volume estimation*, Ph.D. thesis, Georgia Institute of Technology, 2012.

[Dad12b] Daniel Dadush, *A $O(1/\varepsilon^2)^n$ time sieving algorithm for approximate integer programming*, LATIN, 2012.

[DB15] Daniel Dadush and Nicolas Bonifas, *Short paths on the voronoi graph and closest vector problem with preprocessing*, SODA, 2015.

[DFK91] Martin E. Dyer, Alan M. Frieze, and Ravi Kannan, *A random polynomial time algorithm for approximating the volume of convex bodies*, J. ACM 38.1 (1991).

[Din16] Irit Dinur, *Mildly exponential reduction from gap 3sat to polynomial-gap label-cover*, Electron. Colloquium Comput. Complex. 23 (2016).

[DK16] Daniel Dadush and Gábor Kun, *Lattice sparsification and the approximate closest vector problem*, Theory of Computing 12.1 (2016).

[DKL+] Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler, and Damien Stehlé, *Crystals-dilithium – algorithm specifications and supporting documentation (version 3.1)*.

[DKRS03] Irit Dinur, Guy Kindler, Ran Raz, and Shmuel Safra, *Approximating CVP to within almost-polynomial factors is NP-hard*, Combinatorica 23.2 (2003).

[DPV11] Daniel Dadush, Chris Peikert, and Santosh S. Vempala, *Enumerative lattice algorithms in any norm via m-ellipsoid coverings*, FOCS, 2011.

[DV13] Daniel Dadush and Santosh Vempala, *Near-optimal deterministic algorithms for volume computation via m-ellipsoids*, Proceedings of the National Academy of Sciences, 2013.

[EHN11] Friedrich Eisenbrand, Nicolai Hähnle, and Martin Niemeier, *Covering cubes and the closest vector problem*, SoCG, 2011.

[Eis10] Friedrich Eisenbrand, *Integer programming and algorithmic geometry of numbers*, Springer Berlin Heidelberg, 2010.

[EV22] Friedrich Eisenbrand and Moritz Venzin, *Approximate CVP_p in time* $2^{0.802n}$, J. Comput. Syst. Sci. 124 (2022).

[FTJ79] Tadeusz Figiel and Nicole Tomczak-Jaegermann, *Projections onto hilbertian subspaces of banach spaces*, Israel Journal of Mathematics 33.2 (1979).

[Gen09] Craig Gentry, *Fully homomorphic encryption using ideal lattices*, STOC, 2009.

[GG00] Oded Goldreich and Shafi Goldwasser, *On the limits of nonapproximability of lattice problems*, Journal of Computer and System Sciences 60.3 (2000).

[GLS88] Martin Grötschel, László Lovász, and Alexander Schrijver, *Geometric algorithms and combinatorial optimization*, The Journal of the Operational Research Society, 1988.

[GMSS99] Oded Goldreich, Daniele Micciancio, Shmuel Safra, and Jean-Pierre Seifert, *Approximating shortest lattice vectors is not harder than approximating closest lattice vectors*, Inf. Process. Lett. 71.2 (1999).

[GN08] Nicolas Gama and Phong Q. Nguyen, *Finding short lattice vectors within mordell's inequality*, STOC, 2008.

[HR07] Ishay Haviv and Oded Regev, *Tensor-based hardness of the shortest vector problem to within almost polynomial factors*, STOC, 2007.

[HRS20] Christoph Hunkenschröder, Gina Reuland, and Matthias Schymura, *On compact representations of voronoi cells of lattices*, Mathematical Programming 183.1 (2020).

[IP01] Russell Impagliazzo and Ramamohan Paturi, *On the complexity of k-sat*, J. Comput. Syst. Sci. 62.2 (2001).

[Joh48] Fritz John, *Extremum problems with inequalities as subsidiary conditions*, Studies and Essays Presented to R. Courant on his 60th Birthday, January 8, 1948, Interscience Publishers, Inc., 1948.

[Kan87] Ravi Kannan, *Minkowski's convex body theorem and integer programming*, Math. Oper. Res. 12.3 (1987).

[Kar72] Richard M. Karp, *Reducibility among combinatorial problems*, Complexity of Computer Computations, The IBM Research Symposia Series, 1972.

[Kho05] Subhash Khot, *Hardness of approximating the shortest vector problem in lattices*, J. ACM 52.5 (2005).

[KL78] Grigorii Anatolevich Kabatiansky and Vladimir Iosifovich Levenshtein, *On bounds for packings on a sphere and in space*, Problemy Peredachi Informatsii 14.1 (1978).

[KPV12] Subhash A. Khot, Preyas Popat, and Nisheeth K. Vishnoi, $2^{\log^{1-\varepsilon}(n)}$ *hardness for the closest vector problem with preprocessing*, STOC, 2012.

[Len83] Hendrik W. Lenstra, *Integer programming with a fixed number of variables*, Math. Oper. Res. 8.4 (1983).

[Lew79] Daniel R. Lewis, *Ellipsoids defined by banach ideal norms*, Mathematika 26.1 (1979).

[Lin63] Joram Lindenstrauss, *On the modulus of smoothness and divergent series in banach spaces.*, Michigan Math. J. 10.3 (1963).

[LLL82] Arjen K. Lenstra, Hendrik W. Lenstra, and László Lovász, *Factoring polynomials with rational coefficients*, Mathematische Annalen 261.4 (1982).

[LMvdP15] Thijs Laarhoven, Michele Mosca, and Joop van de Pol, *Finding shortest lattice vectors faster using quantum search*, Designs, Codes and Cryptography 77.2 (2015).

[LWXZ11] Mingjie Liu, Xiaoyun Wang, Guangwu Xu, and Xuexin Zheng, *Shortest lattice vectors in the presence of gaps*, IACR Cryptology ePrint Archive (2011).

[Mic01] Daniele Micciancio, *The shortest vector in a lattice is hard to approximate to within some constant*, SIAM Journal on Computing 30.6 (2001).

[Mil88] V. D. Milman, *Isomorphic symmetrization and geometric inequalities*, Geometric aspects of functional analysis (1986/87), Springer, Berlin, 1988.

[Min10] Hermann Minkowski, *Geometrie der Zahlen*, 1910.

[MR17] Pasin Manurangsi and Prasad Raghavendra, *A birthday repetition theorem and complexity of approximating dense CSPs*, ICALP, 2017.

[MU05] Michael Mitzenmacher and Eli Upfal, *Probability and computing: Randomized algorithms and probabilistic analysis*, Cambridge University Press, 2005.

[Muk21] Priyanka Mukhopadhyay, *Faster provable sieving algorithms for the shortest vector problem and the closest vector problem on lattices in ℓ_p norm*, Algorithms 14.12 (2021).

[MV10a] Daniele Micciancio and Panagiotis Voulgaris, *A deterministic single exponential time algorithm for most lattice problems based on voronoi cell computations*, STOC, 2010.

[MV10b] Daniele Micciancio and Panagiotis Voulgaris, *Faster exponential time algorithms for the shortest vector problem*, SODA, 2010.

[Nas14] Márton Naszódi, *On some covering problems in geometry*, Proceedings of the American Mathematical Society 144 (2014).

[NV22] Márton Naszódi and Moritz Venzin, *Covering convex bodies and the closest vector problem*, Discrete & Computational Geometry 67.4 (2022).

[Odl90] Andrew M. Odlyzko, *The rise and fall of knapsack cryptosystems*, In Cryptology and Computational Number Theory, 1990.

[Pei08] Chris Peikert, *Limits on the hardness of lattice problems in ℓ_p norms*, Comput. Complex. 17.2 (2008).

[Pis80] Gilles Pisier, *Sur les espaces de banach k-convexes*, Seminar on Functional Analysis, 1979–1980 (French), École Polytech., Palaiseau, 1980.

[Pis89] Gilles Pisier, *A new approach to several results of V. Milman*, J. Reine Angew. Math. 393 (1989).

[PS09] Xavier Pujol and Damien Stehlé, *Solving the shortest lattice vector problem in time $2^{2.465n}$*, IACR Cryptology ePrint Archive (2009).

[Reg04] Oded Regev, *Lattices in computer science, lecture 8: $2^{O(n)}$ algorithm for svp*, 2004.

[Reg09] Oded Regev, *On lattices, learning with errors, random linear codes, and cryptography*, J. ACM 56.6 (2009).

[Rot21] Thomas Rothvoss, *Asymptotic convex geometry, lecture notes*, 2021.

[RR06] Oded Regev and Ricky Rosen, *Lattice problems and norm embeddings*, STOC, 2006.

[RV22] Thomas Rothvoss and Moritz Venzin, *Approximate CVP in time* $2^{0.802n}$ *- now in any norm!*, IPCO, 2022.

[Sch87] Claus-Peter Schnorr, *A hierarchy of polynomial time lattice basis reduction algorithms*, Theoretical computer science 53.2-3 (1987).

[Ste16] Noah Stephens-Davidowitz, *Discrete gaussian sampling reduces to CVP and SVP*, SODA, 2016.

[Tal98] István Talata, *Exponential lower bound for the translative kissing numbers of d -dimensional convex bodies*, Discrete & Computational Geometry 19.3 (1998).

[vEB81] Peter van Emde Boas, *Another NP-complete problem and the complexity of computing short vectors in a lattice*, Technical Report 81-04, Mathematische Instituut, University of Amsterdam (1981).

Moritz Venzin

Personal Data

Nationality and Date of Birth:	Swiss | 20 Oktober 1993
Contact information:	moritz.venzin@epfl.ch | +41 078 640 84 87

Education

2022 Sep -2018 Sep	PhD in Applied Mathematics École Polytechnique Fédérale de Lausanne (EPFL), Lausanne Advisor: Prof. Friedrich Eisenbrand
2018 Feb - 2013 Sep	Bachelor and Master's Degree in Pure Mathematics École Polytechnique Fédérale de Lausanne (EPFL), Lausanne

Publications

- "Approximate CVP in time $2^{0.802n}$ - now in any norm!" with T. Rothvoss.
  Integer Programming and Combinatorial Optimization (IPCO), 2022.
- "A QPTAS for stabbing rectangles" with F. Eisenbrand, M. Gallato, O. Svensson.
- "Efficient Sequential and Parallel Algorithms for Multistage Stochastic Integer Programming Using Proximity" with J. Cslovjecsek, F. Eisenbrand, P. Pilipczuk, R. Weismantel.
  29th Annual European Symposium on Algorithms (ESA), 2021.
- "Approximate CVP_p in time $2^{0.802n}$" with F. Eisenbrand.
  28th Annual European Symposium on Algorithms (ESA), 2020.
  Winner of Best Paper Award.
- "Covering Convex Bodies and the Closest Vector Problem" with M. Naszódi.
  Discrete and Computational Geometry (DCG).

Awards

- Best Paper Award (ESA 2020)
- Doc.Mobility Fellowship SNF
  I was awarded 25'000 CHF from the Swiss National Foundation to fund a 6 month research visit to the University of Washington to work with Prof. Thomas Rothvoss.