Enumerative Lattice Algorithms in Any Norm via M-Ellipsoid Coverings

Daniel Dadush* Chris Peikert* Santosh Vempala*

April 13, 2011

Abstract

We give a novel algorithm for enumerating lattice points in any convex body, and give applications to several classic lattice problems, including the Shortest and Closest Vector Problems (SVP and CVP, respectively) and Integer Programming (IP). Our enumeration technique relies on a classical concept from asymptotic convex geometry known as the *M-ellipsoid*, and uses as a crucial subroutine the recent algorithm of Micciancio and Voulgaris (STOC 2010) for lattice problems in the ℓ_2 norm. As a main technical contribution, which may be of independent interest, we build on the techniques of Klartag (Geometric and Functional Analysis, 2006) to give an expected $2^{O(n)}$-time algorithm for computing an M-ellipsoid for any n-dimensional convex body.

As applications, we give deterministic $2^{O(n)}$-time and -space algorithms for solving exact SVP, and exact CVP when the target point is sufficiently close to the lattice, on n-dimensional lattices *in any (semi-)norm* given an M-ellipsoid of the unit ball. In many norms of interest, including all ℓ_p norms, an M-ellipsoid is computable in deterministic $\mathrm{poly}(n)$ time, in which case these algorithms are fully deterministic. Here our approach may be seen as a derandomization of the "AKS sieve" for exact SVP and CVP (Ajtai, Kumar, and Sivakumar; STOC 2001 and CCC 2002).

As a further application of our SVP algorithm, we derive an expected $O(f^*(n))^n$-time algorithm for Integer Programming, where $f^*(n)$ denotes the optimal bound in the so-called "flatness theorem," which satisfies $f^*(n) = O(n^{4/3}\,\mathrm{polylog}(n))$ and is conjectured to be $f^*(n) = \Theta(n)$. Our runtime improves upon the previous best of $O(n^2)^n$ by Hildebrand and Köppe (2010).

Keywords. Shortest/Closest Vector Problem, Integer Programming, lattice point enumeration, M-ellipsoid.

*School of Computer Science, Georgia Institute of Technology.

1 Introduction

The Shortest and Closest Vector Problems (SVP and CVP, respectively) on lattices are central algorithmic problems in the geometry of numbers, with applications to Integer Programming [Len83], factoring polynomials over the rationals [LLL82], cryptanalysis (e.g., [Odl90, JS98, NS01]), and much more. (An n-dimensional *lattice* L is a discrete additive subgroup of $\mathbb{R}^n$, and is generated as the set of integer linear combinations of some basis vectors $b_1, \ldots, b_k \in \mathbb{R}^n$, for some $k \leq n$.) The SVP is simply: given a lattice L represented by a basis, find a nonzero $v \in L$ such that $\|v\|$ is minimized, where $\|\cdot\|$ denotes a particular norm on $\mathbb{R}^n$. The CVP is an inhomogeneous analogue of SVP: given a lattice L and a point $t \in \mathbb{R}^n$, find some $v \in L$ that minimizes $\|v - t\|$. In these problems, one often uses the Euclidean (ℓ_2) norm, but many applications require other norms like ℓ_p or, most generally, the semi-norm defined by a convex body $K \ni 0$ as $\|x\|_K = \inf\{r \geq 0 : x \in rK\}$. Indeed, general (semi-)norms arise quite often in the study of lattices; for example, the "flatness theorem" in Integer Programming — which states that every lattice-free convex body has lattice width bounded by a function of the dimension alone — is a statement about SVP in general norms.

Much is known about the computational complexity of SVP and CVP, in both their exact and approximation versions. On the negative side, SVP is NP-hard (in ℓ_2, under randomized reductions) to solve exactly, or even to approximate to within any constant factor [Ajt98, CN98, Mic98, Kho03]. Many more hardness results are known for other ℓ_p norms and under stronger complexity assumptions than P $\neq$ NP (see, e.g., [vEB81, Din00, RR06, HR07]). CVP is NP-hard to approximate to within $n^{c/\log\log n}$ factors for some constant $c > 0$ [ABSS93, DKRS98, Din00], where n is the dimension of the lattice. Therefore, we do not expect to solve (or even closely approximate) these problems efficiently in high dimensions. Still, algorithms providing weak approximations or having super-polynomial running times are the foundations for the many applications mentioned above.

The celebrated LLL algorithm [LLL82] and variants [Sch87] give $2^{n/\mathrm{polylog}(n)}$ approximations to SVP and CVP in ℓ_2, in $\mathrm{poly}(n)$ time. For exact SVP and CVP in the ℓ_2 norm, Kannan's algorithm [Kan87] gives a solution in deterministic $2^{O(n \log n)}$ time and $\mathrm{poly}(n)$ space. This performance remained essentially unchallenged until the breakthrough randomized "sieve" algorithm of Ajtai, Kumar, and Sivakumar [AKS01], which provides a $2^{O(n)}$-time and -space solution for exact SVP; moreover, the algorithm generalizes straightforwardly to ℓ_p and other norms [BN07, AJ08]. For CVP, in a sequence of works [AKS02, BN07, AJ08] it was shown that a modified version of the AKS sieve can approximate CVP in any ℓ_p norm to within a $(1+\epsilon)$ factor in time and space $(1/\epsilon)^{O(n)}$ for any $\epsilon > 0$. Furthermore, these algorithms can solve CVP exactly in $2^{O(n)}$ time as long as the target point is "very close" to the lattice. It is worth noting that the AKS sieve is a *Monte Carlo* algorithm: while the output solution is correct with high probability, it is not guaranteed.

In a more recent breakthrough, Micciancio and Voulgaris [MV10] gave a *deterministic* $2^{O(n)}$-time (and space) algorithm for exact SVP and CVP in the ℓ_2 norm, among many other lattice problems in NP. Interestingly, their algorithm works very differently from the AKS sieve, by computing an explicit description of the Voronoi cell of the lattice. (The Voronoi cell is the set of all points in $\mathbb{R}^n$ that are closer to the origin than to any other lattice point.) In contrast to the AKS sieve, however, the algorithm of [MV10] appears to be quite specialized to ℓ_2 (or any norm defined by an ellipsoid, simply by applying a linear transformation). This is in part because in ℓ_2 the Voronoi cell is convex and has $2^{O(n)}$ facets, but in general norms this is not the case. A main problem left open in [MV10] was to find deterministic $2^{O(n)}$-time algorithms for lattice problems in ℓ_p and other norms.

1.1 Results and Techniques

Our main contribution is a novel algorithm for enumerating lattice points in any convex body. It uses as a crucial subroutine the Micciancio-Voulgaris (MV) algorithm [MV10] for the ℓ_2 norm that enumerates lattice points in an ellipsoid, and relies on a classical concept from asymptotic convex geometry known as the M-ellipsoid. This connection between lattice algorithms and convex geometry appears to be a fertile direction for further research.

For a lattice L and convex body K in $\mathbb{R}^n$, let $G(K, L)$ be the largest number of lattice points contained in any translate of K, i.e.,

$$G(K, L) = \max_{x \in \mathbb{R}^n} |(K + x) \cap L|. \tag{1.1}$$

Our starting point is the following guarantee on the enumeration of $K \cap L$.[1]

Theorem 1.1 (Enumeration in convex bodies, informal)**.** *Given any convex body $K \subseteq \mathbb{R}^n$ along with an* M-ellipsoid *E of K, and any n-dimensional lattice $L \subseteq \mathbb{R}^n$, the set $K \cap L$ can be computed in deterministic time $G(K, L) \cdot 2^{O(n)}$.*

As we describe later, an M-ellipsoid E of a convex body $K \subseteq \mathbb{R}^n$ is an ellipsoid with roughly the same 'size' and 'shape' as K. We will show that it can generated in randomized $\mathrm{poly}(n)$ time with high probability, and verified in deterministic $2^{O(n)}$ time, and hence can always be computed in expected $2^{O(n)}$ time. Moreover, in many specific cases of interest, such as the unit ball of any ℓ_p norm, an M-ellipsoid is deterministically computable in $\mathrm{poly}(n)$ time.

Our enumeration algorithm is at the core of the following applications. We begin with the Shortest Vector Problem in *any* "well-centered" semi-norm.[2]

Theorem 1.2 (SVP in any (semi-)norm, informal)**.** *There is a deterministic $2^{O(n)}$-time (and -space) algorithm that, given any well-centered n-dimensional convex body K and an M-ellipsoid E of K, solves SVP exactly on any n-dimensional lattice L in the semi-norm $\|\cdot\|_K$ defined by K.*

Besides being a novel algorithm, the improvement over previous approaches is in the generalization to (semi-)norms defined by arbitrary convex bodies, the use of much less randomness (if any), and in having a Las Vegas algorithm whose output is guaranteed to be correct.

We get a similar algorithm for the Closest Vector Problem, but its complexity grows with the distance from the target point to the lattice.

Theorem 1.3 (CVP in any (semi-)norm, informal)**.** *There is a deterministic algorithm that, given any well-centered n-dimensional convex body K and an M-ellipsoid E of K, solves CVP exactly on any n-dimensional lattice L in the semi-norm $\|\cdot\|_K$ defined by K, in $(1 + 2\alpha)^n \cdot 2^{O(n)}$ time and space, provided that the distance from the query point x to L is at most α times the length of the shortest nonzero vector of L (under $\|\cdot\|_K$).*

A main motivation of our work is to develop more powerful tools for solving Integer Programming. We note that solving IP reduces to solving CVP in any well-centered semi-norm: to decide if $K \cap L \neq \emptyset$, first approximate the centroid b of K, then solve CVP with respect to the well-centered body $K - b$ on lattice L and target point b. Then $K \cap L \neq \emptyset$ if and only if there exists $y \in L$ such that $\|y - b\|_{K-b} \leq 1$. However,

[1] For simplicity, throughout this introduction the claimed running times will omit polynomial factors in the lengths of the algorithms' inputs, which are represented in the usual way.

[2] "Well-centered" means that $\mathrm{vol}(K \cap -K) \geq 4^{-n} \mathrm{vol}(K)$; this clearly holds for centrally symmetric K, which corresponds to a standard norm. It also holds for any convex body K with centroid at or very near the origin.

unless we have a bound on the ratio α from the above theorem, we may not get a satisfactory guarantee on the running time of our CVP algorithm in this setting.

For the general case, we can still get an unqualified improvement in the state of the art for IP using our SVP algorithm for general norms.

Theorem 1.4 (Integer Programming, informal)**.** *There exists a randomized algorithm that, given a convex body $K \subseteq \mathbb{R}^n$ and an n-dimensional lattice $L \subset \mathbb{R}^n$, either decides that $K \cap L = \emptyset$ or returns a point $y \in K \cap L$ in expected $O(f^*(n))^n$ time, where $f^*(n)$ is the optimal bound for the "flatness theorem."*

The flatness theorem, a fundamental result in the geometry of numbers, says that every lattice-free convex body has lattice width bounded by a function of the dimension alone (see Equation (4.8) for a precise statement). As first noticed by Lenstra [Len83], it suggests a recursive algorithm for IP that uses a subroutine for finding good flatness directions. Finding an optimal flatness direction directly reduces to solving an SVP in a general norm, which was solved only approximately in previous refinements of Lenstra's algorithm. The above is therefore an essentially "optimal" Lenstra-type algorithm with respect to the classical analysis.

Using the current best known bounds on $f^*(n)$ [BLPS99, Rud00], our IP algorithm has a main complexity term of order $O(n^{4/3}\log^c n)^n$. This improves on the previous fastest algorithm of Hildebrand and Köppe [HK10] which gives a leading complexity term of $O(n^2)^n$; the previous best before that is due to Kannan [Kan87] and achieves a leading complexity term of $O(n^{2.5})^n$. It is conjectured that $f^*(n) = \Theta(n)$ [BLPS99], and this would give a bound of $O(n)^n$ for IP.

In the rest of this introduction we give an overview of our enumeration technique and its application to SVP, CVP, and IP.

Enumeration via M-ellipsoid coverings. We now explain the main technique underlying Theorem 1.1 (enumeration of lattice points in a convex body K). The key concept we use is a classical notion from asymptotic convex geometry, known as the *M-ellipsoid*. An M-ellipsoid E for a convex body K has the property that $2^{O(n)}$ copies (translates) of E can be used to cover K, and $2^{O(n)}$ copies of K suffice to cover E. The latter condition immediately implies that

$$G(E, L) \leq 2^{O(n)} \cdot G(K, L). \tag{1.2}$$

Using the former condition, enumerating $K \cap L$ therefore reduces to enumerating $(E+t) \cap L$ for at most $2^{O(n)}$ values of t (and keeping only those lattice points in K), which can be done in deterministic $2^{O(n)} \cdot G(E, L)$ time by (an extension of) the MV algorithm [MV10].

The existence of an M-ellipsoid for any convex body K was established by Milman [Mil86, MP00], and there are now multiple proofs. Under the famous *slicing conjecture* [Bou86], an appropriate scaling of K's *inertial ellipsoid* (defined by the covariance matrix of a uniform random point from K) is in fact an M-ellipsoid. When K is an ℓ_p ball, an M-ellipsoid is simply the scaled ℓ_2 ball $n^{1/2-1/p} \cdot B_2^n$.

For general convex bodies K, we give an algorithm for computing an M-ellipsoid of K, along with a covering by copies of the ellipsoid. Under the slicing conjecture, the former task is straightforward: simply estimate the covariance matrix of K using an algorithm for sampling uniformly from a convex body (e.g., [DFK89]). To avoid assuming the slicing conjecture, we use an alternative proof of M-ellipsoid existence due to Klartag [Kla06]. The resulting guarantees can be stated as follows.

Theorem 1.5 (M-ellipsoid generator, informal)**.** *There is a polynomial-time randomized algorithm that with high probability computes an M-ellipsoid E of a given n-dimensional convex body K.*[3]

[3]We thank Bo'az Klartag for suggesting to us that the techniques in [Kla06] could be used to algorithmically construct an M-ellipsoid.

Theorem 1.6 (M-ellipsoid covering algorithm, informal)**.** *Given an ellipsoid E and convex body K, there is a deterministic $2^{O(n)}$-time algorithm which certifies that E is an M-ellipsoid of K, and if so returns a covering of K by $2^{O(n)}$ copies of E.*[4]

Combining these two theorems, we get an expected $2^{O(n)}$-time algorithm that is guaranteed to output an M-ellipsoid and its implied covering for any given convex body K. It is an interesting open problem to find a *deterministic* $2^{O(n)}$-time algorithm. We note that deterministic algorithms must have complexity $2^{\Omega(n)}$, since an M-ellipsoid gives a $2^{O(n)}$ approximation to the volume of K, and such an approximation is known to require $2^{\Omega(n)}$ time when K is specified by an oracle [FB86].

Shortest and Closest Vector Problems. Here we outline our deterministic $2^{O(n)}$-time algorithm for SVP in any norm defined by a symmetric convex body K, given an M-ellipsoid of K. (Well-centered semi-norms are dealt with similarly.) For instance, as noted above the scaled ℓ_2 ball $E_p = n^{1/2-1/p} \cdot B_2^n$ is an M-ellipsoid for any ℓ_p ball $K = B_p^n$. Moreover, a good covering of B_p^n by E_p is straightforward to obtain: for $p \geq 2$, just one copy of E_p works (since $B_p^n \subseteq E_p$), while for $1 \leq p < 2$, we can cover B_p^n by a tiling of E_p's axis-aligned inscribed cuboid.

Let L be an n-dimensional lattice, and let $\lambda_1 = \lambda_1(K, L)$ be the length of its shortest vector under $\|\cdot\|_K$. We can assume by rescaling that $1/2 < \lambda_1 \leq 1$, so K contains an SVP solution. Our algorithm simply enumerates all nonzero points in $K \cap L$ (using Theorem 1.1), and outputs one of the shortest. For the running time, it suffices to show that $G(K, L) \leq 2^{O(n)}$, which follows by a simple packing argument: for any $x \in \mathbb{R}^n$, copies of $\frac{1}{4}K$ centered at each point in $(K + x) \cap L$ are pairwise disjoint (because $\lambda_1 > 1/2$) and contained in $\frac{5}{4}K + x$, so $|(K + x) \cap L| \leq \mathrm{vol}(\frac{5}{4}K)/\mathrm{vol}(\frac{1}{4}K) = 5^n$.

For CVP with target point x, the strategy is exactly the same as above, but we use a scaling dK so that $(dK - x) \cap L \neq \emptyset$ and $(\frac{d}{2}K - x) \cap L = \emptyset$ (i.e., d is a 2-approximation of the distance from x to L). In this case, the packing argument gives a bound of $G(dK, L) \leq (1 + 2d/\lambda_1)^n$.

In retrospect, the above algorithms can be seen as a derandomization (and generalization to semi-norms) of the AKS sieve-based algorithms for exact SVP in general norms, and exact CVP in ℓ_p norms [AKS01, AKS02, BN07, AJ08], with matching running times (up to $2^{O(n)}$ factors). Specifically, our algorithms deterministically enumerate all lattice points in a convex region, rather than repeatedly sampling until all such points are found with high probability. However, we do not know whether our techniques can derandomize the $(1 + \epsilon)$-approximate CVP algorithms of [AKS02, BN07] in asymptotically the same running time.

Integer Programming. Our algorithm for Integer Programming (finding a point in $K \cap L$, if it exists) follows the basic outline of all algorithms since that of Lenstra [Len83]. It begins with two pre-processing steps: one to refine the basis of the lattice, and the other to find an ellipsoidal approximation of K. If the ellipsoid volume is sufficiently small compared to the lattice determinant, then we can directly reduce to a lower-dimensional problem. The main step of the algorithm (and Lenstra's key insight, refined dramatically by Kannan [Kan87]) is to find a direction along which the lattice width of K is small. Given such a direction, we recurse on the lattice hyperplanes orthogonal to this direction that intersect K, thus reducing the dimension of the problem by one.

In previous work, a small lattice-width direction was found by replacing K by an ellipsoid E *containing* K, then solving SVP in the norm defined by the dual ellipsoid E^* on the dual lattice L^*. Here we instead use our SVP algorithm for general norms, solving it directly for the norm induced by $(K - K)^*$ on L^*. This refinement allows us to use the best-known bounds on $f^*(n)$ (from the flatness theorem) for the number of hyperplanes on which we have to recurse.

[4] Gideon Schechtman suggested a construction of the covering using parallelepiped tilings.

1.2 Organization

The remainder of the paper is organized as follows. In Section 2 we recall basic concepts from convex geometry that are needed to understand our M-ellipsoid algorithms. In Section 3 we give the M-ellipsoid construction (formalizing Theorems 1.5 and 1.6). In Section 4 we formalize our enumeration technique (Theorem 1.1) and apply it to give algorithms for SVP, CVP and IP. Appendix A contains the proofs of correctness for our M-ellipsoid construction, and Appendix B contains supporting technical material.

2 Convex Geometry Background

Convex bodies. $K \subseteq \mathbb{R}^n$ is a convex body if K is convex, compact and full-dimensional. We say that a body is centrally symmetric, or 0-symmetric, if $K = -K$.

For sets $A, B \in \mathbb{R}^n$ we define the Minkowski sum of A and B as

$$A + B = \{x + y : x \in A, y \in B\}. \tag{2.1}$$

For a vector $t \in \mathbb{R}^n$, we define $t + A = \{t\} + A$ for notational convenience.

Let $K \subseteq \mathbb{R}^n$ be a convex body such that $0 \in K$. We define the gauge function, or Minkowski functional, of K as

$$\|x\|_K = \inf\{r \geq 0 : x \in rK\}, \quad x \in \mathbb{R}^n. \tag{2.2}$$

From classical convex analysis, we have that the functional $\|\cdot\|_K$ is a semi-norm, i.e., it satisfies the triangle inequality and $\|tx\|_K = t\|x\|_K$ for $t \geq 0$, $x \in \mathbb{R}^n$. If K is centrally symmetric, then $\|.\|_K$ is a norm in the usual sense.

The *polar* (or *dual*) body K^* is defined as

$$K^* = \{x \in \mathbb{R}^n : \forall y \in K,\ \langle x, y\rangle \leq 1\}. \tag{2.3}$$

A basic result in convex geometry is that K^* is convex and that $(K^*)^* = K$.

Define the ℓ_p norm on $\mathbb{R}^n$ as

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}. \tag{2.4}$$

For convenience we write $\|x\|$ for $\|x\|_2$. Let $B_p^n = \{x \in \mathbb{R}^n : \|x\|_p \leq 1\}$ denote the ℓ_p ball in $\mathbb{R}^n$. Note from our definitions that $\|x\|_{B_p^n} = \|x\|_p$ for $x \in \mathbb{R}^n$.

For a positive definite matrix $A \in \mathbb{R}^{n\times n}$, we define the inner product with respect to A as

$$\langle x, y\rangle_A = x^t A y \quad x, y \in \mathbb{R}^n. \tag{2.5}$$

We define the norm generated by A as $\|x\|_A = \sqrt{\langle x, x\rangle_A} = \sqrt{x^t A x}$. For a vector $a \in \mathbb{R}^n$, we define the ellipsoid $E(A, a) = \{x \in \mathbb{R}^n : \|x - a\|_A \leq 1\}$. For convenience we shall let $E(A) = E(A, 0)$. Note that with our notation, $\|x\|_A = \|x\|_{E(A)}$. The volume of an ellipsoid $E(A, a)$ is given by the formula

$$\mathrm{vol}(E(A, a)) = \mathrm{vol}(E(A)) = \mathrm{vol}(B_2^n) \cdot \sqrt{\det(A^{-1})}. \tag{2.6}$$

Lastly, an elementary computation gives the useful fact that $E(A)^* = E(A^{-1})$.

We define the *centroid* (or *barycenter*) $b(K) \in \mathbb{R}^n$ and *covariance* matrix $\mathrm{cov}(K) \in \mathbb{R}^{n\times n}$ as

$$b(K) = \int_K \frac{x\,dx}{\mathrm{vol}(K)} \qquad\qquad \mathrm{cov}(K) = \int_K (x-b(K))(x-b(K))^t \frac{dx}{\mathrm{vol}(K)}.$$

We note that $\mathrm{cov}(K)$ is always positive definite and symmetric. The *inertial ellipsoid* of K is defined as $E_K = E(\mathrm{cov}(K)^{-1})$. The *isotropic constant* of K is

$$L_K = \det(\mathrm{cov}(K))^{\frac{1}{2n}} / \mathrm{vol}(K)^{\frac{1}{n}}. \tag{2.7}$$

A major open conjecture in convex geometry is the following:

Conjecture 2.1 (Slicing Conjecture [Bou86])**.** There exists an absolute constant $C > 0$, such that $L_K \le C$ for all $n \ge 0$ and any convex body $K \subseteq \mathbb{R}^n$.

The original bound computed by Bourgain [Bou86] was $L_K = O(n^{1/4}\log n)$. This has since been improved by Klartag [Kla06] to $L_k = O(n^{1/4})$. In addition, the conjecture has been verified for many classes of bodies including the ℓ_p norm balls.

The above concepts (centroid, covariance, isotropic constant, inertial ellipsoid) all generalize easily to logconcave functions in lieu of convex bodies; see Appendix B for details.

Computational model. All our algorithms will work with convex bodies and norms presented by oracles in the standard way. The complexity of our algorithms will be measured by the number of arithmetic operations as well as the number of calls to the oracle. See Appendix B for a more detailed description of the kinds of oracles we use.

3 Computing M-Ellipsoids and Coverings

An M-ellipsoid of a convex body K is an ellipsoid E with the property that at most $2^{O(n)}$ translated copies of E are sufficient to cover all of K, and at most $2^{O(n)}$ copies of K are sufficient to cover E. More precisely, for any two subsets $A, B \in \mathbb{R}^n$, define the covering number

$$N(A,B) = \min\{|\Lambda| : \Lambda \subseteq \mathbb{R}^n, A \subseteq B + \Lambda\}. \tag{3.1}$$

Hence $N(A,B)$ is the minimum number of translates of B needed to cover A. The following theorem was first proved for symmetric bodies by Milman [Mil86] and extended by Milman and Pajor [MP00] to the general case.

Theorem 3.1 ([MP00])**.** *There exists an absolute constant $C > 0$, such that for all $n \ge 1$ and any convex body $K \subseteq \mathbb{R}^n$, there exists an ellipsoid E satisfying*

$$N(K,E)\cdot N(E,K) \le C^n. \tag{3.2}$$

Definition 3.2 (M-ellipsoid)**.** Let $K \subseteq \mathbb{R}^n$ be a convex body. If E is an ellipsoid satisfying Equation (3.2) (for some particular fixed C) with respect to K, then we say that E is an M-ellipsoid of K.

There are many equivalent ways of understanding the M-ellipsoid; here we list a few (proofs of many of these equivalences can be found in [MP00]).

Theorem 3.3. *Let $K \subseteq \mathbb{R}^n$ be a convex body with $b(K) = 0$ (centroid at the origin), and let $E \subseteq \mathbb{R}^n$ be an origin-centered ellipsoid. Then the following conditions are equivalent, where the absolute constant C may vary from line to line:*

1. $N(K, E) \cdot N(E, K) \leq C^n$.
2. $\mathrm{vol}(K + E) \leq C^n \cdot \min\{\mathrm{vol}(E), \mathrm{vol}(K)\}$.
3. $\sup_{t \in \mathbb{R}^n} \mathrm{vol}(K \cap (t + E)) \geq C^{-n} \cdot \max\{\mathrm{vol}(E), \mathrm{vol}(K)\}$.
4. *E^* is an M-ellipsoid of K^*.*

From the above we see that the M-ellipsoid is very robust object, and in particular is stable under polarity (assuming K is well-centered). We will use this fact (or a slight variant of it) in what follows, to help us certify a candidate M-ellipsoid.

As mentioned in the introduction, an M-ellipsoid for an ℓ_p ball is trivial to compute. Using condition 3 of Theorem 3.3 above and standard volume estimates for ℓ_p balls, i.e., that $\mathrm{vol}(B_p^n)^{1/n} = \Theta(n^{-1/p})$, we have the following:

Lemma 3.4. *Let B_p^n denote the n-dimensional ℓ_p ball. Then*

- *For $1 \leq p \leq 2$, $n^{\frac{1}{2}-\frac{1}{p}} \cdot B_2^n \subseteq B_p^n$ (the largest inscribed ball in B_p^n) is an M-ellipsoid for B_p^n.*
- *For $p \geq 2$, $n^{\frac{1}{2}-\frac{1}{p}} \cdot B_2^n \supseteq B_p^n$ (the smallest containing ball of B_p^n) is an M-ellipsoid for B_p^n.*

For general convex bodies, the proofs of existence of an M-ellipsoid in [Mil86] and [MP00] are non-constructive. It is worth noting, however, that under the *slicing conjecture* (also known as the *hyperplane conjecture*), a $\sqrt{n}$ scaling of K's inertial ellipsoid is an M-ellipsoid — indeed, this is an equivalent form of the slicing conjecture. For many norms, including ℓ_p, absolutely symmetric norms (where the norm is preserved under coordinate sign flips), and other classes, the slicing conjecture has been proved. Therefore, for such norms, an M-ellipsoid computation is straightforward: using random walk techniques, estimate the covariance matrix $\mathrm{cov}(K)$ of K, the unit ball of the norm, and return a $\sqrt{n}$ scaling of K's inertial ellipsoid.

In the rest of this section, we describe how to generate an M-ellipsoid in general, without directly relying on the slicing conjecture, with good probability in probabilistic polynomial time. Moreover, we show how to certify that an ellipsoid is an M-ellipsoid in deterministic $2^{O(n)}$ time. A by-product of the certification is a covering of the target body by at most $2^{O(n)}$ translates of the candidate M-ellipsoid. Such a covering will be used by all the lattice algorithms in this paper.

Proofs for all the theorems in this section can be found in Appendix A.

3.1 The Main Algorithm

The main result of this section is Algorithm 1 (M-Ellipsoid), whose correctness is proved in Theorem 3.5. The algorithm uses two main subroutines. The first, M-Gen, described in Section 3.2 below, produces a candidate ellipsoid that is an M-ellipsoid with good probability. The second, Build-Cover, described in Section 3.3, is used to check that both $N(K, E), N((K - K)^*, E^*) = 2^{O(n)}$ by constructing explicit coverings (if possible). Because $N(E, K) \approx N((K - K)^*, E^*)$ (up to $2^{\Theta(n)}$ factors) by the duality of entropy (Theorem A.2), such coverings suffice to prove that E is an M-ellipsoid for K.

Algorithm 1 M-Ellipsoid: Generate a guaranteed M-ellipsoid and its implied covering.

Input: A weak membership oracle O_K for a $(0, r, R)$-centered convex body K.

Output: An M-ellipsoid E of K, and a covering of K by $2^{O(n)}$ copies of E.

1: Approximate the centroid of K using algorithm Estimate-Centroid (Lemma B.9). If Estimate-Centroid fails, restart; otherwise, let b denote returned estimate for $b(K)$.
2: Generate a candidate M-ellipsoid E of K using Algorithm 2 (M-Gen) on $K - b$.
3: Check if $N(K, E) > (13e)^n$ using Algorithm 3 (Build-Cover). If yes, restart; otherwise, let T denote the returned covering of K by E.
4: Check if $N((K - K)^*, E^*) > (25e \cdot 13)^n$ using Algorithm 3 (Build-Cover). If yes, restart; otherwise, return (E, T).

Theorem 3.5 (Correctness of M-Ellipsoid)**.** *For large enough n, Algorithm 1 (M-Ellipsoid) outputs an ellipsoid E satisfying*

$$N(K, E) \le \left(\sqrt{8\pi e} \cdot 13e\right)^n \quad \textit{and} \quad N(E, K) \le \left(\sqrt{8\pi e} \cdot 25e \cdot 13 \cdot 289\right)^n \tag{3.3}$$

along with a set $T \subseteq \mathbb{Q}^n$, $|T| \le \left(\sqrt{8\pi e} \cdot 13e\right)^n$ such that $K \subseteq T + E$, in expected time $\left(\sqrt{8\pi e} \cdot 25e \cdot 13\right)^n \cdot \mathrm{poly}(n, \log(\frac{R}{r}))$.

3.2 Generating a Candidate M-Ellipsoid

Our algorithm for generating a candidate M-ellipsoid is based on a constructive proof of Theorem 3.1 by Klartag [Kla06], who suggested to us the idea of using these techniques to build an M-ellipsoid algorithmically. The main theorem of [Kla06], reproduced below, does not explicitly refer to M-ellipsoids; instead, it shows that for every convex body K, there is another convex body K' that sandwiches K between two small scalings and satisfies the slicing conjecture.

Theorem 3.6 ([Kla06])**.** *Let $K \subseteq \mathbb{R}^n$ be a convex body. Then for every real $\epsilon \in (0, 1)$, there exists a convex body $K' \subseteq \mathbb{R}^n$ such that*

$$d(K, K') = \inf\{\frac{b}{a} : \exists\, t \in \mathbb{R}^n \text{ s.t. } aK' \subseteq K - t \subseteq bK'\} \le 1 + \epsilon \quad \textit{and} \quad L_{K'} \le \frac{c}{\sqrt{\epsilon}}. \tag{3.4}$$

where $c > 0$ is an absolute constant and $L_{K'}$ is the isotropic constant of K'.

From the closeness of K and K' it follows that an M-ellipsoid for K' is an M-ellipsoid for K, and from the bound on $L_{K'}$ the inertial ellipsoid of K' is an M-ellipsoid for K'.

Here we will not need to construct K' itself, but only an ellipsoid very close to its inertial ellipsoid (which as just mentioned is an M-ellipsoid for K). The body K' is derived from a certain family of reweighted densities over K. These densities are given by exponential reweightings of the uniform density along some vector $s \in \mathbb{R}^n$, i.e., $f_s(x) = e^{\langle s, x\rangle}$ for $x \in K$ (and 0 otherwise). For s chosen uniformly from $n \cdot \mathrm{conv}\{K - b(K), b(K) - K\}^*$, the reweighting f_s has two important properties: (i) it is not too highly biased away from uniform over K, and (ii) it has bounded isotropic constant (independent of n) with very high probability. Let E be the inertial ellipsoid of f_s (or any reasonably good approximation to it), which can be found by sampling from f_s. The first property of f_s allows us to prove that E can be covered by $2^{O(n)}$ copies of K, while the second property lets us cover K by $2^{O(n)}$ copies of E (see Lemma A.3).

To make everything work algorithmically, we need robust versions of Klartag's main lemmas, since we will only be able to compute an approximate centroid of K, sample s from a distribution close to uniform, and estimate the covariance matrix of f_s.

Algorithm 2 makes the above description more formal. Note that given an oracle for a convex body, an oracle for the polar body can be constructed in polynomial time [GLS88]. Sampling, both from the uniform and exponentially reweighted distributions, can be done in polynomial time using the random walk algorithm of [LV06b, LV06a]. Theorem A.3 together with Lemma A.4 implies that the algorithm's output is indeed an M-ellipsoid with good probability.

Algorithm 2 M-Gen: Randomized generation of a candidate M-ellipsoid.

Input: A weak membership oracle O_K for a $(0, r, R)$-centered convex body K with $b(K) \in \frac{1}{n+1} E_K$.
Output: With probability $1 - o(1)$, an M-ellipsoid of K.

1: Estimate the centroid $b = b(K)$ using uniform samples from K.
2: Construct a membership oracle for $n\left(\mathrm{conv}\{K - b, b - K\}\right)^*$.
3: Sample a random vector s from $n\left(\mathrm{conv}\{K - b, b - K\}\right)^*$.
4: Estimate the covariance matrix A of the density proportional to $e^{\langle s,x\rangle}$, restricted to K.
5: Output the ellipsoid $E(A^{-1}) = \{x : x^t A^{-1} x \leq 1\}$.

Theorem 3.7 (Correctness of M-Gen)**.** *For large enough n, Algorithm 2 (M-Gen) outputs an ellipsoid E satisfying*

$$N(E, K) \leq (25e)^n \quad \textit{and} \quad N(K, E) \leq (13e)^n \tag{3.5}$$

with probability at least $1 - \frac{3}{n}$ in time $\mathrm{poly}(n, \log(\frac{R}{r}))$.

3.3 Building a Covering

The next theorem yields an algorithm to approximately decide (up to single exponential factors) whether a given convex body K can be covered by a specified number of translates of an ellipsoid E. The algorithm is constructive and proceeds by constructing a simple parallelepiped tiling of K, where the parallelepiped in question is a maximum volume inscribed parallelepiped of E.

Algorithm 3 Build-Cover: Deterministic construction of an ellipsoid covering of a convex body.

Input: A weak membership oracle O_K for an $(0, r, R)$-centered convex body K, an ellipsoid $E = E(A)$, and some $H \geq 1$.
Output: Either a covering of K by $(\sqrt{8\pi e} H)^n$ translates of E, or a declaration that K cannot be covered by H^n copies of E.

1: Let C_E be any maximum-volume inscribed parallelepiped of E (e.g., a maximum-volume inscribed cuboid with the same axes as the ellipsoid).
2: Attempt to cover K using translates of C_E with respect to the natural parallelepiped tiling, via a breadth-first search over the tiling lattice, starting from the origin.
3: If the attempted covering grows larger than $(\sqrt{8\pi e} H)^n$, abort. Otherwise, output the covering.

Theorem 3.8. *Algorithm 3 (Build-Cover) is correct, and runs in time $\left(\sqrt{8\pi e} H\right)^n \cdot \mathrm{poly}(n, \langle A \rangle, \log(\frac{R}{r}))$.*

4 Lattice Algorithms

In this section we prove our general enumeration theorem for convex bodies (Theorem 1.1, formalized in Theorem 4.2) and give its application to the Shortest and Closest Vector Problems, and Integer Programming.

4.1 Lattice Background

An n-dimensional lattice $L \subset \mathbb{R}^n$ is a discrete subgroup under addition. It can be written as

$$L = \left\{ \sum_{i=1}^{k} z_i b_i : z_i \in \mathbb{Z} \right\} \tag{4.1}$$

for some (not necessarily unique) *basis* $B = (b_1, \ldots, b_k)$ of $k \leq n$ linearly independent vectors in $\mathbb{R}^n$. The determinant of L is defined as

$$\det(L) = \sqrt{\det(B^t B)}. \tag{4.2}$$

The *dual lattice* L^* of L is defined as

$$L^* = \{y \in \operatorname{span}(b_1, \ldots, b_k) : \forall x \in L, \langle x, y \rangle \in \mathbb{Z}\}. \tag{4.3}$$

The *minimum distance* of L with respect to K is $\lambda_1(K, L) = \min_{y \in L \setminus \{0\}} \|y\|_K$. The *covering radius* of L with respect to K is $\mu(K, L) = \inf\{s \geq 0 : L + sK = \mathbb{R}^n\}$. Note that from the definition, we see that $\mu(K + t, L) = \mu(K, L)$ for $t \in \mathbb{R}^n$ and that $\mu(-K, L) = \mu(K, L)$. We also define $d_K(L, x) = \inf_{y \in L} \|y - x\|_K$. We define the i^{th} *minimum* of L with respect to the ℓ_2 norm as

$$\lambda_i(L) = \inf\{r \geq 0 : \dim(\operatorname{span}(rB_2^n \cap L)) \geq i\}$$

where span denotes the linear span.

The *shortest vector problem* (SVP) with respect to K is the following: given a basis of an n-dimensional lattice L, compute an element of

$$\mathrm{SVP}(K, L) = \arg\min_{y \in L \setminus \{0\}} \|y\|_K. \tag{4.4}$$

The *closest vector problem* (CVP) with respect to K is: given a basis of an n-dimensional lattice L and a point $x \in \mathbb{R}^n$, compute an element of

$$\mathrm{CVP}(K, L, x) = \arg\min_{y \in L} \|y - x\|_K. \tag{4.5}$$

To denote the sets of *approximate* minimizers for SVP and CVP, we define for any $\epsilon > 0$

$$\mathrm{SVP}_\epsilon(K, L) = \{z \in L \setminus \{0\} : \|z\|_K \leq (1 + \epsilon) \cdot \min_{y \in L \setminus \{0\}} \|y\|_K\} \tag{4.6}$$

$$\mathrm{CVP}_\epsilon(K, L, x) = \{z \in L : \|z - x\|_K \leq (1 + \epsilon) \cdot \min_{y \in L} \|y - x\|_K\}. \tag{4.7}$$

Integer programming. A fundamental tool in integer programming is the so-called "flatness theorem," which says that for any convex body $K \subseteq \mathbb{R}^n$ and n-dimensional lattice $L \subseteq \mathbb{R}^n$,

$$1 \leq \mu(K, L) \cdot \lambda_1((K-K)^*, L^*) \leq f(n), \tag{4.8}$$

where $\mu(K, L) = \inf\{s \geq 0 : L + sK = \mathbb{R}^n\}$ is the covering radius of L, and

$$\lambda_1((K-K)^*, L^*) = \inf_{y \in L^* \setminus \{0\}} \big(\sup_{x \in K} \langle x, y \rangle - \inf_{x \in K} \langle x, y \rangle \big)$$

is the lattice width of K. The flatness theorem is most easily interpreted as follows: either K certainly contains a lattice point in L, or there exist at most $\lfloor f(n) \rfloor + 1$ hyperplanes of the form $H_k = \{x \in \mathbb{R}^n : \langle y, x \rangle = k\}$, $y \in L^* \setminus \{0\}$, $k \in \mathbb{Z}$ and $\inf_{x \in K} \langle y, x \rangle \leq k \leq \sup_{x \in K} \langle y, x \rangle$, such that any lattice point in K must lie on one of these hyperplanes. Crucially, we note that computing $\lambda_1((K-K)^*, L^*)$ for a general convex body K is exactly a shortest non-zero vector computation with respect to a general norm.

The asymptotic growth (and even the finiteness) of the function $f(n)$ in (4.8) has been the source of intense study over the past century. Restricting to the important special case where $K = B_2^n$, the optimal growth rate has been settled at $f(n) = \Theta(n)$ [Ban93]. When K is centrally symmetric, the best known bound is $f(n) = O(n \log n)$ [Ban96]. For the general case, the current best bound is $f(n) = O(n^{\frac{4}{3}} \log^c n)$ [BLPS99, Rud00] for some fixed $c > 0$. We let $f^*(n)$ denote best possible upper bound for the general flatness theorem.

4.2 Lattice Point Enumeration in Convex Bodies

We now use enumeration via the M-ellipsoid covering to solve the Shortest and Closest Vector Problems. To do this we will need the recent algorithm of Micciancio and Voulgaris [MV10] for the Closest Vector Problem under the ℓ_2 norm (and hence any ellipsoidal norm), which we call the MV algorithm for short. The following is an immediate extension of their graph-traversal approach [Vou].

Proposition 4.1 ([MV10], Algorithm Ellipsoid-Enum)**.** *There is an algorithm Ellipsoid-Enum that, given any positive definite $A \in \mathbb{Q}^{n \times n}$, any basis B of an n-dimensional lattice $L \subseteq \mathbb{R}^n$, and any $t \in \mathbb{R}^n$, computes the set $L \cap (E(A) + t)$ in deterministic time*

$$2^{O(n)} \cdot (|L \cap (E(A) + t)| + 1) \cdot \mathrm{poly}(\langle A \rangle, \langle B \rangle, \langle t \rangle). \tag{4.9}$$

Here the idea is that the points inside $(E(A) + t) \cap L$ form a connected subgraph, where we consider two lattice points adjacent if they differ by a Voronoi-relevant vector of L, where Voronoi relevance is defined with respect to the inner product defined by A (see [MV10] for formal definitions). An initial point inside $(E(A) + t) \cap L$ can be computed (if it exists) in a single call to the MV algorithm, and the rest can be computed by a standard breadth-first search of the graph.

For a convex body $K \subseteq \mathbb{R}^n$ and a lattice $L \subseteq \mathbb{R}^n$ define

$$G(K, L) = \max_{x \in \mathbb{R}^n} |(K + x) \cap L|, \tag{4.10}$$

the maximum number of lattice points in K under any translation.

We can now state our enumeration theorem, which formalizes Theorem 1.1 from the introduction.

Algorithm 4 Algorithm Lattice-Enum(K, L, x, d, ϵ)

Input: An $(0, r, R)$-centered convex body K presented by a weak distance oracle D_K for $\|\cdot\|_K$, a basis B for a lattice L, an input point x, distance $d \geq 0$, and $0 < \epsilon < 1$.

Output: $S \subseteq L$ satisfying (4.11).

1: Let $(E, T) \leftarrow$ M-Ellipsoid(K) ▷ This covering need only be computed once for repeated calls.
2: Let $S \leftarrow \emptyset$
3: **for all** $s \in T$ **do**
4: Let $U_s \leftarrow$ Ellipsoid-Enum$(dE,\ L, x + ds)$
5: $S \leftarrow S \cup \{y : y \in U_S, D_K(y - x, \frac{\epsilon}{2}) \leq d + \frac{\epsilon}{2}\}$
6: **return** S

Theorem 4.2 (Enumeration in convex bodies)**.** *Algorithm 4 (Lattice-Enum) outputs a set $S \subseteq L$ such that*

$$\{y \in L : \|y - x\|_K \leq d\} \subseteq S \subseteq \{y \in L : \|y - x\|_K \leq d + \epsilon\} \tag{4.11}$$

in expected time $G(dK, L) \cdot 2^{O(n)} \cdot \mathrm{poly}(\log(\frac{R}{r}), \log(\frac{1}{\epsilon}), \langle B\rangle, \langle x\rangle)$.

Proof.

Correctness: We first note that $K \subseteq \cup_{s\in T}\ s + E$ then $x + dK \subseteq \cup_{s\in T}\ x + d(s + E)$. Hence given a covering for K, we have a covering of $dK + t$. Now on input $(dE,\ L,\ x + ds)$ the algorithm Ellipsoid-Enum returns the set $(x + ds) + dE \cap L = x + d(s + E) \cap L$.

Now we first show that for all $y \in x + dK \cap L$, $y \in S$. By the covering property, we know that for some $s \in T$, $y \in x + (s + E) \cap L$. Finally, by the properties of the weak-semi norm oracle since $y \in dK + x \Leftrightarrow \|y - t\|_K \leq d$, we have that

$$D_K(y - x, \frac{\epsilon}{2}) \leq \|y - x\|_K + \frac{\epsilon}{2} \leq d + \frac{\epsilon}{2},$$

and hence y is correctly placed in S as needed. Lastly, we must show that if $y \notin (d + \epsilon)K + x \Leftrightarrow \|y - t\| > d + \epsilon$, then $y \notin S$. Again, from the properties of the weak distance oracle we see that

$$D_K(y - x, \frac{\epsilon}{2}) \geq \|y - x\|_K - \frac{\epsilon}{2} > d + \epsilon - \frac{\epsilon}{2} = d + \frac{\epsilon}{2}$$

as needed. Lastly, by construction, the set S only contains lattice points, and so by the above arguments U satisfies the required properties.

Runtime: By Theorem 3.5, M-Ellipsoid computes an M-ellipsoid in expected time $\mathrm{polylog}(\frac{R}{r})C_1^n$. Let E denote an M-ellipsoid of K and let $T \subseteq \mathbb{R}^n$ be as above. From Theorem 3.5, we know that $|T| \leq C_2^n$, hence the algorithm makes at most C_2^n calls to Ellipsoid-Enum. Now to bound the complexity of enumerating $x + d(s + E) \cap L$ for each $s \in T$, we need to bound $|x + d(s + E) \cap L| \leq G(dE, L)$. Now we note that

$$G(dE, L) \leq N(dE, dK)G(dK, L) = N(E, K)G(dK, L) \leq C_2^n G(dK, L)$$

by Theorem 3.5. Hence for any $s \in T$, Ellipsoid-Enum takes at most $C_3^n\ \mathrm{poly}(\langle B\rangle, \langle x\rangle)\ (C_2^n G(dK, L)) \leq \mathrm{poly}(\langle B\rangle, \langle x\rangle)\ C_4^n\ G(dK, L)$ time to compute $x + d(s + E) \cap L$. Hence the total running time is bounded by

$$\mathrm{polylog}(\tfrac{R}{r})\ C_1^n + C_2^n\ \mathrm{poly}(\langle B\rangle)\ C_4^n\ G(dK, L)\ \leq\ \mathrm{poly}(\log \tfrac{R}{r}, \langle B\rangle, \langle x\rangle)\ C_5^n\ G(dK, L) \tag{4.12}$$

where $C_5 > 0$ is an absolute constant. □

We remark that the only randomness in the algorithm is to build the M-ellipsoid; once this has been achieved the rest of the algorithm is deterministic. Hence, in the cases where the M-ellipsoid is known explicitly, as it is for the ℓ_p balls (where an appropriately scaled Euclidean ball suffices), the algorithm can be in fact made completely deterministic. The algorithms for the shortest vector and closest vector problem described in the next sections will only depend on the Lattice-Enum algorithm, and hence they will be deterministic as long as Lattice-Enum is deterministic.

4.3 Shortest Vector Problem

Our main goal will be to use the above enumeration algorithm to solve the Shortest Vector Problem. The following gives a useful bound on $G(K, L)$ for a general convex body.

Lemma 4.3. *Let $K \subseteq \mathbb{R}^n$ be a convex body satisfying $\mathrm{vol}(K \cap -K) \geq \gamma^{-n} \mathrm{vol}(K)$, $\gamma \geq 1$, and let L be an n-dimensional lattice. Then for $d > 0$ we have that*

$$G(dK, L) \leq \left(\gamma\left(1 + \frac{2d}{\lambda_1(K, L)}\right)\right)^n. \tag{4.13}$$

We note γ above is easily bounded in many natural situations. When K is centrally symmetric we can set $\gamma = 1$ since $K \cap -K = K$, and if K is a general convex body with $b(K) = 0$ setting $\gamma = 2$ is valid by Theorem A.1. Hence the notion of "well-centered", i.e., $\gamma \leq 4$, is quite robust.

Proof of Lemma 4.3. Let $s = \frac{1}{2}\lambda_1(K, L)$. For $x \in L$, we examine

$$x + \mathrm{int}(s(K \cap -K)) = \{z \in \mathbb{R}^n : \|z - x\|_{K \cap -K} < s\}.$$

Now for $x, y \in L$, $x \neq y$, we claim that

$$x + \mathrm{int}(s(K \cap -K)) \cap y + \mathrm{int}(s(K \cap -K)) = \emptyset \tag{4.14}$$

Assume not, then $\exists\, z \in \mathbb{R}^n$ such that $\|z - x\|_{K \cap -K}, \|z - y\|_{K \cap -K} < s$. Since $K \cap -K$ is symmetric, we note that $\|y - z\|_{K \cap -K} = \|z - y\|_{K \cap -K} < s$. But now, since $K \cap -K \subseteq K$, we see that

$$\begin{aligned}\|y - x\|_K &= \|y - z + z - x\|_K \leq \|y - z\|_K + \|z - x\|_K \\ &\leq \|y - z\|_{K \cap -K} + \|z - x\|_{K \cap -K} < s + s = 2s = \lambda_1(K, L)\end{aligned}$$

a clear contradiction since $y - x \neq 0$.

Take $c \in \mathbb{R}^n$. To bound $G(dK, L)$ we must bound $|(c + dK) \cap L|$. For $x \in c + dK$, we note that $x + s(K \cap -K) \subseteq c + (d + s)K$. Therefore,

$$\mathrm{vol}((d+s)K) = \mathrm{vol}(c+(d+s)K) \geq \mathrm{vol}\left(((c + dK) \cap L) + s(K \cap -K)\right) = |(c+dK) \cap L| \,\mathrm{vol}(s(K \cap -K)) \tag{4.15}$$

where the last equality follows from (4.14). Therefore, we have that

$$|(c + dK) \cap L| \leq \frac{\mathrm{vol}((d + s)K)}{\mathrm{vol}(s(K \cap -K))} = \left(\frac{d + s}{\gamma^{-1}s}\right)^n = \left(\gamma\left(1 + \frac{2d}{\lambda_1(K, L)}\right)\right)^n \tag{4.16}$$

as needed. □

Algorithm 5 Shortest-Vectors(K, L, ϵ)

Input: A $(0, r, R)$-centered convex body K presented by a weak distance oracle D_K for $\|\cdot\|_K$, a basis B for a lattice L, and $0 < \epsilon < 1$.

Output: $S \subseteq L$ such that $\mathrm{SVP}(K, L) \subseteq S \subseteq \mathrm{SVP}_\epsilon(K, L)$

Compute $z \in \mathrm{SVP}(B_2^n, L)$ using the MV algorithm. Set $t, d \leftarrow \frac{\|z\|}{R}$.
repeat
  $U \leftarrow$ Lattice-Enum$(K, L, 0, d, t) \setminus \{0\}$
  if $U = \emptyset$ **then**
    $d \leftarrow 2d$
until $U \neq \emptyset$
$U \leftarrow$ Lattice-Enum$(K, L, 0, d + t, t) \setminus \{0\}$
$m \leftarrow \min\{D_K(y, \frac{\epsilon}{4}t) : y \in U\}$
$S \leftarrow \{y : D_K(y, \frac{\epsilon}{4}t) \leq m + \frac{\epsilon}{2}\, t, y \in U\}$
return S

We can now state the algorithm and main theorem of this section.

Theorem 4.4 (Correctness of Shortest-Vectors)**.** *If K is well-centered, i.e.,* $\mathrm{vol}(K \cap -K) \geq 4^{-n}\,\mathrm{vol}(K)$*, then Algorithm 5 (Shortest-Vectors) outputs a set $S \subseteq L$ satisfying*

$$\mathrm{SVP}(K, L) \subseteq S \subseteq \mathrm{SVP}_\epsilon(K, L) \tag{4.17}$$

in expected time

$$2^{O(n)} \cdot \mathrm{poly}(\log(\tfrac{R}{r}), \log(\tfrac{1}{\epsilon}), \langle B \rangle). \tag{4.18}$$

Proof.

Correctness: First note that since K is $(0, r, R)$-centered, we know that $\frac{\|y\|}{R} \leq \|y\|_K \leq \frac{\|y\|}{r}$ for all $y \in \mathbb{R}^n$. Now take $z \in \mathrm{SVP}(K, L)$ and $z' \in \mathrm{SVP}(B_2^n, L)$. Let $\omega = \|z\|_K$, and as in the algorithm let $t = \frac{\|z'\|}{R}$. Now we have that

$$t = \frac{\|z'\|}{R} \leq \frac{\|z\|}{R} \leq \|z\|_K \leq \|z'\|_K \leq \frac{\|z'\|}{r} = t\frac{R}{r} \tag{4.19}$$

Therefore $t \leq \omega \leq t\frac{R}{r}$.

Now for $z \in \mathrm{SVP}(K, L)$, we must show that $z \in S$. Let d_f denote the final value of d after the while loop terminates. Since $U \neq \emptyset$ and $0 \notin U$ after the while loop terminates, and since the enumeration algorithm guarantees that $U \subseteq \{y \in L : \|y - x\|_K \leq d_f + t\}$, we have that $\omega \leq d_f + t$. Now let $U_f =$ Lattice-Enum$(K, L, 0, d_f + t, t) \setminus \{0\}$, i.e. the final setting of the set U. By the properties of Lattice-Enum, we know that $\{y \in L : \|y - x\|_K \leq d_f + t\} \subseteq U_f$, and hence we have that $\mathrm{SVP}(K, L) \subseteq U_f$. From the computation of the number m, during the final stage of the algorithm, we now see that $\omega - \frac{\epsilon}{4}t \leq m \leq \omega + \frac{\epsilon}{4}t$. Therefore for $z \in \mathrm{SVP}(K, L)$, we have that

$$D_K(z, \frac{\epsilon}{4}t) \leq \omega + \frac{\epsilon}{4}t \leq m + \frac{\epsilon}{2}t \tag{4.20}$$

and hence z will correctly be placed in S as needed.

Now assume that $z \in L \setminus \{0\}$ and $z \notin \mathrm{SVP}(K, L)_\epsilon$. We must show that $z \notin S$. Since $\omega \geq t$ from above, we have that $\|z\|_K > (1 + \epsilon)\omega \geq \omega + \epsilon t$. Therefore, we see that

$$D_K(z, \frac{\epsilon}{4}t) \geq \|z\|_K - \frac{\epsilon}{4}t > \omega + \frac{3\epsilon}{4}t \geq m + \frac{\epsilon}{2}t \tag{4.21}$$

and hence z will never be added to S as needed.

Runtime: First we run MV to compute an element of $\mathrm{SVP}(B_2^n, L)$ which takes $\mathrm{poly}(\langle B\rangle, \langle x\rangle)2^{O(n)}$ time. Next since $\omega \geq t$ (ω, t as above), we have that $\lambda_1(K, L) \geq t$. Now the enumeration algorithm is seeded with $d = t \leq \lambda_1(K, L)$. From here we see that the moment d is pushed above $\lambda_1(K, L)$, the set U returned by Lattice-Enum will be non-empty. Hence during the execution of the while loop, the value of d is never more that $2\lambda_1(K, L)$. Furthermore, the last execution of the enumeration algorithm is run on $d + t \leq 3\lambda_1(K, L)$. Hence every run of the enumeration algorithm happens for distances less than $3\lambda_1(K, L)$. Therefore by Lemma 4.3 and Theorem 1.1, we have that each run of the enumeration algorithm takes at most

$$\mathrm{polylog}(\tfrac{R}{r}, \tfrac{1}{t})\ \mathrm{poly}(\langle B\rangle)\, C^n\, G(3\lambda_1(K, L)K, L) \leq \mathrm{polylog}(\tfrac{R}{r}, \tfrac{1}{t})\ \mathrm{poly}(\langle B\rangle)\, C^n\, (4 \cdot 7)^n \tag{4.22}$$

Next, since $t \leq \omega \leq t\frac{R}{r}$, we see that we will execute the enumeration algorithm at most $\log_2 \frac{R}{r} + 1$ times. Remembering that $t = \frac{\|z'\|}{R}$, we have that all the lattice points of L generated by the algorithm lie inside a ball of radius at most $3\ \frac{R}{r}\ \|z'\| \leq 3\ \frac{R}{r}\ \sqrt{n}\|B\|$ around x. Hence, these lattices points as well as the number t can be represented using at most $\mathrm{poly}(\langle B\rangle, \langle x\rangle, \ln(\frac{R}{r}))$ bits. Therefore, apart from in the enumeration algorithm, we only evaluate the weak norm oracle on inputs of size $\mathrm{poly}(\langle B\rangle, \ln(\frac{R}{r}), \langle x\rangle, \ln\frac{1}{\epsilon})$ which is polynomial in the input. Finally, we filter the list U_f into S, which requires exactly $2|U_f|$ evaluations of the norm-oracle, where the cardinality of U_f is bounded by (4.22). Combining all of the above bounds, yields the desired result. □

4.4 Closest Vector Problem

Before presenting our CVP algorithm, we again need a simple enumeration bound.

Lemma 4.5. *Let $K \subseteq \mathbb{R}^n$ be a convex body, and let $L \subseteq \mathbb{R}^n$ denote an n-dimensional lattice. Then for $t > 0$ we have*

$$G(tK, L) \leq (4t + 2)^n \cdot G(K, L) \tag{4.23}$$

Proof. Since $G(tK, L)$ is invariant under shifts of K, we may assume that $b(K) = 0$. Since $b(K) = 0$, from [MP00] we know that $\mathrm{vol}(K) \leq 2^n \mathrm{vol}(K \cap -K)$ (Theorem (B.13)). We remember that $N(tK, K \cap -K)$ denotes the minimum number of translates of $K \cap -K$ needed to cover tK. Since $K \cap -K$ is symmetric, by a standard packing argument we have that

$$\begin{aligned} N(tK, K \cap -K) &\leq \frac{\mathrm{vol}(tK + \frac{1}{2}(K \cap -K))}{\mathrm{vol}(\frac{1}{2}(K \cap -K))} \leq \frac{\mathrm{vol}(tK + \frac{1}{2}K)}{\mathrm{vol}(\frac{1}{2}(K \cap -K))} \\ &= \left(\frac{t + \frac{1}{2}}{\frac{1}{2}}\right)^n \frac{\mathrm{vol}(K)}{\mathrm{vol}(K \cap -K)} \leq (2t + 1)^n 2^n = (4t + 2)^n. \end{aligned} \tag{4.24}$$

Next since $K \cap -K \subseteq K$, we have that $N(tK, K) \leq N(tK, K \cap -K)$. Now let $\Lambda \subseteq \mathbb{R}^n$ denote a set satisfying $|\Lambda| = N(tK, K)$ and $tK \subseteq \bigcup_{x\in\Lambda} x + K$. Then for $c \in \mathbb{R}^n$ we have that

$$\begin{aligned} |tK + c \cap L| &\leq |(\Lambda + c + K) \cap L| \leq \sum_{x\in\Lambda} |(x + c + K) \cap L| \\ &\leq |\Lambda| \cdot G(K, L) = N(tK, K) \cdot G(K, L) \leq (4t + 2)^n \cdot G(K, L) \end{aligned} \tag{4.25}$$

as needed. □

Algorithm 6 Closest-Vectors(K, L, x, ϵ)

Input: An $(0, r, R)$-centered convex body K with weak distance oracle D_K for $\|\cdot\|_K$, a basis B for a lattice L, an input point x, and $0 < \epsilon < 1$.

Output: $S \subseteq L$, $\mathrm{CVP}(K, L, t) \subseteq S \subseteq \mathrm{CVP}_\epsilon(K, L, t)$

if $x \in L$ **then**
  return $\{x\}$
Compute $z \in \mathrm{CVP}(B_2^n, L)$ using the MV algorithm. Set $t, d \leftarrow \frac{\|z\|}{R}$
repeat
  $U \leftarrow$ Lattice-Enum(K, L, x, d, t)
  if $U = \emptyset$ **then**
    $d \leftarrow 2d$
until $U \neq \emptyset$
$U \leftarrow$ Lattice-Enum$(K, L, x, d + t, t)$
$m \leftarrow \min\{D_K(y - x, \frac{\epsilon}{4}t) : y \in U\}$
$S \leftarrow \{y : D_K(y - x, \frac{\epsilon}{4}t) \leq m + \frac{\epsilon}{2}\, t, y \in U\}$
return S

We can now state the algorithm and main theorem of this section.

Theorem 4.6 (Correctness of Closest-Vectors)**.** *If K is well-centered, i.e.,* $\mathrm{vol}(K \cap -K) \geq 4^{-n}\,\mathrm{vol}(K)$*, then Algorithm 6 computes a set $S \subseteq L$ such that*

$$\mathrm{CVP}(K, L, x) \subseteq S \subseteq \mathrm{CVP}_\epsilon(K, L, x) \tag{4.26}$$

in expected time

$$2^{O(n)} \cdot G(dK, L) \cdot \mathrm{poly}(\log(\tfrac{1}{\epsilon}), \log(\tfrac{R}{r}), \langle B\rangle, \langle x\rangle), \tag{4.27}$$

where $d = d_K(L, x)$.

The proof is essentially identical to the one for SVP.

Proof.

Correctness: If $x \in L$, clearly there is nothing to do, so assume $x \notin L$. First note that since K is $(0, r, R)$-centered, we know that $\frac{\|y\|}{R} \leq \|y\|_K \leq \frac{\|y\|}{r}$ for all $y \in \mathbb{R}^n$. Now take $z \in \mathrm{CVP}(K, L, x)$ and $z' \in \mathrm{CVP}(B_2^n, L, x)$. Let $\omega = \|z - x\|_K$, and as in the algorithm let $t = \frac{\|z'-x\|}{R}$. Now we have that

$$t = \frac{\|z' - x\|}{R} \leq \frac{\|z - x\|}{R} \leq \|z - x\|_K \leq \|z' - x\|_K \leq \frac{\|z' - x\|}{r} = t\frac{R}{r} \tag{4.28}$$

Therefore $t \leq \omega \leq t\frac{R}{r}$. Now for $z \in \mathrm{CVP}(K, L, x)$, we must show that $z \in S$. Let d_f denote the final value of d after the while loop terminates. Since $U \neq \emptyset$ after the while loop terminates, and since the enumeration algorithm guarantees that $U \subseteq \{y \in L : \|y - x\|_K \leq d_f + t\}$, we have that $\omega \leq d_f + t$. Now let $U_f = \mathrm{Enumerate}(K, L, x, d_f + t, t)$, i.e. the final setting of the set U. By the properties Lattice-Enum, we know that $\{y \in L : \|y - x\|_K \leq d_f + t\} \subseteq U_f$, and hence we have that $\mathrm{CVP}(K, L, x) \subseteq U_f$. From the computation of the number m, during the final stage of the algorithm, we now see that $\omega - \frac{\epsilon}{4}t \leq m \leq \omega + \frac{\epsilon}{4}t$. Therefore for $z \in \mathrm{CVP}(K, L, x)$, we have that

$$D_K(z - x, \frac{\epsilon}{4}t) \leq \omega + \frac{\epsilon}{4}t \leq m + \frac{\epsilon}{2}t \tag{4.29}$$

and hence z will correctly be placed in S as needed.

Now assume that $z \in L$ and $z \notin \mathrm{CVP}(K, L, x)_\epsilon$. We must show that $z \notin S$. Since $\omega \geq t$ from above, we have that $\|z - x\|_K > (1+\epsilon)\omega \geq \omega + \epsilon t$. Therefore, we see that

$$D_K(z - x, \frac{\epsilon}{4}t) \geq \|z - x\|_K - \frac{\epsilon}{4}t > \omega + \frac{3\epsilon}{4}t \geq m + \frac{\epsilon}{2}t \tag{4.30}$$

and hence z will never be added to S as needed.

Runtime: We first check if $x \in L$, this take $\mathrm{poly}(\langle B\rangle, \langle x\rangle)$ time. Next, we run the MV algorithm to compute an element of $\mathrm{CVP}(B_2^n, L, x)$ which takes $\mathrm{poly}(\langle B\rangle, \langle x\rangle)2^{O(n)}$ time. Next, note that since $\omega \geq t$ (ω, t as above), we have that $d_K(L, x) \geq t$. Now the enumeration algorithm is seeded with $d = t \leq d_K(L, x)$. Now we note that the moment d is pushed above $d_K(L, x)$, the set U returned by the enumeration algorithm will be non-empty. Hence during the execution of the while loop, the value of d is never more that $2d_K(L, x)$. Furthermore, the last execution of the enumeration algorithm is run on $d + t \leq 3d_K(L, x)$. Hence every run of the enumeration algorithm happens for distances less than $3d_K(L, x)$. Therefore by Lemma 4.5 and Theorem 4.2, we have that each run of the enumeration algorithm takes at most

$$\begin{aligned}\mathrm{polylog}(\frac{R}{r}, \frac{1}{t})\ \mathrm{poly}(\langle B\rangle)\, C^n\, G(3d_K(L, x)K, L) \\ \leq \mathrm{polylog}(\frac{R}{r}, \frac{1}{t})\ \mathrm{poly}(\langle B\rangle)\, C^n\, 14^n G(d_K(L, x)K, L)\end{aligned} \tag{4.31}$$

Next since $t \leq \omega \leq t\frac{R}{r}$, we see that we will execute the enumeration algorithm at most $\ln_2 \frac{R}{r} + 1$ times. Now remembering that $t = \frac{\|z'-x\|}{R}$, we see that all the lattice points of L generated by the algorithm lie inside a ball of radius at most $3\frac{R}{r}\|z' - x\| \leq 3\frac{R}{r}\sqrt{n}\|B\|$ around x. Hence, these lattices points as well as the number t can be represented using at most $\mathrm{poly}(\langle B\rangle, \langle x\rangle, \ln(\frac{R}{r}))$ bits. Therefore, apart from in the enumeration algorithm, we only evaluate the weak norm oracle on inputs of size $\mathrm{poly}(\langle B\rangle, \ln(\frac{R}{r}), \langle x\rangle, \ln\frac{1}{\epsilon})$ which is polynomial in the input. Finally, we filter the list U_f into S, which requires exactly $2|U_f|$ evaluations of the norm-oracle, where the cardinality of U_f is bounded by (4.31). Combining all of the above bounds, yields the desired result. □

Though the runtime of the Closest-Vectors algorithm cannot be bounded bounded in general due to the $G(dK, L)$ term, its running time can be controlled in interesting special cases. For example, if K is well-centered and $d_K(L, x) \leq \alpha\lambda_1(K, L)$, i.e. the target point is relatively close to the lattice, then by Lemma 4.3 the main complexity term of the Closest-Vectors algorithm on K, L, x becomes

$$G(d_K(L, x)K, L) \leq \left(4\left(1 + \frac{2d_K(L, x)}{\lambda_1(K, L)}\right)\right)^n \leq (4 + 8\alpha)^n \tag{4.32}$$

which is of order $2^{O(n)}$ when $\alpha = O(1)$. With this bound, we recover (up to large C^n factors) the running time of the AKS sieve for exact CVP when the target point is close.

4.5 Integer Programming

In this section, we present an algorithm for integer programming feasibility based on a general norm SVP solver. Relying on the best known bounds for the flatness theorem (see Equation 4.8), we show that our algorithm achieves a modest improvement in complexity of IP. For a brief history, the first fixed

dimension polynomial time algorithm for integer linear programming is due to Lenstra in [Len83] which achieved an essential complexity is $2^{O(n^3)}$. This was dramatically improved by Kannan in [Kan87] reducing the complexity to $O(n^{2.5})^n$. The next improvement is due Köppe and Hildebrand [HK10] reducing the complexity to $O(n^2)^n$ while generalizing to feasible regions defined by quasi-convex polynomials. Here we present an algorithm which runs in $O(n^{\frac{4}{3}} \log^{O(1)} n)^n$, for feasible regions equipped with a strong separation oracle (see Definition B.2).

Let $f^*(n)$ denote the optimal function for the flatness theorem. Our main result here is as follows:

Theorem 4.7 (Integer Programming)**.** *Let $K \subseteq RB_2^n$ be a convex body given by a strong separation oracle SEP_K. Let $L \subseteq \mathbb{R}^n$ be a n-dimensional lattice given by a basis $B \in \mathbb{Q}^{n \times n}$. Then there exists an algorithm which either decides that $K \cap L = \emptyset$, or returns a point $x \in K \cap L$ in expected time*

$$O(f^*(n))^n \operatorname{poly}(\langle R \rangle, \langle a_0 \rangle, \langle B \rangle)$$

Unfortunately, the algorithm described above is not agnostic to the value of f^*, its exact value (or any known upper bound) is needed in the code of to guarantee the algorithm's correctness. Hence using the best known bounds on $f^*(n)$ (see [BLPS99, Rud00]), we get an algorithm of essential complexity $(c_1 n^{\frac{4}{3}} \log^{c_2} n)^n$ for absolute constants c_1, c_2.

We give an outline of the algorithm. The algorithm works as almost all previous IP algorithms do, i.e. by finding a "thinnest" width direction of K with respect to L. More precisely, we adopt a recursive solution strategy, where given K and L as above, we seek to find a small collection of parallel hyperplanes H_k, $k \in A$, such that if $K \cap L \neq \emptyset$ then for some $k \in A$ we have that $K \cap L \cap H_k \neq \emptyset$. At this point, we simply solve the integer program with respect to $K \cap H_k$, $L \cap H_k$ recursively for each $k \in A$, and decide that $K \cap L$ is empty if all the subproblems return empty and return any found lattice point otherwise. As we will explain below, finding the above set of hyperplanes reduces to solving a shortest vector problem with respect to a general norm, in particular the "width" norm of K, i.e. $\|x\|_{(K-K)^*} = \sup_{y \in K} \langle y, x \rangle - \inf_{y \in K} \langle y, x \rangle$. In previous IP algorithms, the alluded to SVP problem is solved only approximately via a reduction to ℓ_2 (i.e. via an ellipsoidal approximation of the norm). The main source of improvement for our algorithm comes from the fact the we solve the associated SVP exactly using a general norm SVP solver.

Proof of Theorem 4.7.

IP ALGORITHM:

Basis Refinement: As a first step, we will reduce to working with a lattice admitting a basis of length at most $2\sqrt{n}R$. This will allow us to control the encoding length of the basis after each recursive invocation of the IP algorithm. To begin, we use the MV algorithm for CVP to compute a closest vector $p \in L$ to a_0 in the ℓ_2 norm. If $\|p - a_0\|_2 > R$ we declare that $K \cap L = \emptyset$ (since $K \subseteq a_0 + RB_2^n$). Otherwise, we again use the MV algorithm to compute linearly independent lattice vectors $v_1, \ldots, v_n$ achieving the successive minima of L, i.e. where $\|v_i\|_2 = \lambda_i(L)$. Both invocations of the MV algorithm here take at most $2^{O(n)} \operatorname{poly}(\langle B \rangle)$ time. Letting $v_0 = 0$, compute the largest index k, $0 \leq k \leq n$, such that $\|v_k\| \leq 2R$. Now let $L' = L \cap \operatorname{span}(v_0, v_1, \ldots, v_k)$.

Claim: $L \cap a_0 + RB_2^n \subseteq p + L'$.

Proof. Take $y \in L \cap a_0 + RB_2^n$. Since $p \in a_0 + RB_2^n$, we have that $\|y - p\|_2 \leq 2R$. Assume that $y - p \notin \operatorname{span}(v_0, v_1, \ldots, v_k)$. Since $y - p \in L$ and $y - p$ is linearly independent from $v_0, v_1, \ldots, v_k$, we get that $\lambda_{k+1}(L) \leq 2R$. But by our choice of k, we know that $\lambda_{k+1}(L) > 2R$, a clear contradiction. Therefore $y - p \in L \cap \operatorname{span}(v_0, v_1, \ldots, v_k) \cap L = L'$, as needed. □

Since $K \subseteq a_0 + RB_2^n$, from the above claim we get that it suffices to check whether $K - p \cap L' = \emptyset$ to solve IP feasibility with respect to K and L. Now using standard techniques (Chris: reference needed), we may compute a basis B' for L' using $v_0, v_1, \ldots, v_k$ sastifying $\|B'\|_2 \leq \sqrt{k}\|v_k\|_2 \leq 2\sqrt{k}R$ in polynomial time. Let $W = \operatorname{span}(L')$ denote the linear span of L', a_0' denote the orthogonal projection of $a_0 - p$ onto W, and let $R' = \sqrt{R^2 - \|a_0 - a_0'\|^2}$. It is easy to check that $K - p \cap W$ is (a_0', R')-circumscribed in W. Given that may restrict our attention to points in $K - p \cap L'$, for the rest of the algorithm we replace L by L', K by $K - p \cap W$ (for which a strong separation oracle is readily available via Lemma B.7), and (a_0, R) by (a_0', R').

Localizing K***:*** For the next step, we compute a strong enough ellipsoidal approximation of K to begin inferring about how K interacts with L. To do this, we use algorithm GLS-Round (Theorem B.5), running against K with parameter $\epsilon = \left(\frac{1}{4n}\right)^n \det(L)$ to deterministically compute an ellipsoid $E + t$ such that either (1) $\operatorname{vol}(E) \leq \epsilon$ (i.e. E is tiny compared to the 'sparsity' of L), or (2) E sandwiches K well, i.e. $t + \frac{1}{\sqrt{n}(n+1)}E \subseteq K \subseteq t + E$. This step can be done in $\operatorname{poly}(n, \log \frac{R}{\epsilon}) = \operatorname{poly}(n, \langle R \rangle, \langle \det(L) \rangle)$ time.

Branching on a "thinnest" width direction of K***:*** Here we wish to find a dual vector $y \in L^*$, such that there exists a small number of hyperplanes of the form $H_k = \{x : \langle x, y \rangle = k\}$, $k \in A \subseteq \mathbb{Z}$, with the property that if K contains a point of L then there exists a lattice point in $H_k \cap K \cap L \neq \emptyset$ for some $k \in A$. At this point, as explained previously, we recurse on $K \cap H_k$, $L \cap H_k$, for all $k \in A$. To implement such a recursive call for a specific H_k, $k \in A$, we compute a basis for $L \cap H_0$ and a point $p \in H_k \cap L$, and call the IP procedure on $K - p \cap H_0$ and $L \cap H_0$. All the preprocessing here can be done in polynomial time via standard methods, where as above we note that a strong separation oracle for $K - p \cap H_0$ is readily computable via Lemma B.7.

Now to find such a y and set A, we proceed as follows. If we are in case (1) above, we use the MV algorithm to compute a vector $y \in \operatorname{SVP}(E^*, L^*)$, which can be done in $2^{O(n)} \operatorname{poly}(\langle B \rangle)$ time. Noting that $(E - E)^* = \frac{1}{2}E^*$, we see that

$$\operatorname{vol}((E - E)^*) = \left(\frac{1}{2}\right)^n \operatorname{vol}(E^*) = \left(\frac{1}{2}\right)^n \operatorname{vol}(B_2^n)^2 \frac{1}{\operatorname{vol}(E)} > \left(\frac{1}{2n}\right)^n \frac{1}{\operatorname{vol}(E)}$$

Given that $\operatorname{vol}(E)^{\frac{1}{n}} \leq \epsilon = \frac{1}{4n} \det(L)^{\frac{1}{n}}$, from the above we see that

$$\operatorname{vol}((E - E)^*)^{\frac{1}{n}} > \frac{1}{2n} \frac{1}{\operatorname{vol}(E)^{\frac{1}{n}}} \geq 2 \frac{1}{\det(L)^{\frac{1}{n}}} = 2 \det(L^*)^{\frac{1}{n}}$$

Since $(E - E)^* = \frac{1}{2}E^*$ is centrally symmetric, by Minkowski's first theorem (Theorem B.10) we have that $2\|y\|_{E^*} = \|y\|_{(E-E)^*} = \lambda_1((E - E)^*, L^*) < 1$. We remember that $\|y\|_{(E-E)^*} = \sup_{x \in E} \langle y, x \rangle - \inf_{x \in E} \langle y, x \rangle$ is the width of E with respect to y. Since $y \in L^*$ we note that for any $x \in E + t \cap L$, we must have that $\langle x, y \rangle \in (\langle y, t \rangle + [\inf_{x \in E} \langle y, x \rangle, \sup_{x \in E} \langle y, x \rangle]) \cap \mathbb{Z}$. Since E has width < 1 with respect to y, it is easy to see that if $E + t \cap L$ is non-empty then all the lattice points in $E + t \cap L$ must lie on the hyperplane $H = \{x \in \mathbb{R}^n : \langle x, y \rangle = \lfloor t \rceil\}$. Since $K \subseteq E + t$, it is also clearly the case that $K \cap L \subseteq H \cap L$. To finish

with this case, we now recursively solve the Integer Program with respect $K \cap H$ and $L \cap H$, returning empty iff $K \cap L \cap H = \emptyset$.

If we are in case (2), we know K is well-sandwiched by E, i.e. $t + \frac{1}{\sqrt{n}(n+1)}E \subseteq K \subseteq t + E$. To find a thin direction for K, we shall compute $y \in \mathrm{SVP}((K-K)^*, L^*, 1)$. To do this, we must build a weak distance oracle for $(K-K)^*$. Given that K is well sandwiched by E, using the Ellipsoid Method (theorem B.4), for any $y \in \mathbb{Q}^n$ and $\epsilon > 0$, we may compute $l, u \in \mathbb{Q}$ satisfying

$$l - \frac{\epsilon}{2} \leq \inf_{x \in K} \langle y, x \rangle \leq l \quad u \leq \sup_{x \in K} \langle y, x \rangle \leq u + \frac{\epsilon}{2}$$

in polynomial time. We note now that

$$|\|y\|_{(K-K)^*} - (u-l)| = |\sup_{x \in K} \langle y, x \rangle - \inf_{x \in K} \langle y, x \rangle - (u-l)| \leq \epsilon$$

as needed. Next, the SVP algorithm needs sandwiching guarantees on $(K-K)^*$. Given our guarantees on K, we see that $\frac{1}{2}E^* = (E-E)^* \subseteq K - K \subseteq \frac{1}{2}(n+1)\sqrt{n}E^*$. Technically, the algorithm Shortest-Vectors requires the sandwiching ratio with respect to euclidean balls, but this type of sandwiching is equivalent to ellipsoidal sandwiching after linear transformation. Having constructed a weak distance oracle for $(K-K)^*$ and computed the sandwiching guarantees, we may now call Shortest-Vectors$((K-K)^*, L^*, 1)$ (Theorem 4.4) and retrieve $y \in L^*$ from the output. Since the sandwiching guarantees are polynomial in n and the required accuracy is $O(1)$, this call be executed in expected time $2^{O(n)}\,\mathrm{poly}(\langle B \rangle, \langle E \rangle)$ time. Using the Ellipsoid Method (theorem B.4) as above, we compute bounds $u, l \in \mathbb{Q}$ satisfying $u \leq \sup_{x \in K} \langle y, x \rangle \leq u+1$ and $l-1 \leq \inf_{x \in K} \langle y, x \rangle \leq l$ in polynomial time. Now compute $A = [l-1, \min\{u+1, l+f^*(n)+1\}] \cap \mathbb{Z}$. We now show that it suffices to restrict our attention to the hyperplanes $H_k = \{x \in \mathbb{R}^n : \langle x, y \rangle = k\}$ for $k \in A$.

Claim: If $K \cap L \neq \emptyset$, then there exists $x \in K \cap L$ such that $\langle y, x \rangle \in A$.

Proof. First, if $l + f^*(n) + 1 \geq \sup_{x \in K} \langle y, x \rangle$, then by our guarantees on u and l we have that $A \supseteq [\inf_{x \in K} \langle y, x \rangle, \sup_{x \in K} \langle y, x \rangle] \cap \mathbb{Z}$. Since $\langle y, x \rangle \in \mathbb{Z}$ for any $x \in L$, we clearly have that $x \in K \cap L \Rightarrow \langle x, y \rangle \in A$. Next, if $l + f^*(n) + 1 \leq \sup_{x \in K} \langle y, x \rangle$, then we have that

$$f^*(n) \leq \sup_{x \in} \langle y, x \rangle - l - 1 \leq \|y\|_{(K-K)^*} - 1 \leq \lambda_1((K-K)^*, L^*)$$

by our assumption that $y \in \mathrm{SVP}((K-K)^*, L^*, 1)$. Take $x_0 \in \arg\min_{x \in K} \langle y, x \rangle$ and examine the convex body

$$\tilde{K} = \left(1 - \frac{f^*(n)+1}{\|y\|_{(K-K)^*}}\right) x_0 + \left(\frac{f^*(n)+1}{\|y\|_{(K-K)^*}}\right) K.$$

Since $x_0 \in K$ and $f^*(n) + 1 \leq \|y\|_{(K-K)^*}$ we get by convexity that $\tilde{K} \subseteq K$. Furthermore, we can see that

$$\tilde{K} \subseteq \{z \in \mathbb{R}^n : \inf_{x \in K} \langle y, x \rangle \leq \langle z, y \rangle \leq \inf_{x \in K} \langle y, x \rangle + f^*(n) + 1\}.$$

Therefore any $x \in \tilde{K} \cap L$ must satisfy $\langle x, y \rangle \in A$. Hence if $\tilde{K} \cap L \neq \emptyset$, since $\tilde{K} \subseteq K$ there exists $x \in K \cap L$ such that $\langle y, x \rangle \in A$. We now show that $\tilde{K} \cap L \neq \emptyset$ to complete the claim.

By homogeneity, we see that

$$\lambda_1((\tilde{K}-\tilde{K})^*, L^*) = \lambda_1\left(\left(\frac{f^*(n)+1}{\|y\|_{(K-K)^*}}(K-K)\right)^*, L^*\right) = (f^*(n)+1)\,\frac{\lambda_1((K-K)^*, L^*)}{\|y\|_{(K-K)^*}}$$
$$\geq (f^*(n)+1)\frac{\|y\|_{(K-K)^*}-1}{\|y\|_{(K-K)^*}} \geq (f^*(n)+1)\frac{f^*(n)}{f^*(n)+1} = f^*(n)$$

Applying the flatness theorem to $\tilde{K}$, we now get that $\mu(\tilde{K}, L) \leq 1$ and hence that $\tilde{K} \cap L \neq \emptyset$ as needed. □

Given the claim, we may complete the algorithm by recursively solving the integer programs with respect to $K \cap H_k$ and $L \cap H_k$, for all $k \in A$. We return EMPTY if all calls return EMPTY, and return any found lattice point otherwise.

RUNTIME: The correctness of the algorithm has already been discussed above, so it only remains to check that the runtime of the algorithm is bounded by $O(f^*(n))^n \operatorname{poly}(\langle a_0\rangle, \langle R\rangle, \langle B\rangle)$ on expectation (we note that the only source of randomness in the algorithm comes from the calls to the Shortest-Vectors algorithm). The algorithm above is recursive, where at each node of the recursion we perform the 3 named procedures above and then break the problem into at most $\lceil f^*(n)\rceil + 2$ subproblems which we solve recursively (the calls to IP on $K \cap H_k$, $L \cap H_k$ as above). Now if we can show that the processing at each recursive node takes at most expected $2^{O(n)} \operatorname{poly}(\langle a_0\rangle, \langle R\rangle, \langle B\rangle)$ time - where a_0, R, B are the *original* parameters provided to the top level call of the IP algorithm - then by solving a standard recurrence relation we get that the whole running time is indeed $O(f^*(n))^n \operatorname{poly}(\langle a_0\rangle, \langle R\rangle, \langle B\rangle)$ on expectation as needed.

Let us examine a specific recursion node with associated convex body $\bar{K}$, $(\bar{a}_0, \bar{R})$-circumscribed in $\mathbb{R}^{\bar{n}}$, and $\bar{n}$-dimensional lattice $\bar{L}$ with basis $\bar{B}$. Now it is straightforward to see that at this recursion node, the amount of computation is certainly bounded by $2^{O(\bar{n})} \operatorname{poly}(\langle \bar{a}_0\rangle, \langle \bar{R}\rangle, \langle \bar{B}\rangle)$ on expectation, since the above procedures only make calls to subroutines with either polynomial runtimes (such as the GLS-Round algorithm, the Ellipsoid Method, and standard linear algebraic procedures) or single exponential runtimes (such as the MV algorithm and the Shortest-Vectors algorithm). The main issue is therefore whether the lattice basis and affine subspace passed to the next level recursion nodes have bit size bounded by a fixed polynomial (i.e. whose degree does not depend on n) in the size of the original parameters. For clarity, we only sketch the argument here. The main reason this is true is because of the Basis Refine step. Most crucially, after the refine step, we end up with a lattice basis whose length is bounded by $2\sqrt{\bar{n}}\bar{R} \leq 2\sqrt{n}R$. Since $\bar{L}$ is a sub-lattice of our original lattice L, it is not hard to verify that any vector of $\bar{L}$ (and in fact of L) of length less than $2\sqrt{n}R$ has bit size bounded by $\operatorname{poly}(\langle R\rangle, \langle B\rangle)$ (for a fixed polynomial). Hence the Basis Refine step "smooths" any incoming basis and subspace to ones whose bit description is well bounded by the original parameters. Since the bit description of the lattice basis and subspace passed to the next child node is only a fixed polynomial larger than that of the "smoothed" basis after our refine step, the claim follows. The runtime is therefore bounded by $O(f^*(n))^n \operatorname{poly}(\langle a_0\rangle, \langle R\rangle, \langle B\rangle)$ on expectation as desired. □

5 Acknowledgments

We gratefully thank Bo'az Klartag, Gideon Schechtman, Daniele Micciancio, Oded Regev, and Panagiotis Voulgaris for fruitful discussions and critical ideas. In particular, Klartag suggested to us that the techniques of [Kla06] could be used for an algorithmic construction of an M-ellipsoid, and Schechtman suggested the use of parallelepiped tilings to construct an explicit covering.

References

[ABSS93] S. Arora, L. Babai, J. Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *J. Comput. Syst. Sci.*, 54(2):317–331, 1997. Preliminary version in FOCS 1993.

[AJ08] V. Arvind and P. S. Joglekar. Some sieving algorithms for lattice problems. In *FSTTCS*, pages 25–36. 2008.

[Ajt98] M. Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract). In *STOC*, pages 10–19. 1998.

[AKS01] M. Ajtai, R. Kumar, and D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *STOC*, pages 601–610. 2001.

[AKS02] M. Ajtai, R. Kumar, and D. Sivakumar. Sampling short lattice vectors and the closest lattice vector problem. In *IEEE Conference on Computational Complexity*, pages 53–57. 2002.

[Bal88] K. M. Ball. Logarithmically concave functions and sections of convex sets in rn. *Studia Mathematica*, 88:69–84, 1988.

[Ban93] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(4):625–635, 1993.

[Ban96] W. Banaszczyk. Inequalities for convex bodies and polar reciprocal lattices in R^n II: Application of k-convexity. *Discrete and Computational Geometry*, 16:305–311, 1996. ISSN 0179-5376.

[Bla18] W. Blaschke. Über affine geometry xiv: eine minimum aufgabe für legendres trägheits ellipsoid. *Ber. verh. sächs. Akad. d. Wiss.*, 70:72–75, 1918.

[BLPS99] W. Banaszczyk, A. Litvak, A. Pajor, and S. Szarek. The flatness theorem for nonsymmetric convex bodies via the local theory of Banach spaces. *Mathematics of Operations Research*, 24(3):728–750, 1999.

[BM87] J. Bourgain and V. D. Milman. New volume ratio properties for convex symmetric bodies in $\mathbb{R}^n$. *Inventiones Mathematicae*, 88:319–340, 1987.

[BN07] J. Blömer and S. Naewe. Sampling methods for shortest vectors, closest vectors and successive minima. In *ICALP*, pages 65–77. 2007.

[Bou86] J. Bourgain. On high-dimensional maximal functions associated to convex bodies. *Amer. J. Math*, 108(6):1467–1476, 1986.

[CN98] J.-Y. Cai and A. Nerurkar. Approximating the SVP to within a factor (1+1/dim$^\epsilon$) is NP-hard under randomized reductions. *J. Comput. Syst. Sci.*, 59(2):221–239, 1999. Preliminary version in CCC 1998.

[DFK89] M. E. Dyer, A. M. Frieze, and R. Kannan. A random polynomial-time algorithm for approximating the volume of convex bodies. *J. ACM*, 38(1):1–17, 1991. Preliminary version in STOC 1989.

[Din00] I. Dinur. Approximating SVP_∞ to within almost-polynomial factors is NP-hard. *Theor. Comput. Sci.*, 285(1):55–71, 2002. Preliminary version in CIAC 2000.

[DKRS98] I. Dinur, G. Kindler, R. Raz, and S. Safra. Approximating CVP to within almost-polynomial factors is NP-hard. *Combinatorica*, 23(2):205–243, 2003. Preliminary version in FOCS 1998.

[FB86] Z. Furedi and I. Barany. Computing the volume is difficult. In *STOC '86: Proceedings of the eighteenth annual ACM symposium on Theory of computing*, pages 442–447. ACM, New York, NY, USA, 1986. ISBN 0-89791-193-8. doi:http://doi.acm.org/10.1145/12130.12176.

[GLS88] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer, 1988.

[HK10] R. Hildebrand and M. Köppe. A faster algorithm for quasi-convex integer polynomial optimization. Arxiv, Report 1006.4661, 2010. `http://arxiv.org`.

[HR07] I. Haviv and O. Regev. Tensor-based hardness of the shortest vector problem to within almost polynomial factors. In *STOC*, pages 469–477. 2007.

[JS98] A. Joux and J. Stern. Lattice reduction: A toolbox for the cryptanalyst. *J. Cryptology*, 11(3):161–185, 1998.

[Kan87] R. Kannan. Minkowski's convex body theorem and integer programming. *Mathematics of operations research*, 12(3):415–440, August 1987. 1987.

[Kho03] S. Khot. Hardness of approximating the shortest vector problem in lattices. *J. ACM*, 52(5):789–808, 2005. Preliminary version in FOCS 2003.

[Kla06] B. Klartag. On convex perturbations with a bounded isotropic constant. *Geometric And Functional Analysis*, 16:1274–1290, 2006. ISSN 1016-443X.

[KLS95] R. Kannan, L. Lovász, and M. Simonovits. Isoperimetric problems for convex bodies and a localization lemma. *Discrete & Computational Geometry*, 13:541–559, 1995.

[Kup08] G. Kuperberg. From the mahler conjecture to gauss linking integrals. *Geometric And Functional Analysis*, 18:870–892, 2008.

[Len83] H. W. Lenstra. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4):538–548, November 1983.

[LLL82] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, December 1982.

[LV06a] L. Lovász and S. Vempala. Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization. In *FOCS '06: Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 57–68. IEEE Computer Society, Washington, DC, USA, 2006.

[LV06b] L. Lovász and S. Vempala. Hit-and-run from a corner. *SIAM J. Computing*, 35:985–1005, 2006.

[LV06c] L. Lovász and S. Vempala. Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm. *J. Comput. Syst. Sci.*, 72(2):392–417, 2006.

[Mic98] D. Micciancio. The shortest vector in a lattice is hard to approximate to within some constant. *SIAM J. Comput.*, 30(6):2008–2035, 2000. Preliminary version in FOCS 1998.

[Mil86] V. Milman. Inegalites de brunn-minkowski inverse et applications at la theorie locales des espaces normes. *C. R. Acad. Sci. Paris*, 302(1):25–28, 1986.

[MP89] V. Milman and A. Pajor. Isotropic position and inertia ellipsoids and zonoids of the unit ball of a normed n-dimensional space. *Geometric Aspects of Functional Analysis*, pages 64–104, 1989.

[MP00] V. Milman and A. Pajor. Entropy and asymptotic geometry of non-symmetric convex bodies. *Advances in Mathematics*, 152(2):314 – 335, 2000.

[MV10] D. Micciancio and P. Voulgaris. A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations. In *STOC*, pages 351–358. 2010.

[Naz09] F. Nazarov. The Hörmander proof of the Bourgain-Milman theorem, 2009. Preprint.

[NS01] P. Q. Nguyen and J. Stern. The two faces of lattices in cryptology. In *CaLC*, pages 146–180. 2001.

[Odl90] A. M. Odlyzko. The rise and fall of knapsack cryptosystems. In C. Pomerance, editor, *Cryptology and Computational Number Theory*, volume 42 of *Proceedings of Symposia in Applied Mathematics*, pages 75–88. 1990.

[RR06] O. Regev and R. Rosen. Lattice problems and norm embeddings. In *STOC*, pages 447–456. 2006.

[RS57] C. Rogers and G. Shephard. The difference body of a convex body. *Arch. Math.*, 8:220–233, 1957.

[RS58] C. Rogers and G. Shephard. Convex bodies associated with a given convex body. *J. London Soc.*, 33:270–281, 1958.

[Rud00] M. Rudelson. Distance between non-symmetric convex bodies and the MM*-estimate. *Positivity*, 4(8):161–178, 2000.

[San49] L. A. Santaló. Un invariante afin para los cuerpos convexos del espacio de n dimensiones. *Portugalie Math.*, 8:155–161, 1949.

[Sch87] C.-P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theor. Comput. Sci.*, 53:201–224, 1987.

[Son90] G. Sonnevend. Applications of analytic centers for the numerical solution of semi-infinite convex programs arising in control theory. In H. Sebastian and K. Tammer, editors, *System Modelling and Optimization*, volume 143 of *Lecture Notes in Control and Information Sciences*, pages 413–422. Springer Berlin / Heidelberg, 1990.

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

[Vou] P. Voulgaris. Personal Communication.

[YN76] D. B. Yudin and A. S. Nemirovski. Evaluation of the information complexity of mathematical programming problems (in russian). *Ekonomika i Matematicheskie Metody*, 13(2):3–45, 1976.

A M-Ellipsoid Proofs

Here we prove correctness of the all the main M-ellipsoid algorithms from Section 3. We rely heavily on several geometric estimates, which are listed and proved in Section A.1 below, and on standard algorithms from convex optimization and convex geometry, which are described in Section B.3.

Proof of Theorem 3.5 (Correctness of M-Ellipsoid). Here we give more detail as to the implementation of each of the steps of Algorithm 1:

- Step 1: Make a direct call to algorithm Estimate-Centroid (Lemma B.9) on K.
- Step 2: If Estimate-Centroid returns an estimate b of $b(K)$, we have the guarantee that

$$b + \frac{r}{2(n+1)\sqrt{n}} B_2^n \subseteq K \subseteq b + 2R \tag{A.1}$$

  Since the guarantees about b in K are polynomial in the input, we can build a weak membership oracle O_{K-b} for $K - b$, where $K - b$ is $(0, \frac{r}{2(n+1)\sqrt{n}}, 2R)$-centered, in polynomial time from O_K. Now we run the algorithm of Theorem 3.7 on the oracle O_{K-b} and retrieve the tentative M-ellipsoid $E(A)$ of K.
- Step 3: Here we make a direct call to the algorithm Build-Cover on $(K, E(A))$ where we ask whether $N(K, E(A)) > (13e)^n$.
- Step 4: First, we implement a weak membership oracle $O_{(K-K)^*}$ for $(K - K)^*$ from O_K using the ellipsoid algorithm, where we can guarantee that $(K - K)^*$ is $(0, r\frac{1}{2R}, \frac{1}{2r})$-centered. To do this, we note that

$$x \in (K - K)^* \Leftrightarrow \sup_{y \in K} \langle y, x \rangle - \inf_{y \in K} \langle y, x \rangle \leq 1$$

  Hence we can build a weak membership oracle for $(K - K)^*$ by approximately maximizing and minimizing with respect to x over K. This can readily be done via the ellipsoid algorithm (see Theorem B.4). The guarantees we get on $(K - K)^*$ are seen as follows:

$$a_0 + rB_2^n \subseteq K \subseteq RB_2^n \Rightarrow 2rB_2^n \subseteq K - K \subseteq 2RB_2^n \Rightarrow \frac{1}{2R} B_2^n \subseteq (K - K)^* \subseteq \frac{1}{2r} B_2^n$$

  Next, we note that $E(A)^* = E(A^{-1})$, and hence can be computed in polynomial time. Next, we call the algorithm Build-Cover on ($(K - K)^*$,$E(A)^*$) where we ask whether $N((K - K)^*, E(A)^*) > (25e \cdot 13)^n$.

Correctness: We must show that if the algorithm succeeds, returning the ellipsoid $E(A)$, that $E(A)$ indeed satisfies

$$N(K, E) \leq \left(\sqrt{8\pi e} \cdot 13e\right)^n \quad N(E, K) \leq \left(\sqrt{8\pi e} \cdot 25e \cdot 13 \cdot 289\right)^n \tag{A.2}$$

These guarantees depend only on the correctness of the algorithm Build-Cover. In, step 3, if the test passes, we are guaranteed to get a covering T of K by E where $|T| \leq \left(\sqrt{8\pi e} \cdot 13e\right)^n$. Hence the first requirement

is met. In step 4, if the test passes, we are guaranteed that $N((K-K)^*, E^*) \leq \left(4\sqrt{\frac{\pi e}{2}} \cdot 25e \cdot 13\right)^n$. Now by Theorem A.2, since E^* is centrally symmetric, for n large enough, we have that

$$N(E, K) \leq 289^n N((K-K)^*, E^*) \leq \left(\sqrt{8\pi e} \cdot 25e \cdot 13 \cdot 289\right)^n \tag{A.3}$$

as needed.

Runtime: We note that of each the steps $1-4$ already have a running time bounded by the desired runtime. Hence, it suffices to show that the main loop is executed on expectation only $O(1)$ times. To do this, we first condition on the event that in step 1, the returned estimate b satisfies that $b - b(K) \in \frac{1}{n+1}E_K$. This occurs with probability at least $1 - \frac{1}{n}$. Next, in step 2, given that $K - b$ satisfies the conditions of Theorem 3.7, i.e. that $b(K-b) = b(K) - b \in \frac{1}{n+1}E_K$, we may condition on the event that the returned ellipsoid $E(A)$ satisfies

$$N(K, E(A)) \leq (13e)^n \quad N(E(A), K) \leq (25e)^n. \tag{A.4}$$

Since this event occurs with probability $1 - \frac{3}{n}$, our total success probability is $1 - \frac{4}{n}$. Now in step 3, given that $N(K, E(A)) \leq (13e)^n$, the test is guaranteed to pass. Since E is centrally symmetric, for n large enough, we have that

$$N((K-K)^*, E^*) \leq (12(1+o(1)))^n N(E, K) \leq (25e \cdot 13)^n. \tag{A.5}$$

Therefore, the test in step 4 is also guaranteed to succeed. Finally, we see that the probability that each execution of the loop terminates successfully is at least $1 - \frac{4}{n}$, therefore the expected number of runs of the loop is $O(1)$ as needed. □

Proof of Theorem 3.7 (Correctness of M-Gen). The proof has two parts, first building the right oracle, then using it to sample and estimate the inertial ellipsoid.

Building a membership oracle for the polar: We first show that a polynomial time weak membership oracle for $S = n\left(\text{conv}\{K, -K\}\right)^*$ can be built from O_K. We note that

$$v \in n\left(\text{conv}\{K, -K\}\right)^* \Leftrightarrow \max\left\{\sup_{x \in K} \langle v, x\rangle, \sup_{x \in K} \langle -v, x\rangle\right\} \leq n \tag{A.6}$$

Given the guarantees on O_K, we have that

$$\frac{n}{R}B_2^n \subseteq n\left(\text{conv}\{K, -K\}\right)^* \subseteq \frac{n}{r}B_2^n \tag{A.7}$$

Constructing a weak membership oracle for S therefore requires only the ability to perform 2 different approximate optimizations over K. This can achieved using the standard optimization techniques described in Theorem B.4. Hence, a polynomial time weak membership oracle for S can be built as claimed.

Building the M-ellipsoid: Le π_S denote the uniform distribution on S. Equipped with a weak membership oracle for S, we may use the sampling algorithm of Theorem B.6, to sample a point $Y \in S$ with distribution σ satisfying $\mathrm{d_{TV}}(\sigma, \pi_S) \leq \frac{1}{n}$ in time $\text{poly}(n)\,\text{polylog}(\frac{R}{r}, n)$. Set $s = Y$, where Y is the computed sample. We shall use s to specify a reweighting of the uniform distribution on K. Let $f_s(x) = e^{\langle s, x\rangle}$ for $x \in K$ and 0 otherwise. Using the algorithm described by Corollary B.8, we may compute a matrix $A \in \mathbb{R}^{n \times n}$ satisfying

$$e^{-\frac{1}{n}}E_{f_s} \subseteq E(A) \subseteq e^{\frac{1}{n}}E_{f_s} \tag{A.8}$$

with probability $1 - \frac{1}{n}$ in time $\text{poly}(n)\,\text{polylog}(\frac{R}{r})$. We return the ellipsoid $\sqrt{n}E(A)$ as our candidate M-ellipsoid for K.

Analysis: We now show that for n large enough, the ellipsoid returned by this algorithm satisfies with high probability the covering conditions

$$N(K, \sqrt{n}E(A)) \leq (13e)^n \quad N(\sqrt{n}E(A), K)) \leq (25e)^n \tag{A.9}$$

First, we condition on the event (A.8), i.e. that we get a good estimate of E_{f_s}. Hence at this point, our success probability is at least $1 - \frac{1}{n}$.

Let $\eta > 0$ be a constant to be decided later. Let X be uniformly distributed on S, and let Y denote the approximately uniform sample the above algorithm computes on S, remembering that $S = n\left(\mathrm{conv}\{K, -K\}\right)^*$. Given the guarantee that $b(K) \in \frac{1}{n+1}E_K$, from Lemma A.4 setting $\epsilon = 1$, for n large enough we have that

$$\mathrm{E}[L_{f_X}^{2n}] \leq \left((1 + o(1))\sqrt{\frac{2}{\pi e}}\,\frac{e^\epsilon}{\sqrt{\epsilon}}\right)^{2n} \leq \left((1+\eta)\sqrt{\frac{2e}{\pi}}\right)^{2n} \tag{A.10}$$

Using Markov's inequality, we see that

$$\Pr\left[L_{f_X} > (1+\eta)^2\sqrt{\frac{2e}{\pi}}\right] \leq \frac{\mathrm{E}[L_{f_X}^{2n}]}{\left((1+\eta)^2\sqrt{\frac{2e}{\pi}}\right)^{2n}} \leq \frac{1}{(1+\eta)^{2n}}. \tag{A.11}$$

Now since $\mathrm{d_{TV}}(X, Y) \leq \frac{1}{n}$, we see that

$$\Pr\left[L_{f_Y} > (1+\eta)^2\sqrt{\frac{2e}{\pi}}\right] \leq \frac{1}{(1+\eta)^{2n}} + \frac{1}{n} \leq \frac{2}{n} \tag{A.12}$$

for n large enough (η will be chosen to be constant). Hence after additionally conditioning on the complement of event $A.12$, our success probabiblity is at least $1 - \frac{3}{n}$. At this point, letting $s = Y$, we see that s specifies a density f_s on K satisfying

$$L_{f_s} \leq (1+\eta)^2\sqrt{\frac{2e}{\pi}}. \tag{A.13}$$

Furthermore since $s \in n\left(\mathrm{conv}\{K, -K\}\right)^*$, $b(K) \in \frac{1}{n+1}E_K$ and $E_K \subseteq K$, we have that

$$\frac{\sup_{x\in K} f_s(x)}{f_s(b(K))} = \sup_{x\in K} e^{\langle s, x - b(K)\rangle} = \sup_{x\in K} e^{\langle s,x\rangle + \langle -s, b(K)\rangle} \leq e^{n+1}. \tag{A.14}$$

Hence by Lemma A.3, letting $\sqrt{n}E(A) = T$, and $\delta = e^{\frac{1}{n}}$, we get that

$$N(K, \sqrt{n}E(A)) \leq (12\delta)^n\,\frac{4}{3}\,\frac{\sup_{x\in K} f_s(x)}{f_s(b(K))} \leq 12^n e\,\frac{4}{3}\,e^{n+1} \leq (12e(1+\eta))^n \tag{A.15}$$

and

$$\begin{aligned} N(\sqrt{n}E(A), K) &\leq (12\delta^2)^n\ \mathrm{vol}(\sqrt{n}B_2^n)\,\frac{4}{3}\,L_{f_s}^n \\ &\leq 12^n e^2\,(\sqrt{2\pi e}(1+o(1)))^n\,\frac{4}{3}\,\left((1+\eta)^3\sqrt{2}\right)^n \leq (24e(1+\eta)^3)^n \end{aligned} \tag{A.16}$$

for n large enough. Choosing $\eta > 0$ such that $(1+\eta)^3 = \frac{25}{24}$ yields the result. □

Proof of Theorem 3.8 (Correctness of Build-Cover). The goal here is to either compute a covering of K by E, or conclude that $N(K, E)$ is large. To make this task easier, we will replace E by a parallelepiped P inscribed in E, and use a tiling procedure (since P can be used to tile space) to cover K. We will show any cover produced in this way is not much larger than $N(K, E)$, and hence will help provide a lower bound on $N(K, E)$. Furthermore since $P \subseteq E$, any cover of K by P immediately translates into a cover of K by E.

Building P: To compute P we will need to perform some standard matrix algebra. First we compute the Cholesky Factorization of A, i.e. we compute $V \in \mathbb{R}^{n\times n}$ such that $A = V^tV$. Next we compute $B = V^{-1}$, the inverse of V, and label the columns of B as $B = (b_1, \ldots, b_n)$. Both of the computations here can be done in time $\mathrm{poly}(\langle A\rangle)$ via standard methods. Now we note that

$$\langle b_i, b_j\rangle_A = b_i^t A b_j = (B^t A B)_{ij} = (V^{-t}V^tVV^{-1})_{ij} = (\mathrm{Id}_n)_{ij}. \tag{A.17}$$

Hence the vectors $(b_1, \ldots, b_n)$ form an orthonormal basis of with respect to the dot product $\langle\cdot,\cdot\rangle_A$. Therefore the ellipsoid $E(A)$ may be expressed as

$$E(A) = \{x \in \mathbb{R}^n : x^tAx \le 1\} = \{x \in \mathbb{R}^n : \sum_{i=1}^n \langle b_i, x\rangle_A^2 \le 1\}. \tag{A.18}$$

Now define P as

$$P = \left\{x \in \mathbb{R}^n : |\langle b_i, x\rangle_A| \le \frac{1}{\sqrt{n}}\right\} = \left\{\sum_{i=1}^n a_i b_i : |a_i| \le \frac{1}{\sqrt{n}},\ 1 \le i \le m\right\} \tag{A.19}$$

where the second equality follows since the b_is are orthonormal under $\langle\cdot,\cdot\rangle_A$. Now for $x \in \mathbb{R}^n$, we see that

$$\max_{1\le i\le n} |\langle b_i, x\rangle_A| \le \left(\sum_{i=1}^n \langle b_i, x\rangle_A^2\right)^{1/2} \le \sqrt{n}\max_{i\le i\le n} |\langle b_i, x\rangle_A| \quad \Rightarrow \quad P \subseteq E(A) \subseteq \sqrt{n}P. \tag{A.20}$$

Now a standard computation yields that

$$\mathrm{vol}(P) = \left(\frac{2}{\sqrt{n}}\right)^n \det(A)^{-1/2} \quad \mathrm{vol}(E(A)) = \left(\frac{\sqrt{2\pi e}(1+o(1))}{\sqrt{n}}\right)^n \det(A)^{-1/2} \tag{A.21}$$

where we remember here that $\det(B) = \det(V^{-1}) = \det(V)^{-1} = \det(A)^{-1/2}$. Therefore we have that $\mathrm{vol}(E(A)) \le \left(\sqrt{\frac{\pi e}{2}}(1+o(1))\right)^n \mathrm{vol}(P)$.

Tiling K with P: Define the lattice

$$L = \left\{\sum_{i=1}^n \frac{2}{\sqrt{n}} z_i b_i : z_i \in \mathbb{Z},\ 1 \le i \le m\right\}, \tag{A.22}$$

so L is the lattice spanned by the vectors $\frac{2}{\sqrt{n}}(b_1, \ldots, b_n)$. From here it is straightforward to verify that P tiles space with respect to L, so $L + P = \mathbb{R}^n$ and for $x, y \in L$, $x \ne y$, $x + \mathrm{int}(P) \cap y + \mathrm{int}(P) = \emptyset$, i.e. the interiors are disjoint. In fact, one can see that P is simply a shift of the fundamental parallelepiped of L with respect to the basis $\frac{2}{\sqrt{n}}(b_1, \ldots, b_n)$.

We now wish to tile K with copies of P. To do this we examine the set $H = \{x \in L : x + P \cap K \ne \emptyset\}$. Since $P + L = \mathbb{R}^n$, it is easy to see that

$$K \subseteq H + P. \tag{A.23}$$

Hence, we shall want to decide for $x \in L$, whether $x + P \cap K \ne \emptyset$. Since we only have a weak membership oracle for K, we will only be able to decide whether $x + P$ approximately intersects K. To formalize this, we build an weak intersection oracle INT which queried on $x \in \mathbb{R}^n$, $\epsilon > 0$ satisfies

$$\mathrm{INT}(x, \epsilon) = \begin{cases} 0: & x + P \cap K = \emptyset \\ 1: & x + (1+\epsilon)P \cap K \ne \emptyset \end{cases}. \tag{A.24}$$

Using this oracle we will be able to overestimate T, and compute a set $S \subseteq L$ such that

$$H \subseteq S \subseteq \{x : x + (1+\epsilon)P \cap K \neq \emptyset\} \tag{A.25}$$

which will suffice for our purposes. Now to build INT, we first remark that for $x \in \mathbb{R}^n$, $t \geq 0$

$$x + tP \cap K \neq \emptyset \Leftrightarrow \inf_{y \in K} \|y - x\|_P \leq t \Leftrightarrow \inf_{y \in K} \sqrt{n} \max_{1 \leq i \leq n} |\langle b_i, y - x \rangle_A| \leq t. \tag{A.26}$$

Hence deciding the minimum scaling t of P for which $x + tP \cap K \neq \emptyset$ is equivalent to solving a simple convex program. The above convex program is exactly in the form described in Theorem B.4, hence for $\epsilon > 0$, and $x \in \mathbb{Q}^n$, we may compute a number $\omega \geq 0$ such that

$$|\omega - \inf_{y \in K} \|y - x\|_K| \leq \epsilon \tag{A.27}$$

in time $\text{poly}(n, \langle x \rangle, \langle A \rangle) \, \text{polylog}(\frac{R}{r}, \frac{1}{\epsilon})$. We now build INT. On query $x \in \mathbb{Q}^n$, $\epsilon > 0$, we do the following:

1. Compute $\omega \geq 0$ satisfying $|\omega - \inf_{y \in K} \|y - x\|_K| \leq \frac{\epsilon}{2}$.
2. If $\omega \leq 1 + \frac{\epsilon}{2}$ return 1, otherwise return 0.

From $(A.27)$ the above procedure clearly runs in polytime. To prove correctness, we must show that $\text{INT}(x, \epsilon) = 1$ if $x + P \cap K \neq \emptyset$ and $\text{INT}(x, \epsilon) = 0$ if $x + (1+\epsilon)P \cap K = \emptyset$. If $x + P \cap K \neq \emptyset$, we note that $\inf_{y \in K} \|y - x\|_K \leq 1$, hence by the guarantee on ω we have that

$$\omega \leq \inf_{y \in K} \|y - x\|_K + \frac{\epsilon}{2} \leq 1 + \frac{\epsilon}{2}, \tag{A.28}$$

and so we correctly classify x. If $x + (1+\epsilon)P \cap K = \emptyset$, then $\inf_{y \in K} \|y - x\|_K > 1 + \epsilon$ and so

$$\omega \geq \inf_{y \in K} \|y - x\|_K - \frac{\epsilon}{2} > 1 + \frac{\epsilon}{2} \tag{A.29}$$

as needed.

We now compute a tiling of K. The idea here is simple. We define a graph G on the lattice L, where for $x, y \in L$, $x \sim y$ iff $x - y \in \frac{2}{\sqrt{n}}\{\pm b_1, \ldots, \pm b_n\}$. We identify each lattice point $x \in L$ with the tile $x + P$. Starting from the tile centered at 0, we begin a breadth first search on G of the tiles intersecting K. In this way, we will compute the connected component containing 0 in G of tiles intersecting K. Lastly, if the number of intersecting K tiles exceeds $\left(4\sqrt{\frac{\pi e}{2}}H\right)^n$, we abort and return that $N(K, E) \geq H^n$. The algorithm is given in Algorithm 7.

Correctness: To argue correctness of the above algorithm, we must guarantee that the algorithm either computes a valid covering of K or that it proves that $N(K, E) > H^n$. For $\epsilon \geq 0$, let

$$H_\epsilon = \{x \in L : x + (1+\epsilon)P \cap K \neq \emptyset\} \quad \text{and} \quad H'_\epsilon = \{x \in L : \text{INT}(x, \epsilon) = 1\} \tag{A.30}$$

From the description above, we see that the algorithm performs a breath first search on G starting from 0 of the tiles in $H'_{\frac{1}{n}}$. From the properties of the weak intersection oracle INT, we know that $H_0 \subseteq H'_{\frac{1}{n}} \subseteq H_{\frac{1}{n}}$.

The goal of the algorithm is to discover a super-set of H_0. Since $H_0 \subseteq H'_{\frac{1}{n}}$, the algorithm will correctly add elements of H_0 to the cover T if it finds them. Since we perform a breadth first search from 0, to

Algorithm 7 Computing a tiling.

$M \leftarrow \{0\}, N \leftarrow \{0\}, T \leftarrow \emptyset$.
while $N \neq \emptyset$ **do**
  choose $x \in N$
  $N \leftarrow N \setminus \{c\}$
  if $\mathrm{INT}(x, \frac{1}{n}) = 1$ **then**
    $T \leftarrow T \cup \{x\}$
    if $|T| > \left(4\sqrt{\frac{\pi e}{2}}H\right)^n$ **then**
      return FAIL
    for all $\delta \in \frac{2}{\sqrt{n}}\{\pm b_1, \ldots, \pm b_n\}$ **do**
      if $x + \delta \notin M$ **then**
        $N \leftarrow N \cup \{x\}$, $M \leftarrow M \cup \{x\}$
return T

guarantee we find all of H_0 we need only ensure that H_0 forms a connected subgraph of G. As noted before, the set of tiles indexed by H_0 are just lattice shifts of the fundamental parallelepiped of L with respect to the basis $\frac{2}{\sqrt{n}}(b_1, \ldots, b_n)$. In this setting, the connectivity of H_0 with respect the edges defined by the basis (i.e. the set of tiles touching any convex set), is a classical fact. Therefore, the algorithm will indeed discover all of H_0, provided that the partial cover T remains no larger than $\left(4\sqrt{\frac{\pi e}{2}}H\right)^n$.

Now we must justify that if the algorithm aborts, i.e. if $|T| > \left(4\sqrt{\frac{\pi e}{2}}H\right)^n$, that indeed $N(K, E) > H^n$. Now at every timestep we have that $T \subseteq H'_{\frac{1}{n}} \subseteq H_{\frac{1}{n}}$. Therefore, to show correctness, it suffices to show that $|H_{\frac{1}{n}}| \leq \left(4\sqrt{\frac{\pi e}{2}}\right)^n N(K, E)$. Now for $x \in H_{\frac{1}{n}}$, we have that

$$x + (1 + \frac{1}{n})P \cap K \neq \emptyset \Rightarrow x \in K + (1 + \frac{1}{n})P \Rightarrow x + P \in K + (2 + \frac{1}{n})P \tag{A.31}$$

Furthermore, since for $x, y \in H_{\frac{1}{n}}$, $x \neq y$, $x + \mathrm{int}(P) \cap y + \mathrm{int}(P) = \emptyset$, we have that

$$\mathrm{vol}(K + (2 + \frac{1}{n})P) \geq \mathrm{vol}(\cup_{x \in H_{\frac{1}{n}}} x + P) = |H_{\frac{1}{n}}| \,\mathrm{vol}(P) \tag{A.32}$$

Using that $P \subseteq E$, and $\mathrm{vol}(E) \leq \left(\sqrt{\frac{\pi e}{2}}(1 + o(1))\right)^n \mathrm{vol}(P)$ we get

$$\begin{aligned} |H_{\frac{1}{n}}| &\leq \frac{\mathrm{vol}(K + (2 + \frac{1}{n})P)}{\mathrm{vol}(P)} \leq \left(\sqrt{\frac{\pi e}{2}}(1 + o(1))\right)^n \frac{\mathrm{vol}(K + (2 + \frac{1}{n})E)}{\mathrm{vol}(E)} \\ &\leq \left(\sqrt{\frac{\pi e}{2}}(1 + o(1))(3 + \frac{1}{n})\right)^n N(K, E) \leq \left(4\sqrt{\frac{\pi e}{2}}\right)^n N(K, E) \end{aligned} \tag{A.33}$$

for n large enough. Hence the algorithm correctly decides whether $N(K, E) > H^n$.

Runtime: The running time of the algorithm is proportional to the number of tiles visited and the number of edges crossed during the search phase. Since all the tiles visited in the algorithm are adjacent to the tiles in the set T, and the number of edges is $2n$, the total number of tiles visited is at most $2n|T| \leq 2n\left(4\sqrt{\frac{\pi e}{2}}H\right)^n$. Furthermore, the edges traversed correspond to all the outgoing edges from T, and hence is bounded by

the same number. Now at every visited tile, we make a call to $\mathrm{INT}(x, \frac{1}{n})$ for some $x \in L$, which takes $\mathrm{poly}(n, \langle A \rangle) \, \mathrm{polylog}(\frac{R}{r})$ time. Hence the total running time is

$$\mathrm{poly}(n, \langle A \rangle) \, \mathrm{polylog}(\frac{R}{r}) \left(4\sqrt{\frac{\pi e}{2}} H \right)^n \tag{A.34}$$

as needed. □

A.1 Geometric Estimates

Here we list and prove the necessary geometric inequalities that we used in the proofs above. We begin with a slight extension of Theorem B.13.

Theorem A.1. *Let K be a convex body such that $b(K) \in tE_K$, for some $t \in [0, 1)$. Then*

$$\mathrm{vol}(K \cap -K) \geq \left(\frac{1-t}{2} \right)^n \mathrm{vol}(K) \tag{A.35}$$

Proof. From Theorem B.13 we have that

$$\frac{1}{2^n} \mathrm{vol}(K) \leq \mathrm{vol}(K - b(K) \cap -K + b(K)) = \mathrm{vol}(K \cap -K + 2b(K)) \tag{A.36}$$

Next, we note that for $x \in \mathbb{R}^n$

$$K \cap -K + 2x \neq \emptyset \Leftrightarrow 2x \in K + K \Leftrightarrow x \in K \tag{A.37}$$

Since $b(K) \in tE_K$ and $b(K) + E_K \subseteq K$, we see that $(1-t)E_K \subseteq K$. Hence we can write

$$0 = t(-2nb(K)) + (1-t)2b(K), \tag{A.38}$$

where $-nb(K) \in -(1-t)E_K = (1-t)E_K \subseteq K$. Now we see that

$$t\left(K \cap (-K + -2nb(K))\right) + (1-t)\left(K \cap (-K + 2b(K))\right) \subseteq K \cap -K \tag{A.39}$$

where both sets on the left hand side are non-empty by (A.37). Therefore by the Brunn-Minkowski inequality, we have that

$$\begin{aligned} \mathrm{vol}(K \cap -K)^{\frac{1}{n}} &\geq t \, \mathrm{vol}\left(K \cap (-K + -n2b(K))\right)^{\frac{1}{n}} + (1-t) \, \mathrm{vol}\left(K \cap (-K + 2b(K))\right)^{\frac{1}{n}} \\ &\geq (1-t) \, \mathrm{vol}\left(K \cap (-K + 2b(K))\right)^{\frac{1}{n}} \geq \frac{1-t}{2} \mathrm{vol}(K)^{\frac{1}{n}} \end{aligned} \tag{A.40}$$

Therefore we get that

$$\mathrm{vol}(K \cap -K) \geq \left(\frac{1-t}{2} \right)^n \mathrm{vol}(K)$$

as needed. □

The next lemma is a slight specialization of [MP00, Theorem 5]. We require this inequality for the M-ellipsoid certification procedure.

Theorem A.2 (Duality of Entropy)**.** *Let $K, T \subseteq \mathbb{R}^n$ be convex bodies where T is centrally symmetric. Then*

$$N(T, K) \leq ((1 + o(1))288)^n \cdot N((K - K)^*, T^*) \tag{A.41}$$

and

$$N((K - K)^*, T^*) \leq (12(1 + o(1)))^n \cdot N(T, K). \tag{A.42}$$

Proof. Since the above quantities are invariant under shifts of K, we may shift K so that $b(K) = 0$. Applying Theorem B.13, we see that that $\mathrm{vol}(K - K) \leq 4^n \,\mathrm{vol}(K) \leq 8^n \,\mathrm{vol}(K \cap -K)$, where we note that since $0 \in K$ we have that $K \cap -K \subseteq K \subseteq K - K$. Next applying the covering estimates from Lemma B.14, we get that

$$N(K - K, K) \leq N(K - K, K \cap -K) \leq 3^n \frac{\mathrm{vol}(K - K)}{\mathrm{vol}(K \cap -K)} \leq 24^n.$$

From here, we see that

$$N(T, K) \leq N(T, K - K)N(K - K, K) \leq 24^n N(T, K - K). \tag{A.43}$$

Next since both T and $K - K$ are centrally symmetric, we apply Lemma B.14 to get that

$$N(T, (K - K)) \leq 3^n \frac{\mathrm{vol}(T)}{\mathrm{vol}((K - K) \cap T)}.$$

Now we note that $((K - K) \cap T)^* = \mathrm{conv}\{(K - K)^*, T^*\}$. Hence applying the Blashke-Santaló inequality to $\mathrm{vol}(T)$ and the Bourgain-Milman inequality to $\mathrm{vol}((K - K) \cap T)$ we get that

$$3^n \frac{\mathrm{vol}(T)}{\mathrm{vol}((K - K) \cap T)} \leq (6(1 + o(1)))^n \frac{\mathrm{vol}(\mathrm{conv}\{(K - K)^*, T^*\})}{\mathrm{vol}(T^*)}$$

Since 0 is both in $(K - K)^*$ and T^*, we see that $\mathrm{conv}\{(K - K)^*, T^*)\} \subseteq (K - K)^* + T^*$ and hence

$$(6(1 + o(1)))^n \frac{\mathrm{vol}(\mathrm{conv}\{(K - K)^*, T^*\})}{\mathrm{vol}(T^*)} \leq (6(1 + o(1)))^n \frac{\mathrm{vol}((K - K)^* + T^*)}{\mathrm{vol}(T^*)}.$$

Lastly, applying Lemma B.14 to the last estimate, we get that

$$(6(1 + o(1)))^n \frac{\mathrm{vol}((K - K)^* + T^*)}{\mathrm{vol}(T^*)} \leq (12(1 + o(1)))^n N((K - K)^*, T^*).$$

Combining the above estimates yields the first desired inequality.

Now switching the roles $(K - K)$ and T with $(K - K)^*$ and T^*, we have that

$$N((K - K)^*, T^*) \leq (12(1 + o(1))^n N(T, K - K) \leq (12(1 + o(1))^n N(T, K),$$

yielding the second inequality. □

We now make precise the relationship between the isotropic constant of the exponential reweightings defined by Klartag [Kla06] and the M-ellipsoid.

Lemma A.3. *Let $K \subseteq \mathbb{R}^n$ be a convex body. Take $s \in \mathbb{R}^n$ and let $f_s(x) = e^{\langle s,x\rangle}$ for $x \in K$ and 0 otherwise. Let $T \subseteq \mathbb{R}^n$ be a convex body such that for some $\delta \geq 1$ we have that*

$$\frac{\sqrt{n}}{\delta} E_{f_s} \subseteq T \subseteq \delta\sqrt{n} E_{f_s} \tag{A.44}$$

where E_{f_s} is the inertial ellipsoid of f_s. Then we have that

$$N(K,T) \leq (12\delta)^n \, \frac{4}{3} \, \frac{\sup_{x\in K} f_s(x)}{f_s(b(K))} \quad \text{and} \quad N(T,K) \leq (12\delta^2)^n \, \operatorname{vol}(\sqrt{n}B_2^n) \, \frac{4}{3} \, L_{f_s}^n \tag{A.45}$$

where $b(K)$ is the centroid of K, and L_{f_s} is the isotropic constant of f_s.

Proof. Since the above estimates are all invariant under shifts of K, we may assume that $b(f_s) = 0$ (centroid of f_s). We note that $b(f_s) \in K$ always and hence $0 \in K$. Let X be distributed as π_{f_s}, where π_{f_s} is the probability measure induced by f_s. So we have that $\mathrm{E}[X] = b(f_s) = 0$ and $\mathrm{E}[XX^t] = \operatorname{cov}(f_s)$.

Remember that $E_{f_s} = \{x : x^t \operatorname{cov}(f_s)^{-1} x \leq 1\}$, therefore $\|x\|_{E_{f_s}} = \sqrt{x^t \operatorname{cov}(f_s)^{-1} x}$. Now note that

$$\begin{aligned} \mathrm{E}[\|X\|_{E_{f_s}}^2] &= \mathrm{E}[X^t \operatorname{cov}(f_s)^{-1} X] = \mathrm{E}[\operatorname{trace}[\operatorname{cov}(f_s)^{-1} XX^t]] = \operatorname{trace}[\operatorname{cov}(f_s)^{-1} \mathrm{E}[XX^t]] && \text{(A.46)} \\ &= \operatorname{trace}[\operatorname{cov}(f_s)^{-1} \operatorname{cov}(f_s)] = \operatorname{trace}[\mathrm{Id}_n] = n. && \text{(A.47)} \end{aligned}$$

Now by Markov's inequality, we have that

$$\pi_{f_s}(2\sqrt{n}E_{f_s}) = 1 - \Pr[\|X\|_{E_{f_s}} > 2\sqrt{n}] \geq 1 - \frac{\mathrm{E}[\|X\|_{E_{f_s}}^2]}{4n} = 1 - \frac{n}{4n} = \frac{3}{4}. \tag{A.48}$$

By Jensen's inequality, we see that

$$\int_K f_s(x)dx = \int_K e^{\langle s,x\rangle} dx = \operatorname{vol}(K) \int_K e^{\langle s,x\rangle} \frac{dx}{\operatorname{vol}(K)} \geq \operatorname{vol}(K) e^{\langle s, b(K)\rangle} = \operatorname{vol}(K) f_s(b(K)), \tag{A.49}$$

where $b(K)$ is the centroid of K.

Using (A.49) and (A.48) we see that

$$\operatorname{vol}(2\sqrt{n}E_{f_s} \cap K) \geq \frac{\int_{2\sqrt{n}E_{f_s}} f_s(x)dx}{\sup_{x\in K} f_s(x)} \geq \frac{3}{4} \, \frac{\int_K f_s(x)dx}{\sup_{x\in K} f_s(x)} \geq \frac{3}{4} \, \frac{f_s(b(K))}{\sup_{x\in K} f(x)} \operatorname{vol}(K). \tag{A.50}$$

Using that $\frac{\sqrt{n}}{\delta} E_{f_s} \subseteq T$, $0 \in K$, $\delta \geq 1$, and by (A.50) we get that

$$\begin{aligned} \operatorname{vol}(T \cap K) &\geq \operatorname{vol}\left(\frac{\sqrt{n}}{\delta} E_{f_s} \cap K\right) = \left(\frac{1}{\delta}\right)^n \operatorname{vol}(\sqrt{n}E_{f_s} \cap \delta K) \geq \left(\frac{1}{\delta}\right)^n \operatorname{vol}\left(\sqrt{n}E_{f_s} \cap \frac{1}{2}K\right) \\ &= \left(\frac{1}{2\delta}\right)^n \operatorname{vol}(2\sqrt{n}E_{f_s} \cap K) \geq \left(\frac{1}{2\delta}\right)^n \, \frac{3}{4} \, \frac{f_s(b(K))}{\sup_{x\in K} f(x)} \, \operatorname{vol}(K). \end{aligned} \tag{A.51}$$

Using the definition of L_{f_s}, (A.48), $\sqrt{n}E_{f_s} \subseteq \delta T$ and that $0 \in K$, we get that

$$\begin{aligned} \det(\operatorname{cov}(f_s))^{\frac{1}{2}} &= L_K^n \, \frac{\int_K f_s(x)dx}{\sup_{x\in K} f_s(x)} \leq L_K^n \, \frac{4}{3} \, \frac{\int_{2\sqrt{n}E_{f_s}} f_s(x)dx}{\sup_{x\in K} f_s(x)} \leq L_K^n \, \frac{4}{3} \, \operatorname{vol}(2\sqrt{n}E_{f_s} \cap K) \\ &\leq L_K^n \, \frac{4}{3} \, \operatorname{vol}(2\delta T \cap K) \leq (2\delta L_K)^n \, \frac{4}{3} \, \operatorname{vol}(T \cap K). \end{aligned} \tag{A.52}$$

Using that $T \subseteq \delta\sqrt{n}E_{f_s}$ and the ellipsoid volume formula (2.6), we have that

$$\mathrm{vol}(T) \leq \mathrm{vol}(\delta\sqrt{n}E_{f_s}) = \delta^n \,\mathrm{vol}(\sqrt{n}B_2^n)\det(\mathrm{cov}(f_s))^{\frac{1}{2}}. \tag{A.53}$$

Combining equations (A.52),(A.53) we get that

$$\mathrm{vol}(T) \leq (2\delta^2 L_K)^n \;\mathrm{vol}(\sqrt{n}B_2^n)\,\frac{4}{3}\;\mathrm{vol}(T\cap K). \tag{A.54}$$

Now applying Lemma B.14 to the inequalities (A.51),(A.54) the theorem follows. □

From Lemma A.3, we see that if the slicing conjecture is true, then for any convex body, its inertial ellipsoid appropriately scaled is an M-ellipsoid. To bypass this, Klartag shows that for any convex body K, there exists a "mild" exponential reweighting f_s of the uniform density on K with bounded isotropic constant. As one can see from Lemma A.3, the severity of the reweighting controls $N(K, \sqrt{n}E_{f_s})$ whereas the isotropic constant of f_s controls $N(\sqrt{n}E_{f_s}, K)$.

The main tool to establish the existence of "good" exponential reweightings for K is the following lemma, which one can extract from the proof of Theorem 3.6 in [Kla06]. We will use it here for $\epsilon = 1$, in which case the expectation below is of order $2^{O(n)}$. The argument is essentially identical to that of [Kla06]; we include it for completeness.

Lemma A.4 ([Kla06])**.** *Let $K \subseteq \mathbb{R}^n$ be a convex body such that $b(K) \in \frac{1}{n+1}E_K$. For $s \in \mathbb{R}^n$, let $f_s : K \to \mathbb{R}^+$ denote the function $f_s(x) = e^{\langle s,x\rangle}$, $x \in K$. Let X be distributed as $\epsilon n\,(\mathrm{conv}\{K,-K\})^*$ for some real $\epsilon > 0$. Then we have*

$$\mathrm{E}[L_{f_X}^{2n}] \leq \left((1+o(1))\,\sqrt{\frac{2}{\pi e}}\,\frac{e^\epsilon}{\sqrt{\epsilon}}\right)^{2n} \tag{A.55}$$

Proof. For $s \in \mathbb{R}^n$ define $f_s : K \to \mathbb{R}_+$ by $f_s(x) = e^{\langle s,x\rangle}$ for $x \in K$. In Lemma 3.2 of [Kla06] is it shown that

$$\int_{\mathbb{R}^n} \det(\mathrm{cov}(f_s))ds = \mathrm{vol}(K) \tag{A.56}$$

By Theorem B.11, we have that $E_K + b(K) \subseteq K$. Since $b(K) \in \frac{1}{n+1}E_K$ by assumption, we see that $\frac{n}{n+1}E_K \subseteq E_K + b(K) \subseteq K$. Hence $0 \in K$. From [RS58], we know that for any convex body K such that $0 \in K$, we have that $\mathrm{vol}(\mathrm{conv}\{K,-K\}) \leq 2^n \,\mathrm{vol}(K)$.

Let $L = \mathrm{conv}\{K,-K\}$. Note that

$$L^* = (\mathrm{conv}\{K,-K\})^* = \{y : |\,\langle x,y\rangle\,| \leq 1,\ \forall x \in K\} \tag{A.57}$$

Since L is centrally symmetric by the Bourgain-Milman inequality (Theorem B.12), we have that

$$\mathrm{vol}(L^*)\,\mathrm{vol}(L) \geq \left((1+o(1))\frac{\pi e}{n}\right)^n \tag{A.58}$$

Hence we get that

$$\mathrm{vol}(L^*) \geq \left(\frac{(1+o(1))\pi e}{n\,\mathrm{vol}(L)^{\frac{1}{n}}}\right)^n \geq \left(\frac{(1+o(1))\pi e}{2n\,\mathrm{vol}(K)^{\frac{1}{n}}}\right)^n \tag{A.59}$$

Take $s \in \epsilon n L^*$. We examine the properties of $f_s : K \to \mathbb{R}_+$. Since $s \in \epsilon n L^*$, we see that

$$\sup_{x \in K} f_s(x) = e^{\sup_{x \in K} \langle s, x \rangle} \leq e^{\epsilon n} \tag{A.60}$$

Since $b(K) \subseteq \frac{1}{n+1} E_K \subseteq \frac{1}{n} K$ and $s \in \epsilon n \left(\operatorname{conv}\{K, -K\}\right)^*$, we see that $| \langle s, b(K) \rangle | \leq \epsilon$. Now by Jensen's inequality, we have that

$$\begin{aligned} \int_K e^{\langle s, x \rangle} dx &= \operatorname{vol}(K) \left(\int_K e^{\langle s, x \rangle} \frac{dx}{\operatorname{vol}(K)} \right) \geq \operatorname{vol}(K) e^{\int_K \langle s, x \rangle \frac{dx}{\operatorname{vol}(K)}} \\ &= \operatorname{vol}(K) e^{\langle s, b(K) \rangle} \geq \operatorname{vol}(K) e^{-\epsilon} \end{aligned} \tag{A.61}$$

Now we see that

$$L_{f_s}^{2n} = \left(\sup_{x \in K} \frac{f_s(x)}{\int_K f_s(x) dx} \right)^2 \det(\operatorname{cov}(f_s)) \leq \left(\frac{e^{\epsilon n}}{\operatorname{vol}(K) e^{-\epsilon}} \right)^2 \det(\operatorname{cov}(f_s)) = \frac{e^{2(n+1)\epsilon}}{\operatorname{vol}(K)^2} \det(\operatorname{cov}(f_s)) \tag{A.62}$$

Applying inequality (A.62), Lemma 3.2 of [Kla06], and equation (A.59), we get that

$$\begin{aligned} \frac{1}{\operatorname{vol}(\epsilon n L^*)} \int_{\epsilon n L^*} L_{f_s}^{2n} ds &\leq \frac{e^{2(n+1)\epsilon}}{\operatorname{vol}(\epsilon n L^*) \operatorname{vol}(K)^2} \int_{\epsilon n L^*} \operatorname{vol}(K)^2 \det(\operatorname{cov}(f_s)) ds \\ &\leq \frac{e^{2(n+1)\epsilon}}{\operatorname{vol}(\epsilon n L^*) \operatorname{vol}(K)^2} \operatorname{vol}(K) \leq \left(\frac{(1 + o(1)) e^{2\epsilon}}{\epsilon n \operatorname{vol}(L^*)^{\frac{1}{n}} \operatorname{vol}(K)^{\frac{1}{n}}} \right)^n \\ &\leq \left(\frac{(1 + o(1)) 2 e^{2\epsilon}}{\pi e \epsilon} \right)^n = \left((1 + o(1)) \sqrt{\frac{2}{\pi e}} \frac{e^{\epsilon}}{\sqrt{\epsilon}} \right)^{2n} \end{aligned} \tag{A.63}$$

The above quantity is exactly $\mathrm{E}[L_{f_X}]$ since X is uniform over $\epsilon n L^*$. The statement thus follows. □

B Additional Background

For two probability distributions σ_1, σ_2 over a domain $\mathcal{X}$, their *total variation* (or *statistical*) *distance* is

$$\mathrm{d_{TV}}(\sigma_1, \sigma_2) = \sup_{A \subseteq \mathcal{X}} |\sigma_1(A) - \sigma_2(A)|. \tag{B.1}$$

B.1 Logconcave functions

We will need to work with the generalization of convex bodies to logconcave functions. A function $f : \mathbb{R}^n \to \mathbb{R}_+$ is logconcave if for all $x, y \in \mathbb{R}^n$, and $0 \leq \alpha \leq 1$, we have that

$$f(\alpha x + (1 - \alpha) y) \geq f(x)^{\alpha} f(y)^{1 - \alpha} \tag{B.2}$$

The canonical examples of logconcave functions are the indicator functions of convex bodies as well as the Gaussian distributions. We will now generalize the concepts defined before for convex bodies to logconcave functions.

For a logconcave function f on $\mathbb{R}^n$ such that $0 < \int_{\mathbb{R}^n} f(x)\,dx < \infty$, we define the associated probability measure (distribution) π_f, where for measurable $A \subseteq \mathbb{R}^n$, we have

$$\pi_f(A) = \frac{\int_A f(x)\,dx}{\int_{\mathbb{R}^n} f(x)\,dx}. \tag{B.3}$$

We define the *centroid* (or barycenter) and *covariance* matrix of f as

$$b(f) = \frac{\int_{\mathbb{R}^n} xf(x)dx}{\int_{\mathbb{R}^n} f(x)\,dx} \qquad \mathrm{cov}(f)_{ij} = \frac{\int_{\mathbb{R}^n}(x_i - b(f)_i)(x_j - b(f)_j)f(x)\,dx}{\int_{\mathbb{R}^n} f(x)\,dx} \;\; 1 \le i,j \le n$$

The matrix $\mathrm{cov}(f)$ is positive semi-definite and symmetric. We say that f is isotropic, or in isotropic position, if $b(f) = 0$ and $\mathrm{cov}(f)$ is the identity matrix. Define the *inertial ellipsoid* of f as

$$E_f = E(\mathrm{cov}(f)^{-1}) = \{x : x^t\,\mathrm{cov}(f)^{-1}x \le 1\}$$

The *isotropic constant* of f is defined as

$$L_f = \left(\sup_{x\in\mathbb{R}^n} \frac{f(x)}{\int_{\mathbb{R}^n} f(x)dx}\right)^{\frac{1}{n}} \cdot \det(\mathrm{cov}(f))^{\frac{1}{2n}}.$$

A natural extension of the slicing conjecture (Conjecture 2.1) is that L_f is bounded by a universal constant. This generalized slicing conjecture was shown by Ball [Bal88] to be equivalent to the slicing conjecture for convex bodies, up to a constant factor in the precise bound.

For a convex body K, let π_K denote the uniform measure (distribution) over K. Let f_K denote the associated density, i.e.,

$$f_K(x) = \frac{1}{\mathrm{vol}(K)} I[x \in K],$$

We note that the definitions coincide exactly if we replace K by f_K, i.e., $\mathrm{cov}(K) = \mathrm{cov}(f_K)$, $b(K) = b(f_K)$, $L_K = L_{f_K}$, etc. We extend all the notions defined above for log-concave functions to convex bodies in the same way, e.g. we let $E_K = E_{f_K}$. We say that K is in isotropic position if $b(K) = 0$ and $\mathrm{cov}(K)$ is the identity (a different normalization is sometimes used in asymptotic convex geometry, namely, $b(K) = 0$, $\mathrm{vol}(K) = 1$, and $\mathrm{cov}(K)$ is constant diagonal).

B.2 Computational model

For a rational matrix A, we define $\langle A\rangle$ as the length of the binary encoding of A. The lattice algorithms presented will have complexity depending on the dimension n of the lattice and the bit length of the description of the input basis.

Since we work with general (semi-)norms, we shall need an appropriate way to represent them. We now define the three different types of oracles that we will need. For convenience, our semi-norms will always be indexed by a convex body K. With some slight modifications, we will adopt the terminology from [GLS88].

Let $K \subseteq \mathbb{R}^n$ be a convex body. For $\epsilon \ge 0$, we define

$$K^\epsilon = K + \epsilon B_2^n \quad \text{and} \quad K^{-\epsilon} = \{x \in K : x + \epsilon B_2^n \subseteq K\} \tag{B.4}$$

We say that K is (a_0, R)-*circumscribed* if $K \subseteq a_0 + RB_2^n$ for some $a_0 \in \mathbb{Q}^n$ and $R \in \mathbb{Q}$. We say that K is (a_0, r, R)-*centered* if $a_0 + rB_2^n \subseteq K \subseteq a_0 + RB_2^n$ for $a_0 \in \mathbb{Q}^n$, $r, R \in \mathbb{Q}$. We will always assume that the above parameters are given explicitly as part of the input to our problems, and hence our algorithms will be allowed to depend polynomially in $\langle a_0\rangle, \langle r\rangle, \langle R\rangle$.

Definition B.1. A *weak membership oracle* O_K for K is function which takes as input a point $x \in \mathbb{Q}^n$ and real $\epsilon > 0$, and returns

$$O_K(x, \epsilon) = \begin{cases} 1 & : x \in K^{\epsilon} \\ 0 & : x \notin K^{-\epsilon} \end{cases} \tag{B.5}$$

where any answer is acceptable if $x \in K^{\epsilon} \setminus K^{-\epsilon}$.

Definition B.2. A *strong separation oracle* SEP_K for K on input $y \in \mathbb{Q}^n$ either returns YES if $y \in K$, or some $c \in \mathbb{Q}^n$ such that $\langle c, x\rangle < \langle c, y\rangle$, $\forall x \in K$.

When working with the above oracle, we assume that there is a polynomial ϕ, such that on input y as above, the output of SEP_K has size bounded by $\phi(\langle y\rangle)$. The runtimes of algorithms using SEP_K will therefore depend on ϕ.

Let K be a convex body containing the origin.

Definition B.3. A *weak distance oracle* D_K for K is a function that takes as input a point $x \in \mathbb{Q}^n$ and $\epsilon > 0$, and returns a rational number satisfying

$$|D_K(x, \epsilon) - \|x\|_K| \leq \epsilon. \tag{B.6}$$

As above, we assume the existence of a polynomial ϕ, such that the size of the output of D_K on (x, ϵ) is bounded by $\phi(\langle x\rangle, \langle \epsilon\rangle)$. For a $(0, r, R)$-centered body K, $\forall x \in \mathbb{R}^n$, we crucially have that

$$\frac{1}{R}\|x\| \leq \|x\|_K \leq \frac{1}{r}\|x\|.$$

B.3 Standard Algorithms

Here we list some of the algorithmic tools we will require.

The following theorem is essentially the classical equivalence between weak membership and weak optimization [YN76, GLS88].

Theorem B.4 (Convex Optimization via Ellipsoid Method)**.** *Let $K \subseteq \mathbb{R}^n$ an (a_0, r, R)-centered convex body given by a weak membership oracle O_K. Let $A \in \mathbb{Q}^{m\times n}$, $c \in \mathbb{Q}^m$. Define $f : \mathbb{R}^n \to \mathbb{R}$ as*

$$f(x) = \max_{1\leq i\leq m} \langle A_i, x\rangle + c_i \tag{B.7}$$

where A_i is the i^{th} row of A. Then for $\epsilon > 0$, a number $\omega \in \mathbb{Q}$ satisfying

$$|\omega - \inf_{x\in K} f(x)| \leq \epsilon \tag{B.8}$$

can be computed using O_K in time

$$\mathrm{poly}(n, \langle A\rangle, \langle a_0\rangle, \langle c\rangle)\,\mathrm{polylog}(\frac{R}{r}, \frac{1}{\epsilon}) \tag{B.9}$$

We will also need an algorithm from [GLS88], which allows one to deterministically compute an ellipsoid with relatively good "sandwiching" guarantees for a convex body K. We present a small modification of the result in GLS:

Theorem B.5 (Algorithm GLS-Round)**.** *Let $K \subseteq R^n$ be an (a_0, R)-circumscribed convex body given by a strong-separation oracle SEP_K. Then for any $\epsilon > 0$, in $\mathrm{poly}(\log \frac{R}{\epsilon}, \langle a_0 \rangle n)$ time one can compute $A \succ 0$, $A \in \mathbb{Q}^{n \times n}$ and $t \in \mathbb{R}^n$, such that the ellipsoid $E = E(A)$ satisfies $K \subseteq E + t$, and one of the following: (a) $\mathrm{vol}(E) \leq \epsilon$, or (b) $\frac{1}{(n+1)n^{\frac{1}{2}}} E + t \subseteq K$.*

The next theorem comes from the literature on random walks on convex bodies [LV06b, LV06a, LV06c].

Theorem B.6 (Algorithm Logconcave-Sampler, [LV06a])**.** *Let $K \subseteq \mathbb{R}^n$ be a (a_0, r, R)-centered convex body given by a weak membership oracle O_K. Let $f : K \to \mathbb{R}_+$ be a polynomial time computable log-concave function satisfying*

$$\sup_{x \in K} f(x) \leq \beta^n f(0) \tag{B.10}$$

for some $\beta > 1$. Let $\epsilon, \tau > 0$. Then the following can be computed:

1. *A random point $X \in K$ with distribution σ satisfying $\mathrm{d_{TV}}(\sigma, \pi_{f_s}) \leq \tau$ in time*

$$\mathrm{poly}(n, \langle a_0 \rangle) \, \mathrm{polylog}(n, \frac{R}{r}, \beta, \frac{1}{\tau}) \tag{B.11}$$

2. *A point $b \in K$ and a matrix $A \in \mathbb{Q}^{n \times n}$ such that $\forall \, x \in \mathbb{R}^n$*

$$|\, \langle x, b - b(f_s) \rangle \,| \leq \epsilon \, x^t \, \mathrm{cov}(f_s) x \quad \textit{and} \quad |x^t (A - \mathrm{cov}(f_s)) x| \leq \epsilon \, x^t \, \mathrm{cov}(f_s) x, \tag{B.12}$$

with probability $1 - \delta$ in time

$$\mathrm{poly}(n, \langle a_0 \rangle, \frac{1}{\epsilon}) \, \mathrm{polylog}(n, \frac{R}{r}, \beta, \frac{1}{\delta}). \tag{B.13}$$

The following simple lemma allows us to construct a strong separation oracle for any hyperplane section of a convex body already equipped with a strong separation oracle.

Lemma B.7. *Let $K \subseteq \mathbb{R}^n$ be a convex body presented by a strong separation oracle SEP_K. Let $H = \{x \in \mathbb{R}^n : Ax = b\}$ denote an affine subspace, where $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$. Then one can construct a separation oracle for $K \cap H$, such that on input $y \in H$, the oracle executes in time $\mathrm{poly}(\langle y \rangle, \langle A \rangle, \langle b \rangle)$ using a single call to SEP_K.*

Proof. We wish to construct a strong separation oracle for $K \cap H$, where $H = \{x \in \mathbb{R}^n : Ax = b\}$ is an affine subspace, given a strong separation oracle for K. To do this given $y \in H$, we do the following. First, we call SEP_K on y. If SEP_K returns that $y \in K$, we return YES. If SEP_K returns a separator $c \in \mathbb{R}^n$ such that $\sup_{x \in K} \langle c, x \rangle < \langle c, y \rangle$, we compute $\bar{c}$ the orthogonal projection of c onto $W = \{x \in R^n : Ax = 0\}$ (the lineality space of H). If $\bar{c} = 0$, we note that $\langle \bar{c}, \cdot \rangle$ is constant over H. Therefore if $K \cap H \neq \emptyset$, there exists $x \in K \cap H \subseteq K$ such that $\langle c, x \rangle = \langle c, y \rangle$, a contradiction. Hence if $\bar{c} = 0$, we return that $K \cap H$ is EMPTY. Otherwise, we simply return $\bar{c}$. Since the derived oracle simply calls SEP_K once and projects any found separator onto the lineality space of H, the runtime is clearly $\mathrm{poly}(\langle A \rangle, \langle b \rangle, \langle y \rangle)$ as needed. □

We now derive some straightforward applications of the above fundamental tools.

Corollary B.8 (Algorithm Estimate-Covariance)**.** *Let $K \subseteq \mathbb{R}^n$ be an (a_0, r, R)-centered convex body given by a weak membership oracle O_K. Let $f : K \to \mathbb{R}_+$ be a polynomial time computable log-concave function satisfying*

$$\sup_{x \in K} f(x) \leq e^{2n} f(0). \tag{B.14}$$

Then an ellipsoid $E(A)$, $A \in \mathbb{Q}^{n \times n}$, can be computed satisfying

$$e^{-\frac{1}{n}} E_{f_s} \subseteq E(A) \subseteq e^{\frac{1}{n}} E_{f_s} \tag{B.15}$$

with probability $1 - \delta$ in time $\mathrm{poly}(n, \langle a_0 \rangle, \log(\frac{R}{r}), \log(\frac{1}{\delta}))$.

Proof. Using Theorem B.6, we can compute a matrix $B \subseteq \mathbb{Q}^{n \times n}$ satisfying

$$|x^t (B - \mathrm{cov}(f_s))x| \leq \frac{1}{n}\, x^t \,\mathrm{cov}(f_s) x \quad \forall\, x \in \mathbb{R}^n, \tag{B.16}$$

with probabiliy $1-\delta$ in time $\mathrm{poly}(n)\,\mathrm{polylog}(n, \frac{R}{r}, \frac{1}{\delta})$. We now condition on the event (B.16). Remembering that $x^t B x = \|x\|_B^2$ and $x^t \,\mathrm{cov}(f_s) x = \|x\|^2_{\mathrm{cov}(f_s)}$, we may rewrite (B.16) as

$$\sqrt{\frac{n-1}{n}} \|x\|_{\mathrm{cov}(f_s)} \leq \|x\|_B \leq \sqrt{\frac{n+1}{n}} \|x\|_{\mathrm{cov}(f_s)} \tag{B.17}$$

From the above, we see that the ellipsoid $E(\mathrm{cov}(f_s)) = \{x : \|x\|_{\mathrm{cov}(f_s)} \leq 1\}$ and $E(B) = \{x : \|x\|_B \leq 1\}$ satisfy

$$\sqrt{\frac{n}{n+1}} E(\mathrm{cov}(f_s)) \subseteq E(B) \subseteq \sqrt{\frac{n}{n-1}} E(\mathrm{cov}(f_s)) \tag{B.18}$$

Remembering that the polar ellipsoids satisfy

$$E(B)^* = E(B^{-1}) \quad \text{and} \quad E(\mathrm{cov}(f_s))^{-1} = E(\mathrm{cov}(f_s)^{-1}) = E_{f_s}. \tag{B.19}$$

where the last equality follows by the definition of E_{f_s}. Taking the polars of the above ellipsoids, the containment relationships in (B.18) flip, and we get

$$\sqrt{\frac{n-1}{n}} E_{f_s} \subseteq E(B^{-1}) \subseteq \sqrt{\frac{n+1}{n}} E_{f_s} \tag{B.20}$$

Now using the inequalities $1 - \frac{1}{n} \geq e^{-\frac{2}{n}}$ for $n \geq 3$ and $1 + \frac{1}{n} \leq e^{\frac{2}{n}}$, we see that (B.20) implies

$$e^{-\frac{1}{n}} E_{f_s} \subseteq E(B^{-1}) \subseteq e^{\frac{1}{n}} E_{f_s} \tag{B.21}$$

as needed. Letting $A = B^{-1}$, the ellipsoid $E(A)$ satisfies the desired requirements. □

Corollary B.9 (Algorithm Estimate-Centroid)**.** *There is a probabilistic algorithm Estimate-Centroid that, given a $(0, r, R)$-centered convex body K presented by a weak membership oracle O_K and some $\delta > 0$, in time* $\mathrm{poly}(n)\,\mathrm{polylog}(n, \frac{R}{r}, \frac{1}{\delta})$ *either outputs FAIL (with probability at most δ) or some $b \in K$ such that:*

$$b + \frac{r}{2(n+1)\sqrt{n}} B_2^n \subseteq K \subseteq b + 2R B_2^n \tag{B.22}$$

and with probability at least $1 - \delta$,

$$b - b(K) \in \frac{1}{n+1} E_K. \tag{B.23}$$

Proof. Using Theorem B.6, we compute a center $b \in K$ satisfying

$$|\langle x, b - b(K)\rangle| \leq \frac{1}{(n+1)^2}\, x^t \operatorname{cov}(K) x \quad \forall\, x \in \mathbb{R}^n, \tag{B.24}$$

with probability $1-\delta$ in time $\operatorname{poly}(n)\operatorname{polylog}(n, \frac{R}{r}, \frac{1}{\delta})$.

First, check whether

$$O_K\left(b \pm \frac{3r}{4(n+1)} e_i, \frac{r}{4(n+1)\sqrt{n}}\right) = 1 \quad \text{for } 1 \leq i \leq n \tag{B.25}$$

If any of the above tests fail, abort and return FAIL.

Let $\delta = \frac{r}{n+1}$. If these tests pass, by the properties of O_K we know that

$$b + \frac{3\delta}{4}\operatorname{conv}\{\pm e_1, \ldots, \pm e_n\} \subseteq K^{\frac{\delta}{4\sqrt{n}}} \Rightarrow b + \frac{3\delta}{4\sqrt{n}} B_2^n \subseteq K^{\frac{\delta}{4\sqrt{n}}} \Rightarrow b + \frac{\delta}{2\sqrt{n}} B_2^n \subseteq K \tag{B.26}$$

Since $b \in K \subseteq RB_2^n$, we clearly also have that $K \subseteq b + 2RB_2^n$. Hence conditioned up outputting b, we have that

$$b + \frac{r}{2(n+1)\sqrt{n}} B_2^n \subseteq K \subseteq b + 2RB_2^n \tag{B.27}$$

as needed.

We now show that if the event (B.24) holds, then the above test will pass and condition (b) will also be satisfied. Since this event holds with probability $1-\delta$, this will suffice to prove the statement.

For the center b, we note that for all $x \in (n+1)E(\operatorname{cov}(f_s))$, by equation (B.24) we have that

$$|\langle b - b(K), x\rangle| \leq \frac{1}{(n+1)^2} x^t \operatorname{cov}(K) x \leq \frac{1}{(n+1)^2}(n+1)^2 = 1 \tag{B.28}$$

Therefore, we have that $b - b(K) \in ((n+1)E(\operatorname{cov}(K)))^* = \frac{1}{n+1} E_K$ as needed.

We now show that the tests must all pass. From Theorem B.11, we know that

$$b(K) + \sqrt{\frac{n+2}{n}} E_K \subseteq K \subseteq b(K) + \sqrt{n(n+2)} E_K \tag{B.29}$$

By the guarantee on O_K, we know that $rB_2^n \subseteq b(K) + \sqrt{n(n+2)} E_K$. But we have that

$$\begin{aligned} rB_2^n - b(K) \subseteq \sqrt{n(n+2)} E_K &\Rightarrow rB_2^n + b(K) \subseteq \sqrt{n(n+2)} E_K \\ &\Rightarrow \frac{1}{2}(rB_2^n - b(K)) + \frac{1}{2}(rB_2^n + b(K)) \subseteq \sqrt{n(n+2)} E_K \\ &\Rightarrow rB_2^n \subseteq \sqrt{n(n+2)} E_K \end{aligned} \tag{B.30}$$

since both E_K and B_2^n are symmetric. From the inequality $n+1 \geq \sqrt{n(n+2)}$, we have that

$$\frac{r}{n+1} B_2^n \subseteq \frac{\sqrt{n(n+2)}}{n+1} E_K \subseteq E_K \tag{B.31}$$

Since $b - b(K) \in \frac{1}{n+1} E_K$ by assumption, and $\sqrt{\frac{n+2}{n}} E_K + b(K) \subseteq K$, we get that

$$b \in b(K) + \frac{1}{n+1} E_K \Rightarrow b + E_K \subseteq b(K) + \frac{n+2}{n+1} E_K \Rightarrow b + E_K \subseteq b(K) + \sqrt{\frac{n+2}{n}} E_K \subseteq K \tag{B.32}$$

Therefore by B.31) we have that $b + \frac{r}{n+1} B_2^n \subseteq K$. Letting $\delta = \frac{r}{n+1}$, from the previous sentence we see that

$$b \pm \frac{3}{4}\delta e_i \in K^{-\frac{\delta}{4}} \subseteq K^{-\frac{\delta}{4\sqrt{n}}} \tag{B.33}$$

Therefore by the properties of O_K, the tests in B.25 must all pass. The claim thus holds. □

B.4 Geometric Inequalities

Perhaps the most fundamental inequality in the geometry of numbers is Minkowski's first theorem, which is stated as follows:

Theorem B.10. *Let $L \subseteq \mathbb{R}^n$ be an n dimensional lattice and let $K \subseteq \mathbb{R}^n$ denote a centrally symmetric convex body. Then*

$$\lambda_1(K, L) \leq 2 \left(\frac{\det(L)}{\mathrm{vol}(K)} \right)^{\frac{1}{n}}$$

The following gives bounds on how well the inertial ellipsoid approximates a convex body. The estimates below are from [KLS95]:

Theorem B.11. *For a convex body $K \subseteq \mathbb{R}^n$, the inertial ellipsoid E_K satisfies*

$$\sqrt{\frac{n+2}{n}} \cdot E_K \subseteq K - b(K) \subseteq \sqrt{n(n+2)} \cdot E_K \tag{B.34}$$

where equality holds for any simplex.

The above containment relationship was shown in [MP89] for centrally symmetric bodies (with better bounds), and by [Son90] for general bodies with suboptimal constants.

The next theorem gives estimates on the volume product, a fundamental quantity in Asymptotic Convex Geometry. The upper bound for centrally symmetric bodies follows from the work of Blashke [Bla18], and for general bodies by Santaló [San49]. The lower bound was first established by Bourgain and Milman [BM87], and was recently refined by Kuperberg [Kup08], as well as by Nazarov [Naz09], where Kuperberg achieves the best constants. Finding the exact minimizer of the volume product is a major open problem in Asymptotic Convex Geometry.

Theorem B.12. *Let K be a convex body in $\mathbb{R}^n$. Then we have*

$$\mathrm{vol}(B_2^n)^2 \geq \inf_{x \in K} \mathrm{vol}(K - x)\, \mathrm{vol}((K - x)^*) \geq \left(\frac{\pi e(1 + o(1))}{2n} \right)^n. \tag{B.35}$$

If K is centrally symmetric, then

$$\mathrm{vol}(B_2^n)^2 \geq \mathrm{vol}(K)\, \mathrm{vol}(K^*) \geq \left(\frac{\pi e(1 + o(1))}{n} \right)^n. \tag{B.36}$$

In both cases, the upper bounds are equalities if and only if K is an ellipsoid.

We remark that the upper and lower bounds match within a 4^n factor (2^n for symmetric bodies) since $\mathrm{vol}(B_2^n)^2 = \left(\frac{2\pi e(1+o(1))}{n}\right)^n$. Using the M-ellipsoid, one can directly derive weak bounds (i.e., with sub-optimal constants) on the volume product. Furthermore, as we shall see in Section A , the techniques developed by Klartag [Kla06] can be used to derive the existence of the M-ellipsoid as an essential consequence of the volume product bounds.

The next theorem gives useful volume estimates for some basic operations on a convex body. The first estimate is due to Rogers and Shepard [RS57], and the second is due Milman and Pajor [MP00]:

Theorem B.13. *Let $K \subseteq \mathbb{R}^n$ be a convex body. Then*

$$\mathrm{vol}(K-K) \leq \binom{2n}{n} \mathrm{vol}(K) \leq 4^n \mathrm{vol}(K).$$

If $b(K) = 0$, i.e., the centroid of K is at the origin, then

$$\mathrm{vol}(K) \leq 2^n \mathrm{vol}(K \cap -K).$$

Lastly, we relate some well-known covering estimates. Here $N(K,T) = \min\{|\Lambda| : \Lambda \subseteq \mathbb{R}^n, K \subseteq \Lambda + T\}$, where K, T are convex bodies in $\mathbb{R}^n$.

Lemma B.14. *Let $K, T \subseteq \mathbb{R}^n$ be convex bodies. Then*

$$N(K,T) \leq 6^n \inf_{c \in \mathbb{R}^n} \frac{\mathrm{vol}(K)}{\mathrm{vol}(K \cap (T+c))} \quad \textit{and} \quad \frac{\mathrm{vol}(K+T)}{\mathrm{vol}(T)} \leq 2^n N(K,T). \tag{B.37}$$

If T is centrally symmetric, then

$$N(K,T) \leq \frac{\mathrm{vol}(K+T/2)}{\mathrm{vol}(T/2)}. \tag{B.38}$$

If both K and T are centrally symmetric, then

$$N(K,T) \leq 3^n \frac{\mathrm{vol}(K)}{\mathrm{vol}(K \cap T)}. \tag{B.39}$$

Proof. Let us first examine the case where T is centrally symmetric, where we wish to show that

$$N(K,T) \leq \frac{\mathrm{vol}(K+T/2)}{\mathrm{vol}(T/2)} \tag{B.40}$$

Let $\Lambda \subseteq K$ be a maximal subset of K such that for $x_1, x_2 \in \Lambda$, $x_1 \neq x_2$, $x_1 + T/2 \cap x_2 + T/2 = \emptyset$.

Claim 1: $K \subseteq \cup_{x \in \Lambda}\, x + T$.

Take $y \in K$. By maximality of Λ, there exists $x \in \Lambda$ such that

$$y + T/2 \,\cap\, x + T/2 \neq \emptyset \quad \Rightarrow \quad y \in x + T/2 - T/2 \quad \Rightarrow \quad y \in x + T$$

where the last equality follows since T is centrally symmetric. The claim thus follows.

Claim 2: $|\Lambda| \leq \dfrac{\operatorname{vol}(K + T/2)}{\operatorname{vol}(T/2)}$.

For $x \in \Lambda$, note that since $x \in K$, we have that $x + T/2 \subseteq K + T/2$. Therefore $\Lambda + T/2 \subseteq K$. Since the sets $x + T/2$, $x \in \Lambda$, are disjoint, we have that

$$\operatorname{vol}(K + T/2) \geq \operatorname{vol}(\Lambda + T/2) = |\Lambda| \operatorname{vol}(T/2) \tag{B.41}$$

as needed.

Now let us assume that K is also symmetric. Since both K and T are symmetric, we have that $K \cap T$ is also symmetric. Therefore by the estimate in (B.40) we get that

$$N(K, T) \leq N(K, T \cap K) \leq \frac{\operatorname{vol}(K + \frac{1}{2}(T \cap K))}{\operatorname{vol}(\frac{1}{2}(T \cap K))} \leq \frac{\operatorname{vol}(\frac{3}{2}K)}{\operatorname{vol}(\frac{1}{2}(T \cap K))} = 3^n \frac{\operatorname{vol}(K)}{\operatorname{vol}(T \cap K)} \tag{B.42}$$

as needed.

Now we examine the case where neither K nor T is necessarily symmetric. Since the covering estimate is shift invariant, we may assume that K and T have been shifted such that $\operatorname{vol}(K \cap T)$ is maximized, and that the centroid of $K \cap T$ is at 0. Let $S = (K \cap T) \cap -(K \cap T)$. By Theorem B.13 we have that $\operatorname{vol}(S) \geq 2^{-n} \operatorname{vol}(K \cap T)$. Note that S is a centrally symmetric convex body. Hence by identical reasoning as in (B.42) we get that

$$N(K, T) \leq 3^n \frac{\operatorname{vol}(K)}{\operatorname{vol}(S)} \leq 6^n \frac{\operatorname{vol}(K)}{\operatorname{vol}(K \cap T)}$$

as needed.

Lastly, pick any $\Lambda \subseteq \mathbb{R}^n$ such that $K \subseteq \Lambda + T$ and $|\Lambda| = N(K, T)$. Now we see that

$$\operatorname{vol}(K + T) \leq \operatorname{vol}((\Lambda + T) + T) = \operatorname{vol}(\Lambda + 2T) \leq |\Lambda| \operatorname{vol}(2T) = 2^n \operatorname{vol}(T) N(K, T)$$

as needed. □