EFFICIENT HIGH-DIMENSIONAL SAMPLING AND INTEGRATION

A Dissertation
Presented to
The Academic Faculty

By

Ben Cousins

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in
Algorithms, Combinatorics, and Optimization

Georgia Institute of Technology

April 2017

EFFICIENT HIGH-DIMENSIONAL SAMPLING AND INTEGRATION

Approved by:

Dr. Santosh Vempala, Advisor
School of Computer Science
Georgia Institute of Technology

Dr. A. B. Dieker
Department of Industrial Engineering and Operations Research
Columbia University

Dr. Dana Randall
School of Computer Science
Georgia Institute of Technology

Dr. Prasad Tetali
School of Mathematics
Georgia Institute of Technology

Dr. Eric Vigoda
School of Computer Science
Georgia Institute of Technology

Date Approved: March 23, 2017

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Volume computation is an algorithmic version of the fundamental geometric problem to figure out how much space an object occupies. Related problems of sampling and integration have numerous applications to other fields, thus it is key to develop efficient algorithmic solutions to these problems. This thesis pushes the computational frontier of volume computation, randomized sampling, and integration, both in theory and pracice.

The search for efficient algorithms for volume computation has been an active area of research over the last few decades. Many geometric problems suffer from computational inefficiency in high-dimensions: the so-called *curse of dimensionality*, where the problem efficiency grows exponentially with the dimension. For volume computation, Dyer, Frieze, and Kannan [1] gave a polynomial time randomized algorithm to approximate the volume of any convex body. While their algorithm complexity was prohibitively high, the fundamental ideas inspired further improvements. Building upon the tools and ideas developed in this line of work, we obtain an $O^*(n^3)$ algorithm to approximate the volume of well-rounded convex bodies.

As a crucial tool to our faster volume algorithm, we obtain a faster algorithm for sampling a Gaussian distribution restricted by a convex set. We also generalize the Gaussian sampling restricted by a logconcave function. The Gaussian sampling algorithm additionally yields a faster algorithm for generating a uniform random sample from a convex body.

The ideas for the $O^*(n^3)$ volume algorithm also lead to an efficient algorithm in practice. While the theoretical developments were inspiring, there was still no satisfying real-world implementation of the algorithm. We implement a variant of our algorithm in MATLAB, which can estimate volume in hundreds of dimensions with relative ease. The implementation allows for volume estimations which were previously far out of the realm of computational feasibility. In collaboration with systems biologists, we additionally explore a direct application of the sampling and volume implementations to the analysis of metabolic networks.

CHAPTER 1

INTRODUCTION

High-dimensional convex geometry is rich with beautiful questions concerning the behavior of convex bodies, many of which have important real-world applications. High-dimensional convex bodies are pervasive throughout many fields, most commonly arising as a system of linear inequalities. The ability to integrate over, sample from, and compute the volume of a convex body are fundamental algorithmic tools for such sets. In addition to yielding practical algorithms, the progress towards algorithms for volume computation, random sampling, and integration has shaped our understanding of convex geometry and developed deep connections between high-dimensional geometry and the efficiency of algorithms.

For the problem of volume computation, the goal is to develop an efficient algorithm that estimates the volume of a high-dimensional region within a target relative error. To emphasize the dependence on dimension, algorithms for volume computation are commonly measured in the number of membership queries for the set. Initially, strong negative results were shown for even approximating the volume, where it was proven that any deterministic algorithm must take an exponential number of queries to obtain an exponential approximation to the volume [2, 3]. In spite of these negative results, the first volume algorithm of Dyer, Frieze, and Kannan [1] proved the existence of a polynomial time algorithm to estimate the volume of a convex body. While polynomial time, the algorithm had a complexity of $O^*(n^{23})$ membership oracle queries for an n-dimensional convex body. At the heart of their algorithm was a discrete random walk for generating nearly uniform random samples from a convex body, where the body is approximated by a discretized set of grid points.

Subsequently, a search was spawned searching for faster volume algorithms. To this end, a number of important theoretical tools were developed, such as the proving mixing time bounds for

continuous space random walks as opposed to using a discrete approximation. The Localization Lemma [4, 5] provided a framework to prove properties of convex bodies by arguing about the one-dimensional case. The Localization Lemma is useful for proving isoperimetric inequalities, which in this context are fundamental geometric inequalities that ask for the smallest surface that divides a convex body into two pieces (weighted by the smaller of the two pieces). The efficiency of algorithms for volume computation and isoperimetric inequalities appear to be fundamentally linked, exemplified by the KLS conjecture [5]. The KLS conjecture, informally, states that the most efficient way to cut a convex body, up to constants, is via a hyperplane. It has since been shown that the KLS conjecture is "equivalent" to decades old conjectures in convex geometry [6]. A positive resolution to the KLS conjecture would imply faster sampling algorithms.

Our main contribution is an $O^*(n^3)$ algorithm to estimate the volume of a well-rounded convex body in $\mathbb{R}^n$, improving upon the previous best by a factor of n. Much of the algorithmic progress in this thesis utilizes important properties of high-dimensional Gaussian distributions. It was previously believed that the key technical hurdle to obtaining an $O^*(n^3)$ volume algorithm was a positive resolution of the KLS conjecture. Instead, we utilize a partial resolution of KLS conjecture for Gaussian distributions and combine our insights with the algorithmic framework and tools of [7, 8]. A key component of our algorithm is an accelerated cooling schedule using Gaussians, where we achieve a "perfect" tradeoff with the complexity of Gaussian sampling and the rate at which we approach the uniform distribution, i.e. the volume.

A central ingredient to all efficient volume algorithms is the ability to generate random samples from a high-dimensional distribution. Generally, random sampling is useful for gaining global statistical information about a complex structure. In volume computation and integration, random samples are generated to estimate ratios of integrals, which ultimately yields an estimate for the volume or target integral. In this work, we focus on continuous space random walks, which yield polynomial time algorithms only assuming convexity of the solution space or logconcavity of the target distribution. This robustness allows the tools discussed here to be applied to numerous fields,

including Bayesian inference, operations research, statistical physics, and systems biology. In particular, for systems biology, randomized sampling from high-dimensional models of metabolic networks is a useful tool to capture global information about the set of feasible states of the network, whereas optimization methods provide a comparably more limited view.

We obtain faster algorithms for sampling a Gaussian distribution restricted to a convex set. The previous best sampling algorithm of [9] provided sampling algorithms of $O^*(n^3)$ for any logconcave distribution. We improve the complexity by a factor of n to $O^*(n^2)$ for the important special case of Gaussians restricted to a convex set. Additionally, we obtain a $O^*(n^2)$ sampling algorithm for a Gaussian restricted by a well-rounded logconcave distribution.

Volume computation can be viewed as a special case of integration, i.e. the integration of an indicator function over a convex region. The current fastest complexity for integration of a logconcave function matched the previous best for volume computation, $O^*(n^4)$ [10]. We improve upon the complexity for Gaussian integration over a convex set, obtaining a complexity of $O^*(n^3)$. Gaussian integration is arguably as important as volume computation, especially in statistical applications.

From a practically grounded perspective, there was hope that the theoretical developments for volume computation would eventually lead to an efficient implementation of the algorithm. Using similar ideas from our $O^*(n^3)$ volume algorithm, we develop a MATLAB implementation of this algorithm that performs very favorably to existing implementations. Exact deterministic algorithms for volume computation are severely limited by dimension, with existing implementations limited to around 20 dimensions. By using randomness and allowing for an approximation, we can compute volume in hundreds of dimensions, even for certain convex bodies where exactly computing the volume is $\#P$-hard.

There is a direct, useful application of our sampling algorithms to systems biology. Under a certain model of metabolic networks, the set of feasible flows in the network form a high-dimensional convex body. Random sampling provides an unbiased view of the solution space and has been

shown to provide important biological insight over more traditional optimization methods. The implementation of our MATLAB algorithm allows for efficient sampling in thousands of dimensions, whereas previous approaches in this area were limited to a few hundred dimensions.

CHAPTER 2

BACKGROUND AND PRELIMINARIES

Here we discuss various technical aspects to be used in the analysis of algorithms for volume computation, integration, and sampling. All of these algorithms rely on rapidly mixing Markov chains. We mention some background in this area alongside with a brief survey of currently known and analyzed geometric random walks. The majority of the other tools deal with the behavior of high-dimensional convex bodies and logconcave distributions. We mention some important properties in high-dimensional convex geometry which are regarded as folklore. Additionally, we discuss newer tools and inequalities which were developed alongside efficient algorithms for volume computation, such as the localization lemma, isoperimetric inequalities of convex sets, and the KLS conjecture.

2.1 Computational Model

When considering high-dimensional algorithms for problems such as sampling, integration, and volume computation, their algorithmic complexity is commonly measured in the number of oracle calls. The oracle we consider is a *function oracle*. Suppose we have an oracle for a function $f : \mathbb{R}^n \to \mathbb{R}$. A single oracle query consists of querying a point $x \in \mathbb{R}^n$, and the oracle returns $f(x)$, the function evaluated at x. If $f : \mathbb{R}^n \to \{0, 1\}$ is the indicator function of a convex set or convex body, an oracle for f is referred to as a *membership oracle*.

Such an oracle is usually accompanied by a point in the set $x \in \mathbb{R}^n$, and sandwiching parameters $r, R \in \mathbb{R}$. These parameters satisfy $x + rB_n \subseteq K \subseteq x + RB_n$. For the case of a function oracle for $f : \mathbb{R}^n \to \mathbb{R}$, we are guaranteed the parameters x, r, R are satisfied for the support of f. An additional requirement may be placed on x, e.g. $f(x)$ is close to the maximum of f as in [10]

or a small level set of f contains a ball around x as in Section 3.5.

One benefit of working in the oracle model is that it emphasizes the algorithmic dependence on dimension, as opposed to the complexity of the space. Another reason we work in the oracle model, as opposed to purely counting arithmetic operations, is that the current fastest algorithms for volume computation are essentially the same for all classes of convex bodies (e.g. even by assuming the convex body is a polytope, the volume algorithm remains virtually identical). Thus the arithmetic complexity of the algorithm is bounded number of oracle queries times the number of arithmetic operations in a single query (though it may be possible to improve upon this naive bound for certain walks and membership oracles). For example, in Chapter 4 where we get an $O^*(n^3)$ volume algorithm, the arithmetic complexity of this algorithm for a general polytope with m facets in $\mathbb{R}^n$ will be $O^*(n^4m)$.

2.2 Probability and Markov Chains

We begin with a formal definition of a continuous space Markov chain, adapted from [4]. $(\Omega, \mathcal{A})$ be a σ-algebra, i.e. $\mathcal{A}$ is the set of subsets of Ω and is closed under complement, countable unions and countable intersections. For every state $u \in \Omega$, let P_u denote a probability measure on Ω; assume that for every $A \in \mathcal{A}$, $P_u(A)$ is measurable. The triple $(\Omega, \mathcal{A}, \{P_u : u \in \Omega\})$ together with an initial distribution Q_0 on Ω defines a Markov chain. The Markov chain then fully defines a sequence of random variables $w_0, w_1, w_2, \ldots$ drawn from Ω such that w_0 is chosen from Q_0 and w_{i+1} is chosen from the distribution P_{w_i} for all $i > 0$. Importantly, w_{i+1} is independent of $w_0, \ldots, w_{i-1}$. Thus for all $u_0, \ldots, u_i \in \Omega$ and $A \in \mathcal{A}$,

$$\Pr(w_{i+1} \in A | w_0 = u_0, \ldots, w_i = u_i) = \Pr(w_{i+1} \in A | w_i = u_i) = P_{u_i}(A).$$

We say a probability measure Q on $(\Omega, \mathcal{A})$ is a stationary distribution of the Markov chain if taking one step from the associated distribution of Q yields the same distribution. That is, for all

$A \in \mathcal{A}$,

$$\int_{\Omega} P_u(A) dQ(u) = Q(A).$$

The Markov chain is *irreducible* if for all pairs $x, y \in \Omega$, there exists a sequence of valid transitions to get from x to y. The Markov chain is *lazy* if $P_u(u) \geq 1/2$ for all $u \in \Omega$. This condition is enforced by doing nothing with probability $1/2$. The distribution Q is *atom-free* if there is no $x \in \Omega$ such that $Q(x) > 0$. The following folklore theorem gives sufficient conditions for uniqueness of the stationary distribution Q. The reader is referred to [11, 12] for the details of the proof and more background.

Theorem 2.2.1. *If Q is a stationary distribution for a lazy, irreducible Markov chain, then Q is the unique stationary distribution for that Markov chain.*

In the context of this thesis, the stationary distribution will generally be obvious from the construction of the chain. Of more concern is the mixing time of the Markov chain, which is bounded through the conductance of the chain

The conductance is a bound on the worst cut of the space Ω, with respect to the one-step distribution of the walk. Formally, the conductance of a set $S \subset \Omega$ is

$$\phi(S) = \frac{\int_S P_x(\Omega \setminus S)\, dQ(x)}{\min Q(S), Q(\Omega \setminus S)}.$$

The conductance, ϕ, of a Markov chain is then the minimum conductance of any set:

$$\phi = \min_{S \subseteq \Omega} \phi(S).$$

The *local conductance* ℓ of a point $u \in K$ is $\ell(u) = 1 - P_u(\{u\})$, i.e. the probability that the Markov chain stays at u.

We mention two methods of comparing probability distributions. Let P and Q be two probability distributions with common state space K. The total variation distance of P and Q is defined

as:

$$\|P - Q\|_{tv} = \frac{1}{2}\int_K |P(u) - Q(u)|\, du = \sup_{A \in \mathcal{A}} |P(A) - Q(A)|. \tag{2.1}$$

We define the M-warmness of P and Q as:

$$M(P, Q) = \sup_{S \subseteq K} \frac{P(S)}{Q(S)}. \tag{2.2}$$

The above notions of probabilistic distance have different uses, which we informally mention. The total variation distance is useful for declaring P has converged to Q. If the $d_{tv}(P, Q) \leq \varepsilon$ for some $\varepsilon \in [0, 1]$, then with probability $1 - \varepsilon$, a random sample from P is equivalent to a random sample from Q. The *mixing time* of a Markov chain typically denotes the number of steps until the total variation distance is at most $1/4$. The M-distance is a weaker notion of distance, and P will be a starting distribution for a Markov chain with stationary distribution Q. We say that P is a *warm start* to Q if $M(P, Q) = O^*(1)$.

The following theorem of [4] will be used to bound the mixing time of the Markov chains presented in this thesis.

Theorem 2.2.2 (Corollary 1.5[4]). *Let Q_0 denote a starting distribution and Q denote the unique stationary distribution for a Markov chain. Let $M = M(Q_0, Q)$. Then,*

$$\|Q_t - Q\|_{tv} \leq \sqrt{M}\left(1 - \frac{\phi^2}{2}\right)^t.$$

2.3 Geometric Random Walks

In this section, we will give an overview of six geometric random walks. The primary motivation for the algorithmic development of this area was the search for a faster volume algorithm. The first polynomial time volume algorithm with complexity $O^*(n^{23})$ of Dyer, Frieze, and Kannan [13]

sampled from a very fine discretized grid (using the grid walk) to approximate continuous random points from a convex body. Since this algorithm, the field of continuous random walks has been rigorously developed, and we now have mixing time bounds for more sophisticated methods to sample from convex bodies.

The first four walks below require only function oracle access to be efficiently implemented. The *grid walk* is a discrete walk that, when run on a fine (i.e. points are close together) enough lattice, will approximate the continuous target distribution. Not surprisingly, the grid walk is less efficient than the continuous space Markov chains. The *ball walk* and *hit-and-run* are currently the only two continuous space random walks that have polynomial time mixing guarantees for general convex bodies. *Coordinate hit-and-run*, also known as Gibbs sampling in other literature, is a slight variant on hit-and-run which only selects from a small number of basis directions, but currently little is known of its mixing time. There are many random walks which have unknown mixing times, but we mention coordinate hit-and-run in particular because we observe it to be faster than hit-and-run in practice for volume computation (Section 2.3.4).

The *Dikin walk*, by contrast, requires the convex body to be specified by the intersection of n-dimensional halfspaces, i.e it can sample the set $\{x|Ax \leq b\}$ for $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$. Even though it requires a significantly stronger model, the Dikin walk currently possesses certain advantages over the previously mentioned walks. Namely, it is affine invariant, which removes the need for the expensive preprocessing step to round the convex body or target distribution; this rounding step is provably required for the other walks to run efficiently. The recently analyzed *geodesic walk* operates in a similar setting to the Dikin walk, but obtains slightly faster mixing time guarantees.

We will give a detailed comparison of the currently known theoretical guarantees of each walk in Table 2.1. We now define the walks.

2.3.1 Grid Walk

The grid walk samples from a set of grid points intersected with the convex body K. For a parameter $\delta \in \mathbb{R}$, a point x is in the grid if $x \in K$ and every coordinate of x is an integer multiple of δ. Then for a grid point $x \in K$, define the set of neighbors of x to be those grid points that are distance δ away from x, i.e. the set of $2n$ points defined by $\{y | y = x \pm \delta e_i, i \in [n]\}$ where e_i is the i-th coordinate vector in $\mathbb{R}^n$. Figure 2.1 gives the one-step distribution of the grid walk.

Grid Walk(δ)
At point x:

1. Randomly select one of the $2n$ neighbors of x. Call this point y.
2. Go to y if $y \in K$; otherwise, stay at x.

Figure 2.1: The Grid Walk

2.3.2 Ball Walk

The ball walk is a simple, continuous space random walk defined by a parameter $\delta \in \mathbb{R}$. From the current point $x \in K$, it generates a uniform random point in the ball of radius δ around x, and goes to that point if it is in K.

Ball Walk(δ)
At point x:

1. Pick a random point y from $x + \delta B_n$.
2. Go to y if $y \in K$; otherwise, stay at x.

Figure 2.2: The Ball walk

2.3.3 Hit-and-run

Hit-and-run is a parameter-free random walk that chooses a random chord through the current point x and moves the a random point along that chord. Unlike the previous two walks, there are no "null" steps of hit-and-run where a point generated is outside K and the walk stays at x. The implementation details of hit-and-run are given in Section 5.2.5.

Hit-and-run
At point x:

1. Pick a random line ℓ through the current point x.
2. Go to a random point on $\ell \cap K$.

Figure 2.3: Hit-and-run

2.3.4 Coordinate Hit-and-run

Coordinate hit-and-run is equivalent to hit-and-run, except instead of picking a uniform random direction, the walk only considers coordinate directions. Thus in a single step, at most one coordinate of the current point will change. In principle, the walk could consider sampling directions from any fixed, full-dimensional set of basis directions.

Coordinate Hit-and-run
At point x:

1. Pick a random axis-aligned direction ℓ through the current point x.
2. Go to a random point on $\ell \cap K$.

Figure 2.4: Coordinate Hit-and-run

2.3.5 Dikin Walk

As mentioned above, the Dikin walk assumes that the convex body is an explicit polytope $K = \{x|Ax \leq b\}$. Let m denote the number of inequalities describing K. The one-step distribution of the walk is defined by the Dikin ellipsoid. Let D_x denote the Dikin ellipsoid at point x, defined as follows:

$$D_x := \{z|(z-x)^T \sum_{i=1}^{m} \frac{a_i \cdot a_i^T}{(b_i - a_i^T x)^2}(z-x) \leq 1\}.$$

The Dikin walk at x generates a uniformly chosen random point y from D_x and goes to y with probability

$$\min\{1, \frac{\mathsf{vol}(D_y)}{\mathsf{vol}(D_x)}\}.$$

More generally, the Dikin walk can be implemented for any convex body equipped with a self-concordant barrier function [14].

Dikin Walk(A,b)
At point x:

1. Pick a random point y from the Dikin ellipsoid D_x.
2. Go to y with probability $\min\{1, \mathsf{vol}(D_y)/\mathsf{vol}(D_x)\}$.

Figure 2.5: The Dikin walk in a polytope

2.3.6 Geodesic Walk

The geodesic walk, as with the Dikin walk, assumes the body is an explicit polytope. Unlike the previously mentioned walks, however, the geodesic walk mixes on a higher dimensional Riemannian manifold, then projects the point to lie inside the polytope. The one-step distribution is implemented by solving an ordinary differential equation. The reader is referred to [15] for a complete overview of the process.

2.3.7 Comparison of the Walks

Here we give an overview of the random walks, with their known mixing time bounds. The mixing time of each walk assumes a warm start. $R = O^*(\sqrt{n})$ is obtainable for ball walk and hit-and-run, thus yielding bounds of $O^*(n^3)$ steps and $O^*(mn^4)$ arithmetic operations. Under the additional assumption that K is isotropic, the current best bound on the KLS conjecture [16] implies that the ball walk mixes for an isotropic convex body in $O^*(n^{2.5})$ steps from a warm start.

The parameter ω is the complexity of matrix multiplication, i.e. two $n \times n$ matrices can be multiplied in $O(n^\omega)$ arithmetic operations. The current bound of [17] gives $\omega < 2.3728\ldots$. In practice, using the $O(n^3)$ matrix multiplication algorithm is often preferable until n gets very high, and thus the number of arithmetic operations of the Dikin walk and the geodesic walk would be $O(m^2n^3)$ and $O(m^2n^{11/4})$, respectively.

Walk	Mixing Time	Condition	Arithmetic Operations (polytope)
Grid Walk	$O^*(n^6)$ [18]	$[-1,1]^n \subseteq K \subseteq [-3n,3n]^n$	$O^*(mn^7)$
Ball Walk	$O^*(n^2R^2)$ [7]	$B_n \subseteq K \subseteq RB_n$	$O^*(mn^3R^2)$
Hit-and-run	$O^*(n^2R^2)$ [19]	$B_n \subseteq K \subseteq RB_n$	$O^*(mn^3R^2)$
Coordinate hit-and-run	Open	??	Open
Dikin Walk	$O(mn)$ [20]	K is an explicit polytope	$O(m^2n^\omega)$ [20]
Geodesic Walk	$O^*(mn^{3/4})$ [15]	K is an explicit polytope	$O^*(m^2n^{\omega-1/4})$ [15]

Table 2.1: A comparison of geometric random walks by number of steps and arithmetic operations on a polytope.

2.3.8 Sampling from a Function

While this section focused on generating uniform random samples from a convex body, most of the above walks can be modified to sample from a general function. For the grid walk and the ball walk, the one-step distribution would use a Metropolis filter, where we accept the point y with probability

$$\min\{1, \frac{f(y)}{f(x)}\}.$$

Note that when f is the indicator function of the convex body K, the above procedure is identical to the original ball walk.

For hit-and-run and coordinate hit-and-run, we sample a point y from the chord with probability proportional to $f(y)$. Note than this step now involves sampling from a general one-dimensional logconcave function; the details of this procedure are given in [9] with only a logarithmic (in the error parameter) number of oracle queries.

Provided the Markov chain is lazy and irreducible, the walk is guaranteed to converge to f. Moreover, the ball walk and hit-and-run are fast-mixing from a warm start for logconcave f. The mixing time bounds for both match those for uniform sampling [10], under similar roundedness conditions (see Section 3.5).

2.4 Convex Geometry

In this section, we outline some fundamental properties of convexity and logconcavity in high-dimensions. The facts will be useful for proving various facts throughout this work.

A set $K \subseteq \mathbb{R}^n$ is *convex* if for all pairs $x, y \in K$, their average $(x+y)/2 \in K$. Many properties of convexity carry extend nicely to logconcave functions. A function $f : \mathbb{R}^n \to \mathbb{R}_+$ is logconcave if for all $x, y \in \mathbb{R}^n$ and all $\lambda \in [0, 1]$,

$$f(\lambda x + (1 - \lambda)y) \geq f(x)^{\lambda} f(y)^{1-\lambda}.$$

Observe that the indicator function of a convex set is logconcave. Convexity and logconcavity appear to define the frontier of efficient high-dimensional sampling algorithms.

2.4.1 Fundamental Properties and Inequalities

Define the Minkowski sum of two sets $A, B \subseteq \mathbb{R}^n$ as

$$A + B = \{a + b | a \in A, b \in B\}.$$

The Brunn-Minkowski theorem states that is A, B are measurable, then

$$\mathsf{vol}(A + B)^{1/n} \geq \mathsf{vol}(A)^{1/n} + \mathsf{vol}(B)^{1/n}.$$

The Prékopa-Leindler inequality can be viewed as a functional generalization of Brunn-Minkowski. Let $f, g, h : \mathbb{R}^n \to \mathbb{R}_+$ such that for all $\lambda \in [0, 1]$,

$$h(\lambda x + (1 - \lambda)y) \geq f(x)^\lambda g(x)^{1-\lambda}.$$

Then,

$$\int_{\mathbb{R}^n} h(x)\, dx \geq \left(\int_{\mathbb{R}^n} f(x)\, dx \right)^\lambda \left(\int_{\mathbb{R}^n} g(x)\, dx \right)^{1-\lambda}.$$

It is easy to see that the product and minimum of two logconcave functions. The following theorem due to Dinghas, Leindler, and Prékopa summarizes important properties of logconcave functions.

Theorem 2.4.1. *[21, 22, 23, 24] All marginals as well as the distribution function of a logconcave function are logconcave. The convolution of two logconcave functions is logconcave.*

The Lowner-John ellipsoid E of a convex body $K \subseteq \mathbb{R}^n$ is the ellipsoid of maximum volume contained within K. John [25] showed that E is unique and $K \subseteq nE$.

The following theorem of [5] gives a useful property of isotropic convex bodies. Note that it implies John's theorem.

Theorem 2.4.2. *[5] Let $K \subseteq \mathbb{R}^n$ be an isotropic convex body. Then,*

$$\sqrt{\frac{n+2}{n}} B_n \subseteq K \subseteq \sqrt{n(n+2)}.$$

2.4.2 Isoperimetry

The study of isoperimetry concerns the ratio of the boundary to the volume of a region. In the context of geometric random walks, we study the minimum surface that divides a convex region into two pieces. More precisely, consider a partition of a convex region K into three subsets S_1, S_2, S_3 such that S_1 and S_2 are at least d distance apart, for some chosen distance metric. We then bound the minimum volume, or measure, of S_3 in terms of the parameter d and properties of the underlying region/distribution. These types of isoperimetric inequalities arise naturally when attempting to bound the conductance of a geometric random walk.

We define the notion of distance between sets as the minimum distance between any pair on points. For the case of Euclidean distance,

$$d(A, B) = \min_{a \in A, b \in B} \|a - b\|.$$

Here we give a brief overview of isoperimetric inequalities for proving mixing time bounds for the ball walk and hit-and-run. The following isoperimetric inequality is useful for bounding the mixing time of the ball walk.

Theorem 2.4.3. *[26, 18] Let K be a convex body in $\mathbb{R}^n$ such that $B_n \subseteq RB_n$. Consider a partition of K into S_1, S_2, S_3. Then,*

$$\mathsf{vol}(S_3) \geq \frac{2d(S_1, S_2)}{R} \min\{\mathsf{vol}(S_1), \mathsf{vol}(S_2)\}.$$

The above inequality was stated in terms of the convex body K, but can be generalized to

arbitrary logconcave measures.

Theorem 2.4.4. *Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be a logconcave function with associated measure π_f. Suppose the support of f has diameter D. Consider a partition of K into S_1, S_2, S_3. Then,*

$$\pi_f(S_3) \geq \frac{2d(S_1, S_2)}{D} \min\{\pi_f(S_1), \pi_f(S_2)\}.$$

There are two overarching choices when formulating an isoperimetric inequality: the distance metric and the underlying property of the region/distribution. The one-step distribution of the ball walk suggests Euclidean distance as the natural distance metric. In [5], they get an improved isoperimetric inequality by considering the average distance of a point from the centroid of the distribution. Let z_f denote the centroid of f, i.e.

$$z_f = \mathsf{E}_{X \sim f}(X).$$

Define $M(f)$ as

$$M(f) = \mathsf{E}_{X \sim f}\left(\|X - z_f\|\right).$$

Note that when K is an isotropic simplex, the below isoperimetric inequality represents a substantially better bound as $R = O(n)$, but $M(f) = O^*(\sqrt{n})$.

Theorem 2.4.5. *[5] Let $f : \mathbb{R}^n \to \mathbb{R}$ be a logconcave function with associated measure π_f. Then for any partition S_1, S_2, S_3 of $\mathbb{R}^n$,*

$$\pi_f(S_3) \geq \frac{\ln(2) d(S_1, S_2)}{M(f)} \pi_f(S_1)\pi_f(S_2).$$

For the case of hit-and-run, the isoperimetric inequality uses the cross-ratio distance d_K. For two points $u, v \in K$, let $\ell(u, v)$ denote the unique line through u and v. Then, define p and q as the two endpoints on the chord $\ell(u, v) \cap K$ such that u is between p and v. Then, the cross-ratio

distance d_K is defined as

$$d_K(u,v) = \frac{|u-v||p-q|}{|p-u||v-q|}.$$

The cross-ratio distance between two sets is then the minimum cross-ratio distance between any pair of points from different sets. The following isoperimetry in terms of cross-ratio distance was proven in [19].

Theorem 2.4.6. *[19] Let $K \subseteq \mathbb{R}^n$ be a convex body and S_1, S_2, S_3 be a partition of K. Then,*

$$\mathsf{vol}(S_3) \geq d_K(S_1, S_2) \min\{\mathsf{vol}(S_1), \mathsf{vol}(S_2)\}.$$

The above isoperimetric inequality can similarly be extended to logconcave distributions; such an extension is discussed in Section 3.5.

2.4.3 Rounding

The previous section motivates the need for a preprocessing step to round the convex body K. For instance, Theorem 2.4.3 leads to a mixing time for the ball walk of $O^*(n^2R^2)$. The convex body K could be arbitrarily "skewed" in some directions, which leads to poor mixing times of the random walks that inherently operate in a local space around the current point. We consider two notions of roundedness.

A convex body $K \subseteq \mathbb{R}^n$ is said to be R-sandwiched or R-rounded if $B_n \subseteq K \subseteq RB_n$. Ideally, the parameter R should be made as small as possible. John's theorem [25] proves that $R = n$ is achievable for any convex body, and the isotropic simplex shows that this is tight. The algorithm of Lovász [] computes a transformation that guarantees $R = n^{3/2}$ via an application of the Ellipsoid method, thus losing a $\sqrt{n}$ factor from optimality.

A more refined notion of roundedness relies on the body being close to isotropic. A convex

body $K \subseteq \mathbb{R}^n$ is said to be C-nearly isotropic if

$$\mathsf{E}_{X\sim K}\,(X) \leq C$$

and for all unit vectors $v \in \mathbb{R}^n$,

$$1 - C \leq \mathsf{E}_{X\sim K}\left(v^T x)^2\right) \leq 1 + C.$$

If K is in isotropic position, then $K \cap O^*(\sqrt{n})B_n$ contains all but a negligible fraction of its mass. Therefore if the body K is transformed to approximate isotropic position, its effective radius is $O^*(\sqrt{n})$ for the purposes of computing volume or generating random samples from K. The current fastest algorithm to place K into approximate isotropic position uses $O^*(n^4)$ oracle queries [27].

2.4.4 Localization

The localization lemma of Lovász and Simonovits [4, 5] is an extremely useful analysis tool for proving a certain class of n-dimensional inequalities by reducing them to an equivalent one-dimensional inequality. Isoperimetric inequalities are commonly proved through localization. The fundamental lemma is as follows.

Lemma 2.4.7. *[4] Let $g, h : \mathbb{R}^n \to \mathbb{R}$ be lower semi-continuous Lebesgue integrable functions on $\mathbb{R}^n$ such that*

$$\int_{\mathbb{R}^n} g(x)\,dx > 0 \qquad \textit{and} \qquad \int_{\mathbb{R}^n} h(x)\,dx > 0.$$

Then there exist two points $a, b \in \mathbb{R}^n$ and a linear function $\ell : [0,1] \to \mathbb{R}_+$ such that

$$\int_0^1 \ell(t)^{n-1} g\left((1-t)a + tb\right)\,dt > 0 \qquad \textit{and} \qquad \int_0^1 \ell(t)^{n-1} h\left((1-t)a + tb\right)\,dt > 0.$$

The lemma is proven by constructing a sequence of bisections, arriving at a limiting structure of an infinitesimal, truncated cone (so called *needles*). A needle N is defined by an segment $I = [a, b]$ in $\mathbb{R}^n$ and a nonnegative linear function $\ell : I \to \mathbb{R}^+$. The integral of a function f over a needle is then

$$\int_N f(x)\,dx = \int_0^{\|b-a\|} f(a+tu)\ell(a+tu)^{n-1}\,dt \qquad \text{where } u = \frac{b-a}{\|b-a\|}.$$

The following statements are extensions of the above localization lemma and are sometimes more straightforward to apply.

Lemma 2.4.8. *[5] Let f_1, f_2, f_3, f_4 be four nonnegative continuous functions defined on $\mathbb{R}^n$ and $\alpha, \beta > 0$. Then the following are equivalent:*

(i) For every convex body K in $\mathbb{R}^n$,

$$\left(\int_K f_1\right)^\alpha \left(\int_K f_2\right)^\beta \le \left(\int_K f_3\right)^\alpha \left(\int_K f_4\right)^\beta.$$

(ii) For every needle N in $\mathbb{R}^n$,

$$\left(\int_N f_1\right)^\alpha \left(\int_N f_2\right)^\beta \le \left(\int_N f_3\right)^\alpha \left(\int_N f_4\right)^\beta.$$

For the following theorem, we define an exponential needle E in $\mathbb{R}^n$. An exponential needle is the same as a regular needle, but replaces the linear function with an exponential distribution $e^{\gamma t}$ for some $\gamma \in \mathbb{R}$.

Theorem 2.4.9. *[5] Let f_1, f_2, f_3, f_4 be four nonnegative continuous functions defined on $\mathbb{R}^n$ and $\alpha, \beta > 0$. Then the following are equivalent:*

(i) For every logconcave function F defined on $\mathbb{R}^n$ with compact support,

$$\left(\int_{\mathbb{R}^n} F(t)f_1(t)\,dt\right)^\alpha \left(\int_{\mathbb{R}^n} F(t)f_2(t)\,dt\right)^\beta \le \left(\int_{\mathbb{R}^n} F(t)f_3(t)\,dt\right)^\alpha \left(\int_{\mathbb{R}^n} F(t)f_4(t)\,dt\right)^\beta.$$

(ii) For every exponential needle E in $\mathbb{R}^n$,

$$\left(\int_E f_1\right)^{\alpha}\left(\int_E f_2\right)^{\beta} \leq \left(\int_E f_3\right)^{\alpha}\left(\int_E f_4\right)^{\beta}.$$

While localization is an incredibly useful tool, it cannot be applied in every circumstance. Informally, localization preserves some small amount of information about the distribution when reducing to one dimension. In particular, it is difficult to apply when the property is more "complex", e.g. the distribution is isotropic, has mean zero, etc.

KLS Conjecture

Theorem 2.4.3 and Theorem 2.4.5 can both be shown to be tight by a long box, i.e. a cube that is stretched far along one axis. However, as a precursor to the sampling and volume algorithms, a rounding transformation is computed that makes the body "nice", specifically to place the body in approximate isotropic position. In [5], they formulate their important conjecture that when the convex body K is isotropic, the *isoperimetric coefficient* is an absolute constant independent of dimension.

Conjecture 2.4.10. [5] Let $K \subseteq \mathbb{R}^n$ be an isotropic convex body and $S \subseteq K$. Let $\partial S \cap K$ denote the boundary of S strictly inside K. Then,

$$\mathsf{vol}_{n-1}\left(\partial S \cap K\right) \geq \psi \frac{\mathsf{vol}(S) \cdot \mathsf{vol}(K \backslash S)}{\mathsf{vol}(K)}$$

for some absolute constant ψ.

The previous isoperimetric inequalities were stated in terms of a partition of K into three sets S_1, S_2, S_3; the above formulation can in terms of boundary be can be recovered by letting the distance between S_1 and S_2 tend to 0. When K is isotropic, 2.4.3 and Theorem 2.4.5 imply that $\psi = O(1/n^{-1/2})$ suffices. Note that hyperplanes through the origin show that ψ is at most

a constant; thus the conjecture informally states that up to constants, a hyperplane is the most efficient way to cut an isotropic convex body into two pieces. In the typical method of bounding the conductance of the ball walk, this factor shows up in the square of the conductance, and the KLS conjecture represented a potential $O(n)$ speed-up for the mixing time of the ball walk in an isotropic convex body K.

Since the conjecture was formulated roughly 20 years ago, it has received much attention and substantial partial progress has been made. The current best bound [16] proves that $\psi = O(n^{-1/4})$ is sufficient, thus implying that the ball walk mixes in $O^*(n^{2.5})$ steps from a warm start fo K isotropic.

CHAPTER 3

GAUSSIAN SAMPLING

The driving force behind the theoretical development of efficient sampling algorithms has been the search for fast algorithms for integration and volume computation. Both of these algorithms essentially use a randomized sampler as a black-box, thus the complexity of sampling appears as a multiplicative factor in the complexity of integration or volume. Nevertheless, randomized sampling is an incredibly useful tool in its own right for numerous statistical applications. The ability to generate random samples from a high-dimensional distribution yields a concrete tool to extract information from a complex structure, e.g. mean, covariance, etc.

The current frontier of efficient high-dimension sampling appears to be defined by logconcavity of the distribution [9, 19]. Additionally, if the distribution can be efficiently sampled if it is "close" to logconcave [28, 29]. All of the approaches use an appropriately chosen Markov chain which converge to the target distribution in a polynomial number of steps. In [9], it is proven that $O^*(n^3)$ steps of hit-and-run, from a warm start, suffice to sample from an approximately isotropic distribution.

Our first result is an $O^*(n^2)$ algorithm for sampling from the standard Gaussian distribution in $\mathbb{R}^n$ restricted to an convex set containing the unit ball. Thus, this complexity is an $O^*(n)$ improvement for the special case of Gaussian distributions. To achieve this complexity, we prove the KLS conjecture for such distributions.

The $O^*(n^2)$ mixing time above assumes that the starting distribution gives a warm start to the target distribution. We construct an algorithm that runs in $O^*(n^3)$ oracle queries to generate a warm start for the Gaussian distribution restricted by K. Such a process is a fixed up-front cost as subsequent random points can start from the distribution of the previous point. Thus the sampling complexity will be $O^*(n^3)$ for the first random sample and $O^*(n^2)$ for every subsequent sample.

Additionally, our $O^*(n^2)$ mixing time results extend to sampling a Gaussian multiplied by a logconcave distribution. There is an assumption on the logconcave distribution that it is well-rounded, a weaker condition than being approximately isotropic. The current best complexity for making a logconcave distribution well-rounded is $O^*(n^4)$ [10].

3.1 Results

The following theorem summarizes our main result for generating samples from a Gaussian restricted by a convex body.

Theorem 3.1.1. *Let K be a convex set containing the unit ball, Q_0 be a starting distribution, and Q be the target Gaussian density $\mathcal{N}(0, \sigma^2 I)$ restricted to $K \cap 4\sigma\sqrt{n}B_n$. For any $\varepsilon > 0, p > 0$, the lazy Metropolis ball walk with δ-steps for $\delta = \min\{\sigma, 1\}/(4096\sqrt{n \log n/\varepsilon})$, starting from Q_0, satisfies $d_{tv}(Q_t, Q) \leq \varepsilon$ after*

$$t \geq C \cdot M(Q_0, Q) \cdot \max\{\sigma^2, 1\} \cdot n^2 \log\left(\frac{n}{\varepsilon}\right) \log\left(\frac{M(Q_0, Q)}{\varepsilon}\right)$$

expected steps for an absolute constant C.

To compute the warm start required in Theorem 3.1.1, we use a sequence of spherical Gaussian distributions, starting with a sharp, low-variance Gaussian with almost all of its mass inside the convex body. We then increase the variance slightly, so that the previous Gaussian maintains a warm start for the next Gaussian. We repeat this cooling procedure until we have reach the target Gaussian distribution. The details of this algorithm are given in Algorithm **??**.

Theorem 3.1.2 states our precise result for Gaussian sampling. Obtaining the first random point is more difficult because we need a good starting point for the random walk. However, subsequent points can simply use the previous point as a starting point, thus reducing the number of required oracle calls for each subsequent point by a factor of $O^*(n)$.

Theorem 3.1.2. *For any $\varepsilon > 0$, $p > 0$, and any convex set K in $\mathbb{R}^n$ containing the unit ball, there is an algorithm that, with probability $1 - p$, can generate a random point within total variation distance ε of the Gaussian density $\mathcal{N}(0, \sigma^2 I)$ restricted to K. In the membership oracle model, the complexity of the first random point is*

$$O\left(\max\{\sigma^2, 1\} n^3 \log(n) \log^2\left(\frac{n}{\varepsilon}\right) \log\left(\frac{1}{p}\right)\right) = O^*\left(\max\{\sigma^2, 1\} n^3\right).$$

For subsequent random points, the complexity is

$$O\left(\max\{\sigma^2, 1\} n^2 \log\left(\frac{n}{\varepsilon}\right) \log\left(\frac{1}{\varepsilon}\right)\right) = O^*\left(\max\{\sigma^2, 1\} n^2\right).$$

The set of random points will be ε-independent.

In Chapter 4, we then show that the *Gaussian volume* or Gaussian measure, i.e., the integral of a standard Gaussian over a convex body, of a convex body given by a membership oracle and containing the unit ball can be computed in $O^*(n^3)$ time.

In previous work, the complexity of generating the first nearly uniform random point was always significantly higher than for later points. Here, using a faster cooling schedule, we can generate the first random point in $O^*(n^3)$ steps, under the same assumption that K is well-rounded. Any subsequent uniform random points also require $O^*(n^3)$ steps.

Theorem 3.1.3. *There is an algorithm that, for any $\varepsilon > 0, p > 0$, and any convex body K in $\mathbb{R}^n$ that contains the unit ball and has $\mathsf{E}_K(\|X\|^2) = R^2$, with probability $1 - p$, generates random points from a density ν that is within total variation distance ε from the uniform distribution on K. In the membership oracle model, the complexity of each random point, including the first, is*

$$O\left(\max\{R^2 n^2, n^3\} \log n \log^2 \frac{n}{\varepsilon} \log \frac{1}{p}\right) = O^*\left(\max\{R^2 n^2, n^3\}\right).$$

3.2 Outline of Analysis

To show the random walk quickly reaches its stationary distribution, we will use the standard method of bounding the conductance. For the ball walk, this runs into a hurdle, namely, the local conductance of points near sharp corners of the body can be arbitrarily small, so the walk can get stuck and waste a large number of steps. To avoid this, we could start the walk from a random point chosen from a distribution sufficiently close to the target distribution. But how to generate random points from such a starting distribution? We do this by considering a sequence of distributions, each providing a warm start for the next. The very first distribution is chosen to be a highly concentrated Gaussian so that it almost entirely lies inside the unit ball (inside K). Thus sampling from the initial distribution is easy by standard rejection sampling. Each successive Gaussian is "flatter" with the final one being the target distribution, e.g. standard Gaussian, uniform distribution.

The next challenge is to show that, from a warm start, the expected number of steps to converge to the stationary distribution is only $O^*(n^2)$. This is usually done by bounding the conductance of the Markov chain.

Unfortunately, for the ball walk, this can be arbitrarily small, e.g., for points near corners (but also for points in the interior). To utilize the warm start, we use an idea from [7], namely the *speedy walk*. We emphasize that the speedy walk cannot be implemented efficiently and is only a tool for analysis. It is defined as follows.

At current point x:

1. Pick random point y from $K \cap x + \delta B_n$.

2. Go to y with probability $\min\{1, f(y)/f(x)\}$.

To capture the stationary distribution of the speedy walk with a Metropolis filter we need another parameter, the *local conductance* at x for the speedy walk, without a filter.

The following fact is now easy to verify.

Lemma 3.2.1. *The stationary distribution of the speedy walk with a Metropolis filter applied with a function f has density proportional to $\ell(x)f(x)$.*

For the speedy walk with $\delta = O(1/\sqrt{n})$, we can show that the conductance is $\Omega(1/(\sigma n))$, and so the total number of steps needed is only $O^*(\sigma^2 n^2)$. This is a factor n faster than previous best bounds. We do this by establishing a stronger (and nearly optimal) isoperimetric inequality.

As noted, the speedy walk cannot actually be implemented efficiently. To bound the Metropolis ball walk, we can view it as an interleaving of a speedy walk with wasted steps. Let the Markov chain for the original walk be $w_0, w_1, \ldots, w_i, \ldots,$. The subsequence $w_{i_1}, w_{i_2}, \ldots,$ where we record x if the point y chosen by the Metropolis ball walk is in K, corresponds to the speedy walk. We then need to estimate the number of wasted steps from a warm start. We will show that this is at most a constant factor higher than the number of proper steps. The key ingredient of this analysis is the (known) fact that for a body containing the unit ball average local conductance is high for ball radius $\delta = O(1/\sqrt{n})$ [7]. Even within the speedy walk, there are "null" steps due to the Metropolis filter. However, by restricting the walk to a large ball, we ensure that the probability of rejection by the filter is bounded by a constant, and therefore the number of wasted steps within the speedy walk is at most a constant fraction of all steps. Also, the speedy walk converges to a distribution proportional to $\ell(x)f(x)$, but we can map this to a random sample from f with rejection sampling routine (Section 3.4.4).

To sample efficiently, we need a warm start for each phase, i.e. the M-warmness of the starting to target distribution is at most a constant. To ensure we maintain a warm start, we use a finer-grained cooling schedule so that a random point from one phase is a warm start for the next phase. This cooling schedule is also different in the two parts. In the first part of the algorithm, where we can cool at the rate of $1 + 1/n$ and use $O^*(n^2)$ steps to sample. In the second part, we cool at the rate of $1 + \sigma^2/(2C^2 n)$, and this is fast enough to compensate for the higher sample complexity of

$O^*(\sigma^2 n^2)$. Thus, the overall time to obtain a warm start for every phase of the algorithm is also $O^*(n^3)$. We analyze this in full detail in Section 3.4, including the proof that this cooling rate maintains a warm start from one phase to the next.

If uniform random samples from K are desired, we can use samples from a Gaussian with variance $\sigma^2 = C^2 n$ via a simple rejection sampling routine. Since $K \subseteq C\sqrt{n}B_n$, the two distributions will be within a constant factor of each other, and therefore we can use $O(1)$ expected samples from the Gaussian distribution to obtain a uniform random point.

3.3 Gaussian Isoperimetry

The following theorem is due to Brascamp and Lieb.

Theorem 3.3.1. *[30] Let $\gamma : \mathbb{R}^n \to \mathbb{R}_+$ be the standard Gaussian density in $\mathbb{R}^n$. Let $f : \mathbb{R}^n \to \mathbb{R}_+$ be any logconcave function. Define the density function h over $\mathbb{R}^n$ as follows:*

$$h(x) = \frac{f(x)\gamma(x)}{\int_{\mathbb{R}^n} f(y)\gamma(y)\,dy}.$$

Fix a unit vector $v \in \mathbb{R}^n$, let $\mu = \mathsf{E}_h(x)$. Then, for any $\alpha \geq 1$,

$$\mathsf{E}_h(|v^T(x-\mu)|^\alpha) \leq \mathsf{E}_\gamma(|x_1|^\alpha).$$

We have the following concentration bound.

Corollary 3.3.2. *For h as defined in Theorem 3.3.1, and any $t \geq 1$,*

$$\Pr_h(\|x-\mu\|^2 \geq n + ct\sqrt{n}) \leq e^{-t^2}$$

for an absolute constant c.

The next lemma about one-dimensional isoperimetry is from [5]

Lemma 3.3.3. *[5] For any one-dimensional isotropic logconcave function f, and any partition S_1, S_2, S_3 of the real line,*

$$\pi_f(S_3) \geq \ln(2)\, d(S_1, S_2)\pi_f(S_1)\pi_f(S_2).$$

Proof. By Theorem 5.1 of [5],

$$\pi_f(S_3) \geq \frac{\ln 2}{\mathsf{E}_f(\|x\|)} d(S_1, S_2)\pi_f(S_1)\pi_f(S_2).$$

To prove the lemma we note that

$$\mathsf{E}_f(\|x\|) \leq \sqrt{\mathsf{E}_f(\|x\|^2)} = 1$$

and

$$\pi_f(S_3) \geq c\pi_f(S_1)\pi_f(S_2) \geq c\pi_f(S_1)(1 - \pi_f(S_1) - \pi_f(S_3))$$

which, assuming $\pi_f(S_1) \leq \pi_f(S_2)$ and therefore $\pi_f(S_1) \leq 1/2$, gives

$$\pi_f(S_3) \geq \frac{c}{2+c} \min\{\pi_f(S_1), \pi_f(S_2)\}.$$

□

3.3.1 Isoperimetric Inequality

We can now prove the necessary isoperimetric inequality for Gaussians restricted by a logconcave function, essentially proving the KLS conjecture for such distributions.

Theorem 3.3.4. *Let π be the Gaussian distribution $N(0, \sigma^2 I_n)$ with density function γ restricted by a logconcave function $f : \mathbb{R}^n \to \mathbb{R}_+$, i.e., π has density $d\pi(x)$ proportional to $h(x) =$*

$f(x)d\gamma(x)$. *Let* S_1, S_2, S_3 *partition* $\mathbb{R}^n$ *such that for any* $u \in S_1, v \in S_2$, *either* $\|u-v\| \geq d/\ln(2)$ *or* $d_h(u,v) \geq 4d\sqrt{n}$. *Then,*

$$\pi(S_3) \geq \frac{d}{\sigma}\pi(S_1)\pi(S_2).$$

Proof. We prove the theorem for the case $\sigma = 1$, then note that by applying the scaling $x = y/\sigma$, we get the general case.

Our main tool, as in previous work, is the Localization Lemma of Lovász and Simonovits [4]. Suppose the conclusion is false. Define $h(x) = f(x)\gamma(x)$. Then there exists a partition S_1, S_2, S_3 for which, for some positive real number A,

$$\int_{S_1} h(x)\,dx = A\int_{\mathbb{R}^n} h(x)\,dx$$
$$\text{and } \int_{S_3} h(x)\,dx < dA\int_{S_2} h(x)\,dx.$$

By the localization lemma, there must be a "needle" given by $a, b \in \mathbb{R}^n$ and a nonnegative linear function $l : [0,1] \to \mathbb{R}_+$ for which,

$$\int_{(1-t)a+tb\in S_1\cap[0,1]} h((1-t)a+tb)l(t)^{n-1}\,dt = A\int_{(1-t)a+tb\in[0,1]} h((1-t)a+tb)l(t)^{n-1}\,dt$$

and

$$\int_{(1-t)a+tb\in S_3\cap[0,1]} h((1-t)a+tb)l(t)^{n-1}\,dt < dA\int_{(1-t)a+tb\in S_2\cap[0,1]} h((1-t)a+tb)l(t)^{n-1}\,dt.$$

By a standard combinatorial argument, we can assume that $Z_i = \{t : (1-t)a+tb \in S_i\}$ are intervals that partition $[a,b]$. Thus, to reach a contradiction, it suffices to prove that for a one-dimensional logconcave function $h(t) = f((1-t)a+tb)\gamma((1-t)a+tb)$ with support $[a,b] \subset \mathbb{R}$

and $a \leq u \leq v \leq b$, the following statements hold:

$$\int_a^b h(t)l(t)^{n-1}\,dt \int_u^v h(t)l(t)^{n-1}\,dt \geq \frac{d_h(u,v)}{4\sqrt{n}} \int_a^u h(t)l(t)^{n-1}\,dt \int_v^b h(t)l(t)^{n-1} \tag{3.1}$$

$$\int_a^b h(t)l(t)^{n-1}\,dt \int_u^v h(t)l(t)^{n-1}\,dt \geq \ln(2)\|u-v\| \int_a^u h(t)l(t)^{n-1}\,dt \int_v^b h(t)l(t)^{n-1}. \tag{3.2}$$

The first inequality (3.1) follows directly from Lemma 3.8 in [7]. To see the second inequality (3.2), we first note that by applying Theorem 3.3.1, with $\alpha = 2$, we have that the variance of the distribution proportional to $h(t)l(t)^{n-1}$ is at most 1. This is because $h(t)l(t)^{n-1} = (f((1-t)a + tb)l(t)^{n-1})\gamma((1-t)a+tb)$ and the $f((1-t)a+tb)l(t)^{n-1}$ is itself a logconcave function. Now, we note that by scaling down to increase the variance to exactly 1, the isoperimetric coefficient can only go down. Hence, the second inequality is implied by Lemma 3.3.3. □

3.4 Analyzing the Walk

The analysis of the sampling algorithm is divided into several parts: bounding the conductance of the speedy walk, bounding the warmth of the distribution from one phase to the next, the mixing time of the Metropolis ball walk from a warm start, and finally the complexity of sampling.

3.4.1 Speedy Walk Conductance

First we bound the rate of convergence of the random walk through a lower bound on the conductance. The conductance ϕ of a Markov chain with state space K and next-step distribution P_x is defined as:

$$\phi = \min_{S \subset K} \frac{\int_S P_x(K \backslash S) dQ(x)}{\min\{Q(S), Q(K \backslash S)\}}.$$

We will make use of the following theorem of Lovász and Simonovits [4] to bound the total

variation distance between the current distribution and the target distribution.

Theorem 3.4.1. *[4] Let Q_t be the distribution after t steps of a lazy Markov chain and Q be its stationary distribution. Suppose that Q_0 is M-warm with respect to Q. Then,*

$$d_{tv}(Q_t, Q) \leq \sqrt{M}\left(1 - \frac{\phi^2}{2}\right)^t.$$

For the next lemmas, we let $f : \mathbb{R}^n \to R$ denote the Gaussian density function

$$f(x) = \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right).$$

Lemma 3.4.2. *For the speedy walk applied to a convex body $K \subseteq 4\sigma\sqrt{n}B_n$, $\delta \leq \sigma/(8\sqrt{n})$ and $\sigma^2 \leq 64n$, the acceptance probabilty of the Metropolis filter is at least $\frac{1}{e}$.*

Proof. Assume that $f(x) \leq f(u)$ (otherwise the Metropolis filter always accepts). Then, the acceptance probability is

$$\begin{aligned}
\frac{f(x)}{f(u)} &= \exp\left(-\frac{\|x\|^2 - \|u\|^2}{2\sigma^2}\right) \\
&\geq \exp\left(-\frac{(\|u\| + \delta)^2 - \|u\|^2}{2\sigma^2}\right) \\
&= \exp\left(-\frac{2\delta\|u\| + \delta^2}{2\sigma^2}\right) \\
&\geq \exp\left(-\frac{\sigma^2 + \sigma^2/(64n)}{2\sigma^2}\right) \\
&\geq \frac{1}{e}.
\end{aligned}$$

□

Lemma 3.4.3. *Let $K \subseteq 4\sigma\sqrt{n}B_n$ be a convex body and let $u, v \in K$ such that $\|u - v\| \leq \delta/\sqrt{n}$.*

If $\delta \leq \sigma/(8\sqrt{n})$ *and*

$$\frac{|\ell(u)f(u) - \ell(v)f(v)|}{\max\{\ell(u)f(u), \ell(v)f(v)\}} < \frac{1}{4},$$

then

$$\frac{|\ell(u) - \ell(v)|}{\max\{\ell(u), \ell(v)\}} < \frac{1}{3}.$$

Proof. Assume without loss of generality that $\ell(u)f(u) \geq \ell(v)f(v)$. It follows that

$$\frac{3\gamma(u)\ell(u)}{4} < \gamma(v)\ell(v).$$

Since $\|u - v\| \leq \delta/\sqrt{n}$, we have that $0.9 \leq f(v)/f(u) \leq 1.1$ for n large enough. Thus

$$\ell(u) < \frac{3}{2}\ell(v).$$

By assumption, we have that $\gamma(u)\ell(u) > \gamma(v)\ell(v)$. Thus we also have that

$$\ell(v) < \frac{3}{2}\ell(u).$$

The lemma then follows.

□

The following lemma bounds the overlap for a step of the speedy walk with respect to the speedy walk. We then show that the Gaussian weighting only hurts by a constant factor.

Lemma 3.4.4. *Let* K *be a convex set with* $S \subseteq K$*. Let* $\overline{S} = K \backslash S$ *and let* P_x^{unif} *denote the 1-step distribution from* x *of the speedy walk with respect to the uniform distribution over* K*. Suppose that* $d_\ell(u, v) < 1/3$ *and* $\|u - v\| \leq \delta/\sqrt{n}$*. Then for* $u \in S$ *and* $v \in \overline{S}$*,*

$$P_u^{unif}\left(\overline{S}\right) + P_v^{unif}\left(S\right) > \frac{1}{6}.$$

Proof. Let $B_u = u + \delta B_n$ and let $C = B_u \cap B_v$. By Lemma 3.5 from [5], we know that

$$\mathsf{vol}\left(K \cap C\right) \geq \frac{\mathsf{vol}(\delta B_n)}{e+1} \min\left\{\ell(u), \ell(v)\right\}. \tag{3.3}$$

We have that

$$P_u^{unif}(\overline{S}) = \frac{\mathsf{vol}\left(\overline{S} \cap B_u\right)}{\ell(u)\mathsf{vol}(\delta B_n)} \geq \frac{\mathsf{vol}\left(\overline{S} \cap C\right)}{\ell(u)\mathsf{vol}(\delta B_n)}.$$

Similarly for $P_v^{unif}(S)$. Assume that $\ell(u) \geq \ell(v)$, which implies $\ell(u) \leq 3\ell(v)/2$. Therefore,

$$\begin{aligned} P_u^{unif}(\overline{S}) + P_v^{unif}(S) &\geq \frac{\mathsf{vol}\left(\overline{S} \cap C\right)}{\ell(u)\mathsf{vol}(\delta B_n)} + \frac{\mathsf{vol}\left(S \cap C\right)}{\ell(v)\mathsf{vol}(\delta B_n)} \\ &\geq \frac{2\mathsf{vol}\left(\overline{S} \cap C\right)}{3\ell(v)\mathsf{vol}(\delta B_n)} + \frac{\mathsf{vol}\left(S \cap C\right)}{\ell(v)\mathsf{vol}(\delta B_n)} \\ &\geq \frac{2\mathsf{vol}\left(K \cap C\right)}{3\ell(v)\mathsf{vol}(\delta B_n)} \\ &\geq \frac{2}{3(e+1)} > \frac{1}{6}. \end{aligned}$$

□

It then follows that the Gaussian filter decrease the overlap by at most a constant factor.

Corollary 3.4.5. *Let $S, \overline{S}$ be a partition of a convex body $K \subseteq 4\sigma\sqrt{n}B_n$, and $u \in S, v \in \overline{S}$ be such that $\|u - v\| < \delta/\sqrt{n}$ and $d_h(u, v) < 1/4$, where $h(x) = f(x)\ell(x)$. Then,*

$$P_u(\overline{S}) + P_v(S) > \frac{1}{20}.$$

Proof. By Lemma 3.4.3, we know that $d_\ell(u, v) < 1/3$. We then apply Lemma 3.5.9, while noting that the Gaussian weighting affects the 1-step distributions by at most a $1/e$ factor since that is a lower bound on the acceptance probability of the Metropolis filter (Lemma 3.4.2). □

We can now prove the desired lower bound on the conductance of the speedy walk with respect to a Gaussian weighting over a convex set.

Theorem 3.4.6. *Let K be a convex body such that $B_n \subseteq K \subseteq 4\sigma\sqrt{n}B_n$. The conductance of the speedy walk applied to K with Gaussian density $\mathcal{N}(0, \sigma^2 I)$ and $\delta \leq \sigma/8\sqrt{n}$ steps is $\Omega(\frac{\delta}{\sigma\sqrt{n}})$.*

Proof. Let $S \subset K$ be an arbitrary measurable set of K and let $\overline{S} = K\backslash S$. Assume that $\pi(S) \leq 1/2$. Consider the following partition of K:

$$S_1 = \left\{ x \in S : P_x(\overline{S}) < \frac{1}{20} \right\}$$

$$S_2 = \left\{ x \in \overline{S} : P_x(S) < \frac{1}{20} \right\}$$

$$S_3 = K\backslash S_1 \backslash S_2.$$

Let $h(x) = \ell(x)f(x)$. By Corollary 3.4.5, we have that for any $u \in S_1, v \in S_2$, either $\|u - v\| \geq \delta/\sqrt{n}$ or $d_h(u, v) \geq 1/4$.

We may assume that $\pi(S_1) \geq \pi(S)/2$ and $\pi(S_2) \geq \pi(\overline{S})/2$. If not, we can bound the conductance of S as follows (similarly for $\overline{S}$).

$$\begin{aligned}
\phi(S) &= \frac{\int_S P_x(\overline{S})h(x)\,dx}{\pi(S)} \\
&= \frac{1}{2}\frac{\int_S P_x(\overline{S})h(x)\,dx + \int_{\overline{S}} P_x(S)h(x)\,dx}{\pi(S)} \\
&\geq \frac{1}{2}\frac{\int_{S_3} \frac{h(x)}{20}\,dx}{\pi(S)} \\
&= \frac{1}{40}\frac{\pi(S_3)}{\pi(S)} \\
&\geq \frac{1}{80}.
\end{aligned}$$

Now we can apply Theorem 3.3.4 with

$$d = \min\left\{\frac{\delta \ln 2}{\sqrt{n}}, \frac{1}{16\sqrt{n}}\right\}$$

to the partition S_1, S_2, S_3 to get

$$\pi(S_3) \geq \frac{d}{\sigma}\pi(S_1)\pi(S_2).$$

Using the above, we get that

$$\begin{aligned}
\phi(S) &\geq \frac{1}{40}\frac{\pi(S_3)}{\pi(S_1)} \\
&\geq \frac{d}{40\sigma}\frac{\pi(S_1)\pi(S_2)}{\pi(S_1)} \\
&\geq \frac{d}{160\sigma} \\
&\geq \frac{\delta}{250\sigma\sqrt{n}},
\end{aligned}$$

which proves the theorem. □

3.4.2 Warm Start

The following two lemmas guarantee that the ball walk in the algorithm will always have a *warm start*, i.e. the M-warmness (2.2) is bounded by a constant. The first lemma bounds the warmness under the fixed cooling rate of $1 + 1/n$.

Lemma 3.4.7. *Let $K \subseteq \mathbb{R}^n$, $\sigma_{i+1}^2 = \sigma_i^2(1 + 1/n)$, and $f_i(x) = \exp\left(-\|x\|^2/(2\sigma_i^2)\right)$. Denote Q_i as the associated probability distribution of f_i over K. Then, we can bound the warmness between successive*

$$M(Q_i, Q_{i+1}) \leq \sqrt{e}.$$

The following lemma bounds the warmness when the cooling schedule begins to accelerate, under the roundness condition.

Lemma 3.4.8. *Let $K \subseteq C\sqrt{n} \cdot B_n$, $\sigma_{i+1}^2 = \sigma_i^2(1 + \sigma_i^2/(C^2 n))$, and $f_i(x) = \exp\left(-\|x\|^2/(2\sigma_i^2)\right)$. Denote Q_i as the associated probability distribution of f_i over K. Then we can bound the warmness between successive phases as*

$$M(Q_i, Q_{i+1}) \leq \sqrt{e}.$$

Proof.(of Lemma 3.4.7) Let

$$A = \frac{\int_K e^{-a_{i+1}\|x\|^2} dx}{\int_K e^{-a_i\|x\|^2} dx}.$$

Then,

$$\begin{aligned} M(Q_i, Q_{i+1}) &= \sup_{S \subseteq K} \frac{Q_i(S)}{Q_{i+1}(S)} \\ &\leq \sup_{x \in K} \frac{Q_i(x)}{Q_{i+1}(x)} \\ &= \sup_{x \in K} A \frac{e^{-a_i\|x\|^2}}{e^{-a_{i+1}\|x\|^2}} \\ &= A \cdot \sup_{x \in K} e^{-a_i\|x\|^2/n} \\ &= A, \end{aligned}$$

where the last line follows from the fact that $0 \in K$.

We will now bound A. First, we extend A to be over all $\mathbb{R}^n$ instead of K, and then argue that it

can only decrease when restricted to K.

$$\begin{aligned}\frac{\int_{\mathbb{R}^n} e^{-a_{i+1}\|x\|^2}dx}{\int_{\mathbb{R}^n} e^{-a_i\|x\|^2}dx} &= \frac{(a_i/\pi)^{n/2}}{(a_{i+1}/\pi)^{n/2}}\frac{(a_{i+1}/\pi)^{n/2}}{(a_i/\pi)^{n/2}}\frac{\int_{\mathbb{R}^n} e^{-a_{i+1}\|x\|^2}dx}{\int_{\mathbb{R}^n} e^{-a_i\|x\|^2}dx} \\ &= \frac{a_i^{n/2}}{a_{i+1}^{n/2}} \\ &= \left(\frac{1}{1-1/n}\right)^{n/2} \\ &\leq \sqrt{e}.\end{aligned}$$

Let $\mu_K(r)$ be the proportion of the sphere of radius r centered at 0 that is contained in K. Note that since K is a convex body that contains 0, that $r_1 > r_2 \Rightarrow \mu_K(r_1) \leq \mu_K(r_2)$. Then,

$$A = \frac{\int_0^\infty r^{n-1}e^{-a_{i+1}r^2}\mu_K(r)dr}{\int_0^\infty r^{n-1}e^{-a_ir^2}\mu_K(r)dr}.$$

Note $(r^{n-1}e^{-a_{i+1}r^2})/(r^{n-1}e^{-a_ir^2})$ is a monotonically increasing function in r. Since K is a convex body containing 0, we can partition K into infinitesimally small cones centered at 0. Consider an arbitrary cone C. $\mu_C(r)$ is 1 for $r \in [0, r']$ and then 0 for $r \in (r', \infty)$ since K is convex. Since $(r^{n-1}e^{-a_{i+1}r^2})/(r^{n-1}e^{-a_ir^2})$ is monotonically increasing, the integral over the cone only gets larger by extending $\mu_C(r)$ to be 1 for $r \in [0, \infty)$. Therefore

$$\frac{\int_0^\infty r^{n-1}e^{-a_{i+1}r^2}\mu_C(r)dr}{\int_0^\infty r^{n-1}e^{-a_ir^2}\mu_C(r)dr} \leq \frac{\int_0^\infty r^{n-1}e^{-a_{i+1}r^2}dr}{\int_0^\infty r^{n-1}e^{-a_ir^2}dr} = \sqrt{e}.$$

Since C was an arbitrary cone from a partition of A, we have that $A \leq \sqrt{e}$. □

Proof.(of Lemma 3.4.8). Note that

$$\begin{aligned}
f_{i+1}(x) &= \exp\left(-\frac{\|x\|^2}{2\sigma_{i+1}^2}\right) \\
&= \exp\left(-\frac{\|x\|^2}{2\sigma_i^2(1+\sigma_i^2/(C^2n))}\right) \\
&= \exp\left(-\frac{\|x\|^2}{2\sigma_i^2}\cdot\left(1-\frac{\sigma_i^2/(C^2n)}{1+\sigma_i^2/(C^2n)}\right)\right) \\
&= f_i(x)\cdot\exp\left(\frac{\|x\|^2}{2C^2n}\cdot\frac{1}{1+\sigma_i^2/(C^2n)}\right) \\
&\leq f_i(x)\cdot\exp\left(\frac{\|x\|^2}{2C^2n}\right).
\end{aligned}$$

We have that

$$
\begin{aligned}
M(Q_i, Q_{i+1}) &= \sup_{S \subseteq K} \frac{Q_i(S)}{Q_{i+1}(S)} \\
&\leq \frac{\int_K f_{i+1}(x)\,dx}{\int_K f_i(x)\,dx} \cdot \sup_{x \in K} \frac{f_i(x)}{f_{i+1}(x)} \\
&= \frac{\int_K f_{i+1}(x)\,dx}{\int_K f_i(x)\,dx} \cdot \sup_{x \in K} \left(\exp\left(-\frac{\|x\|^2}{2C^2 n} \cdot \frac{1}{1 + \sigma_i^2/(C^2 n)} \right) \right) \\
&= \frac{\int_K f_{i+1}(x)\,dx}{\int_K f_i(x)\,dx} \\
&\leq \frac{\int_K f_i(x) \exp\left(\|x\|^2/(2C^2 n)\right)\,dx}{\int_K f_i(x)\,dx} \\
&\leq \sup_{x \in K} \left(\exp\left(\frac{\|x\|^2}{2C^2 n} \right) \right) \\
&\leq \sqrt{e}
\end{aligned}
$$

since $\|x\| \leq C\sqrt{n}$. □

3.4.3 Wasted Steps

The speedy walk is defined as the *proper* steps of the ball walk, where the point the ball walk attempts to visit is contained in K. For convenience, we restate the definition of the speedy walk from earlier (Figure 3.2).

Speedy Walk(δ, f)
At current point $x \in K$:

1. Pick random point y from $K \cap (x + \delta B_n)$.
2. Go to y with probability $\min\{1, f(y)/f(x)\}$.

Figure 3.1: The Speedy walk with a Metropolis filter

To prove convergence of the ball with a Metropolis filter, we prove convergence of the speedy walk, then bound the number of "wasted" steps. Note that the speedy walk cannot be implemented as described in Figure 3.2, but is an analysis tool to prove the mixing time of the ball walk.

Next, we bound the average number of wasted steps of the ball walk, i.e., when the ball walk tries to visit a point not in K. The average local conductance of the ball walk is defined as

$$\lambda(f) = \frac{\int_K \ell(x) f(x)\, dx}{\int_K f(x)\, dx}.$$

We say that a density function $f : \mathbb{R}^n \to \mathbb{R}_+$ is a-*rounded* if any level set L contains a ball of radius $a \cdot \mu_f(L)$. We now show that the average local conductance is large, i.e. at least a constant.

Lemma 3.4.9. *For any a-rounded logconcave density function f in $\mathbb{R}^n$,*

$$\lambda(f) \geq 1 - 32\frac{\delta^{1/2} n^{1/4}}{a^{1/2}}.$$

Proof. Define $\hat{f}$ as the following smoothened version of f, obtained by convolving f with a ball of radius δ. Let D be a convex subset of δB_n of half its volume.

$$\hat{f}(x) = \min_D \frac{\int_{y \in x+D} f(y)\, dy}{\mathsf{vol}(D)}.$$

Now Lemma 6.3 from [31] shows that

$$\int_K \hat{f}(x)\,dx \geq 1 - 32\frac{\delta^{1/2}n^{1/4}}{a^{1/2}}.$$

To complete the proof, we observe that for any point x,

$$\ell(x)f(x) \geq \hat{f}(x).$$

To see this, note that

$$\begin{aligned}
\ell(x)f(x) &= f(x)\frac{\int_{x+\delta B_n} 1\,dy}{\mathsf{vol}(\delta B_n)} \\
&\geq \frac{\int_{y\in x+\delta B_n : f(y)\leq f(x)} f(y)\,dy}{\int_{y\in x+\delta B_n : f(y)\leq f(x)} 1\,dy} \\
&\geq \hat{f}(x).
\end{aligned}$$

□

Lemma 3.4.10. *The Gaussian $\mathcal{N}(0,\sigma^2 I)$ restricted to K containing a unit ball centered at 0 is $\min\{\sigma,1\}$-rounded.*

Proof. The level sets of the distribution are balls restricted to K. For the distribution to be $\min\{\sigma,1\}$-rounded, we need that a level set of measure k contains a ball of radius $k\cdot\min\{\sigma,1\}$. Consider the following function of t, which is an upper bound on the measure of the ball of radius $t \leq \min\{\sigma,1\}$, since the unit ball is contained in K:

$$g(t) = \frac{\int_0^t x^{n-1}\exp\left(-\frac{x^2}{2\sigma^2}\right)\,dx}{\int_0^{\min\{\sigma,1\}} x^{n-1}\exp\left(-\frac{x^2}{2\sigma^2}\right)\,dx}.$$

Consider the second derivative of g:

$$g''(t) = \left((n-1) - \frac{t^2}{\sigma^2}\right) \cdot \frac{t^{n-2} \exp\left(-\frac{t^2}{2\sigma^2}\right)}{\int_0^{\min\{\sigma,1\}} x^{n-1} \exp\left(-\frac{x^2}{2\sigma^2}\right)\, dx}.$$

For $g''(t)$ to be nonnegative, we need $\sigma^2(n-1) - t^2 \geq 0$, which it is for $n \geq 2, t \in [0, \min\{\sigma, 1\}]$. Since $g(0) = 0$, $g(\min\{\sigma, 1\}) = 1$, and the second derivative is nonnegative, we then have that $g(t \min\{\sigma, 1\}) \leq t$ for $t \in [0, 1]$, which proves the lemma.

□

We now show that for an appropriate selection of ball radius, the ball walk has large average local conductance.

Lemma 3.4.11. *If $\delta \leq \min\{\sigma, 1\}/(4096\sqrt{n})$, then the average local conductance, $\lambda(f)$, for the density function f proportional to the Gaussian $\mathcal{N}(0, \sigma^2 I_n)$ restricted to K containing the unit ball, is at least $1/2$.*

Proof. Using Lemma 3.4.9 and Lemma 3.4.10, we have that

$$\lambda(f) \geq 1 - 32\frac{\min\{\sigma^{1/2}, 1\} n^{1/4}}{64 n^{1/4} \min\{\sigma^{1/2}, 1\}} = \frac{1}{2}.$$

□

The following lemma is shown in [32].

Lemma 3.4.12. *If the average local conductance is at least λ, $M(Q_0, Q) \leq M$, and the speedy walk takes t steps, then the expected number of steps of the corresponding ball walk is at most Mt/λ.*

Proof. Since $M(Q_0, Q) \leq M$, we have that for all $S \subseteq K$,

$$Q_0(S) \leq MQ(S),$$

and by induction on i, we get that

$$Q_i(S) = \int_K P_x(S) dQ_{i-1}(x) \leq M \int_K P_x(S) dQ(x) = MQ(S).$$

For any point x, the expected number of steps until a proper step is made is $1/\ell(x)$. So, given a point from Q_i, the expected number of steps to obtain a point from Q_{i+1} is

$$\int_K \frac{1}{\ell(x)}\, dQ_i(x) \leq M \int_K \frac{1}{\ell(x)}\, dQ(x) = M \int_K \frac{1}{\lambda}\, d\hat{Q}(x) = \frac{M}{\lambda},$$

where $\hat{Q}$ is the corresponding distribution for the ball walk with a Metropolis filter (i.e., with stationary distribution proportional to $f(x)$). If the speedy walk took t steps, then by linearity of expectation, the expected number of steps for the ball walk is at most Mt/λ. □

3.4.4 Mapping Speedy Distribution to Target

When the speedy walk has converged, we obtain a point approximately from the speedy walk distribution $\ell(x)f(x)$. We will use a rejection routine to map a random point from this distribution to the target distribution $f(x)$ while incurring a small amount of additional sampling error. We adapt the proof of Theorem 4.16 of [5] to the Gaussian setting.

Lemma 3.4.13. *Assume that $\|P - \hat{Q}\|_{tv} \leq \varepsilon$, $B_n \subseteq K$, $\varepsilon \leq 1/10$, and*

$$\delta \leq \frac{\min\{\sigma, 1\}}{8\sqrt{n \log(n/\varepsilon)}}.$$

There is an algorithm that will use a constant number of random samples from P, in expectation, to obtain a distribution R satisfying $\|R - Q\|_{tv} \leq 10\varepsilon$.

Proof. The rejection routine is as follows: let $c = 1 - 1/(2n)$. For a point u from distribution $\ell(x)f(x)$, let $v = (1/c)u$. Accept v with probability $f(v)/f(u)$. Repeat until we accept a v.

The correctness of the above routine follows from the following two facts: (i) with constant probability, the rejection sampling will succeed and (ii) removing a thin shell around the boundary makes $\ell(x)$ look close to uniform on average.

Recall that $\hat{Q}$ is the speedy walk distribution and Q is the ball walk distribution. Consider a level set $\mu_L = \{x : f(x) \geq L\}$. By logconcavity of f, μ_L is convex. From [5], $\hat{Q}_{\mu_L}(c \cdot \mu_L) \geq 1/2$. By applying this to all level sets, it then follows that $\hat{Q}(cK) \geq 1/2$.

Also from [5], if μ_L contains the unit ball, then

$$\int_{\mu_L \cap cK} (1 - \ell(x))\ dx \leq \varepsilon \mathsf{vol}\left(\mu_L \cap cK\right).$$

Recall that the level sets μ_L are balls intersected with K since f is a spherical Gaussian distribution. If μ_L does not contain the unit ball, a standard calculation (using that $B_n \subseteq K$) shows that the local conductance is at least $1 - \varepsilon$ for every point, and thus

$$\int_{\mu_L \cap cK} (1 - \ell(x))\, dx \leq \varepsilon \mathsf{vol}(\mu_L \cap cK).$$

Using the above, we see that

$$\begin{aligned}
\int_{cK} (1 - \ell(x))\, f(x)\, dx &= \int_0^\infty \int_{\mu_L} (1 - \ell(x))\, \{x \in \mathbb{1}_{cK}\}\, dx\, dL \\
&\leq \int_0^\infty \varepsilon \mathsf{vol}(\mu_L \cap cK)\, dL \\
&\leq \varepsilon \int_{cK} f(x)\, dx.
\end{aligned}$$

Then,

$$\begin{aligned}\hat{Q}(cK) &= \frac{\int_{cK} \ell(x) f(x)\,dx}{\int_{K} \ell(x) f(x)\,dx} \\ &= \frac{\int_{cK} f(x)\,dx - \int_{cK} (1-\ell(x))\, f(x)\,dx}{\int_{K} \ell(x) f(x)\,dx} \\ &\geq \frac{\int_{cK} f(x)\,dx - \varepsilon \int_{cK} f(x)\,dx}{\int_{K} \ell(x) f(x)\,dx}\end{aligned}$$

and

$$\hat{Q}(cS) \leq \frac{\int_{cS} f(x)\,dx}{\int_{K} \ell(x) f(x)\,dx}.$$

Let P' be the distribution of the first sample from P which satisfies $(1/c)x \in K$. Define

$$z(x) = \begin{cases} f(cx) \text{ if } x \in K \\ 0 \text{ otherwise} \end{cases}$$

and let Z be the probability distribution corresponding to z. Then,

$$\begin{aligned}
P'(S) - Z(S) &= \frac{P(cS)}{P(cK)} - \frac{Q(cS)}{Q(cK)} \\
&\leq \frac{\hat{Q}(cS) + \varepsilon}{\hat{Q}(cK) - \varepsilon} - \frac{Q(cS)}{Q(cK)} \\
&\leq \frac{\displaystyle\int_{cS} f + \varepsilon \int_K \ell(x) f(x)\, dx}{\displaystyle\int_{cK} f - \varepsilon \int_{cK} f - \varepsilon \int_K \ell(x) f(x)\, dx} - \frac{\displaystyle\int_{cS} f(x)\, dx}{\displaystyle\int_{cK} f(x)\, dx} \\
&\leq \frac{1 + 2\varepsilon}{1 - 2\varepsilon} - 1 \\
&\leq 10\varepsilon.
\end{aligned}$$

Then accept a point x with probability $f(x)/z(x)$, which is at least a constant since $\|x\| \leq 4\sigma\sqrt{n}$. The overall expected number of rejection steps is a constant since $\hat{Q}(cK) \geq 1/2$. □

3.4.5 Proof of Sampling Theorems

We can now prove Theorem 3.1.2 and Theorem 3.1.1 for sampling a Gaussian distribution restricted to a convex body.

Proof.(of Theorem 3.1.1)

By Theorem 3.5.12 and Theorem 5.2.4, we have selecting $\delta = \min\{\sigma, 1\}/(4096\sqrt{n})$ implies that the speedy walk starting from a distribution that is M-warm will be within total variation distance ε of the target distribution in $O(\max\{\sigma^2, 1\} n^2 \log(n/\varepsilon) \log(M/\varepsilon))$ steps.

By Lemma 3.4.12, the ball walk will, in expectation, take at most $2M$ times as many steps since the average local conductance λ is at least $1/2$. Therefore, the total number of expected ball walk steps is $O(M \max\{\sigma^2, 1\} n^2 \log(n/\varepsilon) \log(M/\varepsilon))$. We then repeat this walk $O(1)$ times until we obtain a point from the proper target distribution using Lemma 3.4.13.

□

Proof.(of Theorem 3.1.2.)

Note that here, we are analyzing the sampling phases of Figure 4.1, and only the phases when $\sigma^2 \leq 1$.

By Theorem 3.1.1, we have that the ball walk will take $O(M \max\{\sigma^2, 1\}n^2 \log(n/\varepsilon) \log(M/\varepsilon))$ steps in expectation. By Lemma 3.4.7, each phase will always provide a warm start to the next, i.e. $M = O(1)$. By assigning a sampling error $(\varepsilon/n)^{16}$ to each phase, we ensure that the overall sampling failure is at most ε by a straightforward union bound. Therefore, each sampling phase takes

$$O\left(n^2 \log^2\left(\frac{n}{\varepsilon}\right)\right)$$

expected steps of the ball walk. Adding up across phases introduces an additional $n \log n$ factor since we increase σ^2 by the rate of $1 + 1/n$ between phases.

If we want to instead run for a fixed number of steps, we can keep a global counter of the ball walk steps. Say the expected number of ball walk steps is T. If at any point the number of ball walk steps goes above $2T$, we abandon this run of the algorithm. The probability of a single run failing is at most $1/2$ by Markov's inequality. If we want an overall failure probability of at most p, then we can run $\log(1/p)$ iterations of the algorithm, and with probability $1 - p$, at least one of them will succeed. □

Proof.(of Theorem 3.1.3) The proof of Theorem 3.1.3, which extends Gaussian sampling to uniform sampling, follows along the same lines as Theorem 3.1.2. When $\sigma^2 \leq 1$, the total expected ball walk steps is

$$O\left(n^3 \log(n) \log^2\left(\frac{n}{\varepsilon}\right)\right).$$

When $\sigma^2 > 1$, we additionally use Lemma 3.4.8, which implies that we can accelerate our cooling rate and still maintain a warm start. This accelerated rate allows us to overcome the increased mixing time of $O^*(\max\{\sigma^2, 1\}n^2)$ once $\sigma^2 \geq 1$. Now consider a "chunk" of phases

as a set of phases until σ^2 doubles. There will be $O(C^2 n/\sigma^2)$ phases in a chunk, where each chunk has expected mixing time $O(\sigma^2 n^2 \log(n/\varepsilon))$. Since there are $O(\log n)$ chunks (provided $C = \text{poly}(n)$), the total number of expected ball walk steps when $\sigma^2 > 1$ is

$$O\left(C^2 n^2 \log(n) \log\left(\frac{n}{\varepsilon}\right)\right).$$

Note that this will yield a random sample with respect to a Gaussian with $\sigma^2 = C^2 n$ restricted to K. We can map this point to a uniform random point using simple rejection sampling, which will succeed with probability at least $1/e$ since $K \subseteq C\sqrt{n}$. If it fails, we can restart the algorithm. As with Theorem 3.1.2, we can repeat $\log(1/p)$ times to transform the expected ball walk steps into a fixed number of steps with success probability $1 - p$.

□

3.5 Logconcave Sampling

Here we extend the Gaussian sampling to logconcave functions $f : \mathbb{R}^n \to \mathbb{R}$, provided f is near-isotropic and $f\gamma$ is well-rounded, where γ is the standard Gaussian distribution $\mathcal{N}(0, I)$.

Theorem 3.5.1. *Let $f : \mathbb{R}^n \to \mathbb{R}_+$ be a logconcave function. Let Q_i be the distribution after i steps of the speedy walk and let Q be the target distribution. The target distribution Q is the Gaussian distribution $\mathcal{N}(0, \sigma^2 I)$ multiplied by ℓ_f. For any $\varepsilon > 0$, the lazy speedy walk with δ-steps for $\delta = \min\{\sigma, 1\}/(4096\sqrt{n})$ starting from Q_0 satisfies $d_{tv}(Q_t, Q) \leq \varepsilon$ after*

$$t \geq C \cdot \max\{\sigma^2, 1\} n^2 \log\left(\frac{n}{\varepsilon}\right) \log\left(\frac{M(Q_0, Q)}{\varepsilon}\right)$$

steps of the speedy walk, for some absolute constant C.

3.5.1 Isoperimetric Inequality

For the case of general logconcave sampling and integration, we still use the isoperimetric inequality in Theorem 3.3.4, but our proof requires a slightly stronger isoperimetry. We will also use the following isoperimetry, which we state and prove below. Note the distinction between f and ℓ_f in the theorem statement.

Theorem 3.5.2. *Let $f : \mathbb{R}^n \to \mathbb{R}_+$ be a logconcave function with maximum M_f and define*

$$K = \left\{x : \|x\| \leq t\sqrt{n} \text{ and } f(x) > e^{-2(n-1)-2t} M_f\right\}$$

with $t \leq n$. Let π be the Gaussian distribution $\mathcal{N}(0, \sigma^2 I_n)$ multiplied by ℓ_f restricted to K. Let S_1, S_2, S_3 partition K such that for any $u \in S_1, v \in S_2$,

$$\frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}} \geq \frac{8dn}{\sigma}.$$

Then,

$$\pi(S_3) \geq \frac{d}{\sigma}\pi(S_1)\pi(S_2).$$

Proof. The theorem essentially follows from an isoperimetric inequality of cross-ratio distance. We see that

$$\begin{aligned}
\pi(S_3) &\geq d_K(S_1, S_2)\pi(S_1)\pi(S_2) && \text{by Theorem 2.5 of [33]} \\
&\geq \frac{1}{6n + 2t} d_f(S_1, S_2)\pi(S_1)\pi(S_2) && \text{by Lemma 6.14 of [33]} \\
&\geq \frac{1}{8n} \min_{u \in S_1, v \in S_2} \frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}} \pi(S_1)\pi(S_2) && \text{by Lemma 3.5.3} \\
&\geq \frac{d}{\sigma}\pi(S_1)\pi(S_2)
\end{aligned}$$

as desired. □

Lemma 3.5.3. *Let $f : \mathbb{R}^n \to \mathbb{R}_+$ be a logconcave function. Then for any two points $u, v \in \mathbb{R}^n$,*

$$d_f(u, v) \geq \frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}}.$$

Proof. Assume without loss of generality that $f(u) \leq f(v)$. Then,

$$\frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}} = 1 - \frac{f(u)}{f(v)}.$$

Recall that

$$d_f(u, v) = \frac{f(u, v)f(\ell(u, v))}{f^-(u, v)f^+(u, v)}.$$

We have that

$$f(u, v) \geq f(u) \cdot \|u - v\|,$$

and also

$$f^-(u, v) \leq f(u) \cdot \|u - v\| \cdot \frac{1}{1 - f(u)/f(v)}.$$

Therefore,

$$d_f(u, v) \geq \left(1 - \frac{f(u)}{f(v)}\right) \cdot \frac{f(\ell(u, v))}{f^+(u, v)} \geq 1 - \frac{f(u)}{f(v)}.$$

□

3.5.2 Filtered Speedy Walk

Whereas for Gaussian sampling we used the unfiltered speedy walk, for logconcave sampling we incorporate the function f into the speedy walk. Essentially, this requires that all null steps of the speedy walk are due to the Gaussian filter, which has a constant probability of acceptance. By incorporating f into the speedy walk, we can then bound the wasted steps on average.

Here we prove some basic properties of the filtered speedy walk.

Define the function

$$\ell_f(x) = \int_{x+\delta B_n} \min\{f(x), f(z)\}\, dz.$$

Define the speedy walk as follows.

Speedy Walk(f, g, x)

1. Select y from $x + \delta B_n$ with probability proportional to $\min\{f(y), f(x)\}$.
2. Go to y with probability $\min\{1, g(y)/g(x)\}$.

Figure 3.2: The speedy walk with respect to f and Gaussian weighting g.

Lemma 3.5.4. *The lazy speedy walk converges to a distribution proportional to $\ell_f(x)g(x)$.*

Proof. We show that the walk is time reversible with respect to the distribution $g(x)\ell_f(x)$, i.e. $g(x)\ell_f(x)\Pr(x \to y) = g(y)\ell_f(y)\Pr(y \to x)$ for all $x, y \in \mathbb{R}^n$.

From Figure 3.2, we have that

$$\Pr(x \to y) = \begin{cases} \dfrac{\min\{f(y), f(x)\}}{\displaystyle\int_{x+\delta B_n} \min\{f(x), f(z)\}\, dz} \cdot \min\{1, \tfrac{g(y)}{g(x)}\} & \text{if } \|x - y\| \le \delta \\ 0 & \text{otherwise} \end{cases}$$

So, if $\|x - y\| > \delta$, it is clearly reversible (both sides are zero). If $\|x - y\| \leq \delta$, then

$$g(x)\ell_f(x)\Pr(x \to y) = g(x)\ell_f(x) \cdot \left(\frac{\min\{f(y), f(x)\}}{\displaystyle\int_{x+\delta B_n} \min\{f(x), f(z)\}\, dz} \cdot \min\left\{1, \frac{g(y)}{g(x)}\right\} \right)$$

$$= \min\{g(x), g(y)\} \cdot \min\{f(x), f(y)\}$$
$$= g(y)\ell_f(y)\Pr(y \to x).$$

□

Lemma 3.5.5. *The function ℓ_f is logconcave.*

Proof. Define the function $G(x, z) : \mathbb{R}^{2n} \to \mathbb{R}_+$ as $G(x, z) = \min\{f(x), f(z)\}$. Then, define $G_1(x, z) = f(x)$ and $G_2(x, z) = f(z)$. Since f is logconcave, G_1 and G_2 are clearly logconcave.

So $G(x, z) = \min\{G_1(x, z), G_2(x, z)\}$ is the minimum of two logconcave functions, and therefore it is logconcave as a function of (x, z). Then define

$$G_\delta(x, z) = \begin{cases} G(x, z) & \text{if } \|x - z\| \leq \delta \\ 0 & \text{otherwise} \end{cases}$$

Also define $h : \mathbb{R}^{2n} \to \mathbb{R}_+$ as

$$h(x, z) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$$

It is easy to see that both h and G_δ are logconcave. The logconcavity of h follows immediately from the definition, and the fact that G_δ is logconcave follows from the triangle inequality. Then

we have that

$$\begin{aligned} G_\delta * h(x,z) &= \int_{(t_1,t_2)\in R^{2n}} G_\delta(x-t_1, z-t_2)h(t_1,t_2)\,dt \\ &= \int_{t_2\in\mathbb{R}^n} G_\delta(x, z-t_2)h(0,t_2)\,dt \\ &= \int_{y\in x+\delta B_n} G_\delta(x,y)\,dy \\ &= \ell_f(x). \end{aligned}$$

So we have that ℓ_f is the convolution of two logconcave functions and therefore also logconcave by Theorem 2.4.1.

□

3.5.3 One Step Overlap

The following is the 1-step distribution according to the speedy walk in Figure 3.2.

$$\Pr(x \to y) = \begin{cases} \dfrac{\min\{f(y), f(x)\}}{\displaystyle\int_{x+\delta B_n} \min\{f(x), f(z)\}\,dz} \cdot \min\{1, \frac{g(y)}{g(x)}\} & \text{if } \|x-y\| \le \delta \\ \\ 0 & \text{otherwise} \end{cases}$$

For a point $u \in \mathbb{R}^n$ and $S \subseteq \mathbb{R}^n$, denote the probability that u goes to S after 1 step of the speedy walk as

$$P_u(S) = \int_S P(u \to x)\,dx. \tag{3.4}$$

In this section we develop a coupling argument which argues that if two points are "close", then there is a constant probability that they are equivalent after a step of the speedy walk. By

close, we mean both the Euclidiean and functional distance between the two points is small.

To prove the one step overlap, we use an interesting geometric construction that allows us to apply an isoperimetric inequality in terms of cross-ratio distance.

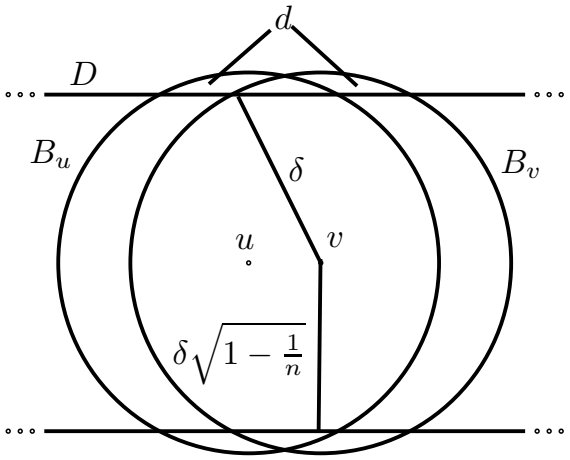


Figure 3.3: The basic situation for the 1-step overlap. We want to argue that under appropriate conditions, the measure of $B_u \cap B_v$ is a constant fraction of the smaller of B_u, B_v.

The following construction is visualized in Figure 3.3. Suppose that $\|u - v\| = d$. Assume without loss of generality that $v = u + (d, 0, 0, \ldots, 0)$. Define $B_u = u + \delta B_n$, $B_v = v + \delta B_n$, and $C = B_u \cap B_v$. The end goal is to show that the mass of C is large with respect to the smaller of B_u and B_v.

Let D be an infinite cylinder with radius $\delta\sqrt{1 - 1/n}$ spanning the x_1 axis, i.e.

$$D = \left\{ x = (x_1, x_2, \ldots, x_n) \,\middle|\, \sqrt{x_2^2 + x_3^2 + \ldots x_n^2} \leq \delta\sqrt{1 - \frac{1}{n}} \right\}.$$

Then let $S_u = B_u \cap D$, $S_v = B_v \cap D$, and $S_C = C \cap D$ (Figure 3.4). The following lemma is easy to verify.

Lemma 3.5.6. *Let $K_S = S_u \cup S_v \cup S_C$. For $d \leq 2\delta/\sqrt{n}$, K_S is convex and $S_u \backslash S_C, S_v \backslash S_C, S_C$ are a partition of K_S.*

The following lemma says that the measure of S_u is close to the measure of B_u (similarly for

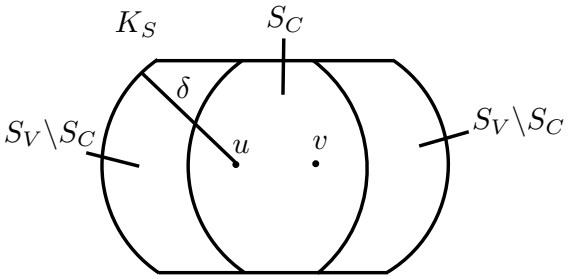


Figure 3.4: The convex body K_S divided into three parts. The point here is that we made the region convex, while also creating some distance between the two sets, by ignoring troublesome small regions.

B_v).

Lemma 3.5.7. *For S_u, B_u as defined above,*

$$\int_{S_u} \min\{f(u), f(z)\}\, dz \geq \frac{1}{2} \int_{B_u} \min\{f(u), f(z)\}\, dz.$$

Proof. Define

$$T = u + \delta \sqrt{1 - \frac{1}{n}} \cdot B_n.$$

We first show that the measure of T is close to the measure of B_u. Consider an arbitrary infinitesi-

mal cone R with tip at u. We then have that

$$\frac{\int_{R\cap T} \min\{f(u), f(z)\}\, dz}{\int_{R\cap B_u} \min\{f(u), f(z)\}\, dz} = \frac{\int_0^{\delta\sqrt{1-1/n}} t^{n-1} \min\{f(u), f(t)\}\, dt}{\int_0^{\delta} t^{n-1} \min\{f(u), f(t)\}\, dt}$$

$$= \frac{\int_0^{\delta} s^{n-1} \left(1-\frac{1}{n}\right)^{(n+1)/2} \min\{f(u), f\left(s\sqrt{1-\frac{1}{n}}\right)\}\, ds}{\int_0^{\delta} t^{n-1} \min\{f(u), f(t)\}\, dt} \qquad \text{where } t = s\sqrt{1-\frac{1}{n}}$$

$$\geq \left(1-\frac{1}{n}\right)^{(n+1)/2}$$

$$\geq \frac{1}{2}.$$

In the second to last step, we used the fact that $\min\{f(u), f(t)\}$ is monotonically non-increasing with respect to t (this follows from the logconcavity of f). Since R was an arbitrary cone, we then have that

$$\int_T \min\{f(u), f(z)\}\, dz \geq \frac{1}{2}\int_{B_u} \min\{f(u), f(z)\}\, dz.$$

Then observe that $T \subseteq S_u$, which proves the lemma.

□

The below lemma states that the cross-ratio distance is at least a constant. Note that a horizontal chord can attain constant cross-ratio distance, so the lemma is tight up to constants.

Lemma 3.5.8. *Let $S'_u = S_u \backslash S_C$ and $S'_v = S_v \backslash S_C$. Suppose that*

$$d = \|u - v\| \leq \frac{\delta}{2\sqrt{n}}.$$

Then,

$$d_K(S'_u, S'_v) \geq \frac{8}{9}.$$

Proof.

Shift the picture so that $v = (0, 0, \ldots, 0)$ and $u = (-\delta/\sqrt{n}, 0, 0, \ldots, 0)$. Then consider an arbitrary chord L which intersects both S'_u and S'_v. Let a be the left endpoint of L, i.e. of the two locations where L intersects the boundary of S'_u, let a denote the point which has minimum x_1 value. Let c denote the unique point along L that has $x_1 = 0$. Then, let b denote the point along L which intersects B_v and is between a and c.

So we have $b = ta + (1 - t)c$ for some $t \in [0, 1]$. The goal now is to show that b is not too close to c, i.e. get a lower bound on t. After we lower bound t, we will then show how it implies a lower bound on the cross-ratio distance.

From our construction, we can simplify the above to a 3-dimensional problem. Note that the construction is symmetric about the x_1 axis. Therefore, without loss of generality, we can apply any rotation orthogonal to the x_1 axis. Thus, we may assume that $c = (0, c_2, 0, \ldots, 0)$ by rotation (recall that $c_1 = 0$ from definition). Once we fix $c = (0, c_2, 0, \ldots, 0)$, by a similar argument, we may rotate again orthogonal to both the x_1 and x_2 axes until $a = (a_1, a_2, a_3, 0, \ldots, 0)$. We may further assume that $c_2 \geq 0$ without loss of generality.

We now have the following constraints:

$$\begin{aligned} c_2 &\leq \delta\sqrt{1 - \frac{1}{n}} \\ a_1 &\leq -\frac{2\delta}{\sqrt{n}} \\ \left(a_1 + \frac{\delta}{\sqrt{n}}\right)^2 + a_2^2 + a_3^2 &= \delta^2 \\ b_1^2 + b_2^2 + b_3^2 &= \delta^2 \end{aligned}$$

Expanding the first quadratic constraint, we observe that

$$a_1^2 + a_2^2 + a_3^2 = \delta^2 - \frac{2a_1\delta}{\sqrt{n}} - \frac{\delta^2}{n}. \tag{3.5}$$

From the definition of b, note that

$$\begin{aligned} b_1^2 + b_2^2 + b_3^2 &= t^2 a_1^2 + (ta_2 + (1-t)c_2)^2 + t^2 a_3^2 \\ &= t^2 a_1^2 + t^2 a_2^2 + 2t(1-t)a_2 c_2 + (1-t)^2 c_2^2 + t^2 a_3^2 \\ &= t^2 \left(a_1^2 + a_2^2 + a_3^2 + \frac{2(1-t)a_2 c_2}{t} + \frac{(1-t)^2 c_2^2}{t^2} \right). \end{aligned} \tag{3.6}$$

We now show that the above implies a lower bound on t. Combining (3.5) and (3.6), we have that

$$\begin{aligned} \delta^2 &= t^2 \left(\delta^2 - \frac{2a_1\delta}{\sqrt{n}} - \frac{\delta^2}{n} + \frac{2(1-t)a_2 c_2}{t} + \frac{(1-t)^2 c_2^2}{t^2} \right) \\ &= t^2 \left(\delta^2 (1 - \frac{1}{n}) - \frac{2a_1\delta}{\sqrt{n}} - 2a_2 c_2 + c_2^2 \right) + t\left(2a_2 c_2 - 2c_2^2\right) + c_2^2. \end{aligned}$$

Define

$$f(t) = t^2 \left(\delta^2 \left(1 - \frac{1}{n} \right) - \frac{2a_1\delta}{\sqrt{n}} - 2a_2 c_2 + c_2^2 \right) + t\left(2a_2 c_2 - 2c_2^2\right).$$

We have that

$$f(t) = \delta^2 - c_2^2$$

and

$$\begin{aligned} f'(t) &= 2t\left(\delta^2\left(1-\frac{1}{n}\right) - \frac{2a_1\delta}{\sqrt{n}} - 2a_2c_2 + c_2^2\right) + \left(2a_2c_2 - 2c_2^2\right) \\ &= 2t\left(\delta^2\left(1-\frac{1}{n}\right)\right) - \frac{4ta_1\delta}{\sqrt{n}} + (2-4t)a_2c_2 + (2t-2)c_2^2. \end{aligned}$$

Since f' is linear, we have that for $y \in [0, 1/4]$, $f'(y) \leq \max\{f'(0), f'(1/4)\}$. We now bound $f'(0)$ and $f'(1/4)$ separately:

$$\begin{aligned} f'(0) &= 2a_2c_2 - 2c_2^2 \\ &\leq 2\delta^2 - 2c_2^2, \end{aligned}$$

and

$$\begin{aligned} f'\left(\frac{1}{4}\right) &= \frac{1}{2}\left(\delta^2\left(1-\frac{1}{n}\right) - \frac{2a_1\delta}{\sqrt{n}} - 2a_2c_2 + c_2^2\right) + 2a_2c_2 - 2c_2^2 \\ &= \frac{\delta^2}{2}\left(1-\frac{1}{n}\right) - \frac{3c_2^2}{2} - \frac{2a_1\delta}{\sqrt{n}} + a_2c_2 \\ &\leq \frac{\delta^2}{2}\left(1-\frac{1}{n}\right) - \frac{3c_2^2}{2} + \delta^2\sqrt{1+\frac{4}{n}} \qquad \text{since } a_1^2 + a_2^2 \leq \delta^2 \\ &\leq \frac{\delta^2}{2}\left(1-\frac{1}{n}\right) - \frac{3c_2^2}{2} + \delta^2\left(1+\frac{2}{n}\right) \\ &\leq \frac{3\delta^2}{2} - \frac{3c_2^2}{2} + \frac{3\delta^2}{n} \\ &\leq 3\delta^2 - 3c_2^2, \end{aligned}$$

where in the last line we used that $\delta^2 - c_2^2 \geq \delta^2/n$. For $y \in [0, 1/4]$, we then have

$$\begin{aligned} f(y) &= f(0) + \int_0^y f'(x)\,dx \\ &= \int_0^y f'(x)\,dx \\ &\leq \int_0^y 3(\delta^2 - c_2^2)\,dx \\ &= 3y(\delta^2 - c_2^2). \end{aligned}$$

Recall that $f(t) = \delta^2 - c_2^2$. Thus we have that $f(y) < f(t)$ for all $y \in [0, 1/4]$, which implies that $t \geq 1/4$ since we know our desired solution is when $t \geq 0$.

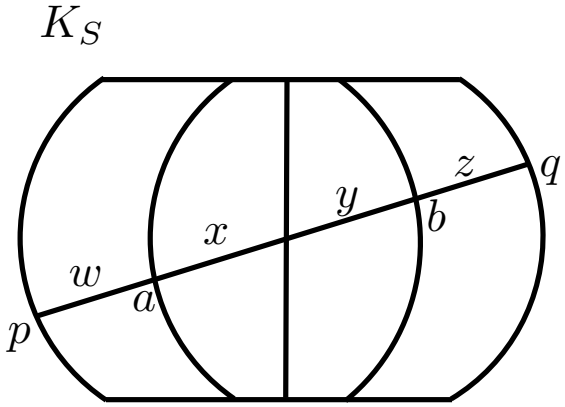


Figure 3.5: We have lower bounds on x/w and y/z. This gives a lower bound on the cross ratio distance.

The lower bound on t then gives an upper bound on the cross ratio distance. Using the notation

as in Figure 3.5, $t \geq 1/4$ implies that $x/w, y/z \geq 1/3$, and thus

$$\begin{aligned} d_K(S'_u, S'_v) &= \frac{|a-b||p-q|}{|p-a||b-q|} \\ &= \frac{(x+y)(w+x+y+z)}{w \cdot z} \\ &\geq \frac{(w/3+z/3)(4w/3+4z/3)}{w \cdot z} \\ &= \frac{4w^2/9 + 8wz/9 + 4z^2/9}{wz} \\ &\geq \frac{8}{9}. \end{aligned}$$

□

Lemma 3.5.9. *Let B_u, B_v, C be defined as above and $|f(u) - f(v)|/\max\{f(u), f(v)\} \leq 1/4$. Then*

$$\int_C \min\{f(u), f(v), f(z)\}\, dz \geq \frac{9}{16} \min\left\{ \int_{B_u} \min\{f(u), f(z)\}\, dz, \int_{B_v} \min\{f(v), f(z)\}\, dz \right\}.$$

Proof. Assume without loss of generality that $f(u) \leq f(v)$, which implies that $f(v) < 4f(u)/3$.

We first show a simpler inequality. Note that $\min\{f(v), f(z)\}$ is a logconcave function, and let π denote its associated measure. We can then apply an isoperimetric inequality for cross-ratio distance. By Theorem 2.5 in [33] and Lemma 3.5.6, we have that

$$\pi(S_C) \geq d_K(S'_u, S'_v) \min\{\pi(S'_u), \pi(S'_v)\}.$$

After applying Lemma 3.5.7 and Lemma 3.5.8, the lemma follows.

□

Lemma 3.5.10. *Let $S, \overline{S}$ be a partition of $\mathbb{R}^n$ with $u \in S$ and $v \in \overline{S}$. Suppose*

$$d(u,v) < \frac{\delta}{\sqrt{n}} \qquad \textit{and} \quad \frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}} < \frac{1}{8} \qquad \textit{and} \quad \frac{|\ell_f(u) - \ell_f(v)|}{\max\{\ell_f(u), \ell_f(v)\}} < \frac{1}{8}.$$

Let P_u^f denote a probability distribution with density at z equal to

$$\begin{cases} \min\{f(u), f(z)\} & \textit{if } \|u - z\| \leq \delta \\ 0 & \textit{otherwise} \end{cases}$$

Then,

$$P_u^f(\overline{S}) + P_v^f(S) > \frac{1}{3}.$$

Proof. Assume without loss of generality that $\ell_f(v) \leq \ell_f(u)$. Then by assumption, we have that $\ell_f(u) \leq 8\ell_f(v)/7$.

We have that

$$P_u^f(\overline{S}) + P_v^f(S) \geq \frac{\int_{C \cap \overline{S}} \min\{f(u), f(z)\}\, dz}{\ell_f(u)} + \frac{\int_{C \cap S} \min\{f(v), f(z)\}\, dz}{\ell_f(v)}$$

$$\geq \frac{7}{8} \cdot \frac{\int_{C \cap \overline{S}} \min\{f(u), f(z)\}\, dz + \int_{C \cap S} \min\{f(v), f(z)\}\, dz}{\ell_f(v)}$$

$$\geq \frac{7 \int_C \min\{f(u), f(v), f(z)\}\, dz}{8\ell_f(v)}$$

$$\geq \frac{63}{128}. \qquad \text{by Lemma 3.5.9}$$

□

Corollary 3.5.11. *Let $S, \overline{S}$ be a partition of $\mathbb{R}^n$ with $u \in S$ and $v \in \overline{S}$. Suppose*

$$d(u,v) < \frac{\delta}{\sqrt{n}} \qquad \textit{and} \quad \frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}} < \frac{1}{8} \qquad \textit{and} \quad \frac{|\ell_f(u)\gamma(u) - \ell_f(v)\gamma(v)|}{\max\{\ell_f(u)\gamma(u), \ell_f(v)\gamma(v)\}} < \frac{1}{10}.$$

Let P_u be as defined in Equation (3.4)*. Then,*

$$P_u(\overline{S}) + P_v(S) > \frac{1}{3e}.$$

Proof. By applying Lemma 3.4.3, we satisfy the conditions of Lemma 3.5.10. By Lemma 3.4.2, the Gaussian filter can only decrease the probability of crossing over by at most a $1/e$ factor. □

3.5.4 Conductance

Theorem 3.5.12. *Let $f : \mathbb{R}^n \to \mathbb{R}_+$ be a logconcave function and let γ denote the Gaussian density corresponding to $\mathcal{N}(0, \sigma^2 I)$. The conductance of the speedy walk with target distribution proportional to $\ell_f \cdot \gamma$ and $\delta \leq \sigma/50\sqrt{n}$ steps is $\Omega(\frac{\delta}{\sigma\sqrt{n}})$.*

Proof. Let

$$h(x) = \frac{\ell_f(x)\gamma(x)}{\displaystyle\int_{\mathbb{R}^n} \ell_f(x)\gamma(x)\,dx},$$

and let π be the associated measure of h.

Let $S \subset \mathbb{R}^n$ be an arbitrary measurable subset of $\mathbb{R}^n$ and let $\overline{S} = \mathbb{R}^n \backslash S$. Assume that $\pi(S) \leq$

$1/2$. Consider the following partition of $\mathbb{R}^n$:

$$S_1 = \left\{ x \in S : P_x(\overline{S}) < \frac{1}{20e} \right\}$$

$$S_2 = \left\{ x \in \overline{S} : P_x(S) < \frac{1}{20e} \right\}$$

$$S_3 = \mathbb{R}^n \backslash S_1 \backslash S_2.$$

We may assume that $\pi(S_1) \geq \pi(S)/2$ and $\pi(S_2) \geq \pi(\overline{S})/2$. If not, we can bound the conductance of S as follows (similarly for $\overline{S}$).

$$\begin{aligned}
\phi(S) &= \frac{\int_S P_x(\overline{S}) h(x)\, dx}{\pi(S)} \\
&= \frac{1}{2} \frac{\int_S P_x(\overline{S}) h(x)\, dx + \int_{\overline{S}} P_x(S) h(x)\, dx}{\pi(S)} \\
&\geq \frac{1}{2} \frac{\int_{S_3} \frac{h(x)}{20e}\, dx}{\pi(S)} \\
&= \frac{1}{40e} \frac{\pi(S_3)}{\pi(S)} \\
&\geq \frac{1}{80e}.
\end{aligned}$$

By Lemma 3.5.10, we have that for any $u \in S_1, v \in S_2$, either

$$\|u - v\| \geq \frac{\delta}{\sqrt{n}} \quad \text{or} \quad \frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}} \geq \frac{1}{8} \quad \text{or} \quad \frac{|\ell_f(u)\gamma(u) - \ell_f(v)\gamma(v)|}{\max\{\ell_f(u)\gamma(u), \ell_f(v)\gamma(v)\}} \geq \frac{1}{10}.$$

We now prove a lower bound on the conductance by applying both Theorem 3.3.4, and Theo-

rem 3.5.2. By Theorem 3.3.4, we have that if

$$\|u - v\| \geq \frac{\delta}{\sqrt{n}} \qquad \text{or} \quad \frac{|\ell_f(u)\gamma(u) - \ell_f(v)\gamma(v)|}{\max\{\ell_f(u)\gamma(u), \ell_f(v)\gamma(v)\}} \geq \frac{1}{10}$$

then

$$\pi(S_3) \geq \min\left\{\frac{\delta \ln 2}{\sigma\sqrt{n}}, \frac{2}{5\sqrt{n}}\right\} \pi(S_1)\pi(S_2).$$

By Theorem 3.5.2, if

$$\frac{|f(u) - f(v)|}{\max\{f(u), f(v)\}} \geq \frac{1}{8},$$

then

$$\pi(S_3) \geq \frac{1}{64n}\pi(S_1)\pi(S_2).$$

Combining the above and using that $\delta \leq \sigma/(50\sqrt{n})$, we see that

$$\pi(S_3) \geq \frac{\delta \ln 2}{\sigma\sqrt{n}}\pi(S_1)\pi(S_2).$$

This then bounds the conductance as

$$\begin{aligned}
\phi(S) &\geq \frac{1}{80e}\frac{\pi(S_3)}{\pi(S_1)} \\
&\geq \frac{\delta \ln 2 \cdot \pi(S_1)\pi(S_2)}{80e \cdot \sigma\sqrt{n}\pi(S_1)} \\
&\geq \frac{\delta \ln 2}{320e \cdot \sigma\sqrt{n}},
\end{aligned}$$

which proves the theorem. □

We can now bound the mixing time of the speedy walk. Recall Theorem 3.5.1.

Proof. By Theorem 3.5.12, we know that the conductance of the speedy walk ϕ_{SW} satisfies

$$\phi_{SW} = \Omega\left(\frac{\delta}{\sigma\sqrt{n}}\right).$$

By Theorem 5.2.4, we know that after t steps of the speedy walk starting from a distribution Q_0, we have $d_{tv}(Q_t, Q) \leq \varepsilon$ after

$$t \geq \phi_{SW}^{-2} \ln\left(\frac{\sqrt{M(Q_0, Q)}}{\varepsilon}\right)$$

steps. Selecting

$$\delta = \frac{\min\{1, \sigma\}}{4096\sqrt{n}}$$

then proves the theorem. □

3.5.5 Wasted Steps

In this section, we show that the speedy walk (Figure 3.2) can be efficiently implemented by its corresponding ball walk (Figure 2.2). We emphasize that the speedy walk is an analysis tool for analyzing the mixing time of the ball walk and cannot be implemented directly. The ball walk, however, is easy to implement, but it is difficult to directly analyze. Here we show that in expectation, the number of ball walk steps is at most a constant factor times the number of speedy walk steps, provided that $f\gamma$ is well-rounded

Recall the definition of ℓ_f:

$$\ell_f(x) = \frac{1}{\mathsf{vol}(\delta B_n)} \int_{x+\delta B_n} \min\{f(x), f(z)\}\, dz.$$

Note the lack of the Gaussian function from the definition of ℓ_f. This choice is for technical reasons, such as for isoperimetry where we need the speedy walk to sample from a Gaussian times

a logconcave distribution. In this section, we will analyze the wasted steps of f. The Gaussian distribution also can constribute wasted steps, but we choose δ so that there is always a constant probability that the Gaussian Metropolis filter accepts (Lemma 3.4.2).

When we say the "local conductance" $\lambda(x)$ of a point $x \in \mathbb{R}^n$, we are referring to the probability that the filter of f accepts if we choose a point uniformly at random from $x + \delta B_n$, i.e.

$$\lambda(x) = \frac{\ell_f(x)}{f(x)} = \frac{\int_{x+\delta B_n} \min\{1, \frac{f(z)}{f(x)}\}\, dz}{\mathsf{vol}(\delta B_n)}.$$

We now show that the local conductance is high on average. We will use the following lemma from [5], which bounds the average local conductance with respect to the uniform distribution over a convex body.

Lemma 3.5.13. *[5] Let K be a convex body containing a ball of radius r. Let X be a random point in K and let Z be a uniform random point in $X + \delta B$. Then*

$$\Pr(Z \notin K) \leq \frac{\delta\sqrt{n}}{2r}.$$

Lemma 3.5.14. *Let h be a logconcave function, and suppose that the level set of h with measure $1/8$ contains the the unit ball. Then the average local conductance with respect to $h(x)$ is at least $1/2$ for $\delta \leq 1/4\sqrt{n}$. More precisely,*

$$\frac{\int \ell_h(x)\, dx}{\int h(x)\, dx} \geq \frac{1}{2}.$$

Proof. Here we will adapt the proof of Lemma 6.3 from [33].

Consider selecting a random point X from the distribution corresponding to f, and then selecting a point Z uniformly at random from a ball of radius δ around X. According to the Metropo-

lis filter, we accept the point Z with probability $\min\{1, f(Z)/f(X)\}$. We can also look at the Metropolis filter from the following perspective. Select a number T uniformly at random from the interval $[0, f(X)]$. Then, accept the point Z if $T \leq f(Z)$. Therefore, the average local conductance is

$$\mathsf{E}_{X\sim f}\left(\ell(X)\right) = \Pr_{X,Z,T}\left(T \leq f(Z)\right).$$

We now estimate the probability that $T > f(Z)$. First, fix some t such that $\pi_h(L_h(t)) = 1/8$, and let T be chosen from its marginal distribution with respect to h. Then, select X uniformly from the level set $L_h(T)$ and choose Z uniformly in a ball around X.

First, observe that

$$\Pr(T \geq t) \leq \Pr(f(X) \geq t) = \pi_h(L_h(t)) = \frac{1}{8}. \tag{3.7}$$

We then estimate the probability that $t \geq T > f(Z)$. If $T \leq t$, then $\pi_h(L_h(T)) \geq 1/8$ and thus $L_h(T)$ contains a ball of radius 1. Therefore by Lemma 3.5.13, $\Pr(Z \notin L_h(T)) \leq \delta\sqrt{n}/2$, and thus

$$\Pr\left(t \geq T \geq f(Z)\right) \leq \frac{\delta\sqrt{n}}{2}. \tag{3.8}$$

Combining (3.7), (3.8), we have

$$\begin{aligned}\Pr\left(T \geq f(Z)\right) &\leq \Pr(T \geq t) + \Pr(t \geq T > f(Z)) \\ &\leq \pi_h(L_h(t)) + \frac{\delta\sqrt{n}}{2}.\end{aligned}$$

Recall that t was chosen such that $\pi_h(L_h(t)) = 1/8$, and choosing $\delta = 1/4\sqrt{n}$ gives

$$\Pr\left(T > f(Z)\right) \leq \frac{1}{4}.$$

□

3.6 Future Work and Open Questions

One of the biggest and most fundamental open questions in high-dimensional sampling is the KLS conjecture(Section 2.4.4), which bounds the mixing time of the ball walk for an isotropic logconcave function. We list below some more open questions in this area, each of which gives algorithmic improvements for sampling or useful theoretical insights.

1. What is the mixing time of hit-and-run for the standard Gaussian distribution $\mathcal{N}(0, I)$ restricted to a convex body K containing the unit ball? To be more precise, we would like an analogous isoperimetric inequality to Theorem 3.3.4 that would imply hit-and-run mixes in $O^*(n^2)$ steps from a warm start for Gaussians. The current best mixing time for hit-and-run is $O^*(n^3)$ [10] for any well-rounded logconcave function, proven via an isoperimetric inequality involving cross-ratio distance.

   The heart of the above question is to develop a connection between hit-and-run and the KLS conjecture. It is widely believed that hit-and-run should be no slower than the ball walk. However, an improvement in the bound for the KLS conjecture does not currently imply any faster mixing time for hit-and-run.

2. Consider the coordinate hit-and-run walk described in Figure 2.4. Coordinate hit-and-run is commonly known as Gibbs sampling, and the walk is used throughout practice. What is the mixing time of coordinate hit-and-run? Currently, nothing is known of its mixing time.

   The potential benefit of coordinate hit-and-run is that for many natural convex bodies (polytopes, ellipsoids, etc.), a step of coordinate hit-and-run can be implemented with $O(n)$ fewer arithmetic operations than a step of hit-and-run. We give surprising experimental evidence in Section 5.4.7 that for the purposes of computing volume, coordinate hit-and-run and hit-and-run take roughly the same number of steps to return a similar volume estimate.

3. Finding a warm start for the random walk is a cumbersome technical component, and it is

unclear whether a warm start is necessary. In practice, such a step is commonly ignored and replaced by running the walk for a short "burn-in" phase.

For hit-and-run, it is proven in [19] that the walk will mix in $O^*(n^4)$ from any starting point, provided the point is not exponentially close to the boundary. Thus, there is a penalty of $O^*(n)$ compared to the mixing time of $O^*(n^3)$ from a warm start. Is this penalty necessary?

For the ball walk, one difficulty is that there are certain points that are bad starting points. If the first point of the walk is too close to a corner, then it will take exponential time, in expectation, to take a single proper step. Such an example can be easily seen for the cube, where as a point approaches a corner, the probability of making a proper step is 2^{-n}. But can we get a mixing time bound if we start the ball walk from a "nice" point (say, starting from the origin for K isotropic)?

4. Let K be an arbitrary convex body. Consider starting the ball walk from a point $x \in K$, with uniform target distribution over K. Let X_t denote the probability distribution of the walk after t steps. Is the expected distance of the walk from x monotonically increasing with t? More precisely, is

$$\mathsf{E}(\|X_t - x\|) \geq \mathsf{E}(\|X_{t-1} - x\|)$$

for all $t \geq 1$?

Note that without the boundary effects of K, such a statement would be trivial, as the distribution of X_t would look like a spherical Gaussian of increasing variance centered at x. An analogous conjecture can be made for hit-and-run.

This conjecture could be helpful in declaring convergence of a sampling procedure, similar to the convergence test mentioned in 5.3.1.

CHAPTER 4

VOLUME COMPUTATION VIA GAUSSIAN COOLING

Computing the volume of a convex body is an ancient and fundamental problem; it is also a *difficult* problem, as evidenced by both the #P-hardness of computing the volume of an explicit polytope [34] and exponential lower bounds for deterministic algorithms in the general oracle model, even to approximate the volume to within an exponential factor in the dimension [3, 35]. Against this backdrop, the breakthrough result of Dyer, Frieze and Kannan [13, 1] established a randomized polynomial-time algorithm for estimating the volume to within any desired accuracy. In the quarter-century since then, the quest for faster volume algorithms has revealed an array of powerful and elegant techniques for the design and analysis of algorithms, and influenced the development of asymptotic convex geometry [36, 37, 18, 4, 5, 7, 38, 8, 10, 39] .

The DFK algorithm for computing the volume of a convex body K in $\mathbb{R}^n$ given by a membership oracle uses a sequence of convex bodies $K_0, K_1, \ldots, K_m = K$, starting with the unit ball fully contained in K and ending with K. Each successive body $K_i = 2^{i/n}B_n \cap K$ is a slightly larger ball intersected with K. Using random sampling, the algorithm estimates the ratios of volumes of consecutive bodies. The product of these ratios times the volume of the unit ball was the estimate of the volume of K. Sampling is achieved by a random walk in the convex body. There were many technical issues to be addressed, but the central challenge was to show a random walk that "mixed" rapidly, i.e. converged to its stationary distribution in a polynomial number of steps. The overall complexity of the algorithm was $O^*(n^{23})$ oracle calls[1].

Since then researchers have improved the complexity of volume computation and sampling for convex bodies considerably, to $O^*(n^4)$ for volume estimation and for obtaining the first random sample [8, 10] and to $O^*(n^3)$ per sample for subsequent samples [10, 19]. These improvements rely

[1]The O^* notation suppresses error terms and logarithmic factors.

on continuous random walks, the use of affine transformations, improved isoperimetric inequalities and several other developments. However, throughout the course of these developments, the outer DFK algorithm using a chain of bodies remained unchanged till the most recent improvement in 2003 [8]. The LV algorithm [8] relies on sampling a sequence of logconcave distributions, akin to simulated annealing, starting with one that is highly concentrated around a point deep inside the convex body and ending with the uniform distribution (we will discuss these ideas in more detail presently). The total number of random points needed is only $O^*(n)$, down from $\Omega(n^2)$ needed by all previous algorithms. Combining this with the $O^*(n^3)$ complexity for each sample yielded the overall $O^*(n^4)$ complexity for volume computation. Before running this algorithm, there is a pre-processing step where the convex body is placed in nearly-isotropic position, ensuring in particular that most of the body is contained in a ball of radius $O(\sqrt{n})$. Crucially, this well-roundedness property is maintained during the course of the algorithm.

Is there a faster algorithm? In 1995, Kannan, Lovász and Simonovits, while analyzing the convergence of the ball walk for sampling, proposed a beautiful geometric conjecture now known as the KLS hyperplane conjecture [5]. Roughly speaking, it says that the worst-case isoperimetric ratio for a subset of a convex body is achieved by a hyperplane to within a constant factor. They were able to show that hyperplanes are within $O(\sqrt{n})$ of the minimum. The convergence of the ball walk depends on the square of the reciprocal of the isoperimetric ratio; thus the KLS conjecture had the potential to improve the sampling time by a factor of n to $O^*(n^2)$ per sample and thereby indicated the possibility of an $O^*(n^3)$ volume algorithm (such an algorithm would have to surmount other substantial hurdles). Thus the KLS conjecture appeared to be a fundamental bottleneck to obtaining an $O^*(n^3)$ volume algorithm.

The KLS hyperplane conjecture remains unresolved, in spite of intensive efforts and partial progress towards its resolution [40, 41, 39]. Indeed, it captures two well-known and much older conjectures from convex geometry, the slicing (or hyperplane) conjecture and the thin-shell conjecture (these were all shown to be equivalent in a certain sense recently [6, 39]), and thus has

effectively evaded resolution for nearly a half-century.

In Chapter 3, we showed $O^*(n^2)$ oracle calls suffice to generate a random sample from the standard Gaussian distribution $\mathcal{N}(0, I)$ in $\mathbb{R}^n$ restricted to an arbitrary convex body containing the unit ball. More generally, if a convex body K contains the unit ball, then we showed that $O^*(\max 1, \sigma^2 n^2)$ samples suffice to generate a random sample from $K \cap \mathcal{N}(0, \sigma^2 I)$ (see Theorem 3.1.1). Here the improved Gaussian sampling is first used to obtain an $O^*(n^3)$ algorithm for approximating the integral of a standard Gaussian over K, which we hereafter refer to as the *Gaussian volume* or Gaussian measure of K. This extension of Gaussian sampling to Gaussian volume follows a similar structure of the algorithms in [7, 8, 10], but applied to the Gaussian case. We now outline the high-level approach.

The improved complexity for Gaussian volume estimation is achieved by using a sequence of Gaussians (rather than exponentials as in LV), starting with a highly concentrated Gaussian centered inside K and ending with the standard Gaussian. The cooling schedule is analogous to the schedule in the LV algorithm, but each sample takes only $O^*(n^2)$ time. For a Gaussian with covariance $\sigma^2 I$, the mixing time is $O^*(\max\{\sigma^2, 1\}n^2)$. Since the starting σ is small and the last σ is 1, this bound is $O^*(n^2)$ throughout the algorithm. (We encounter additional technical issues such as maintaining a warm start for the random walks.)

Building upon the Gaussian volume result, we find an $O^*(n^3)$ algorithm for computing the volume of any convex body containing a unit ball and mostly contained in a ball of radius $O^*(\sqrt{n})$. Equivalently, it suffices to have $\mathsf{E}(\|X\|^2) = O^*(n)$ for a uniform random point X from the body. Note that this condition is satisfied if K is in isotropic position. Assuming the body is well-rounded (or sandwiched) in this sense, no further affine transformation is used, and there is no need to assume or maintain near-isotropy during the course of the volume algorithm.

To describe the main ideas behind the improvement, we recall the LV algorithm in more detail. It uses a sequence of $O^*(\sqrt{n})$ exponential distributions, starting with a distribution that is concentrated inside the unit ball contained in K, then "flattening" this distribution to the uniform

by adjusting a multiplicative factor in the exponent[2]. In each phase, samples from the previous distribution are used to estimate the ratio of the integrals of two consecutive exponential functions (by simply averaging the ratio of the function values at the sample points). It is crucial to keep the variance of this ratio estimator bounded, and to do this, the distributions could be cooled by a factor of $1 + \frac{1}{\sqrt{n}}$ in each phase. This leads to $O^*(\sqrt{n})$ phases in total, and to $O^*(\sqrt{n})$ samples per phase. Along with the sample complexity of $O^*(n^3)$ per sample, this gives the bound of $O^*(n^4)$.

Returning to the usual Lebesgue volume, how could we possibly improve the LV algorithm, without relying on the KLS conjecture? We will also use Gaussian cooling, starting with a highly concentrated Gaussian and flattening it (i.e., increasing σ) till we reach the uniform distribution. In the beginning, this is similar to the algorithm of [32]. But after σ becomes higher than 1 (or some constant), we no longer have quadratic sampling time, as the mixing time of the ball work grows as $\max\{\sigma^2, 1\}n^2$. Moreover, we need to go till $\sigma^2 = \Omega(n)$, so cooling at the fixed rate of $1 + 1/n$ would be too slow. The main new idea is that for $\sigma > 1$, the cooling rate can be made higher, in fact about $1 + \sigma^2/n$ instead of only $1 + 1/n$. This means that the number of phases to double σ^2 is only n/σ^2. It can be shown that the number of samples per "doubling" phase is only $O^*(1)$, giving n/σ^2 samples in total. Multiplying by the sampling time, we have $\frac{n}{\sigma^2} \cdot \sigma^2 n^2 = n^3$, a cubic algorithm! The key technical component of the analysis is to show that the variance of the ratio estimator remains bounded even at this higher cooling rate of $1 + \sigma^2/n$.

We now formally state the problems.

Problem 4.0.1. [Volume] Given a membership oracle for a convex set K in $\mathbb{R}^n$ containing the unit ball B_n, and error parameter $\varepsilon > 0$, give an algorithm that computes a number V such that with probability at least $3/4$,

$$(1 - \varepsilon)\mathsf{vol}(K) \leq V \leq (1 + \varepsilon)\mathsf{vol}(K).$$

We denote the Gaussian density function as $\gamma(x) = (2\pi)^{-n/2} \cdot \exp\left(-\|x\|^2/2\right)$.

[2]In the original description, the algorithm first created a "pencil" using an extra dimension, but this can be avoided [10].

Problem 4.0.2. [Gaussian Volume] Given a membership oracle for a convex set K in $\mathbb{R}^n$ containing the unit ball B_n, and error parameter $\varepsilon > 0$, give an algorithm that computes a number V such that with probability at least $3/4$,

$$(1-\varepsilon)\int_K \gamma(x)\,dx \le V \le (1+\varepsilon)\int_K \gamma(x)\,dx.$$

4.1 Results

Our main result can be stated precisely as follows, which solves Problem 4.0.1 in $O^*(n^3)$ assuming the input body K is well-rounded. We note that the roundness condition can be achieved for any convex body by a preprocessing step consisting of an affine transformation. It is a significantly weaker condition than isotropic position.

Theorem 4.1.1. *There is an algorithm that, for any $\varepsilon > 0, p > 0$ and convex body K in $\mathbb{R}^n$ that contains the unit ball and has $\mathsf{E}_K(\|X\|^2) = O(n)$, with probability $1-p$, approximates the volume of K within relative error ε and has complexity*

$$O\left(\frac{n^3}{\varepsilon^2}\cdot \log^2 n \log^2\frac{1}{\varepsilon}\log^2\frac{n}{\varepsilon}\log\frac{1}{p}\right) = O^*\left(n^3\right).$$

in the membership oracle model.

More generally, if $B_n \subset K$ and $\mathsf{E}_K(\|X\|^2) = R^2$, then the algorithm has complexity

$$O\left(\frac{\max\{R^2n^2, n^3\}}{\varepsilon^2}\cdot \log^2 n \log^2\frac{1}{\varepsilon}\log^2\frac{n}{\varepsilon}\log\frac{1}{p}\right) = O^*\left(\max\{R^2n^2, n^3\}\right).$$

The following result essentially follows as a corollary of Theorem 4.1.1, but we state it explicitly as computing the Gaussian measure of a convex body is an important problem, arguably even more important than uniform volume.

Theorem 4.1.2. *For any $\varepsilon > 0$, $p > 0$, and any convex set K in $\mathbb{R}^n$ containing the unit ball, there is an algorithm that, with probability $1 - p$, approximates the Gaussian volume $\mathcal{N}(0, \sigma^2 I)$ of K within relative error ε and has complexity*

$$O\left(\frac{n^3}{\varepsilon^2} \cdot \log^2(n) \log^2\left(\frac{n}{\varepsilon}\right) \log\left(\frac{1}{p}\right)\right) = O^*(n^3)$$

in the membership oracle model.

4.2 Outline of Analysis

The sampling time when the variance is σ^2 is $\max\{1, \sigma^2\}n^2$. If we cooled at a rate of $1 + 1/n$ throughout the algorithm, we would get an $O^*(n^4)$ algorithm since the last doubling phase, i.e. the set of phases until σ^2 doubles, takes $\Omega(n)$ samples, each mixing for $\Omega(n^3)$ steps. The main insight that speeds up our algorithm is the cooling rate of $1 + \sigma^2/(2C^2n)$ once $\sigma^2 > 1$. Cooling at a faster rate once $\sigma^2 > 1$ will allow us to compute volume in time $O^*(n^3)$ by having fewer phases when the mixing time of the ball walk increases.

The volume algorithm proceeds as a series of phases, where each phase seeks to estimate a ratio of Gaussian integrals over the convex body K. More precisely, let

$$f(\sigma^2, x) = \begin{cases} \exp\left(-\|x\|^2/(2\sigma^2)\right) & \text{if } x \in K \\ 0 & \text{otherwise} \end{cases}$$

and

$$F(\sigma^2) = \int_{\mathbb{R}^n} f(\sigma^2, x)\, dx.$$

Define μ_i as the probability distribution proportional to $f(\sigma_i^2, x)$; that is, μ_i is a symmetric Gaussian distribution with variance σ_i^2 restricted to K. Let X be a random sample point from μ_i and let

$Y = f(\sigma_{i+1}^2, X)/f(\sigma_i^2, X)$. We see that the expectation of Y is the ratio of $F(\sigma_{i+1}^2)/F(\sigma_i^2)$:

$$\begin{aligned}
\mathsf{E}(Y) &= \int_K \exp\left(\frac{\|x\|^2}{2\sigma_i^2} - \frac{\|x\|^2}{2\sigma_{i+1}^2}\right) d\mu_i(x) \\
&= \int_K \exp\left(\frac{\|x\|^2}{2\sigma_i^2} - \frac{\|x\|^2}{2\sigma_{i+1}^2}\right) \cdot \frac{\exp\left(-\|x\|^2/(2\sigma_i^2)\right)}{F(\sigma_i^2)}\, dx \\
&= \frac{1}{F(\sigma_i^2)} \cdot \int_K \exp\left(-\frac{\|x\|^2}{2\sigma_{i+1}^2}\right) dx = \frac{F(\sigma_{i+1}^2)}{F(\sigma_i^2)}.
\end{aligned}$$

Our goal is to estimate $\mathsf{E}(Y)$ within some target relative error. The algorithm estimates the quantity $\mathsf{E}(Y)$ by taking random sample points $X_1, \ldots, X_k$ and computing the empirical estimate for $\mathsf{E}(Y)$ from the corresponding $Y_1, \ldots, Y_k$:

$$W = \frac{1}{k}\sum_{j=1}^{k} Y_j = \frac{1}{k}\sum_{j=1}^{k} \frac{f_{i+1}(X_j)}{f_i(X_j)}.$$

The variance of Y divided by its expectation squared will give a bound on how many independent samples X_i are needed to estimate $\mathsf{E}(Y)$ within the target accuracy. Thus we seek to bound $\mathsf{E}(Y^2)/\mathsf{E}(Y)^2$. We have that

$$\mathsf{E}(Y^2) = \frac{\int_K \exp\left(\frac{\|x\|^2}{2\sigma_i^2} - \frac{\|x\|^2}{\sigma_{i+1}^2}\right) dx}{\int_K \exp\left(-\frac{\|x\|^2}{2\sigma_i^2}\right) dx} = \frac{F(\frac{\sigma_{i+1}^2\sigma_i^2}{2\sigma_i^2 - \sigma_{i+1}^2})}{F(\sigma_i^2)}$$

and

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} = \frac{F(\sigma_i^2)F(\frac{\sigma_{i+1}^2\sigma_i^2}{2\sigma_i^2 - \sigma_{i+1}^2})}{F(\sigma_{i+1}^2)^2}$$

If we let $\sigma^2 = \sigma_{i+1}^2$ and $\sigma_i^2 = \sigma^2/(1+\alpha)$, then we can further simplify as

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} = \frac{F\left(\frac{\sigma^2}{1+\alpha}\right) F\left(\frac{\sigma^2}{1-\alpha}\right)}{F\left(\sigma^2\right)^2}.$$

The algorithm has two parts, and the cooling rate α_i is different for them. In the first part,

starting with a Gaussian of variance $\sigma^2 = 1/(4n)$, which has almost all its measure inside the ball contained in K, we increase σ^2 by a fixed factor of $1 + 1/n$ in each phase till the variance σ^2 reaches 1. For each σ, we sample random points from the corresponding distribution and estimate the ratio of the densities for the current phase and the next phase by averaging over samples. The total complexity for the first part is thus

$$O^*(n) \text{ phases} \times O^*(1) \text{ samples per phase} \times O^*(n^2) \text{ time per sample} = O^*(n^3).$$

In the second part, we increase the variance till it reaches $C^2 n$, after which one final phase suffices to compare with the target uniform distribution. However, we cannot afford to cool at the same rate of $1 + 1/n$ because the time per sample goes to $O^*(\sigma^2 n^2)$ for $\sigma > 1$. By the end of this part, we would be using $O^*(n^3)$ per sample, and the overall complexity would be $O^*(n^4)$. Instead we observe that we can cool at a faster rate of $1 + \sigma^2/(2C^2 n)$ and still maintain that the variance of the ratio estimator is a constant. The following bound on the variance, proved in Section 4.4, allows us to cool at a faster rate as σ increases and overcome the increased sampling cost of $O^*(\sigma^2 n^2)$.

Lemma 4.2.1. *Let $K \subseteq C\sqrt{n}B_n$ and $\alpha \leq 1/2$. Then,*

$$\frac{F\left(\frac{\sigma^2}{1+\alpha}\right) F\left(\frac{\sigma^2}{1-\alpha}\right)}{F\left(\sigma^2\right)} \leq \exp\left(2 \cdot \frac{C^2\alpha^2 n}{\sigma^2}\right).$$

Note that the above RHS is $\leq 1 + \sigma^2/(Cn)$ if we select $\alpha = \sigma^2/(2C^2 n)$. With this rate, the number of phases needed to double the variance is only $O(C^2 n/\sigma^2)$, and the number of samples per phase will be $O^*(1)$. Together, they compensate for the higher complexity of obtaining each sample. The complexity of the second part of the algorithm is thus

$$O^*\left(\frac{C^2 n}{\sigma^2}\right) \text{ phases} \times O^*\left(1\right) \text{ samples per phase} \times O^*(\sigma^2 n^2) \text{ time per sample} = O^*(C^2 n^3).$$

In Section 4.4, we prove that cooling at this accelerated rate still keeps the variance of the ratio estimator appropriately bounded.

We note that with respect to estimating the volume using Lemma 4.2.1, there is a range of cooling rates that we could select to obtain an $O^*(n^3)$ algorithm. We need to select $\alpha \leq \sigma/(C\sqrt{n})$ to maintain $\alpha \leq 1/2$ and satisfy the condition of Lemma 4.2.1. We need $\alpha \geq \sigma^2/(C^2 n)$ because otherwise there would be too many phases. So for any α such that $c_1\sigma^2/(C^2 n) \leq \alpha \leq c_2\sigma/(C\sqrt{n})$, we get that the complexity of the volume algorithm is

$$O^*\left(\frac{1}{\alpha}\right) \text{ phases} \times O^*\left(\frac{C^2\alpha^2 n}{\sigma^2} \cdot \frac{1}{\alpha}\right) \text{ samples per phase} \times O^*(\sigma^2 n^2) \text{ time per sample} = O^*(C^2 n^3).$$

We select the cooling rate of $\alpha = \sigma^2/(2C^2 n)$ for simplicity of the algorithm since this cooling rate also maintains a warm start for the ball walk sampler, as shown in Lemma 3.4.8.

4.3 Volume Algorithm

At a high level, the algorithm relies on sampling random points from a sequence of distributions using the ball walk with a Metropolis filter. For a target density proportional to the function f, the ball walk with δ-steps is defined in Figure 2.2.

After a suitable number of steps, the point x obtained will be from a distribution close to the one whose density is proportional to f. However, this process is slightly complicated by the fact that we only know that the point is mixed once a certain number of *proper* steps have been taken, i.e. steps where $y \in K$ or alternatively where $f(y) \neq 0$.

The algorithm in Figure 4.1 starts with a Gaussian of variance $1/(4n)$, with mean at the center of the unit ball inside K. This variance is increased over a sequence of phases till the distribution becomes uniform over K. Until the variance σ^2 reaches 1, it is increased by a fixed factor of $1 + 1/n$ in each phase. After the variance reaches 1, the variance accelerates, increasing by a factor of $1 + \sigma^2/(2C^2 n)$ where σ^2 is the current variance. This process is continued till the vari-

ance becomes linear in C^2n, at which point one final phase can be used to jump to the uniform distribution. In each phase, we pick a sample of random points from the current distribution and compute the average of the ratio of the current density to the next density for each point. The product of these ratios times a fixed term to account for the integral of the initial function is the estimate output by the algorithm.

Let $f(\sigma^2, K)$ be the function that assigns value $\exp(-\|x\|^2/(2\sigma^2))$ to points in a convex set K and zero to points outside. The algorithm in Figure 4.1 uses a series of such functions.

Volume(K, ε) We assume $B_n \subseteq K \subseteq C\sqrt{n}B_n$.

1. Initialize: $\nu = \left(\frac{\varepsilon}{n}\right)^{16}, \sigma_0^2 = \frac{1}{4n}, k = \frac{512 \log C^2 n}{\varepsilon^2}, i = 0$; x_0 is a random point from $N(0, \sigma_0^2 I) \cap K$.

   Define
   $$\beta(\sigma) = \begin{cases} 1 + \dfrac{1}{n} \text{ if } \sigma^2 \leq 1 \\ 1 + \dfrac{\sigma^2}{2C^2 n} \text{ otherwise} \end{cases}$$

2. While $\sigma_i^2 \leq C^2 n$:

   (a) Get k points $\{X_1, \ldots, X_k\}$ using the **Ball Walk** with
   $$\begin{cases} \delta = \min\{\sigma_i, 1\}/(4096\sqrt{n \log n/\varepsilon}) \text{ ball radius} \\ f = f(\sigma_i^2, K \cap 4\sigma_i\sqrt{n}B_n) \text{ target density} \\ 10^{16} \max\{\sigma^2, 1\} n^2 \cdot \log(1/\nu) \text{ proper steps} \end{cases}$$

   (b) Set $\sigma_{i+1}^2 = \sigma_i^2 \cdot \beta(\sigma_i)$; if $\sigma_{i+1}^2 > C^2 n$, set $\sigma_{i+1}^2 = \infty$.

   (c) Compute the ratio estimate
   $$W_{i+1} = \frac{1}{k} \cdot \sum_{j=1}^{k} \frac{f_{i+1}(X_j)}{f_i(X_j)}.$$

   (d) Increment i.

3. Return $(2\pi\sigma_0^2)^{n/2} W_1 \ldots W_i$ as the volume estimate for K.

Figure 4.1: The Volume algorithm

4.4 Accelerated Cooling Schedule

Here we present the $O^*(n^3)$ volume algorithm for well-rounded convex bodies using the accelerated cooling schedule. The key technical component is bounding the variance of the volume estimator. In Lemma 4.5.3, we prove that the variance is bounded when we cool at the rate of $1 + 1/n$, and its proof is relatively straightforward. In this section, we prove the variance remains bounded when we cool at the rate $1 + \sigma/n$ when $K \subseteq \sqrt{n}B_n$, and its proof is more involved.

The goal of this section is to prove Lemma 4.2.1, which gives a bound on the variance of the random variable we use to estimate the ratio of Gaussian integrals in the volume algorithm in Figure 4.1. Here we will prove the inequality to be true for all logconcave functions, but only apply it to an indicator function of a convex body for the volume algorithm.

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a logconcave function such that $E_f(\|X\|^2) = R^2$.

Define

$$g(x, \sigma^2) = f(x) \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right)$$

and also define

$$G(\sigma^2) = \int_{\mathbb{R}^n} g(x, \sigma^2)\, dx.$$

Define μ_i as the probability distribution proportional to $g(x, \sigma_i^2)$. Let X be a random sample from μ_i and let $Y = g(X, \sigma_{i+1}^2)/g(X, \sigma_i^2)$. From a standard calculation, we have that

$$\mathsf{E}(Y) = \frac{G(\sigma_{i+1}^2)}{G(\sigma_i^2)}.$$

The second moment of Y is given by

$$\begin{aligned}
\mathsf{E}(Y^2) &= \int_{\mathbb{R}^n} \left(\frac{g(x,\sigma_{i+1}^2)}{g(x,\sigma_i^2)}\right)^2 d\mu_i(x) \\
&= \int_{\mathbb{R}^n} \left(\frac{g(x,\sigma_{i+1}^2)}{g(x,\sigma_i^2)}\right)^2 \cdot \frac{g(x,\sigma_i^2)}{G(\sigma_i^2)}\, dx \\
&= \frac{1}{G(\sigma_i^2)} \int_{\mathbb{R}^n} \frac{g(x,\sigma_{i+1}^2)^2}{g(x,\sigma_i^2)}\, dx \\
&= \frac{1}{G(\sigma_i^2)} \int_{\mathbb{R}^n} g\left(x, \frac{\sigma_{i+1}^2\sigma_i^2}{2\sigma_i^2 - \sigma_{i+1}^2}\right) dx \\
&= \frac{G(\frac{\sigma_{i+1}^2\sigma_i^2}{2\sigma_i^2-\sigma_{i+1}^2})}{G(\sigma_i^2)}.
\end{aligned}$$

To bound the number of samples X needed to estimate Y within a target relative error, we will bound $\mathsf{E}(Y^2)/\mathsf{E}(Y)^2$, which is given by

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} = \frac{G(\frac{\sigma_{i+1}^2\sigma_i^2}{2\sigma_i^2-\sigma_{i+1}^2})G(\sigma_i^2)}{G(\sigma_{i+1}^2)^2}.$$

Then letting $\sigma^2 = \sigma_{i+1}^2$ and $\sigma_i^2 = \sigma^2/(1+\alpha)$, we can further simplify as

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} = \frac{G\left(\frac{\sigma^2}{1+\alpha}\right) G\left(\frac{\sigma^2}{1-\alpha}\right)}{G(\sigma^2)^2}.$$

The above n-dimensional inequality is difficult to analyze directly. We will reduce it to a simpler 1-dimensional inequality via localization. Define an exponential needle $E = (a, b, \gamma)$ as a segment $[a, b] \subseteq \mathbb{R}^n$ and $\gamma \in \mathbb{R}$ corresponding to the weight function $e^{\gamma t}$ applied the segment $[a, b]$. The integral of an n-dimensional function f over this one dimensional needle is

$$\int_E f = \int_0^{|b-a|} f(a+tu)e^{\gamma t}\, dt \qquad \text{where} \quad u = \frac{b-a}{|b-a|}.$$

We use the following theorem from [5].

Theorem 4.4.1. *([5]) Let f_1, f_2, f_3, f_3 be four nonnegative continuous functions defined on $\mathbb{R}^n$, and $\alpha, \beta > 0$. Then, the following are equivalent:*

1. *For every logconcave function F defined on $\mathbb{R}^n$ with compact support,*

$$\left(\int_{\mathbb{R}^n} F(t) f_1(t)\, dt\right)^{\alpha} \left(\int_{\mathbb{R}^n} F(t) f_2(t)\, dt\right)^{\beta} \leq \left(\int_{\mathbb{R}^n} F(t) f_3(t)\, dt\right)^{\alpha} \left(\int_{\mathbb{R}^n} F(t) f_4(t)\, dt\right)^{\beta}$$

2. *For every exponential needle E,*

$$\left(\int_E f_1\right)^{\alpha} \left(\int_E f_2\right)^{\beta} \leq \left(\int_E f_3\right)^{\alpha} \left(\int_E f_4\right)^{\beta}$$

A crucial aspect of our proof is that we can restrict the support of our target logconcave function f, which then allows us to consider a restricted family of needles. Recall that we assumed $\mathsf{E}_f(\|X\|^2) = R^2$. Set $R_1 = 2R \cdot \log(1/\varepsilon)$. By the following lemma from [31], if we restrict the support of f to be $R_1 \cdot B_n$, we only lose an $\varepsilon/2$ fraction of the mass.

Lemma 4.4.2. *[31] Let $X \in \mathbb{R}^n$ be a random point from a logconcave distribution with $\mathsf{E}(X^2) = R^2$. Then for any $t > 1, \Pr(\|X\| > tR) < \exp(-R + 1)$.*

We can now reduce the desired inequality to a simpler form of exponential needles, which are restricted to lie in the interval $[-R_1, R_1]$.

Lemma 4.4.3. *If for all intervals $[\ell, u] \subseteq [-R_1, R_1]$ and $\gamma > 0$,*

$$\frac{\int_{\ell}^{u} \exp(\gamma t) \exp\left(-\frac{t^2(1+\alpha)}{2\sigma^2}\right)\, dt \cdot \int_{\ell}^{u} \exp(\gamma t) \exp\left(-\frac{t^2(1-\alpha)}{2\sigma^2}\right)\, dt}{\left(\int_{\ell}^{u} \exp(\gamma t) \exp\left(-\frac{t^2}{2\sigma^2}\right)\, dt\right)^2} \leq c,$$

then for all logconcave functions f defined on $\mathbb{R}^n$ whose support is a compact subset of $R_1 \cdot B_n$,

$$\frac{G(\frac{\sigma^2}{1+\alpha}) G(\frac{\sigma^2}{1-\alpha})}{G(\sigma^2)^2} \leq c.$$

Proof. Applying Theorem 4.4.1 and setting $f_1(x) = g(\sigma^2/(1+\alpha), x)$, $f_2(x) = g(\sigma^2/(1-\alpha), x)$, $f_3(x) = f_4(x) = \sqrt{c} \cdot g(\sigma^2, x)$, $\beta = \gamma = 1$, we have that

$$\frac{G(\frac{\sigma^2}{1+\alpha})G(\frac{\sigma^2}{1-\alpha})}{G(\sigma^2)^2} \leq c$$

if and only if for all exponential needles $E \subseteq \mathbb{R}^n$,

$$\frac{\int_E g\left(\frac{\sigma^2}{1+\alpha}, x\right)\, dx \int_E g\left(\frac{\sigma^2}{1-\alpha}, x\right)\, dx}{\left(\int_E g\left(\sigma^2, x\right)\, dx\right)^2} \leq c.$$

To prove the lemma, we will show that we can reduce the inequality for an arbitrary exponential needle $E \subseteq \mathbb{R}^n$ to the simpler form. E is defined by an interval $\mathcal{I}$ in $\mathbb{R}^n$ and an arbitrary exponential function $\exp(\gamma t)$ on $\mathcal{I}$. Define z as the closest distance from the origin to the extension of the $\mathcal{I}$ in both directions. Parameterize the interval $\mathcal{I}$ in terms of t, where $t = 0$ gives the closest point along the extension of $\mathcal{I}$ to the origin (note $t = 0$ does not necessarily have to be on $\mathcal{I}$). Also define the minimum and maximum values of t on $\mathcal{I}$ as ℓ and u respectively. We can assume that $-R_1 \leq \ell \leq u \leq R_1$ since f is 0 outside of $R_1 \cdot B_n$. We then have that

$$\begin{aligned}\int_E g(\sigma^2, x)\, dx &= \int_\ell^u \exp(\gamma t) \exp\left(-\frac{t^2+z^2}{2\sigma^2}\right))\, dt \\ &= \exp\left(-\frac{z^2}{2\sigma^2}\right) \cdot \int_\ell^u \exp(\gamma t) \exp\left(-\frac{t^2}{2\sigma^2}\right)\, dt.\end{aligned}$$

Note that in the integral ratio, the terms with z cancel out since

$$\exp\left(-\frac{(1+\alpha)z^2}{2\sigma^2} - \frac{(1-\alpha)z^2}{2\sigma^2} + \frac{2z^2}{2\sigma^2}\right) = 1,$$

which then proves the lemma. □

Before bounding the desired inequality, we first prove the following two helper lemmas.

Lemma 4.4.4. *Let X be a random variable with $\mathsf{E}(X^4) < \infty$ and $a \leq X \leq b$. Then,*

$$\mathsf{E}(X^4) - \mathsf{E}(X^2)^2 \leq 4\max\{a^2, b^2\}\mathsf{Var}(\mathsf{X}).$$

Proof. Let Y be an independent random variable drawn from the same distribution as X. Then,

$$\begin{aligned}
2\mathsf{Var}(X^2) &= \mathsf{Var}(X^2) + \mathsf{Var}(Y^2) \\
&= \mathsf{E}(X^4) - \mathsf{E}(X^2)^2 + \mathsf{E}(Y^4) - \mathsf{E}(Y^2)^2 \\
&= \mathsf{E}(X^4) - 2\mathsf{E}(X^2)\mathsf{E}(Y^2) + \mathsf{E}(Y^4) \\
&= \mathsf{E}\left((X^2 - Y^2)^2\right) \\
&= \mathsf{E}\left((X+Y)^2(X-Y)^2\right) \\
&\leq 4\max\{a^2, b^2\}\mathsf{E}\left((X-Y)^2\right) \\
&= 4\max\{a^2, b^2\}\mathsf{E}\left(X^2 - 2XY + Y^2\right) \\
&= 8\max\{a^2, b^2\}\mathsf{Var}(X).
\end{aligned}$$

□

Lemma 4.4.5. *Let $[\ell, u] \subseteq [-R_1, R_1]$ and*

$$v(x) = \frac{\int_\ell^u t^2 \exp(\gamma t) \exp\left(-\frac{t^2 x}{2\sigma^2}\right) dt}{\int_\ell^u \exp(\gamma t) \exp\left(-\frac{t^2 x}{2\sigma^2}\right) dt}.$$

Then, $v'(x) \geq -2R_1^2/x$.

Proof. For convenience, define

$$s(x,t) = \exp(\gamma t) \exp\left(-\frac{t^2 x}{2\sigma^2}\right).$$

We have that

$$v'(x) = \left(\frac{1}{2\sigma^2}\right) \cdot \frac{\left(\int_\ell^u t^2 s(x,t)\,dt\right)^2 - \int_\ell^u s(x,t)\,dt \int_\ell^u t^4 s(x,t)\,dt}{\left(\int_\ell^u s(x,t)\,dt\right)^2}.$$

Observe that the above quantity is the difference of moments of a truncated Gaussian distribution.

We then have that

$$\begin{aligned}
v'(x) &= \left(\frac{1}{2\sigma^2}\right) \cdot \left(\mathsf{E}(X^2)^2 - \mathsf{E}(X^4)\right) && \text{where } X \sim \mathcal{N}(\frac{\gamma\sigma^2}{x}, \frac{\sigma^2}{x}) \Big| \ell \le X \le u \\
&\ge -\frac{2R_1^2}{\sigma^2} \cdot \mathsf{Var}(X) && \text{by Lemma 4.4.4} \\
&\ge -\frac{2R_1^2}{\sigma^2} \cdot \frac{\sigma^2}{x} && \text{by Theorem 3.3.1} \\
&= -\frac{2R_1^2}{x}.
\end{aligned}$$

□

The following lemma now proves the variance bound.

Lemma 4.4.6. *Let* $[\ell, u] \subseteq [-R_1, R_1]$ *and* $\alpha \le 1/2$*. Then*

$$\frac{\int_\ell^u \exp(\gamma t) \exp\left(-\frac{t^2(1+\alpha)}{2\sigma^2}\right)\,dt \cdot \int_\ell^u \exp(\gamma t) \exp\left(-\frac{t^2(1-\alpha)}{2\sigma^2}\right)\,dt}{\left(\int_\ell^u \exp(\gamma t) \exp\left(-\frac{t^2}{2\sigma^2}\right)\,dt\right)^2} \le \exp\left(2 \cdot \frac{R_1^2\alpha^2}{\sigma^2}\right).$$

Proof.

Again for convenience, define

$$s(x,t) = \exp(\gamma t) \exp\left(-\frac{t^2 x}{2\sigma^2}\right).$$

Define

$$h(\alpha) := \frac{\int_\ell^u s(1+\alpha,t)\,dt \cdot \int_\ell^u s(1-\alpha,t)\,dt}{\left(\int_\ell^u s(1,t)\,dt\right)^2}.$$

Note that the lemma is equivalent to bounding $h(\alpha)$. We first prove the following claim, from which the lemma will easily follow.

Claim 4.4.7. *For $\alpha \le 1/2$,*

$$h'(\alpha) \le \frac{4 \cdot \alpha R^2 \cdot h(\alpha)}{\sigma^2}.$$

Proof. First, observe that

$$\frac{\partial}{\partial\alpha}\Big(s(1+\alpha,t)\Big) = \frac{\partial}{\partial\alpha}\left(\int_\ell^u \exp(\gamma t)\exp\left(-\frac{t^2(1+\alpha)}{2\sigma^2}\right)\,dt\right)$$

$$= \frac{-1}{2\sigma^2}\left(\int_\ell^u t^2\exp(\gamma t)\exp\left(-\frac{t^2(1+\alpha)}{2\sigma^2}\right)\,dt\right)$$

and similarly

$$\frac{\partial}{\partial\alpha}\Big(s(1-\alpha,t)\Big) = \frac{1}{2\sigma^2}\left(\int_\ell^u t^2\exp(\gamma t)\exp\left(-\frac{t^2(1-\alpha)}{2\sigma^2}\right)\,dt\right).$$

Then taking the derivative of $h(\alpha)$ with respect to α gives

$$\frac{\partial}{\partial\alpha}\left(h(\alpha)\right) = \frac{\partial}{\partial\alpha}\left(\frac{\int_\ell^u s(1+\alpha,t)\,dt \cdot \int_\ell^u s(1-\alpha,t)\,dt}{\left(\int_\ell^u s(1,t)\,dt\right)^2}\right)$$

$$= \frac{1}{2\sigma^2}\cdot\frac{\int_\ell^u s(1+\alpha,t)\,dt \cdot \int_\ell^u t^2 s(1-\alpha,t)\,dt - \int_\ell^u s(1-\alpha,t)\,dt \cdot \int_\ell^u t^2 s(1+\alpha,t)\,dt}{\left(\int_\ell^u s(1,t)\,dt\right)^2}.$$

We now have that

$$\frac{h'(\alpha)}{h(\alpha)} = \frac{1}{2\sigma^2}\cdot\left(\frac{\int_\ell^u t^2\exp\left(\gamma t\right)\exp\left(-\frac{t^2(1-\alpha)}{2\sigma^2}\right)\,dt}{\int_\ell^u \exp\left(\gamma t\right)\exp\left(-\frac{t^2(1-\alpha)}{2\sigma^2}\right)\,dt} - \frac{\int_\ell^u t^2\exp\left(\gamma t\right)\exp\left(-\frac{t^2(1+\alpha)}{2\sigma^2}\right)\,dt}{\int_\ell^u \exp\left(\gamma t\right)\exp\left(-\frac{t^2(1+\alpha)}{2\sigma^2}\right)\,dt}\right).$$

Let

$$v(x) = \frac{\int_\ell^u t^2\exp\left(\gamma t\right)\exp\left(-\frac{t^2 nx}{2\sigma^2}\right)\,dt}{\int_\ell^u \exp\left(\gamma t\right)\exp\left(-\frac{t^2 nx}{2\sigma^2}\right)\,dt}.$$

We then have that

$$\begin{aligned}
\frac{h'(\alpha)}{h(\alpha)} &= \frac{1}{2\sigma^2}\left(\left(v(1+\alpha) + \int_{1+\alpha}^{1-\alpha} v'(x)\,dx\right) - v(1+\alpha)\right) \\
&= -\frac{1}{2\sigma^2}\left(\int_{1-\alpha}^{1+\alpha} v'(x)\,dx\right) \\
&\leq \frac{1}{2\sigma^2}\left(\int_{1-\alpha}^{1+\alpha} \frac{2R_1^2}{x}\,dx\right) && \text{by Lemma 4.4.5} \\
&= \frac{R_1^2}{2\sigma^2}\left(-\frac{1}{x^2}\Big|_{1-\alpha}^{1+\alpha}\right) \\
&= \frac{R_1^2}{2\sigma^2}\left(\frac{1}{(1-\alpha)^2} - \frac{1}{(1+\alpha)^2}\right) \\
&= \frac{R_1^2}{2\sigma^2}\left(\frac{4\alpha}{(1-\alpha^2)^2}\right) \\
&\leq \frac{4R_1^2\alpha}{\sigma^2}.
\end{aligned}$$

□

By Claim 4.4.7, we then have a bound on $h(\alpha)$ as follows:

$$\begin{aligned}
\ln h(\alpha) &= \ln h(0) + \int_0^{\alpha} \frac{d}{dx}\left(\ln h(x)\right)\,dx \\
&= \ln(1) + \int_0^{\alpha} \frac{h'(x)}{h(x)}\,dx \\
&\leq \int_0^{\alpha} \frac{4R_1^2 x}{\sigma^2}\,dx \\
&= \frac{2R_1^2 x^2}{\sigma^2}\Big|_0^{\alpha} \\
&= \frac{2R_1^2\alpha^2}{\sigma^2},
\end{aligned}$$

and thus

$$h(\alpha) \leq \exp\left(\frac{2R_1^2\alpha^2}{\sigma^2}\right).$$

□

Lemma 4.4.8. *Suppose a logconcave function $f : \mathbb{R}^n \to \mathbb{R}$ has support contained in $R_1 \cdot B_n$. Let $g(x, \sigma^2) = f(x) \exp\left(-\|x\|^2/(2\sigma^2)\right)$. Let X be drawn from a distribution proportional to $g(x, \sigma^2)$ and $Y = g(X, \sigma^2(1+\alpha))/g(X, \sigma^2)$. Then for $\alpha \leq 1/2$,*

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} \leq \exp\left(\frac{2R_1^2\alpha^2}{\sigma^2}\right).$$

Proof. Follows immediately from Lemma 4.4.3 and Lemma 4.4.6. □

The bound in Lemma 4.2.1 then follows by applying Lemma 4.4.8 with the indicator function of a convex body.

4.5 Further Analysis

4.5.1 Completing the Cooling Schedule

The following lemma shows that the starting distribution of our algorithm will (1) have a volume that can be approximated by a standard integral computation and (2) can be efficiently sampled from using simple rejection sampling, i.e. generate a point from the full Gaussian and reject if it is not in K.

Lemma 4.5.1. *If $\sigma^2 \leq (n + \sqrt{8n\ln(1/\varepsilon)})/2$ and $B_n \subseteq K$, then $\int_K \exp\left(-\|x\|^2\right) 2\sigma^2\, dx \geq (1-\varepsilon)\int_{\mathbb{R}^n} \exp\left(-\|x\|^2\right) 2\sigma^2\, dx$.*

Proof. We will use the following concentration bound on a spherical Gaussian in $\mathbb{R}^n$ with mean μ and variance σ^2, which is valid for $t > 1$:

$$\Pr(\|X-\mu\|^2 - \sigma^2 n > t\sigma^2\sqrt{n}) \leq e^{-t^2/8}.$$

Selecting $\mu = 0$, $t = \sqrt{8\ln(1/\varepsilon)}$, and $\sigma^2 = 1/(n + t\sqrt{n})$ gives

$$\Pr(\|X\|^2 > 1) \leq \varepsilon$$

and therefore all but an ε-fraction of the Gaussian is contained inside B_n (and therefore K). $\square$

We now bound the variance when cooling at the fixed rate of $1 + 1/n$, which will be useful when σ^2 is small. First, we will need the following lemma that is proved in [8].

Lemma 4.5.2. *Let $K \subseteq \mathbb{R}^n$ be a convex body and $f : K \to \mathbb{R}$ be a logconcave function. For any $a > 0$, define*

$$Z(a) = \int_K f(ax)dx.$$

Then $a^n Z(a)$ is a logconcave function of a.

Lemma 4.5.3. *Assume $n \geq 3$. Let X be a random point in K with density proportional to $f_i(x) = \exp\left(-\frac{\|x\|^2}{2\sigma_i^2}\right)$, $\sigma_{i+1}^2 = \sigma_i^2(1 + 1/n)$, and $Y = f_{i+1}(X)/f_i(X)$. Then,*

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} < 1 + \frac{2}{n}.$$

Proof. We have

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} = \frac{\int_K \exp\left(-\frac{\|x\|^2(1-\alpha)}{2\sigma^2}\right)\, dx \int_K \exp\left(-\frac{\|x\|^2(1+\alpha)}{2\sigma^2}\right)\, dx}{\left(\int_K \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right)\, dx\right)^2}.$$

By Lemma 4.5.2, the function $z(a) = a^{n+1} \int_K \exp\left(-a\|x\|^2/2\right)\, dx$ is logconcave, and thus

$$z\left(\frac{1-\alpha}{\sigma^2}\right) z\left(\frac{1+\alpha}{\sigma^2}\right) \leq z\left(\frac{1}{\sigma^2}\right)^2.$$

Therefore

$$\int_K \exp\left(-\frac{\|x\|^2(1-\alpha)}{2\sigma^2}\right) dx \int_K \exp\left(-\frac{\|x\|^2(1+\alpha)}{2\sigma^2}\right) dx \leq \left(\frac{1}{1-\alpha^2}\right)^{n+1} \left(\int_K \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right) dx\right)^2.$$

Setting $\alpha = 1/n$, we have that

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} \leq \left(\frac{1}{1-1/n^2}\right)^{n+1} = \left(1+\frac{1}{n^2-1}\right)^{n+1} \leq \exp\left(\frac{1}{n-1}\right) \leq 1+\frac{2}{n}.$$

□

The following lemma says that going to a Gaussian of variance $C^2 n$ for $K \subseteq C\sqrt{n}B_n$ is sufficient to provide a good volume estimate for K.

Lemma 4.5.4. *Let $K \subseteq C\sqrt{n}B_n$, $f_i(x) = \exp(-\frac{\|x\|^2}{2\sigma_i^2})$, $\sigma_i^2 \geq C^2 n$, and $\sigma_{i+1}^2 = \infty$. Then for X drawn from distribution proportional to $f_i \cap K$ and $Y = f_{i+1}(X)/f_i(X)$,*

$$\frac{\mathsf{E}(Y^2)}{\mathsf{E}(Y)^2} \leq e^2.$$

Proof. To show that $\sigma_i^2 \geq C^2 n$ is sufficient to switch to the uniform distribution, observe that $f_i(X) \geq 1/e$ since $\|X\|^2 \leq C^2 n$ and thus $Y \leq e$. Also note $Y \geq 1$. Therefore $\mathsf{E}(Y^2)/\mathsf{E}(Y)^2 \leq e^2$. □

4.5.2 Bounding the Dependence

We now show that the volume estimate computed in Algorithm 4.1 is accurate. The analysis here is involved due to the dependence between samples used. The dependence is a seemingly unavoidable byproduct of our approach in order to efficiently provide a warm start to every run of the random walk. If we had independence, the accuracy of the algorithm would essentially follow from a straightforward application of Chebyshev's inequality. The purpose of this section

is to bound the effect that the dependence between samples has on the accuracy of our volume estimator.

Define R_i as the i-th integral ratio, i.e.

$$R_i := \frac{F(\sigma_{i+1}^2)}{F(\sigma_i^2)} = \frac{\int_K \exp\left(-\|x\|^2/(2\sigma_{i+1}^2)\right)\, dx}{\int_K \exp\left(-\|x\|^2/(2\sigma_i^2)\right)\, dx},$$

and let W_i denote the estimate of the algorithm for R_i.

For two random variables X, Y, we will measure their independence by the following:

$$\mu(X,Y) = \sup_{A,B} |P(X \in A, Y \in B) - P(X \in A)P(Y \in B)|,$$

where A, B range over measurable subsets of the ranges of X, Y.

We will give an argument similar to [8], and use the following lemmas that were proved there.

Lemma 4.5.5. *[8] If f and g are two measurable functions, then*

$$\mu(f(X), g(Y)) \leq \mu(X,Y).$$

Lemma 4.5.6. *[8] Let X, Y be random variables such that $0 \leq X \leq a$ and $0 \leq Y \leq b$. Then*

$$|\mathsf{E}(XY) - \mathsf{E}(X)E(Y)| \leq ab\mu(X,Y).$$

Lemma 4.5.7. *[8] Let $X \geq 0$ be a random variable, $a > 0$, and $X' = \min(X, a)$. Then*

$$\mathsf{E}(X') \geq E(X) - \frac{\mathsf{E}(X^2)}{4a}.$$

Lemma 4.5.8. *With probability at least $4/5$,*

$$(1-\varepsilon)R_1 \dots R_m \leq W_1 \dots W_m \leq (1+\varepsilon)R_1 \dots R_m.$$

Proof. Let $(X_0^i, X_1^i, X_2^i, \ldots, X_k^i)$ be the sequence of sample points for the ith volume phase. The distribution of each X_i is approximately the correct distribution, but slightly off based on the error parameter ν in each phase that bounds the total variation distance. We will define new random variables $\overline{X}_j^i$ that have the correct distribution for each phase.

Note that X_j^0 would be sampled from the exact distribution, and then rejected if outside of K. Therefore $\Pr(X_j^0 = \overline{X}_j^0) = 1$. Suppose that the total number of sample points throughout the algorithm is t. Using induction and the definition of total variation distance, we see that

$$\Pr(X_i^j = \overline{X}_i^j, \forall i, j) \geq 1 - t\nu. \tag{4.1}$$

Let

$$Y_j^i = \frac{\exp\left(-\frac{\|X_j^i\|^2}{2\sigma_{i+1}^2}\right)}{\exp\left(-\frac{\|X_j^i\|^2}{2\sigma_i^2}\right)} \quad \text{and} \quad \overline{W}_i = \frac{1}{k_i}\sum_{j=1}^{k_i} Y_j^i.$$

Note that for a fixed i, all of the Y_j^i have the same expectation since they are from the exact distribution, and it is equal to $\mathsf{E}(\overline{W}_i)$. Suppose that we have $\mathsf{E}((Y_j^i)^2) \leq c_i \mathsf{E}(Y_j^i)^2$. Then

$$\begin{aligned}\mathsf{E}(\overline{W}_i^2) &= \frac{1}{k_i^2}\left(\sum_{j=1}^{k_i} \mathsf{E}((Y_j^i)^2) + k_i(k_i-1)R_i^2\right) \\ &\leq \left(1 + \frac{c_i - 1}{k_i}\right) \cdot \mathsf{E}(\overline{W}_i)^2. \end{aligned} \tag{4.2}$$

The following claim bounds the variance of our ratio estimator under a faster cooling rate; combined with Lemma 4.5.3, we have a bounds on the variance throughout our algorithm. It follows from Lemma 4.2.1.

Claim 4.5.9. *Suppose that $K \subseteq C\sqrt{n}B_n$ and let $\alpha = \sigma^2/(2C^2n)$. Then,*

$$\frac{\mathsf{E}\left((Y^i)^2\right)}{\mathsf{E}(Y^i)^2} < 1 + \frac{\sigma^2}{C^2 n}.$$

Suppose that we had independence between samples and consider bounding the cumulative error for all phases of the algorithm. When $\sigma^2 \leq 1$, we can bound the number of phases for the first part as $m_1 \leq 2n \log 4n$. When $\sigma^2 > 1$, we will analyze the phases in chunks, where a chunk is the set of phases until σ^2 doubles. Note that the number of phases in a chunk starting with variance σ^2 is at most $2C^2n/\sigma^2$. Also there are at most $\log(C^2n)$ chunks. Observe that for a single chunk with starting variance σ^2, where i, j are the starting and ending phases of the chunk, we have

$$\frac{\mathsf{E}(\overline{W}_i^2 \dots \overline{W}_j^2)}{R_i^2 \dots R_j^2} \leq \left(1 + \frac{2\sigma^2}{kC^2n}\right)^{2C^2n/\sigma^2} \leq \left(1 + \frac{5}{k}\right).$$

Then, there is one final phases when we switch to the uniform distribution, which has variance at most $1 + e^2$ by Lemma 4.5.4.

Let m denote the total number of phases. If we had independence between samples, then we can use Lemma 4.5.3 and Claim 4.5.9 with Chebyshev's inequality to bound the probability of failure:

$$\begin{aligned}
\Pr\left(\frac{|\overline{W}_1 \dots \overline{W}_m - R_1 \dots R_m|}{R_1 \dots R_m} \geq \frac{\varepsilon}{2}\right) &\leq \frac{4\mathsf{Var}(\overline{W}_1 \dots \overline{W}_m)}{\varepsilon^2 R_1^2 \dots R_m^2} \\
&= \frac{4}{\varepsilon^2}\left(\frac{\mathsf{E}(\overline{W}_1^2 \dots \overline{W}_m^2)}{R_1^2 \dots R_m^2} - 1\right) \\
&\leq \frac{4}{\varepsilon^2}\left(\left(1 + \frac{2}{kn}\right)^{m_1}\left(1 + \frac{5}{k}\right)^{\log C^2 n}\left(1 + \frac{e^2}{k}\right) - 1\right) \\
&\leq \frac{4}{\varepsilon^2}\left(\exp\left(\frac{2m_1}{kn} + \frac{5\log(C^2n)}{k} + \frac{e^2}{k}\right) - 1\right) \\
&\leq \frac{4}{\varepsilon^2}\left(\exp\left(\frac{\varepsilon^2}{50}\right) - 1\right) \\
&\leq \frac{4}{\varepsilon^2}\left(\left(1 + \frac{\varepsilon^2}{40}\right) - 1\right) \\
&= \frac{1}{10}
\end{aligned}$$

However, subsequent samples are dependent, and we must carefully bound the dependence. The analysis is somewhat involved, but will follow essentially the sample template as in [8, 32] which utilizes the following lemma to bound dependence between subsequent samples, where ν is the target total variation distance for each sample point. For convenience, denote the entire sequence of t samples points used in the algorithm as $(Z_0, Z_1, \ldots, Z_{t-1})$.

Lemma 4.5.10. *(a) For $0 \leq i < t$, the random variables Z_i and Z_{i+1} are ν-independent, and the random variables $\overline{Z}_i$ and $\overline{Z}_{i+1}$ are (3ν)-independent.*

(b) For $0 \leq i < t$, the random variables $(Z_0, \ldots, Z_i)$ and Z_{i+1} are (3ν)-independent.

(c) For $0 \leq i < m$, the random variables $\overline{W}_1 \ldots \overline{W}_i$ and $\overline{W}_{i+1}$ are $(3km\nu)$-independent.

The variables $\overline{W}_i$ are not bounded, but we will introduce a new set of random variables based on $\overline{W}_i$ that are bounded so we can later apply Lemma 4.5.6. Let

$$\alpha = \frac{\varepsilon^{1/2}}{8(m\mu)^{1/4}},$$

where $\mu = 3km\nu$. Note that α is much larger than one. Define

$$V_i = \min\{\overline{W}_i, \alpha \mathsf{E}(\overline{W}_i)\}.$$

It is clear that $\mathsf{E}(V_i) \leq \mathsf{E}(\overline{W}_i)$, and by Lemma 4.5.7, we also have

$$\mathsf{E}(V_i) \geq \mathsf{E}(\overline{W}_i) - \frac{\mathsf{E}(\overline{W}_i^2)}{4\alpha \mathsf{E}(\overline{W}_i)} \geq (1 - \frac{1}{4\alpha}(1 + \frac{7}{k}))\mathsf{E}(\overline{W}_i) \geq (1 - \frac{1}{2\alpha})\mathsf{E}(\overline{W}_i).$$

Let $U_0 = 1$ and define recursively

$$U_{i+1} = \min\{U_i V_{i+1}, \alpha \mathsf{E}(V_1) \ldots \mathsf{E}(V_{i+1})\}.$$

We will now show that

$$(1 - \frac{i-1}{\alpha})\mathsf{E}(V_1)\dots\mathsf{E}(V_i) \leq \mathsf{E}(U_i) \leq (1 + 2\mu\alpha^2 i)\mathsf{E}(V_1)\dots\mathsf{E}(V_i). \tag{4.3}$$

By Lemma 4.5.5, the random variables U_i and V_{i+1} are μ-independent, and by Lemma 4.5.6 and since $\alpha \geq 1$,

$$|\mathsf{E}(U_iV_{i+1} - \mathsf{E}(U_i)\mathsf{E}(V_{i+1})| \leq \mu\alpha\mathsf{E}(V_1)\dots\mathsf{E}(V_i)\alpha\mathsf{E}(\overline{W}_{i+1}) \leq 2\mu\alpha^2\mathsf{E}(V_1)\dots\mathsf{E}(V_{i+1}). \tag{4.4}$$

From (4.4), we can get the upper bound on $\mathsf{E}(U_{i+1})$ by induction:

$$\begin{aligned}\mathsf{E}(U_{i+1}) &\leq \mathsf{E}(U_iV_{i+1}) \\ &\leq \mathsf{E}(U_i)\mathsf{E}(V_{i+1}) + 2\mu\alpha^2\mathsf{E}(V_1)\dots\mathsf{E}(V_{i+1}) \\ &\leq (1 + 2\mu\alpha^2(i+1))\mathsf{E}(V_1)\dots\mathsf{E}(V_{i+1}).\end{aligned} \tag{4.5}$$

Similarly,

$$\mathsf{E}(U_i^2) \leq (1 + 2\mu\alpha^4 i)\mathsf{E}(V_1^2)\dots\mathsf{E}(V_i^2) \tag{4.6}$$

$$\text{and} \quad \mathsf{E}(U_i^2V_{i+1}^2) \leq (1 + 2\mu\alpha^4 i)\mathsf{E}(V_1^2)\dots\mathsf{E}(V_{i+1}^2). \tag{4.7}$$

For the lower bound, we use Lemma 4.5.7 and (4.7) to get:

$$\begin{aligned}\mathsf{E}(U_{i+1}) &\geq \mathsf{E}(U_iV_{i+1}) - \frac{\mathsf{E}(U_i^2V_{i+1}^2)}{4\alpha\mathsf{E}(V_1)\dots\mathsf{E}(V_{i+1})} \\ &\geq \mathsf{E}(U_iV_{i+1}) - (1 + 2\mu\alpha^4 i)\frac{\mathsf{E}(V_1^2)\dots\mathsf{E}(V_{i+1}^2)}{4\alpha\mathsf{E}(V_1)\dots\mathsf{E}(V_{i+1})}.\end{aligned} \tag{4.8}$$

For $\alpha \geq 3k$, we have that

$$\begin{aligned}
\mathsf{E}(V_i^2) \leq \mathsf{E}(\overline{W}_i^2) &\leq (1+\frac{7}{k})\mathsf{E}(\overline{W}_i)^2 \\
&\leq (1+\frac{7}{k})\frac{1}{(1-1/(2\alpha))^2}\mathsf{E}(V_i)^2 \\
&\leq (1+\frac{7}{k})(1+\frac{1}{2k})\mathsf{E}(V_i)^2 \\
&\leq (1+\frac{8}{k})\mathsf{E}(V_i)^2. \qquad (4.9)
\end{aligned}$$

Combining (4.4), (4.8), and (4.9),

$$\begin{aligned}
\mathsf{E}(U_{i+1}) &\geq \mathsf{E}(U_iV_{i+1}) - \frac{1}{4\alpha}(1+2\mu\alpha^4 i)(1+\frac{8}{k})^i\mathsf{E}(V_1)\ldots\mathsf{E}(V_{i+1}) \\
&\geq \mathsf{E}(U_iV_{i+1}) - \frac{1+2\mu\alpha^4 i}{2\alpha}\mathsf{E}(V_1)\ldots\mathsf{E}(V_{i+1}) \\
&\geq \mathsf{E}(U_i)\mathsf{E}(V_{i+1}) - \frac{1}{\alpha}\mathsf{E}(V_1)\ldots\mathsf{E}(V_{i+1}).
\end{aligned}$$

Then, by induction on i,

$$\mathsf{E}(U_{i+1}) \geq \mathsf{E}(V_1)\ldots\mathsf{E}(V_{i+1}) - \frac{i}{\alpha}\mathsf{E}(V_1)\ldots\mathsf{E}(V_{i+1}). \qquad (4.10)$$

Putting (4.5) and (4.10) together, we now have a proof of (4.3). Thus,

$$\mathsf{E}(U_m) \leq (1+\frac{\varepsilon}{4})\mathsf{E}(V_1)\ldots\mathsf{E}(V_m) \leq (1+\frac{\varepsilon}{4})\mathsf{E}(\overline{W}_1)\ldots\mathsf{E}(\overline{W}_m).$$

We also have that $\alpha \geq 4m/\varepsilon$ implies

$$\mathsf{E}(U_m) \geq (1-\frac{\varepsilon}{4})\mathsf{E}(\overline{W}_1)\ldots\mathsf{E}(\overline{W}_m).$$

From (4.6) and (4.9), and the selection of α, μ, and the lower bound on k, we have that

$$\begin{aligned}
\mathsf{E}(U_m^2) &\leq (1 + 2\mu\alpha^4 m)\mathsf{E}(V_1^2)\ldots\mathsf{E}(V_m^2) \\
&\leq (1 + 2\mu\alpha^4 m)(1 + \frac{8}{k})^m \mathsf{E}(V_1)^2\ldots\mathsf{E}(V_m)^2 \\
&\leq (1 + 2\mu\alpha^4 m)(1 + \frac{8}{k})^m \frac{1}{(1-(m-1)/\alpha)^2}\mathsf{E}(U_m)^2 \\
&\leq (1 + \frac{\varepsilon^2}{64})E(U_m)^2,
\end{aligned}$$

and hence

$$\Pr\left(|U_m - \mathsf{E}(U_m)| \leq \frac{\varepsilon}{2}\mathsf{E}(\overline{W}_1)\ldots\mathsf{E}(\overline{W}_m)\right) \geq 0.9$$

by Chebyshev's inequality. Then, applying Markov's inequality,

$$\Pr(U_{i+1} \neq U_i V_{i+1}) = \Pr\left(U_i V_{i+1} > \alpha\mathsf{E}(V_1)\ldots\mathsf{E}(V_{i+1})\right) \leq \frac{2}{\alpha}$$

and similarly

$$\Pr(V_i \neq \overline{W}_i) \leq \frac{1}{\alpha}.$$

So, with probability at least $1 - 3k/\alpha$, we have $U_m = \overline{W}_1\ldots\overline{W}_m$. Also, from (4.1), we have that $\overline{W}_1\ldots\overline{W}_m = W_1\ldots W_m$ with probability at least $1-2km\nu$. Recall that $\mathsf{E}(\overline{W}_1)\ldots\mathsf{E}(\overline{W}_m) = R_1\ldots R_m$. Therefore, with probability at least $4/5$

$$|W_1\ldots W_m - R_1\ldots R_m| \leq \frac{\varepsilon}{2}R_1\ldots R_m,$$

which proves the lemma. □

4.5.3 Proof of Volume Theorems

We can now prove the volume algorithm runs in $O^*(n^3)$ oracle queries.

Proof.(of Theorem 4.1.1)

We assume that $\varepsilon \geq 2^{-n}$, which only ignores cases which would take exponential time. Then by Lemma 4.5.1, selecting $\sigma_0^2 = 1/(4n)$ implies that all but a negligible amount of volume of the starting Gaussian is contained in K.

Recall that our algorithm only has a bound on the expected number of steps. To account for this, we will run the algorithm $O(1)$ times to obtain a run which takes at most a constant factor of ball walk steps to proper steps, say with probability $1/20$. By Lemma 4.5.8, the answer returned by the algorithm will be within the target relative error with probability at least $4/5$. Thus the overall probability of failure is $3/4$. Note that we can boost this probability of failure to $1 - p$ by the standard trick of repeating the algorithm $\log 1/p$ times and returning the median.

We now analyze the runtime of the algorithm in Figure 4.1. Set $C = R\log(1/\varepsilon)/\sqrt{n}$. Assume that $C \geq 1$ (otherwise arbitrarily increase C). When $\sigma^2 \leq 1$, using the value of k, the mixing time assigned to each phase, and the fact that there are $O(n \log n)$ phases, we see that the total number of ball walk steps taken is $O(n^{2.5} k \log n \log^2(n/\varepsilon)) = O(n^3 \log^2 n \log^2(n/\varepsilon)/\varepsilon^2) = O^*(n^3)$. When $\sigma^2 > 1$, the analysis is very similar if we note that the faster cooling rate and fewer number of samples cancels out the slower mixing time of $O^*(\sigma^2 n^2)$. Thus, it follows that the total number of ball walk steps taken is

$$O\left(\frac{C^2 n^3 \log^2 n \log^2 \frac{n}{\varepsilon}}{\varepsilon^2}\right) = O\left(\frac{R^2 n^2}{\varepsilon^2} \cdot \log^2 n \log^2 \frac{1}{\varepsilon} \log^2 \frac{n}{\varepsilon}\right) = O^*(R^2 n^2).$$

□

The proof of Theorem 4.1.2 follows from Theorem 4.1.1 by stopping Algorithm 4.1 when $\sigma^2 = 1$.

4.6 Future Work and Open Questions

Perhaps the most important open question spawned by this work is to find an $O^*(n^3)$ rounding algorithm for arbitrary convex bodies. The current best rounding complexity is $O^*(n^4)$ oracle queries [8]. A potential avenue for finding such an algorithm is to use the standard Gaussian distribution as a means to approximately round the convex body K.

One potential complication with rounding is that it is unknown whether the KLS conjecture would imply a $O^*(n^3)$ rounding algorithm. The KLS conjecture would imply that sampling using the ball walk is $O^*(n^2)$ for isotropic convex bodies. However, the current $O^*(n^4)$ rounding algorithm of [8] samples from convex bodies which are only well-rounded, i.e. K satisfies $B_n \subseteq K \subseteq O^*(\sqrt{n})B_n$. It is an open question to develop a rigorous connection between the KLS conjecture and the complexity of putting a convex body into approximate isotropic position.

What is the correct complexity for volume estimation of well-rounded convex bodies? In the membership oracle model, the current best algorithm is $O^*(n^3)$ (Theorem 4.1.1) and the best lower bound is $O^*(n^2)$ [42]. The $O^*(n^2)$ lower bound is developed for parallelopipeds, a fairly restrictive class of convex bodies. It seems promising that by considering a more complex family of convex bodies, we can obtain a better lower bound than $O^*(n^2)$.

What about obtaining a faster algorithm than $O^*(n^3)$ for volume estimation? It can be shown than to generate an independent random sample via the ball walk requires $O^*(n^2)$ steps. Thus, the $O^*(n^3)$ volume complexity only has an $O^*(n)$ overhead compared to the currently best possible sample complexity. One possible direction is suggested by our volume implementation in Chapter 5, where we observe that taking many dependent samples is faster for estimating volume than a small number of independent samples in practice. Could such an approach lead to an improved asymptotic complexity for volume computation?

In the volume algorithm in Figure 4.1, the algorithm only assumes membership oracle access to a convex body. Can we get a faster algorithm, deterministic or randomized, for estimating the

volume of an explicit polytope $P = \{x|Ax \leq b\}$? It is known exactly computing the volume of an explicit polytope is $\#\mathcal{P}$-complete [43], but it is open whether there exists a deterministic fully polynomial time algorithm to estimate the volume of a polytope.

CHAPTER 5

IMPLEMENTATION OF SAMPLING AND VOLUME ALGORITHMS

High-dimensional integration and sampling is a fundamental problem of interest across the sciences and engineering [44] [45] [46] [47] [48] [49]. In 1989, Dyer, Frieze and Kannan [13] [1] found an $O^*(n^{23})$ algorithm for computing the volume of a convex body (the O^* notation suppresses dependences on log factors and error parameters). Their approach was extended to polynomial time algorithms for integrating logconcave functions. Since then, over the past quarter century, the theoretical efficiency of algorithms for computing the volume and for integration has gone down from $O^*(n^{23})$ membership tests to $O^*(n^4)$ such tests [8]. Recently, we found an $O^*(n^3)$ algorithm for computing the Gaussian measure of a convex body [32] and an $O^*(n^3)$ algorithm for computing the volume of a well-rounded convex body [50].

Together with today's computation speeds, this progress suggests the possibility of practical multi-dimensional integration. Indeed, the first such implementation of the Lovász-Vempala $O^*(n^4)$ algorithm was reported in 2012 [51]; however, it could only compute the volume of cubes of dimension up to 9, with higher dimensional cubes taking prohibitively long. There are several reasons for this, including (a) the complexities above are for the number of membership tests; each test typically takes quadratic or higher number of arithmetic operations, (b) the theoretical bounds have large constants (e.g., 10^{10} and multiple logarithmic factors), and (c) the algorithms are designed for the the worst-case, i.e., they include routines such as: "run for $f(n)$" steps, where $f(n)$ is independent of the input body. These aspects had to be addressed to some extent in [51] to obtain a working implementation even in very small dimension.

In this paper, we present a more practical algorithm for computing volume and Gaussian volume of a polyhedron, which can handle 100 dimensional bodies in as little as 10 minutes. We also extend this to polyhedra intersected with ellipsoids. The algorithm builds on [8], [32], and [50],

but crucially needs a few more ideas. We present extensive experimental results. The MATLAB implementation is publicly available on MATLAB's File Exchange [52]. To the best of our knowledge, there is no benchmark for high-dimensional volume computation. So we propose a family of test bodies that could serve this purpose for future improvements and algorithms.

In the next section, we describe the main ideas of the algorithm, noting clearly where we extend previous work to obtain a practical algorithm. Following that, we describe the key aspects of our implementation in details, with rigorous justifications to the extent possible. In Section 5.4, we present computational results of our algorithm. We conclude this section with our proposed benchmark for evaluating volume algorithms. Following our work, [53] gave a C++ implementation for volume computation of polytopes and reported even faster results on some bodies in our benchmark.

5.0.1 Benchmark

We propose the following families of test bodies. The first 5 families have volumes that can be computed efficiently by simple formulas, thus serve as testable instances in any dimension. Within these 5, families 2(b), 4 and 5 will typically require that a volume algorithm perform some type of "rounding" step to maintain efficiency.

The last two families of bodies have volumes that can be computed exactly, but the best known algorithms to compute the volume take exponential time. Therefore, approximating the volume appears to be necessary for efficiency. We discuss these test families in more detail when we present the results of our evaluation.

1. **Cube**: a standard $[-1, 1]^n$ cube, which has volume 2^n.

2. **(a) Isotropic Simplex**: a regular n-simplex, which has volume $\sqrt{n+1}/(n!\sqrt{2^n})$.
   (b) Standard Simplex: defined as $\{x \in \mathbb{R}^n : \sum_{i=1}^n x_i \leq 1, x_i \geq 0\}$, that is all coordinates are nonnegative and sum to at most 1. The volume of the standard simplex is $1/n!$.

3. **Half Ball**: the n-dimensional unit ball, with the restriction that $x_1 \geq 0$. The volume of this body is $1/2 \cdot \pi^{n/2}/\Gamma(n/2+1)$.

4. **Transformed Cube**: starting from the **Cube**, we apply a random linear transformation T– each entry of the n by n matrix T was chosen from $\mathcal{N}(0,1)$. The volume is then $|T|2^n$.

5. **Ellipsoid**: an axis-aligned ellipsoid with radius 1 along $n-1$ axes and radius 100 along 1 axis. The volume of the shape is then $100\pi^{n/2}/\Gamma(n/2+1)$.

6. **Zonotope**: the Minkowski sum of m line segments where each line segment is in $\mathbb{R}^n$. The volume of a zonotope can be computed exactly with a direct method, but the algorithm will take exponential time for general m, n (roughly $\binom{m}{n}$). It is, in fact, #P-hard to compute the volume of a zonotope. [43]

7. **Birkhoff polytope**: the polytope of all $n \times n$ doubly stochastic matrices of dimension $n^2 - 2n + 1$. The volume has been computed exactly for values of $n \leq 10$ using specialized algorithms, but is unknown for $n > 10$.

5.1 Volume Algorithm

At a high level, our algorithm is based on that of [8] (henceforth referred to as the LV algorithm). We give an overview here, and a detailed discussion of each component in our implementation in Section **??**. First, note that computing volume is a special case of integration. That is, the volume of K can be expressed as (where $\mathbb{1}_K$ is the indicator function of K):

$$\mathsf{vol}(K) = \int_{\mathbb{R}^n} \mathbb{1}_K(x)\, dx.$$

The above quantity is hard to compute, or even estimate, directly, but there is an insight which makes the problem tractable. Consider the following representation of $\mathsf{vol}(K)$, for any function

$f : \mathbb{R}^n \to \mathbb{R}$:

$$\mathsf{vol}(K) = \frac{\int_K f(x)\,dx}{\int_K f(x)\,dx} \cdot \int_K \mathbb{1} = \int_K f(x)\,dx \cdot \frac{\int_{\mathbb{R}^n} \mathbb{1}_K(x)\,dx}{\int_K f(x)\,dx}.$$

Now the difficult task of computing the volume has been reduced to two, perhaps easier, tasks: (1) compute the ratio of two integrals and (2) integrating the function f over K. Note that the above representation can be extended to any sequence of functions $\{f_0, \ldots, f_{m-1}\}$ where each $f_i : \mathbb{R}^n \to \mathbb{R}$. Given this sequence of functions, the volume of K can be rewritten as

$$\mathsf{vol}(K) = \int_K f_0(x)\,dx \cdot \frac{\int_K f_1(x)\,dx}{\int_K f_0(x)\,dx} \cdot \frac{\int_K f_2(x)\,dx}{\int_K f_1(x)\,dx} \cdots \frac{\int_K \mathbb{1}}{\int_K f_{m-1}(x)\,dx}$$

The functions f_i can be chosen to be anything we like, but should be selected so that each term in the above equation is efficiently estimatable. First, we select f_0 such that $\int_K f_0(x)dx$ is directly computable, up to sufficient precision. For instance, f_0 can be the indicator function for a ball B where $B \subseteq K$, and then $\int_K f_0(x)dx = \mathsf{vol}(B)$, which has a nice, direct formula. Another option is to select f_0 to be a low variance Gaussian centered inside K; that is, the weight of the Gaussian will be highly concentrated around a single point inside K. Then, if the Gaussian is sufficiently "sharp", then $\int_K f_0(x)dx \approx \int_{\mathbb{R}^n} f_0(x)dx$, and integrating a Gaussian over the full space $\mathbb{R}^n$ again has a nice, direct formula. For the implementation, we select f_0 as a low variance Gaussian; further details on how we select f_0 are given in Section 5.2.1.

Next, we want to efficiently compute each integral ratio:

$$\frac{\int_K f_i(x)\,dx}{\int_K f_{i-1}(x)\,dx}.$$

At first glance, the above problem seems just as intractable as volume. However, we do not need an exact answer, and instead want to estimate the above integral ratio within some target relative error. To estimate the integrals, we will use random sampling. Suppose we had a random sample from a distribution proportional to f_{i-1} restricted to K. Let X be a random variable drawn from

the probability distribution proportional to f_{i-1} (letting μ_{i-1} denote its measure), and define a corresponding variable Y as

$$Y = \frac{f_i(X)}{f_{i-1}(X)}.$$

Consider the expected value of this random variable Y, which is equal to the desired quantity we want to estimate:

$$E(Y) = \int_K \frac{f_i(x)}{f_{i-1}(x)}\, d\mu_{i-1}(x) = \int_K \frac{f_i(x)}{f_{i-1}(x)} \cdot \frac{f_{i-1}(x)}{\int_K f_{i-1}(x)}\, dx = \frac{\int_K f_i(x)\, dx}{\int_K f_{i-1}(x)\, dx}.$$

Suppose we have k samples $\{X_1, \ldots, X_k\}$. Then, we can estimate the integral ratio as

$$\frac{1}{k} \sum_{j=1}^{k} \frac{f_i(X_j)}{f_{i-1}(X_j)},$$

which converges to $E(Y)$. For the estimation to be efficient, we need that the number of samples k to get within a target relative error is not too large. For instance, if we went immediately from f_0 to the uniform distribution, an exponential number of samples k would be required. Instead, we construct a cooling schedule from f_0 to the uniform distribution while controlling the variance of Y, which results in a small number of "phases" (i.e. integral ratios) and not too many samples per phase. We select each f_i as a Gaussian, where we slowly increase the variance until the Gaussian is essentially the uniform distribution (i.e. the volume). Further details on how to construct such a sequence are given in Section 5.2.2.

There is still the question of how to obtain samples from the target distribution. We use the *hit-and-run* random walk (Figure 5.1) to generate *approximate* samples from the target distribution. The implementation details of hit-and-run are given in Section 5.2.5. Given enough steps, hit-and-run will converge to the target distribution, and thus can provide sample points to estimate each integral ratio.

But how many steps of hit-and-run are required before the point is from the target distribution?

Hit-and-run(K, f, x): Convex body K, target distribution f, current point x.

- Pick a uniform random line ℓ through the current point x.
- Return a random point on the chord $\ell \cap K$ according to the target distribution f.

Figure 5.1: Hit-and-run sampler

Subsequent points in the random walk may be highly correlated. However, if enough steps of hit-and-run are used, then the current point will "forget" where it started and be an approximately random point from the target distribution. Assuming the current point is somewhat close to the target distribution, it is known $O^*(n^2R^2/r^2)$ hit-and-run steps are required before we are close to target distribution, where $rB_n \subseteq K \subseteq RB_n$; that is, a ball of radius r is contained in K and K is contained in a ball of radius R. The term R/r could unfortunately be very high (e.g. n^{50}) for a general convex body and have a drastic effect on the mixing time. For instance, a long, thin cylinder will have higher mixing time than that of the unit ball. Intuitively, this is because it takes a long time to move from one end of the long cylinder to the other, while it is comparatively easier to move between any two regions in the unit ball.

We can get around this issue by applying a linear transformation T to the convex body K to get a new body $K' = TK$ that is round. Since T is a linear transformation, we have that $\mathsf{vol}(K') = |T| \cdot \mathsf{vol}(K)$ where $|T|$ is the determinant of the matrix corresponding to the transformation. In the case of the long cylinder, while the cylinder could have essentially an arbitrarily high value of R/r, if we instead work with the rounded body, where we shrink the body along the stretched axis and compute $\mathsf{vol}(K')$, then we can efficiently compute the volume. More details of how this transformation is computed are given in Section 5.2.4.

An outline of the algorithm is given in Figure 5.2. Throughout this paper, we assume that the origin lies in K. We start by rounding the convex body K into approximate isotropic position, with respect to the uniform distribution over K (Section 5.2.4). We then compute an annealing schedule $\{a_0, \ldots, a_m\}$ such that almost all of the volume of $e^{-a_0\|x\|^2}$ is contained inside K', and

$a_m = 0$ (i.e. the m-th phase is the uniform distribution) (Section 5.2.2). Once we have the cooling schedule, we compute the volume by estimating each of the ratios

$$\frac{\int_{K'} f_i(x)\,dx}{\int_{K'} f_{i-1}(x)\,dx},$$

where $f_i(x) = e^{-a_i\|x\|^2}$. The ratio is estimated by approximately sampling from $f_{i-1} \cap K'$ (Section 5.2.5) and then averaging the function ratio over the sample points. To test for convergence of this ratio, we use a sliding window over the last W ratios, where if the last W ratios are all within some $C(\varepsilon, m)$ relative error of each other, then we declare convergence (Section 5.2.3). The estimated volume is then the product of the initial integral f_0 over K', the determinant of the linear transformation T that rounded the body, and the ratio estimate R_i for each phase. That is,

$$\begin{aligned}\mathsf{vol}(K) &= \int_K \mathbb{1} = |T| \cdot \int_{K'} \mathbb{1} = |T| \cdot \int_{K'} f_0(x)\,dx \cdot \frac{\int_{K'} f_1(x)\,dx}{\int_{K'} f_0(x)\,dx} \cdots \frac{\int_{K'} \mathbb{1}}{\int_{K'} f_{m-1}(x)\,dx} \\ &= |T| \cdot R_1 R_2 \ldots R_m \cdot \int_{K'} f_0(x)\,dx.\end{aligned}$$

We will now summarize the key optimizations that were made in our implementation to make volume computation practical.

1. For convex bodies that are not sufficiently round, we may need to perform a rounding preprocessing step before estimating the volume. The LV algorithm uses an $O^*(n^4)$ algorithm for rounding, whereas we use a new algorithm which experimentally runs in $O^*(n^3)$ membership calls. (see Section 5.2.4)

2. Instead of using a fixed rate of cooling to the uniform distribution, we *adaptively* compute a cooling schedule according to some constraints. This significantly reduces the number of volume phases required. (see Section 5.2.2)

3. Finally, we only sample from spherical Gaussians, which gives a computationally efficient

Volume(K, ε): Convex body K, error parameter ε.

- $T =$ **Round**(body: K, steps: $8n^3$), set $K' = T \cdot K$.
- $\{a_0, \ldots, a_m\}$ = **GetAnnealingSchedule**(body: K').
- Set x to be random point from $f_0 \cap K'$, $\varepsilon' = \varepsilon/\sqrt{m}$.
- For $i = 1, \ldots, m$,
  - Set $k = 0, x_0 = x, converged = false, W = 4n^2 + 500$.
  - While $converged = false$,
    - $k = k + 1$.
    - x_k=**HitAndRun**(body: K, target distribution: f_{i-1}, current point: x_{k-1}).
    - Set
    $$r_k = \frac{1}{k} \sum_{j=1}^{k} \frac{f_i(x_j)}{f_{i-1}(x_j)}.$$
    - Set $W_{max} = \max\{r_{k-W+1}, \ldots, r_k\}$ and $W_{min} = \min\{r_{k-W+1}, \ldots, r_k\}$.
    - If $W_{max} - W_{min} \leq \varepsilon'/2 \cdot W_{max} \rightarrow converged = true$.
  - Set $R_i = r_k, x = x_k$.
- Return $volume = |T| \cdot (\pi/a_0)^{n/2} \cdot R_1 \ldots R_m$.

Figure 5.2: Volume algorithm

hit-and-run sampler and experimentally improves the mixing time over an arbitrary Gaussian or logconcave function. (see Section 5.2.4)

4. First, we use the empirical distribution of hit-and-run to estimate the volume. In the LV algorithm, hit-and-run is used for some large number of steps, and only a very small fraction of the total steps as sample points, and we experimentally find that using every point from hit-and-run provides a better estimate.

5.2 Implementation Details

In this section we will give a mathematical description of the components of our implementation, and give proofs and/or motivation behind the components. The primary motivation behind our implementation decisions was using as few "hard-coded" constants as possible, and instead try to optimize our runtime for a particular problem instance. For instance, in section 5.2.1, instead of using $a = 2n$ as in [8], we instead use concentration inequalities to binary search for a value of a that is close to optimal for that particular body. As another example, we do not use a fixed cooling schedule, but rather estimate an adaptive cooling schedule based upon samples from the convex body.

An important question about hit-and-run is how fast it converges to the target distribution. The LV algorithm was designed to minimize the asymptotic worst case. Thus, constants of the type 10^{10} and many $\log$ factors are present in the runtime analysis. For instance, it is proven in [19] that $T = 10^{10} n^3 \log 1/\varepsilon$ steps of hit-and-run suffice before we are within distance ε of the target distribution. Combined with the fact that each step of hit-and-run takes $\Omega(n)$ arithmetic operations, this number of steps T is far too large for an actual algorithm, even for very small dimensions. We observe that in practice, one can do much better, but it seems to be difficult to obtain tight bounds on the number of required steps. We instead employ heuristic techniques that try to detect convergence based on the stream of points observed; these techniques experimentally seem to provide a reasonable estimate, but do not give a guarantee of accuracy. For a further discussion, please refer to Sections 5.2.3 and 5.3.1.

In most of the theoretical volume algorithms, there is only an assumption that we have a membership oracle for the convex body. While our experimental evaluations are on explicit polytopes and ellipsoids, our algorithm only needs to compute the intersection of a ray with the body, i.e. the halfspace that first intersects it, and an outer approximation to this would suffice for hit-and-run.

5.2.1 Selecting a Starting Point

Problem: Given a convex body K and error parameter $\varepsilon > 0$, select $a_0 \in \mathbb{R}^n$ such that

$$\int_K e^{-a_0\|x\|^2}\,dx \geq (1-\varepsilon)\int_{\mathbb{R}^n} e^{-a_0\|x\|^2}\,dx.$$

Solution outline:

- Consider a random point X from $e^{-a\|x\|^2}$ over $\mathbb{R}^n$, and bound the probability, as a function of a, that $X \notin K$ using Gaussian tail inequalities. Denote this probability as $p(a)$.
- Binary search of the value of a that gives $p(a) = \varepsilon$, and return $a_0 = a$.

As noted above, we will assume $K = P \cap E$, where P is a polyhedron and E is an ellipsoid. Note that we could select a sufficient value of a_0 without much work, say by assuming that K contains the unit ball and then deriving that $a_0 = (n + \sqrt{8n\ln(1/\varepsilon)})/2$ using Lemma 5.2.1. However, this could significantly increase the time to anneal to Gaussian to the uniform distribution (i.e. the volume). We can use our explicit description of the body K to select a value of a_0 so that K contains close to a $(1-\varepsilon)$ fraction of the volume over $\mathbb{R}^n$. First, note the following two Gaussian concentration inequalities.

Lemma 5.2.1. *Let X be drawn from a spherical Gaussian in $\mathbb{R}^n$ with mean μ and variance σ^2 along any direction. Then for any $t > 1$,*

$$\Pr\left(\|X-\mu\|^2 - \sigma^2 n > t\sigma^2\sqrt{n}\right) \leq e^{-t^2/8}.$$

Lemma 5.2.2. *Let X be drawn from a one-dimensional Gaussian with variance $\sigma^2 = 1/(2a)$. Then,*

$$\Pr(X > t) \leq \frac{e^{-at^2}}{2t\sqrt{a\pi}}.$$

To use these bounds, we will first compute the minimum distance from $0 \in K$ to each hyperplane describing P and to the boundary of E. We then apply Lemma 5.2.1 to the ellipsoid and Lemma 5.2.2 to the polyhedron, and union bound over these probabilities to get a lower bound on the fraction of the Gaussian that lies within the convex body K. That is, for a given a and d the minimum distance from 0 to the surface of the ellipsoid,

$$\begin{aligned}
\Pr(x \notin K) &\leq \Pr(x \notin P) + \Pr(x \notin E) \leq \sum_H \Pr(x \text{ violates } H) + \Pr(\|x\| > d) \\
&\leq \sum_H \Pr\left(\|x_H\| > d(0,H)\right) + e^{-d^2/8} \text{ where } x_H \text{ is the projection of } x \text{ onto the normal of } H \\
&\leq \sum_H \frac{e^{-a \cdot d(0,H)^2}}{2d(0,H)\sqrt{a\pi}} + e^{-d^2/8} =: p(a).
\end{aligned}$$

5.2.2 Computing the Cooling Schedule

Problem: Given a starting function $f_0(x) = e^{-a_0\|x\|^2}$ and a convex body K, construct a sequence of functions that converge to the uniform distribution over K, such that we can efficiently estimate

$$R_i = \frac{\int_K f_i(x)\,dx}{\int_K f_{i-1}(x)\,dx} = \frac{\int_K e^{-a_i\|x\|^2}\,dx}{\int_K e^{-a_{i-1}\|x\|^2}\,dx}.$$

Solution Outline: They key idea, which was used in [8] [32], is that we would like to control the quantity

$$\frac{\mathsf{Var}(Y)}{\mathsf{E}(Y)^2},$$

where $Y = e^{(a_{i-1}-a_i)X}$ and X is drawn from distribution proportional to $f_{i-1} \cap K$. In [8], it is proven that $\mathsf{E}(Y) = R_i$. Therefore, applying Chebyshev's inequality to Y, having $\mathsf{Var}(Y)/\mathsf{E}(Y)^2 \leq 1$ guarantees that only a polynomial number of points are needed to accurately estimate R_i. We estimate $\mathsf{Var}(Y)/\mathsf{E}(Y)^2$ by taking a small number of samples, and stepping as far as we can while

keeping $\mathsf{Var}(Y)/\mathsf{E}(Y)^2 \leq 1$. We take $f_m = f_i$ once f_i appears to be sufficiently close to the uniform distribution.

The following lemma is essentially shown in [8].

Lemma 5.2.3. *Let X be a random point in K with density proportional to $e^{-a_i\|x\|^2}$, $a_{i+1} = a_i(1 - 1/n)$, $n \geq 4$, and*

$$Y = e^{(a_{i+1}-a_i)\|X\|^2}.$$

Then,

$$\frac{\mathsf{Var}(Y)}{\mathsf{E}(Y)^2} \leq \left(\frac{a_i^2}{a_{i+1}(2a_i - a_{i+1})}\right)^{n+1} = \left(1 + \frac{1}{n^2 - 2n}\right)^{n+1} < 1.$$

Our approach seeks to select the sequence of functions $\{f_i\}$ to approach the uniform distribution as quickly as possible while keeping the variance bounded by a constant. We do this by trying to maximize r, where $a_{i+1} = a_i(1 - 1/n)^r$ so that

$$\frac{\mathsf{Var}(Y^2)}{\mathsf{E}(Y)^2} < 1. \tag{5.1}$$

By Lemma 5.2.3, we know that $r = 1$ now suffices. We will then binary search to get, within a factor of 2, the maximum value of r that satisfies (5.1), by taking a small number of sample points and observing their variance.

In [54], for the problem of approximating the partition function of a discrete system, they prove that using a constant number of samples suffice to accurately estimate $\mathsf{Var}(Y)/\mathsf{E}(Y)^2$ for their cooling schedule. We empirically observe a similar behavior for computing the cooling schedule for convex bodies, where a small number of hit-and-run samples suffice, roughly $O(n^2)$, to reasonably estimate $\mathsf{Var}(Y)/\mathsf{E}(Y)^2$.

5.2.3 Declaring Convergence

Problem: Given an error parameter ε' and a stream of dependent random variables $\{X_1, X_2, \ldots\}$,

where $Y_k = 1/k \cdot \sum_{i=1}^{k} X_i$ and

$$R = \lim_{k \to \infty} Y_k,$$

determine a point k such that $Y_k \in [(1-\varepsilon')R, (1+\varepsilon')R]$.

Solution Outline: We use a sliding window of size W and declare convergence once the last W points are all within $\varepsilon'/2$ of each other. That is,

$$\frac{\max_{i:k-W \leq i \leq k} Y_i - \min_{i:k-W \leq i \leq k} Y_i}{\max_{i:k-W \leq i \leq k} Y_i} \leq \varepsilon'/2.$$

The goal of our algorithm is to compute a quantity V' such that $V' \in [(1-\varepsilon)V, (1+\varepsilon)V]$ where V is the true volume of our convex body K. Our algorithm is composed of m phases, each of which approximate the ratio of two integrals over K. By a standard argument, if each of m terms have a relative error $\varepsilon' = \varepsilon/\sqrt{m}$ and have unbiased expectation, then the product of the m terms will have relative error ε. Therefore, we assign relative error $\varepsilon' = \varepsilon/\sqrt{m}$ to each of our integral estimators (the R_i's in Figure 5.2).

We are given a stream of random points in K $\{X_1, X_2, \ldots\}$ drawn approximately from $f_{i-1} \cap K$, which we then relate to $\{Y_1, Y_2, \ldots\}$ by

$$Y_k = \frac{1}{k} \sum_{j=1}^{k} e^{(a_{i-1}-a_i)X_k}. \tag{5.2}$$

Denoting the ratio of integrals in a single phases as R, from [8] we know that

$$R = \lim_{k \to \infty} Y_k. \tag{5.3}$$

Our problem is now: compute a quantity R' with relative error ε' from R. Formally, we want

to estimate the following quantity within a ε'-fraction:

$$\frac{\int_K f_i(x)\,dx}{\int_K f_{i-1}(x)\,dx} = \frac{\int_K e^{-a_i\|x\|^2}\,dx}{\int_k e^{-a_{i-1}\|x\|^2}\,dx}.$$

From hit-and-run, we generate random variables $\{X_1, X_2, \ldots\}$, which are points in K from a distribution approximately proportional to f_{i-1}. Then, applying equations (5.2) and (5.3), we get a sequence of random variables $\{Y_1, Y_2, \ldots\}$ such that

$$R = \lim_{i\to\infty} Y_i.$$

That is, if we take enough steps of hit-and-run, we will converge to the actual ratio R. Computationally, we would like to determine a point where we are within our target accuracy ε'. If the points generated by hit-and-run were independent, then it would be quite easy to determine how many points are necessary to obtain an accurate estimate. However, these points will be highly dependent, and the best known bounds for the number of steps required are far too high for practical computations. Therefore, we use a sliding window approach that stores the last few values of Y_i, and declares convergence once these last few values are all within, say, $\varepsilon'/2$ relative distance of each other.

The benefit of this approach is that it will quickly detect convergence of $\{y_i\}$ to an ε'-fraction of R. However, the drawback is that we can have false positives; that is, we can declare the sequence has converged too soon and we are not within a ε'-fraction of R. There is a clear relationship between the size of this window and how accurately we will estimate R: the more values we store, the longer we will take before declaring convergence. It is unclear how to obtain a good bound on the probability of failure with relation to the window size, but we choose the size of the window based on experimental results (Sections 5.4.2).

5.2.4 Rounding the Convex Body

Problem: Given a convex body K, we would like to find a linear transformation T such that $T \cdot K$ is in approximately isotropic position.

Solution: See Figure 5.3. We assume that K is contained a ball of radius R, because we observe that if enough sample points are taken, each rounding should shrink the maximum singular value by a constant factor. If we do not converge within $\log R$ iterations, we know the number of steps t was not sufficient to accurately estimate the singular values, so we restart with $2t$ steps.

Round(K, t): Convex body: K, rounding steps: t. (**Note**: assume that $B_n \subseteq K \subseteq RB_n$)

- Set $x_0 = 0$, $T = I$, $tries = 0$.
- **Repeat:**
  - $tries = tries + 1$.
  - For $i = 1, \ldots, t$,
    - $x_i =$**HitAndRun**(body: TK, target distribution: $f = 1$, point: x_{i-1}).
  - $(U, S, V^T) = SVD(\{x_1, \ldots, x_r\})$.
  - Set $T = VS^{-1}T$.
- **Until:** $\max(S) \leq 2$ OR $tries > \log R$.
- If $\max(S) \leq 2$:
  - Return T.
- Else:
  - Return **Round**$(K, 2 \cdot t)$.

Figure 5.3: Rounding algorithm

The need for such a transformation is exhibited in the following theorem of [19].

Theorem 5.2.4. *[19] Let K be a convex body that contains a ball of radius r and is contained in a ball of radius R. Let σ be a starting distribution and let σ^m be the distribution of the current*

point after m steps of hit-and-run in K. Let $\varepsilon > 0$, and suppose that the density function $d\sigma/d\pi_K$ is bounded by M except on a set S with $\sigma(S) \leq \varepsilon/2$. Then for

$$m > 10^{10} \frac{n^2 R^2}{r^2} \ln \frac{M}{\varepsilon},$$

the total variation distance of σ^m and π_K is less than ε.

The above mixing time is also shown to be best possible in terms of the quantity R/r, where $rB_n \subseteq K \subseteq RB_n$. This is one measure of how "round" the body K is: if K is a long, thin cylinder, then the ratio R/r can be very high. We can control the ratio R/r by putting the body in approximately isotropic position. We say that a density function is isotropic if its centroid is 0 and its covariance matrix is the identity. That is, for a random variable X drawn from f,

$$\mathsf{E}(X) = 0 \text{ and } \mathsf{E}(XX^T) = I.$$

The above condition is equivalent to saying that for every unit vector $v \in \mathbb{R}^n$,

$$\int_{\mathbb{R}^n} (v^T x)^2 f(x)\, dx = 1.$$

We can now consider a notion of approximately isotropic, and say that f is C-isotropic if

$$\frac{1}{C} \leq \int_{\mathbb{R}^n} (v^T x)^2 f(x)\, dx \leq C.$$

We tested two approaches for rounding the body, one of which performed significantly better. One way is to round the body once beforehand with respect to the uniform distribution. The second method is to round the body in each volume phase, as in [8]. That is, put the body in approximate isotropic position with respect to the current distribution. Based on experimental results, while both methods were comparable in terms of runtime for the actual rounding, the first method of rounding

the body once beforehand made the volume computation much more efficient. The two primary benefits of rounding the body once beforehand are that fewer volume phases are required to keep $\mathsf{Var}(Y^2)/\mathsf{E}(Y)^2 \leq 1$ (as in Section 5.2.2), and we can use spherical Gaussians for every phase, which experimentally appear to mix faster. The second method will apply a linear transformation T to both the target distribution f and the body K, which can make the distribution f very skew. For a brief numerical justification, consider a 10-dimensional randomly transformed hypercube. If the cube is rounded once beforehand, then ~ 5 phases with $\sim 10k$ steps/phase will give $\leq 20\%$ accuracy. If the cube is rounded in each phase, then ~ 12 phases with $\sim 250k$ steps/phase will give $\leq 20\%$ accuracy, with the later (i.e. most "skew") phases requiring $\sim 500k - 1000k$ steps. So, even for small dimensions, we notice rounding the body once beforehand is orders of magnitude faster.

Our goal is to find a linear transformation T such that the TK is 2-isotropic. We do this by obtaining a sequence of points $\{X_1, X_2, \ldots\}$ using hit-and-run with uniform target distribution over K. We then compute the transformation that will put the points from hit-and-run into isotropic position. If enough steps of hit-and-run are taken, then applying that transformation to the body will put the body in approximately isotropic position. Building upon the work of Bourgain [55] and Rudelson [56], Adamczak et al. [57] showed that $O(n)$ random samples suffice to achieve 2-isotropic position. However, the trajectory of points from hit-and-run are not random samples because subsequent points are highly dependent on each other, and to get an adequate "picture" of the distribution, we will need more than $O(n)$ steps of hit-and-run. We observe that taking $8n^3$ steps of hit-and-run seems to suffice for $n \leq 100$ to achieve 2-isotropic position. However, if we note that we are not converging to a sufficiently round body, we restart the rounding process with twice as many steps per rounding. So, this $8n^3$ samples for rounding is not a fixed parameter, but rather a first guess at how many hit-and-run samples are required.

There is one final complexity to the rounding algorithm, which can be seen in the case of a very long box, say a $10^9 \times 1 \times \ldots \times 1$ box. It would require an impractical number of steps of hit-and-run

to round this body to 2-isotropic position, but we can do it in multiple phases. We let hit-and-run mix for $8n^3$ steps on this $10^9 \times 1 \times \ldots \times 1$ box, and (very roughly speaking) we may observe a distribution similar to a $20 \times 1 \times \ldots \times 1$ box, so when we compute the transformation T that puts the sequence of points $\{X_1, X_2, \ldots, \ldots X_r\}$ into isotropic position, and apply that transformation to the body K, we are left with, approximately, a $5 \cdot 10^7 \times 1 \times \ldots \times 1$ box. We then repeat until the sequence of points $\{X_1, X_2, \ldots, X_r\}$ is in 2-isotropic position.

To compute the transformation that points the sequence of points $\{X_1, X_2, \ldots, X_r\}$ into isotropic position, we compute the singular value decomposition of the points. That is, for the matrix of points M, we find matrices such that $M = USV'$ such that S is a diagonal matrix that contains the n singular values. Assume the minimum singular value is 1 by rescaling. Then, if any singular value is $s > 2$, we scale the body K along that axis by s. We ignore the smaller singular values for numerical stability.

5.2.5 Random Walk

This section can be viewed as the "inner-most" section of our algorithm, so the efficiency of a random walk step significantly affects the total runtime of our volume algorithm. For polytopes and ellipsoids, we describe here how to efficiently compute the chord. For other convex bodies, such as zonotopes in Section 5.4.4, the chord computation is less efficient and has a drastic effect on the overall runtime. Once the chord is computed, we then need to efficiently sample from the chord according to a Gaussian distribution. We now discuss how each of the two steps are implemented in our algorithm.

For the majority of our computational results in Section 5.4, we consider the hit-and-run random walk. We choose this walk because it is the fastest walk with provable convergence guarantees for samples. However, there is computational evidence that coordinate hit-and-run might be faster in practice. We present such experimental justification in Section 5.4.7. For many natural convex bodies, coordinate hit-and-run can improve the number of arithmetic operations per step by a factor

of n, but its mixing time is currently open.

Chord Computation

Problem: Given a random direction $u \in \mathbb{R}^n$ and a point $x \in K$, determine the two endpoints (x^-, x^+) of the line $x + \alpha u, \alpha \in \mathbb{R}$ intersected with the convex body K.

Solution Outline: One simple approach is to binary search for the positive α that intersects K (likewise for the negative α value), which only relies on having a membership oracle. However, if we know

$$K = P \cap E = \{x : Ax \leq b\} \cap \{x : (x - v)^T Q^{-1}(x - v) \leq 1\},$$

we can use this description to get a more efficient algorithm by explicitly computing the intersection points of $x + \alpha u$ with both P and E, and taking the closest intersection points in each direction as (x^-, x^+).

For the polyhedron $P = \{x : Ax \leq b\}$, we can compute the distance to each hyperplane of P (row,entry pair of A, b) in the $+u$ and $-u$ direction, and take the minimum distance for each direction. Letting the rows of A and entries of b be labeled $1, \ldots, r$, the values (α_P^-, α_P^+) are given by the following:

$$\alpha_P^- = \max_{1 \leq i \leq r : A_i \cdot u \leq 0} \frac{b_i - A_i \cdot x}{A_i \cdot u} \qquad \alpha_P^+ = \min_{1 \leq i \leq r : A_i \cdot u > 0} \frac{b_i - A_i \cdot x}{A_i \cdot u}. \tag{5.4}$$

For the ellipsoid $E = \{x : (x - v)^T Q^{-1}(x - v) \leq 1\}$, we would again like to determine the two points that intersect E along the line $x + \alpha_E u$. To simplify notation, we will assume $v = 0$ and shift x accordingly. We then would like to solve the equation $(x + \alpha_E)^T Q^{-1}(x + \alpha_E u) = 1$, which is quadratic in α_E and yields a solution pair (α_E^-, α_E^+):

$$u^T Q^{-1} u \cdot \alpha_E^2 + \left(u^T Q^{-1} x + x^T Q^{-1} u\right) \cdot \alpha_E + x^T Q^{-1} x - 1 = 0. \tag{5.5}$$

Combining the values from 5.4 and 5.5, the chord for the convex body is then $(x^-, x^+) = (x + \max\{\alpha_P^-, \alpha_E^-\}u, x + \min\{\alpha_P^+, \alpha_E^+\}u)$. Note that all of the values may not exist, for instance if the polyhedron is unbounded, but if a particular $\alpha_{P/E}^{+/-}$ does not exist, we simply ignore it.

Sampling from the Chord

Problem: Given a description of a chord as $\ell = (u, v) \subset \mathbb{R}^n$, generate a sample according to a density proportional to $f(x) = e^{-a\|x\|^2}$ restricted to ℓ.

Solution outline:

- If $\|u - v\| \geq 2/\sqrt{2a} \rightarrow$ return **GaussianSample**(chord: (u, v), target distribution: f).
- Else $\rightarrow$ return **UniformSample**(chord: (u, v), target distribution: f).

GaussianSample: Since this density is spherical Gaussian, its restriction to the 1-dimensional chord will be a 1-D Gaussian with variance $1/(2a)$. The projection of 0 onto the line extending ℓ will be the mean of this Gaussian, call it μ_Z. Note we can map the points u and v to this 1-dimensional Gaussian in terms of their distance and direction from μ_Z. We can then sample $Z \sim \mathcal{N}(0, 1/(2a))$ until the point $\mu_Z + Z \cdot \overrightarrow{uv}$ lies on $\ell = (u, v)$, where $\overrightarrow{uv}$ is a unit vector in the direction from u to v. Then, return $\mu_Z + Z \cdot \overrightarrow{uv}$.

UniformSample: Another way to sample from this chord is by simple rejection sampling. We enclose the distribution by a rectangle whose width is $\|u - v\|$ and height is $\max_{x \in (u,v)} e^{-a\|x\|^2}$. We then sample uniformly from the box and reject the point if it lies outside the region enclosed by $e^{-a\|x\|^2}$.

To sample efficiently from the chord, we use a combination of these two methods. Note that the two methods perform well in different cases. The first method will perform better when the Gaussian is sharp and mostly contained inside the chord, whereas the second method will perform better when the Gaussian is flat. In the first method, the success probability is the measure of

the Gaussian inside (u, v), and for the second method, the success probability is the ratio of the average value to the maximum value of $e^{-a\|x\|^2}$ in (u, v). We can compute these values, and select the approach that has the highest acceptance probability. However, computing the exact values is somewhat expensive compared to actually generating the random samples. We achieve better performance by a rough approximation which performs well in practice: if the chord length is more than 2 standard deviations long, then we use the Gaussian method. In practice, using this rough approximation for the Gaussian and uniform sampler, we observe an average failure probability of at most 20% over all the test bodies in Section 5.4.

5.2.6 Sampling and Multiple Threads

An unavoidable problem with hit-and-run is that subsequent points of a trajectory are very dependent. In recent volume computation algorithms [7, 8, 32], this dependence was handled by allowing the trajectory to mix for some $c(\varepsilon)$ steps before collecting a new "sample point" to obtain ε-independence between subsequent points. The drawback of this approach is that $c(\varepsilon)$ could be very large. In fact, it can be shown to be $\Omega(n^2)$ [5]. Another approach is to use every point along the chain, with the hope that the greater number of sample points outweighs the dependency. There is theoretical evidence in favor of this approach for different applications of Markov chains [58], and we observe it to be true experimentally for estimating volume with hit-and-run. We then test for convergence by the heuristic approach outlined in Section 5.2.3.

Another natural question is if we can obtain a better estimate of the volume ratio in a single phase by using multiple trajectories of our Markov chain. That is, concurrently run t independent threads of hit-and-run in a circular queue–step thread 1, step thread 2, ..., step thread t, step thread 1, etc. The following intuition gives some insight into how for a fixed number of total steps s (i.e. s/t steps per thread), the number of threads could affect our answer. Running t threads can be thought of as running $2t$ threads, where thread $2i$ starts from the last point of thread $2i-1$. The last point of this thread will be fairly mixed as opposed to whatever point thread $2i$ would start with if

we were actually using $2t$ threads. Therefore, using a smaller number of threads decreases the total mixing time, but a smaller number of threads increases the total dependence between the sample points. So there is a natural trade-off between dependence and mixing time, and it is unclear what the appropriate selection of threads should be. In Section 5.4.6, we see that mixing times seems to have a more significant effect than the dependence, and we should therefore use a very small number of threads. Based on the experiments, we use 5 threads, a constant number independent of dimension, throughout our implementation.

5.3 Sampling Algorithm

Here we consider a different, but related, problem of generating nearly independent random points from a convex body. In practice, it is generally easy to generate a reasonable starting point for the walk (say, the center of the largest ball contained in the convex body). We can then run the random walk starting from this point, but how long should we run the walk before collecting a sample point? In the theoretical literature [19, 50], mixing time bounds are derived in terms of running the random walk for some fixed number of steps. Unfortunately, the current best bounds are far too high for practical use. Therefore, here we introduce a statistical heuristic that could be helpful for declaring that the random walk is mixed.

Once we have a single random point from the target distribution, subsequent independent random points may be desired. For the purposes of computing volume, as outline in Section 5.2.3, we use a large number of dependent random points, which appears to give a better estimate for the volume. Thus in practice, the "dependencies" between the random points are seemingly unbiased with regard to the volume estimate. However, for other applications, it may be desired that the dependence between subsequent sample points be negligible.

5.3.1 Detecting Convergence

Problem: Let f be a logconcave target distribution, $\varepsilon > 0$ error parameter, and let $X_0 \in \mathbb{R}^n$ such

that $f(X_0) > 0$. Let $X_1, X_2, X_3, \ldots$ be a sequence of points from some random walk with unique stationary distribution f. Return a point X_k such that

$$\|X_k - \pi_f\|_{TV} \leq \varepsilon$$

and X_0 and X_k are ε-independent. We use π_f to denote the associated distribution of f.

Solution outline: We first note that the problem stated in such generality is likely to be computationally infeasible. Additional properties of the random walk would be useful in declaring convergence, and we mention some potentially useful properties in Section 4.6 and Section 5.5.

We briefly mention the following statistical test that we use to declare convergence. Note that neither of these tests are used in our volume algorithm.

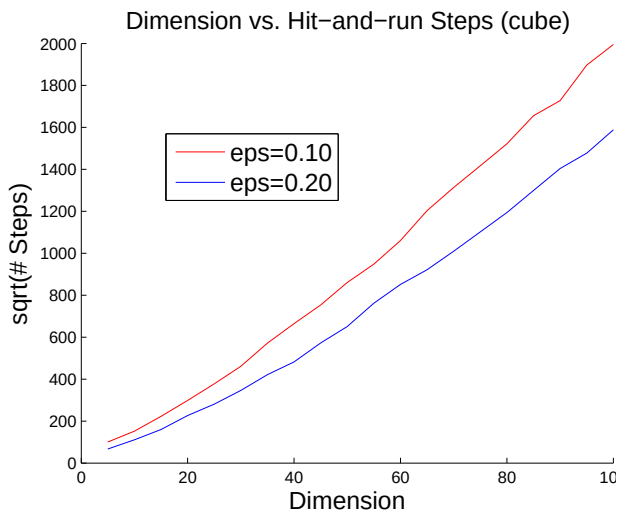
Proportion on halfspace: We assume that the starting distribution is suitably close to the target distribution, i.e. gives a warm start. This step is important for ensuring that for a random hyperplane, a constant fraction of mass lies on both sides of the hyperplane. Then, generate a uniformly chosen random halfspace H through the starting point. Then, measure the proportion of samples that lie in H, and monitor this value as the random walk progresses. Then, use the sliding window test in Section 5.2.3 to declare once this quantity has "converged".

5.4 Computational results

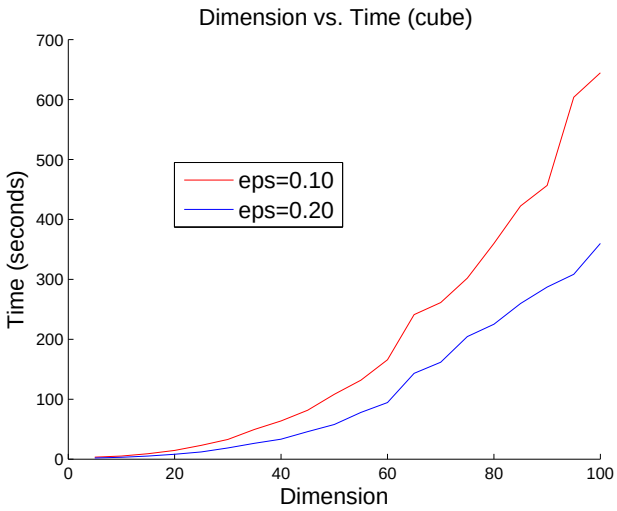
In this section, we will present numerical results of our algorithm over a test set of convex bodies, which are described in Section 5.0.1.

5.4.1 Complexity

The plots in Figure 5.4 and Figure 5.5 show how our program scales with dimension for the n-dimensional **Cube**. We give results over two different hardware configurations. We plot the num-



(a) In the above graph, the correlation coefficient (R-value) is 0.997 for both lines.



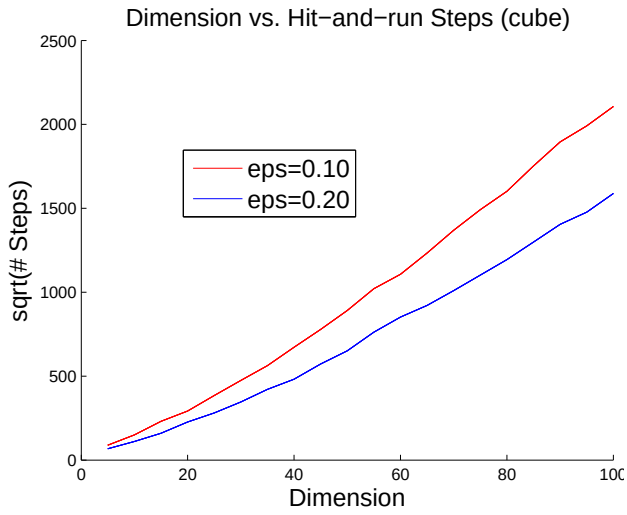
(b) The above runtimes grow at roughly $n^{2.5}$.

Figure 5.4: The above data was computed on a 64-bit Windows 8 machine with a i7-3630QM (8 threads, 2.40 GHz) processor and 8GB RAM using MATLAB R2013a.

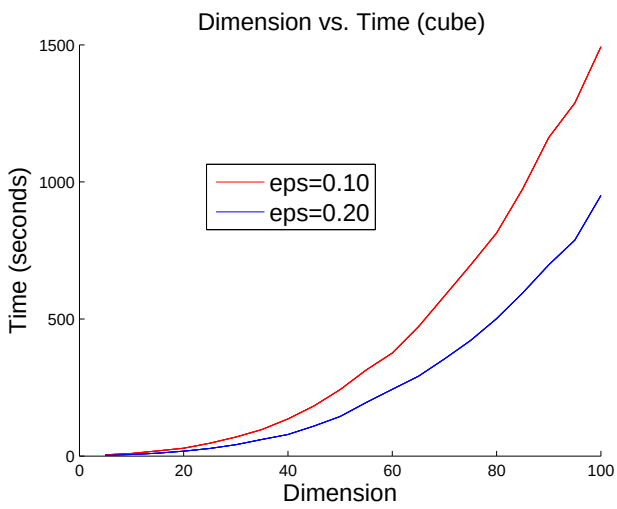
ber of hit-and-run steps and total computation time, for $n = 5, 10, \ldots, 95, 100$. For each value of n, 5 trials were performed. The figures suggest that the volume of the cube can be computed in $O^*(n^2)$ membership oracle calls. For the time complexity, note that each step of hit-and-run requires a chord computation and sampling along that chord. The chord computation requires $\Theta(n^2)$ arithmetic operations, while the sampling steps requires $O(1)$ time. So, we would expect that the time grows as $n^2 \times n^2 = O(n^4)$ based on the number of oracle calls, but the hidden constant in the sampler is quite high. For these small values of n, we instead observe the time to grow roughly like $n^{2.5}$. We note that the current best theoretical complexity is $O^*(n^3)$ oracle calls and $O^*(n^5)$ arithmetic operations [50].

5.4.2 Accuracy and Times

The tables in this section show the numerical results of our implementation on our set of 5 bodies, in 10, 50, and 100 dimensions. The tables show the mean and standard deviation over a certain number of trials, where the standard deviation is reported as a percentage of the mean. Also



(a) As expected, this plot is nearly identical to Figure 5.4a and the number of steps taken is consistent across a different hardware configuration.



(b) The computation time appears to be roughly double that of Figure 5.4b on a different machine, but grows at approximately the same rate.

Figure 5.5: The above data was computed on a 64-bit CentOs 5.6 machine with a X5570 (8 threads, 2.93 GHz) processor and 48 GB RAM using MATLAB R2014b.

included are the average time for one run of the algorithm and the average number of hit-and-run steps that were taken. The total number of hit-and-run steps is equivalent, up to a logarithmic factor, to the number of membership oracle calls, and the best known theoretical algorithm accurately estimates volume in $O^*(n^4)$ membership oracle calls [8]. The majority of the experimental data here was computed on Georgia Tech's Jinx computing cluster, and the computation times given were computed with a i7-3630QM (quad-core, 2.40 GHz) processor and 8GB 1600 MHz RAM using the MATLAB R2013a profiler. Each time given was averaged over 5 trials.

Volume, no rounding

In this section, we will give experimental results for convex bodies without the rounding step. The rounding step is an expensive preprocessing step that ensures the efficiency of our volume algorithm, so these bodies will run much faster than bodies that need to be rounded. We observe that the **Cube** and **Half-Ball** perform noticeably better than the **Isotropic Simplex**. There is theoretical

motivation for a simplex being a "bad" case for sampling and volume computation. For one, the value of R for an isotropic body K that contains the unit ball, where R is the minimum radius ball that contains K, is maximized when K is a simplex [5], where R is essentially n. Most of the mass of the **Isotropic Simplex** will lie near its $n+1$ corners, which are sharper than the vertices of the **Cube**. In a rough sense, sharp corners perform poorly with hit-and-run; it will take longer to visit a sharp corner than, say, a region in the middle of the body. But once a sharp corner is visited, hit-and-run will take small steps that stay near the corner before eventually drifiting away. Since most of the volume of a simplex is contained near its corners, it may take more steps of hit-and-run to get an accurate estimate of its volume.

Table 5.1: **10-Dimensional Results**

Body	**Actual Vol**	**Mean**	**$\frac{\text{Std dev}}{\text{Mean}}$**	**Time**	**# Steps**
Cube	$1.02 \cdot 10^3$	$9.91 \cdot 10^2$	0.165	$6.48 \cdot 10^0$ s	$1.22 \cdot 10^4$
Isotropic Simplex	$1.47 \cdot 10^{-6}$	$1.34 \cdot 10^{-6}$	0.249	$9.10 \cdot 10^0$ s	$1.92 \cdot 10^4$
Half-Ball	$1.28 \cdot 10^0$	$1.22 \cdot 10^0$	0.164	$1.09 \cdot 10^1$ s	$1.73 \cdot 10^4$

Numerical results for 10-dimensional bodies using 1000 trials with error parameter $\varepsilon = 0.20$.

Table 5.2: **50-Dimensional Results**

Body	Actual Vol	Mean	$\frac{\text{Std dev}}{\text{Mean}}$	Time	# Steps
Cube	$1.12 \cdot 10^{15}$	$1.09 \cdot 10^{15}$	0.189	$1.11 \cdot 10^{2}$ s	$4.59 \cdot 10^{5}$
Isotropic Simplex	$3.85 \cdot 10^{-64}$	$3.21 \cdot 10^{-64}$	0.324	$2.02 \cdot 10^{2}$ s	$6.62 \cdot 10^{5}$
Half-Ball	$8.65 \cdot 10^{-14}$	$8.36 \cdot 10^{-14}$	0.159	$1.20 \cdot 10^{2}$ s	$3.95 \cdot 10^{5}$

Numerical results for 50-dimensional bodies using 200 trials with error parameter $\varepsilon = 0.20$.

Table 5.3: **100-Dimensional Results**

Body	Actual Vol	Mean	$\frac{\text{Std dev}}{\text{Mean}}$	Time	# Steps
Cube	$1.26 \cdot 10^{30}$	$1.23 \cdot 10^{30}$	0.172	$4.68 \cdot 10^{2}$ s	$2.12 \cdot 10^{6}$
Isotropic Simplex	$1.77 \cdot 10^{-157}$	$1.41 \cdot 10^{-157}$	0.478	$8.30 \cdot 10^{2}$ s	$3.81 \cdot 10^{6}$
Half-Ball	$1.18 \cdot 10^{-40}$	$1.16 \cdot 10^{-40}$	0.202	$4.56 \cdot 10^{2}$ s	$1.72 \cdot 10^{6}$

Numerical results for 100-dimensional bodies using 100 trials with error parameter $\varepsilon = 0.20$.

Volume with rounding

The following bodies go through a rounding preprocessing step to put the body in approximate isotropic position. To illustrate the rounding complexity, the "# Steps" column will now have two lines: the first line will be "# Volume Steps", and the second line will be "# Rounding Steps". Note

the substantial decrease in efficiency for these bodies, as compared to the round bodies in Section 5.4.2, where this increase in both runtime and membership oracle calls is due to the rounding preprocessing step, which begins to take a prohibitively long time for bodies much higher than 50 dimensions.

After the rounding phase, the efficiency of the volume computation is comparable, but slightly worse, than the corresponding bodies in the previous section. For instance, the 50-dimensional **Transformed Cube** required around 10% more hit-and-run steps than the 50-dimensional **Cube** and had a higher standard deviation in its computed volume; we observe a similar relationship between the 50-dimensional simplices. This behavior is in line with what we expect, because the **Cube** and **Isotropic Simplex** are in isotropic position, whereas the rounding phase will only achieve *approximate* isotropic position for the **Transformed Cube** and **Standard Simplex**.

Table 5.4: **10-Dimensional Results**

Body	**Actual Vol**	**Mean**	**$\frac{\text{Std dev}}{\text{Mean}}$**	**Time**	**# Steps**
Transformed Cube	$5.20 \cdot 10^{0}$	$4.94 \cdot 10^{0}$	0.192	$9.33 \cdot 10^{0}$ s	$1.73 \cdot 10^{4}$ $2.72 \cdot 10^{4}$
Ellipsoid	$2.55 \cdot 10^{2}$	$2.46 \cdot 10^{2}$	0.130	$2.85 \cdot 10^{1}$ s	$1.01 \cdot 10^{4}$ $2.40 \cdot 10^{4}$
Standard Simplex	$2.76 \cdot 10^{-07}$	$2.46 \cdot 10^{-07}$	0.312	$8.60 \cdot 10^{0}$ s	$2.40 \cdot 10^{4}$ $3.13 \cdot 10^{4}$

Numerical results for 10-dimensional bodies using 1000 trials with error parameter $\varepsilon = 0.20$.

Table 5.5: **50-Dimensional Results**

Body	Actual Vol	Mean	Std dev / Mean	Time	# Steps
Transformed Cube	$5.57 \cdot 10^{-17}$	$5.24 \cdot 10^{-17}$	0.286	$1.12 \cdot 10^{3}$ s	$4.93 \cdot 10^{5}$ $6.19 \cdot 10^{6}$
Ellipsoid	$1.73 \cdot 10^{-11}$	$1.69 \cdot 10^{-11}$	0.096	$4.75 \cdot 10^{2}$ s	$1.52 \cdot 10^{5}$ $2.30 \cdot 10^{6}$
Standard Simplex	$3.29 \cdot 10^{-65}$	$2.84 \cdot 10^{-65}$	0.691	$8.21 \cdot 10^{2}$ s	$8.16 \cdot 10^{5}$ $6.39 \cdot 10^{6}$

Numerical results for 50-dimensional bodies using 200 trials with error parameter $\varepsilon = 0.20$.

Improving accuracy

In the above tables, the simplices had the highest standard deviation relative to the mean volume. The standard deviation divided by the mean will give a rough idea of how accurate the reported volume is. The worst case was the 50-dimensional standard simplex, where the observed standard deviation divided by the mean is 0.691, much higher than the target relative error of 0.20. For the results in Table 5.5, only 70 of 200 trials were within 20% of the actual volume. We can improve this accuracy by averaging the result over multiple trials. The average volume over 200 trials was within 0.20 relative error of the actual volume for the 50-dimensional standard simplex; this observation also holds true for all other test bodies reported in this section. The simplex is conjectured to be the body with the smallest isoperimetric coefficient, i.e., where the random walk mixes slowest.

There are two general ways to do increase the probability of an accurate answer. The first is to average over multiple trials. For the 50-dimensional standard simplex, if we group the trials

into pairs and average the volume of the two trials, the probability that we are within our target relative error is now 0.49, as opposed to 0.35 when just considering a single trial. If we group the trials into groups of 10, then this probability improves to 0.65. Another heuristic to obtain a more accurate volume estimate is to lower the error parameter provided to the volume algorithm, until the observed standard deviation divided by mean is approximately the desired relative error.

The main reason we did not incorporate either of these methods in our implementation is that for all other test bodies it seems superfluous and results in a constant factor increase in the runtimes.

5.4.3 Birkhoff Polytope

An interesting application of our volume algorithm was for computing the volume of the Birkhoff polytope. The nth Birkhoff polytope B_n is the polytope of all $n \times n$ doubly stochastic matrices; equivalently, it is the perfect matching polytope of the complete bipartite graph $K_{n,n}$. This polytope has a number of nice combinatorial properties, with one important question being its volume [59] [60] [61] [62] [63]. There has been previous work on computing the volume of B_n for small values of n, where specialized algorithms were developed for this specific polytope [64] [65]. The most recent work of [65] computed B_n for $n = 10$ through a distributed algorithm, with a total computation time of 17 years at 1GHz. To our knowledge, the values for $n > 10$ have not yet been obtained, as the current approaches are too computationally expensive for any value of n higher than 10. We show here that if you relax the requirement for an exact answer, our volume algorithm can obtain a reasonable estimate for the value for $n = 15$ within a few hours.

We now give a more complete description of B_n using its formulation as all $n \times n$ doubly stochastic matrices. Define n^2 variables X_{ij} for $i, j \in \{1, \ldots n\}$ as the values assigned to the corresponding entries of a doubly stochastic matrix. The following equations then define the polytope B_n:

$$\sum_{i=1}^{n} X_{ij} = 1, \quad j \in \{1, \ldots, n\} \tag{5.6}$$

$$\sum_{j=1}^{n} X_{ij} = 1, \quad i \in \{1, \ldots, n\} \tag{5.7}$$

$$X_{ij} \geq 0, \quad i, j \in \{1, \ldots, n\} \tag{5.8}$$

Note that while $B_n \subset \mathbb{R}^{n^2}$, the dimension of the polytope is lower: $\dim(B_n) = (n-1)^2$. We have $2n$ equality constraints above on n^2 variables, but one is redundant. Therefore, to compute the volume of B_n, we compute the volume of the $n^2 - (2n-1) = (n-1)^2$ dimensional subspace spanned by equations (5.6)-(5.7), restricted to the positive orthant by equation (5.8).

One aspect of the Birkhoff polytope that benefits our approach is that the polytope is already round. In Section 5.4.2, we see that the rounding preprocessing step for our convex body is quite expensive and dominates computing its volume. By avoiding this rounding step for the Birkhoff polytope, we can more easily go to high dimensions. Here we go up to $n = 15$, i.e. a 196-dimensional polytope, and each run completes within 5 hours. If desired, it should be computationally feasible to get accurate estimates for $n = 20$ or $n = 25$ within days or weeks, respectively. In Table 5.6, we show the results of 100 independent runs of our volume algorithm for $B_2, \ldots, B_{15}$ with $\varepsilon = 0.20$. In Figure 5.6, we show how our computed volume converges as the number of steps per phase, over multiple trials. In Figure 5.7, we show a similar plot for $n = 15$, where the volume is unknown. The plot indicates that the volume of B_{15} should be close to $5.5 \cdot 10^{-145}$.

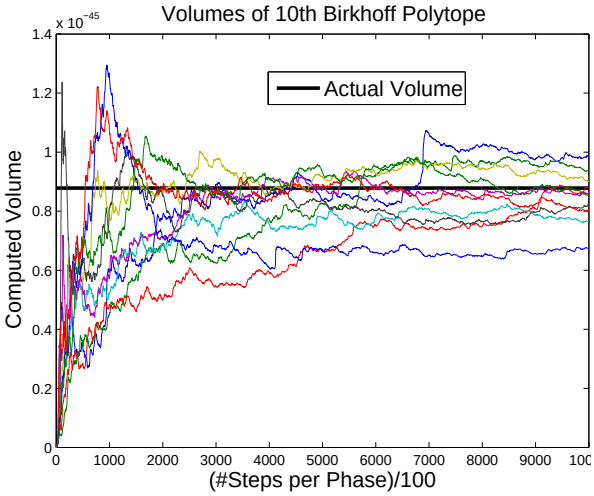
Table 5.6: **Birkhoff Polytope Results**

n	**Actual Vol**	**Mean**	**Std dev / Mean**	**Time**	**# Steps**
2	$2.00 \cdot 10^{+00}$	$2.01 \cdot 10^{+00}$	0.044	$4.80 \cdot 10^{-1}$ s	$8.87 \cdot 10^{2}$
3	$1.12 \cdot 10^{+00}$	$1.14 \cdot 10^{+00}$	0.146	$1.45 \cdot 10^{0}$ s	$4.41 \cdot 10^{3}$
4	$6.21 \cdot 10^{-02}$	$5.79 \cdot 10^{-02}$	0.209	$3.68 \cdot 10^{0}$ s	$1.33 \cdot 10^{4}$
5	$1.41 \cdot 10^{-04}$	$1.32 \cdot 10^{-04}$	0.233	$8.00 \cdot 10^{0}$ s	$4.09 \cdot 10^{4}$
6	$7.35 \cdot 10^{-09}$	$6.67 \cdot 10^{-09}$	0.211	$2.08 \cdot 10^{1}$ s	$1.12 \cdot 10^{5}$
7	$5.64 \cdot 10^{-15}$	$5.34 \cdot 10^{-15}$	0.240	$4.66 \cdot 10^{1}$ s	$2.82 \cdot 10^{5}$
8	$4.42 \cdot 10^{-23}$	$3.92 \cdot 10^{-23}$	0.290	$1.18 \cdot 10^{2}$ s	$6.27 \cdot 10^{5}$
9	$2.60 \cdot 10^{-33}$	$2.36 \cdot 10^{-33}$	0.289	$2.33 \cdot 10^{2}$ s	$1.26 \cdot 10^{6}$
10	$8.78 \cdot 10^{-46}$	$8.04 \cdot 10^{-46}$	0.261	$5.39 \cdot 10^{2}$ s	$2.35 \cdot 10^{6}$
11	???	$1.26 \cdot 10^{-60}$	0.238	$8.62 \cdot 10^{2}$ s	$4.07 \cdot 10^{6}$
12	???	$7.17 \cdot 10^{-78}$	0.254	$1.37 \cdot 10^{3}$ s	$6.49 \cdot 10^{6}$
13	???	$1.21 \cdot 10^{-97}$	0.281	$2.15 \cdot 10^{3}$ s	$1.04 \cdot 10^{7}$
14	???	$5.52 \cdot 10^{-120}$	0.292	$3.51 \cdot 10^{3}$ s	$1.59 \cdot 10^{7}$
15	???	$4.97 \cdot 10^{-145}$	0.277	$5.67 \cdot 10^{3}$ s	$2.35 \cdot 10^{7}$

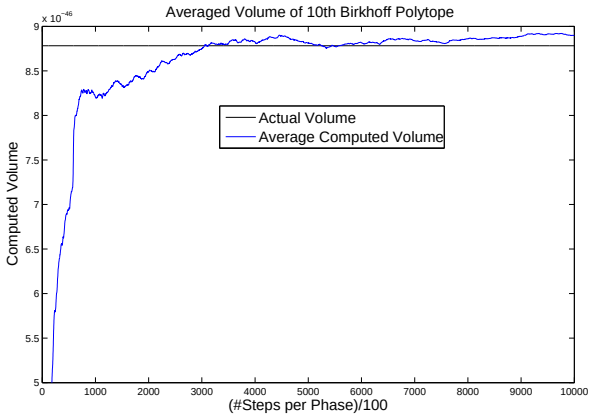
Numerical results for Birkhoff polytopes using 100 trials with error parameter $\varepsilon = 0.20$. Computational times were on a 64-bit Windows 8 machine with a i7-3630QM (8 threads, 2.40 GHz) processor and 8GB RAM using MATLAB R2013a.

5.4.4 Zonotopes

Another application where our algorithm can efficiently provide an accurate estimate, where the exact volume is difficult to compute, is with zonotopes. A zonotope is defined as the Minkowski sum of a set of line segments. Let $\{v_i | 1 \leq i \leq m\}$ be a set of line segments, where each v_i lives in



(a) A plot of 10 independents trials for B_{10} and the computed volume of each trial as a function of the number of steps per volume phase.



(b) A plot of the average computed volume over 100 independent trials for B_{10}. We see that taking the average over trials provides a more accurate estimate.

Figure 5.6

$\mathbb{R}^n$. The zonotope Z is then defined as

$$Z = \{\sum_{i=1}^{m} \lambda_i v_i | 0 \leq \lambda_i \leq 1\}.$$

To compute its exact n-dimensional volume deterministically, consider a set of n line segments and compute the volume of the parallelopiped formed by these n line segments. Then, sum over all $\binom{m}{n}$ sets of n segments to obtain the volume for the zonotope Z. The volume of each parallelopiped is the absolute value of the determinant of the matrix of the n line segments. However, for a fixed dimension n, the runtime of this algorithm will scale as $O(m^n)$. Even for a somewhat small dimension $n = 10$, when the number of line segments m grows to, say, 50 or 100, this approach quickly becomes inefficient. For a further discussion of zonotope volume, we refer the reader to [43], where it is shown that exactly computing the volume of a zonotope is #P-Hard.

The description of a zonotope is different than a polytope or ellipsoid. To use our volume algorithm, we need to compute the chord in one step of hit-and-run. That is, given a current point

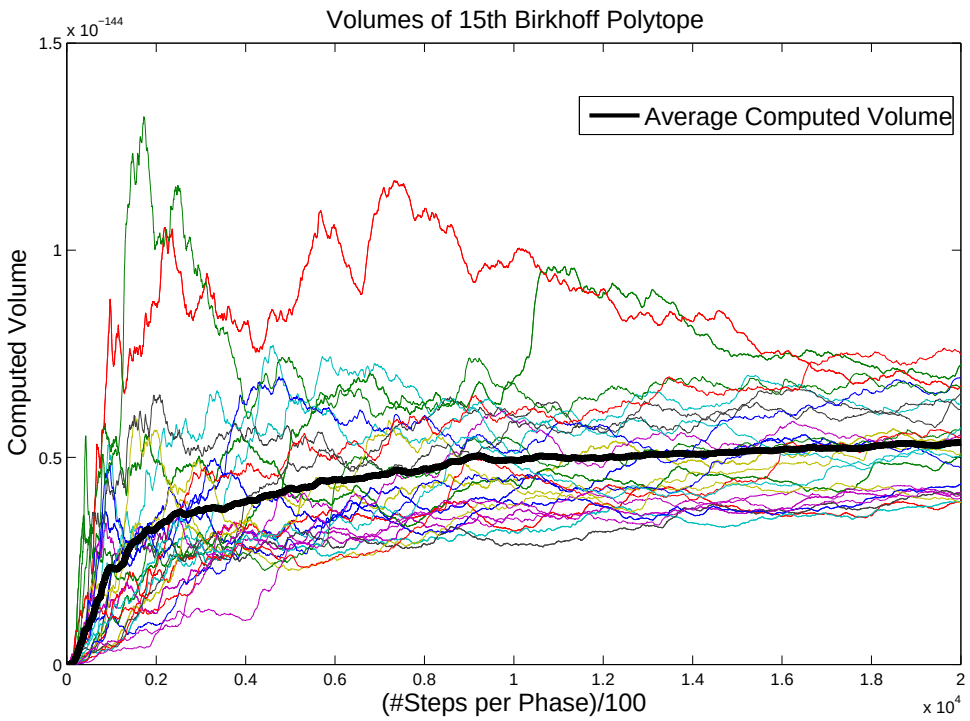


Figure 5.7: A plot of 25 independents trials for B_{15} and the computed volume of each trial as a function of the number of steps per volume phase. The average of the trials is included.

$x \in Z$ and a direction $u \in \mathbb{R}^n$, determine the points at which the line $x + \alpha u$, $\alpha \in \mathbb{R}$, intersects the boundary of Z. We can formulate this as a linear program:

$$\min \quad \alpha \tag{5.9}$$

s.t.

$$x = \sum_{i=1}^{m} \lambda_i v_i - \alpha u \tag{5.10}$$

$$0 \leq \lambda_i \leq 1 \qquad 1 \leq i \leq m \tag{5.11}$$

Solving (5.9)-(5.11) will produce one point at which the line intersects the boundary (i.e. one endpoint of the hit-and-run chord). To find the other, we simply minimize $-\alpha$ in the objective

function (5.9). The chord computation therefore requires solving two linear programs. We note that this is much less efficient than the corresponding algorithms for polytopes or ellipsoids, and in the below computational results, the time to solve these linear programs heavily dominates the runtime.

To generate a zonotope with m line segments, we start with the generators $e_1, \ldots, e_n$ for the unit cube and then add $m - n$ random unit vectors. The zonotope constructed in this fashion may not necessarily be round, so we perform a rounding preprocessing step before computing our estimate for the volume. In Table 5.7, we give computational results for computing volumes of zonotopes. The dimension $n = 10$ is fixed, and we increase the number of line segments m. For $m = 20$, we see that the exact algorithm is much more efficient than our algorithm. However, for $m = 50$, the exact algorithm's runtime drastically increases, while our approximation takes roughly the same amount of time to compute. The exact answer for $m = 50$ took approximately 16 computation hours. The data for $m = 100$ was also included for our algorithm, along with an estimated time for the exact answer, which is approximately 31 years.

Table 5.7: **Zonotope Results**

m	**Exact Volume**	**Exact Time**	**Approx Volume**	**Approx Time**	**# Steps**
20	$4.93 \cdot 10^3$	1.04 s	$4.10 \cdot 10^3$	$3.01 \cdot 10^3$	$9.58 \cdot 10^4$
50	$1.33 \cdot 10^8$	$5.78 \cdot 10^4$ s	$1.17 \cdot 10^8$	$3.73 \cdot 10^3$	$1.04 \cdot 10^5$
100	???	$9.74 \cdot 10^8$ s	$1.62 \cdot 10^{11}$	$4.90 \cdot 10^3$	$1.25 \cdot 10^5$

Numerical results for zonotopes in $\mathbb{R}^{10}$ with 20, 50, and 100 line segments. For each value of m, the data is averaged over 10 independent trials. Computational times were on a 64-bit Windows 8 machine with a i7-3630QM (8 threads, 2.40 GHz) processor and 8GB RAM using MATLAB R2013a.

5.4.5 Convergence

Here we will give experimental results for the rate at which the relative error decreases as a function of the number of hit-and-run steps per volume phase. For a fixed body K and dimension n, we expect that $O(\varepsilon^{-2})$ hit-and-run steps are required to obtain relative error ε [8]. We observe this to be true in expectation in Figure 5.8 for the 20-dimensional **Cube**, where we plot the following quantity as a function of the number of steps per volume phase t:

$$\varepsilon(t) = \mathsf{E}\left(\frac{|V(t) - 2^{20}|}{2^{20}}\right),$$

where $V(t)$ is the observed volume after t hit-and-run steps per phase. That is, $\varepsilon(t)$ is the expected error after taking t steps per phase, and we estimate $\varepsilon(t)$ by averaging over $1,000$ independent trials.

We also observe an additional property of how our estimate converges to the true value, where the estimated value is monotonically increasing, in expectation, as a function of the number of hit-and-run steps. This behavior can be best viewed in Section 5.4.6, Figure 5.9 where a large number of threads estimates the behavior of our estimated volume in expectation, and we see that a large number of threads clearly exhibits the monotonicity property for the 20-dimensional **Cube**. Also, all of the reported volumes in Section 5.4.2 are lower than the actual volume, which suggests we approach the true volume from below.

5.4.6 Number of threads

In this section, we give experimental results on the 20-dimensional **Cube** for how the number of threads, as discussed in Section 5.4.6, affects the accuracy of the volume estimate. Figure 5.9 illustrates the effect of mixing time on the volume estimate. We see that for a greater number of threads, it takes longer to approach the true volume, but it does so more "smoothly" because nearby samples are less dependent. Figure 5.9 also suggests that the computed ratio approaches

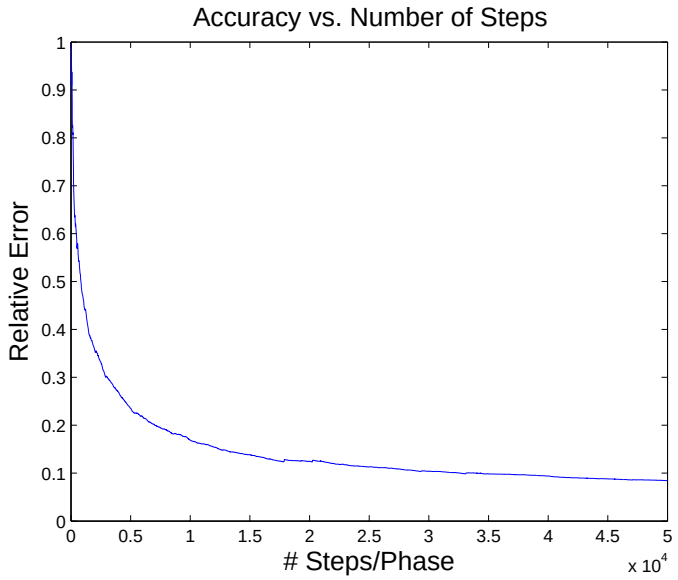


Figure 5.8: In the above plot, the relative error ε decreases as $O(\varepsilon^{-2})$ as a function of the number of steps per phase. The data was averaged over 1,000 trials for the 20 dimensional cube.

the true ratio monotonically, in expectation, but this fact remains to be proven. In Figure 5.10, we see the computed volume after $500,000$ steps per volume phase. It suggests that using a smaller number of threads will provide the most accurate estimate for a fixed number of steps. Using 1 thread will provide the fastest mixing time, but at the cost of high dependence between subsequent sample points; 5 to 10 threads seems to provide an appropriate balance between mixing time and dependence.

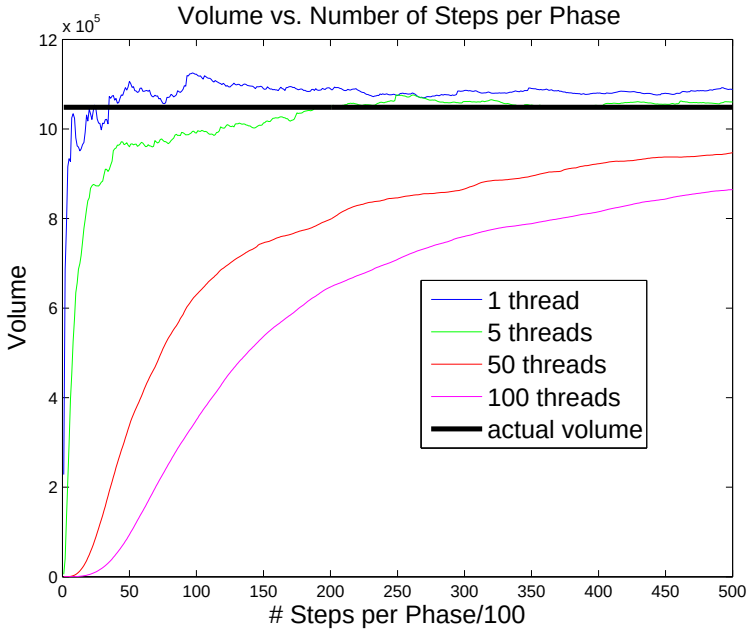


Figure 5.9: In the above graph, we see how the volume estimate approaches the true volume, for varying numbers of threads. The volumes are averaged over 50 trials.

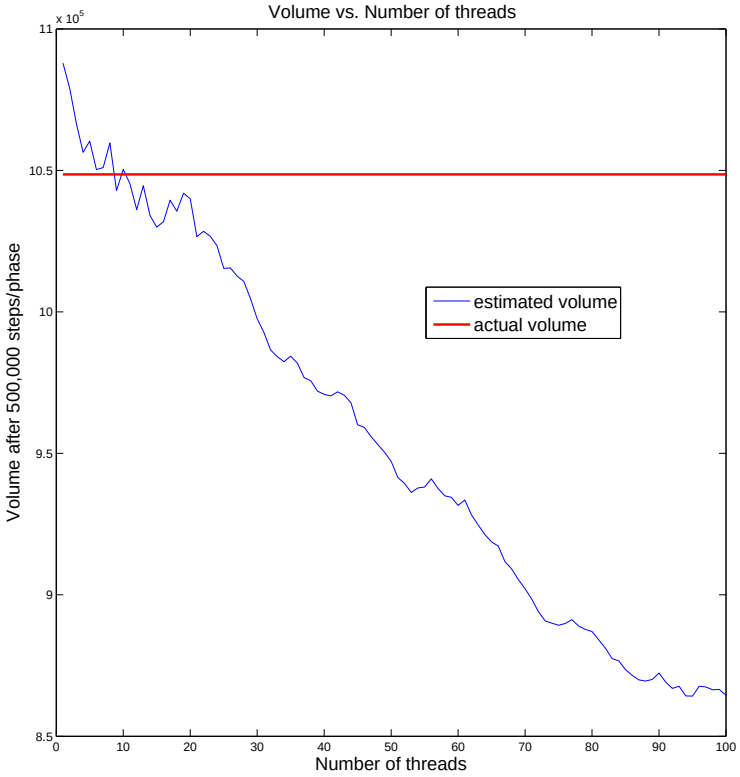


Figure 5.10: This is a snapshot of the computed volumes after $500,000$ steps per phase, for # threads$= 1, \ldots, 100$. The volumes are averaged over 50 trials.

5.4.7 Coordinate Hit-and-run

In this section, we consider the performance of our volume algorithm using a different random walk, coordinate hit-and-run (Section 2.3.4). The only difference between this walk and the hit-and-run random walk we discussed previously is that when choosing a random direction through the current point, only axis-aligned directions are considered. This difference yields a substantial savings for the chord computation step by maintaining the "slacks" of the current point of the random walk. We now briefly overview the implementation of coordinate hit-and-run which yields an $O(n)$ factor (amortized) savings in the number of arithmetic operations of the walk. A single step of hit-and-run can be implemented in $O(mn)$ arithmetic operations, as outlined in Section 5.2.5.

Let x denote the current point of the walk in a polytope $P = \{x|Ax \leq b\}$. Initialize $s = b - Ax$. Suppose coordinate hit-and-run chose to move along coordinate i. Let e_i denote the coordinate vector with a 1 in the i-th coordinate and all other zeros. The bounding facets of P for the line can then be computed by taking the maximum and minimum values of s_j/A_{ji}, which can be implemented in $O(m)$ time. After sampling along this chord to obtain a new point y, update s by adding $(A)_i \cdot (x_i - y_i)$, where $(A)_i$ denotes the i-th column on A. This update steps also requires $O(m)$ time. A similar method can be used for the ellipsoid to obtain an $O(n)$ improvement for the chord computation.

While a step of coordinate hit-and-run can be implemented a factor of $O(n)$ faster than hit-and-run, is it faster for volume computation? Here we give some brief implementation tests to this effect. If the polytope is highly symmetric about the coordinate axes, then we expect coordinate hit-and-run to perform quite favorably to hit-and-run (an almost ideal scenario for coordinate hit-and-run). Thus, we also consider a random rotation of the body to lessen any benefit coordinate hit-and-run might obtain from a degenerate case. Additionally, we consider two real-world polytopes from metabolic networks that might offer a more asymmetric solution space. The results of the computation are in Table 5.8.

Body	Actual Vol	Mean (CHAR)	Steps (CHAR)	Mean (HAR)	Steps (HAR)
Cube (20-dim)	$1.05 \cdot 10^6$	$1.03 \cdot 10^6$	$8.36 \cdot 10^4$	$1.09 \cdot 10^6$	$7.85 \cdot 10^4$
Trans. Cube (20-dim)	$1.05 \cdot 10^6$	$8.56 \cdot 10^5$	$8.36 \cdot 10^4$	$1.01 \cdot 10^6$	$8.92 \cdot 10^4$
Cube (100-dim)	$1.27 \cdot 10^{30}$	$1.23 \cdot 10^{30}$	$4.04 \cdot 10^6$	$1.15 \cdot 10^{30}$	$4.28 \cdot 10^6$
Trans. Cube (100-dim)	$1.27 \cdot 10^{30}$	$1.35 \cdot 10^{30}$	$4.24 \cdot 10^6$	$1.28 \cdot 10^{30}$	$4.27 \cdot 10^6$
E_coli model (22-dim)	??	$8.77 \cdot 10^8$	$1.61 \cdot 10^5$	$6.93 \cdot 10^8$	$1.57 \cdot 10^8$
Mito. model (93-dim)	??	$1.67 \cdot 10^{35}$	$5.40 \cdot 10^6$	$1.86 \cdot 10^{35}$	$5.78 \cdot 10^6$

Table 5.8: Numerical results comparing coordinate hit-and-run to hit-and-run, averaged over 5 trials with $\varepsilon = 0.10$.

5.5 Concluding Remarks and Open Questions

1. Our volume and sampling implementation is available from the MATLAB File Exchange [52] and also from our webpage [66].

2. In the implementation, we use an adaptive cooling schedule; experimentally it performs significantly faster than a fixed cooling schedule. In subsequent work [50], we obtained an $O^*(n^3)$ volume algorithm with an accelerated schedule. Roughly speaking, we start with a Gaussian with $\sigma_0^2 = 1/n$ and cool according to the schedule:

$$\sigma_i^2 = \begin{cases} \sigma_{i-1}^2 \cdot \left(1 + \frac{1}{\sqrt{n}}\right) & \text{if } \sigma_{i-1}^2 = O(1) \\ \sigma_{i-1}^2 \cdot \left(1 + \frac{\sigma_{i-1}}{\sqrt{n}}\right) & \text{otherwise} \end{cases}$$

   Note that this schedule does not depend on the body. In practice, we observe an even faster convergence of $O^*(n^2)$ steps for the convex bodies in our benchmark (for example, in Figures 5.4, 5.5).

3. Hit-and-run was one choice for the random walk to sample from the body. We could also use the well-studied ball walk [7][32]; in each step of the ball walk, we pick a random point y in a ball centered at the current point, and move to y if $y \in K$. In brief implementation tests, it seemed to perform comparable to hit-and-run if the average local conductance (i.e. the

probability that $y \in K$) is kept around $1/2$.

4. For our implementation, the volume computation experimentally runs in $O^*(n^2)$ oracle calls, and the rounding algorithm runs in $O^*(n^3)$ oracle calls. The current best algorithm for volume computation is $O^*(n^3)$ [50] and for rounding is $O^*(n^4)$ [8]. So our empirical results suggest algorithms with better worst-case bounds are possible.

5. One area for improvement in this algorithm is the sliding window convergence test (Sections 5.2.3). The size of the window is based on experimental results to perform well for $n \leq 100$ and $0.10 \leq \varepsilon \leq 0.20$. It would be preferable to have a more robust way to determine convergence of the volume estimator in a single phase that will perform accurately and efficiently for all values of n and ε.

6. We note that we can naively parallelize both the volume and rounding algorithms by assigning each thread of hit-and-run to a different processor. Additionally, for the volume algorithm, we could assign each volume phase (Sections 5.2.2, 5.2.3) to a separate processor. There is the issue of providing a warm start to each phase, but this could be done at a fraction of the cost of volume computation.

Several intriguing theoretical questions arise from our implementation:

1. **Convergence tests.** Our implementation uses empirical convergence tests to test whether we estimate a ratio within a desired error, or to test whether a random walk is suitably close to the target distribution. Could such methods be made rigorous?

2. **Sampling.** Instead of using a random walk to generate approximately independent points and then using these points to estimate the volume as in the theoretical algorithms, is it asymptotically more efficient, in terms of the total number of hit-and-run steps, to use the entire sequence of points visited during the walk to estimate the volume?

3. **Monotonic convergence.** When estimating the integral ratio of two functions f, g over a convex body K (i.e. $\int_K f(x)dx / \int_K g(x)dx$) with a sequence of points from the empirical distribution of hit-and-run, under what conditions on f, g and the starting distribution, does the estimated value approach the true value monotonically from below in expectation? We observed such monotonicity when f, g are spherical Gaussians and f has higher variance than g (Figure 5.9).

CHAPTER 6

SAMPLING FROM METABOLIC NETWORKS

6.1 Background

Metabolic modeling is a subset of systems biology, a field of biology concerned with developing mathematical and computational models for biological systems. We consider a metabolic network to be a set of biochemical compounds known as *metabolites*and reactions that consume/produce these metabolites. One method of metabolic modeling is through constraint-based analysis, which enforces certain thermodynamic principles on the network, such as mass conservation, biologically feasible extremal bounds, etc. Constraint-based analysis has proven a useful tool for analyzing metabolic networks because of its amenability to large-scale models.

More precisely, we are given a biochemical network on m metabolites and n reactions. The network is represented by a stoichiometric matrix $S \in \mathbb{Z}^{m \times n}$, which encodes each reaction and its associated input/output metabolites. Thus each row of S corresponds to a metabolite, and each column corresponds to a reaction. If we examine a single column, i.e. reaction, a negative coefficient corresponds to input and a positive coefficient corresponds to production of a metabolite. Therefore, enforcing mass conservation of each metabolite corresponds to selecting a vector $v \in \mathbb{R}^n$ such that $Sv = 0$, where v_i is the relative rate of reaction i.

Additionally, each reaction has an associated lower and upper bound on its reaction rate given by vectors $\ell, u \in \mathbb{R}^n$ respectively. Therefore we seek solutions to the following system:

$$\begin{aligned} Sv &= 0 \\ \ell \leq v &\leq u \end{aligned} \tag{6.1}$$

It is important to recognize that there are two types of reactions in (6.1): exchange reactions and internal reactions. Exchange reactions represent input to or output of the system; these reactions generally correspond to a column with a single ± 1, and all other entries 0. Internal reactions occur entirely within the model. We enforce that internal reactions must conserve mass and charge, while external reactions necessarily do not.

Our goal is to gain an understanding of the metabolic network, which we will do by analyzing the solutions $v \in \mathbb{R}^n$ to (6.1). In general there will be many feasible solutions. The focus of this paper will be on algorithms for sampling a random v which satisfies the model. A natural question arises of what distribution to use for selecting a v. There are two main high-level approaches in the current biology literature: biased and unbiased.

In biased sampling, we select an optimal flow, for some notion of optimality such as maximizing energy production of the system. Biased sampling is most beneficial for observing the extremal behavior of the system. For simple organisms, e.g. bacteria, looking at the solutions which maximize biomass are likely closer to reality than for the human metabolism. Additionally, there is the added complexity of choosing how to bias the solution space.

For unbiased sampling, we seek to generate uniform, random samples from the set of all possible feasible solutions. If we generate enough random samples, then we can get a picture of the entire solution space, which gives more information into the "typical" behavior of the system. Uniform sampling also allows useful statistics of the solution space to be computed, such as the mean, variances, correlations, etc.

As discussed later in this chapter, we will consider uniformly sampling a solution from the entire space of feasible solutions to the above model, but one could potentially sample from a different distribution imposed upon the solution space, e.g. multivariate Gaussian distribution. Provided the desired target distribution we select is logconcave, we can efficiently generate samples from it [10].

6.1.1 Previous Work

There has been recent work exploring the promising benefits and applications of randomized sampling approaches for the analysis of metabolic networks [67, 49, 68, 69]. If the intrinsic dimension of the metabolic network is small, say at most 20, then standard Monte Carlo methods can be adapted to efficiently generate uniform random samples. But once we get to higher dimensions, we must instead use more sophisticated geometric random walk algorithms.

In [67], they sample from the human red blood cell metabolic network, which has intrinsic dimension 11. They observe that decreasing the availability of pyruvate kinase, an important enzyme for healthy red blood cells and commonly affected by genetic deficiencies, has a drastic effect on reactions in the network. Further, they also observe that it significantly decreases the volume of the solution space: reducing the amount of pyruvate kinase by a factor of 4 decreased the volume by roughly a factor of 100. In [70], they examine the brain metabolism and use random sampling to help identify genes and pathways that can explain the effects of Alzheimer's disease on brain cells.

In addition to observing marginal distributions, random samples can be used to analyze more global behavior of the metabolic network. In [71], they observe, under various distributions, that the distribution of reaction rates in the metabolic network obey a power law distribution.

To extend the sampling methods to moderately high dimensions, implementations of artificial centering hit-and-run, which was based upon [72], have been developed for generating random samples from the set of feasible metabolic flows which experimentally appear to efficiently scale to a few hundred dimensions [73, 74]. These sampling implementations have been successfully used for for a wide variety of applications. In [73], they analyze the effect of disease and diet on metabolic reactions in the human mitochondria and evaluate the potential of treatment options.

However, artificial centering hit-and-run has serious drawbacks, which we elaborate on in Section 6.3. Namely, it is difficult to have confidence in the results, as artificial centering hit-and-run

is not known to converge to the uniform distribution, and there are no theoretical guarantees of its rate of convergence/mixing time. Experimentally, it appears to only be an effective approach when the dimension of the space is at most $100 - 200$.

6.2 Analyzing the Metabolism through Random Sampling

Suppose we had one million random samples $v \in \mathbb{R}^n$ from the set of feasible solutions to the metabolic network. How do we use these samples to gain insights into the metabolic network? One way is to look at marginal distributions, which give the probability distributions over a single reaction. Directly examining the distributions of certain reactions can give concrete, quantitative information about local behavior of the network. However, this essentially considers each reaction to be independent from the others. Thus we can also look at correlations between sets of reactions to evaluate the dependence among reactions.

Metabolic networks are complex and robust structures. Each random sample essentially gives a "snapshot" of the metabolic network, where, say, we fix a time and observe the relative reaction rate for each reaction in the network. However, we do not claim that randomly sampling the feasible space will give a snapshot of reality, as we are working with a simplified model of the metabolism. Nevertheless, the hope is that sampling can provide useful insight that other approaches cannot.

6.2.1 Effect of Changes in External Environment

How does the change in an organism's environment affect their metabolism? This change can correspond to a variety of factors, including diet, diseases, drugs, genetics, etc. One way to reflect these external factors in the model is by changing the bounds on the exchange reactions; recall these correspond to input/output from the internal metabolic network. For example, in [73], they observe the effect that diet and disease has on the marginal distributions of certain key reactions in the human mitochondria, and evaluate the effect of various treatment plans for diabetes and Ischemia on these marginal distributions. In general, we can observe the effect of any factors on

the network via random sampling of the model. So this method has the promising potential to suggest drugs or other treatment of any disease, since we can observe the effect that these external factors has on the diseased metabolic network.

We can also pose the following algorithmic question: can we efficiently determine how to modify the lower and upper bounds $\ell, u \in \mathbb{R}^n$, or even introduce new metabolic pathways [75], to achieve the desired rates for certain key metabolic reactions? We can additionally consider the inverse problem: given a set of changes to the model, how do we properly evaluate the change to the entire system? Can we define an efficiently computable fitness function of a metabolic network?

6.3 Computational Results

Most algorithms developed for uniform metabolic sampling [73, 74, 76] have been based on the artificial centering hit-and-run (ACHR) algorithm [72], where they discuss the notion of "optimal direction choice" for a random process. ACHR is a non-Markovian process that is designed to ease exploration of a poorly structured set and serves as an approximation for this optimal direction choice. However, it has some important drawbacks. Namely, it is not known whether it converges to the uniform distribution [72]. Here, we present a MATLAB implementation of coordinate hit-and-run with rounding (CHRR) that is compatible with the COnstraint-based Reconstruction and Analysis (COBRA) toolbox [68]. A major difference with our approach is a preprocessing step which allows us to use a much simpler Markov chain to explore the set of metabolic flows. Rounding procedures have been used previously prior to sampling[77], but our approach achieves significant improvements for both the quality of the rounding produced and the efficiency of the sampling method (see supplementary methods Section 1). We gain inspiration and guidance from the current state-of-the-art theoretical results for high-dimensional sampling ([LV3, LV2]), while making small modifications which drastically improve efficiency in practice. We compare the performance of CHRR with a comparable implementation of ACHR [68].

As with the implementation in Chapter 5, we also implement a rounding step. However, the isotropy based rounding is prohibitively inefficient in high dimensions. In the case of metabolic networks, our convex body is guaranteed to be a polytope, which can be used to avoid sampling-based rounding. We round the polytope instead by computing the maximum volume ellipsoid contained inside the polytope using the code of [78] and map this ellipsoid to the unit ball prior to sampling the polytope. This ellipsoid is guaranteed to provide an n-rounding to the polytope in the worst case [25], but, experimentally, rounding via the maximum volume ellipsoid performs better than this worst case.

We compared the convergence time of CHRR to the COBRA toolbox implementation of ACHR (Figure 6.1). We found that CHRR converged to a stationary sampling distribution in up to 730 times fewer steps than ACHR (Figure 6.1a) on 15 models with dimensions ranging from 24 to 2430 (see supplementary methods Section 3). Moreover, each step of CHRR was up to 10 times faster than a step of ACHR (Figure 6.1b). Each step of CHRR uses only a small number of arithmetic operations compared to ACHR, and this difference is only exaggerated as the dimension increases. Thus the improved scaling cannot be explained by programmatic differences between the two algorithms. These factors combined to give a 40-3500 fold speedup that tended to increase with model dimension. The constraint-based models used to generate the data in Figure 6.1 are given in Table 6.1.

We now outline the procedure used to generate Figure 6.1. Suppose we have a sequence of points $X_1, X_2, \ldots, X_k$ from a convex body $P = \{x \in \mathbb{R}^n | Ax \leq b\}$, where each X_i is generated from running CHRR for t steps in K. If CHRR has converged, then each X_i should be approximately uniform from K. While exactly testing uniformity of the set of X_i's is computationally intractable, there are a variety of statistical tests which can detect the absence of convergence.

In our experimental tests, we declare convergence once the stream of points passes a chosen set of statistical tests. It is possible that convergence is declared prematurely, and the points X_i are

Table 6.1: **Constraint-based metabolic models.**

Name	Metabolites (m)	Reactions (n)	Dimension ($d = dim\,(\Omega)$)	Reference
e_coli_core	72	95	24	[79]
iLJ478	570	652	59	[80]
iIT341	485	554	65	[81]
iAF692	628	690	79	[82]
iSB619	655	743	83	[83]
iHN637	698	785	88	[84]
iJN678	795	863	91	[85]
iJN746	909	1056	122	[86]
iAB_RBC_283	342	469	130	[87]
iAF987	1109	1285	149	[88]
iYO844	991	1250	167	[89]
iAT_PLT_636	738	1008	289	[90]
iSDY_1059	1890	2540	509	[91]
iJO1366	1805	2583	582	[92]
Recon2_v04	5063	7440	2430	[93]

very far from uniform. This fact serves as motivation to theoretically ground the high-dimensional sampling algorithm as much as possible. For CHRR, we are at least guaranteed that the process will converge to uniform given enough steps, and for hit-and-run, we have precise bounds on the number of steps until convergence.

Convergence to a stationary sampling distribution was determined empirically by computing the potential scale reduction factor (PSRF) [94] of the marginal sampling distributions for all reactions. To compute the PSRF, the set of samples is divided in half and each half is considered a separate chain. The variance of the samples within each chain is computed as well as the variance of the samples between the chains. A low PSRF indicates that these variances are close in magnitude. The PSRF was computed with the MCMC Diagnostics Toolbox for MATLAB [95]. An algorithm was said to have converged when the PSRF of all marginals was stable below 1.1.

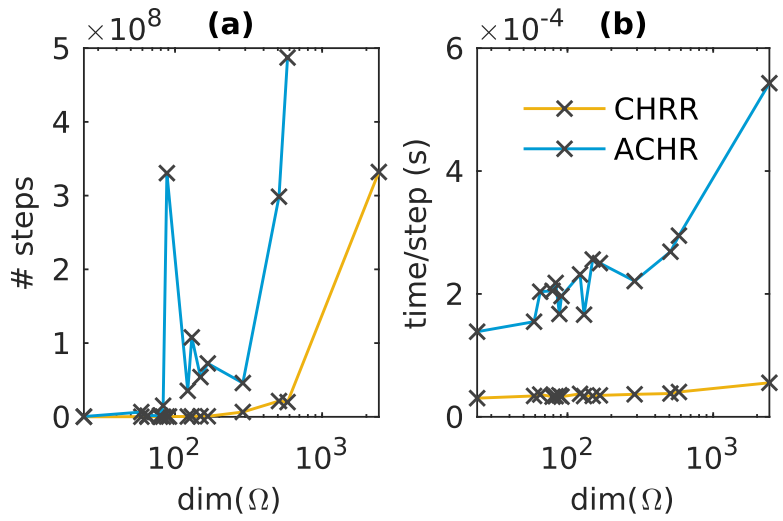


Figure 6.1: A comparison between the convergence times of CHRR and ACHR for 15 constraint-based models. (a) The number of steps until the walk passes the PSRF statistical test. ACHR did not converge in 10^9 steps on 2 of the 15 models. (b) The computational time for each step of the walk.

6.4 Future Work

Now that we are equipped with an improved algorithm for randomized sampling that extends to thousands of dimensions as opposed to hundreds, there are several intriguing computational questions that arise.

6.4.1 Similarity between Solution Sets

Another interesting problem, which could potentially be attacked using random sampling, is quantifying the similarity between two metabolic networks. In the model we are considering, each metabolic network is a high-dimensional polytope. We could directly compare marginal distributions, or we could try to use a more global measure, such as computing the distance between the centroid. A perhaps complicating factor is that two models can lie in different intrinsic dimensions.

Further suppose we are given a labeled population of models, e.g. health/unhealthy, old/young, male/female, etc., and we wish to examine the underlying differences in the metabolic networks

of the two populations. Randomly sampling from the metabolic networks could provide useful biological insight into important differences between the populations of networks.

6.4.2 Target Distribution

From what target distribution should we generate random samples? Two natural choices are optimizing a linear function over the feasible space, and uniform sampling from the entire space of solutions. However, it is quite likely that the actual distribution is much different. Perhaps it is biased towards a few key reactions, and so we should consider a distribution which is skewed towards these metabolites? Two natural choices to consider are exponential and Gaussian distributions, both of which can be efficiently sampled from, but it is unclear if those more closely correspond to the biological reality. Exponential sampling can be viewed as an interpolation between biased and unbiased sampling, where we weight the distribution towards an objective function.

REFERENCES

[1] M. E. Dyer, A. M. Frieze, and R. Kannan, "A random polynomial-time algorithm for approximating the volume of convex bodies," *J. ACM*, 38, no., pp. 1–17, 1991.

[2] G. Elekes, "A geometric inequality and the complexity of computing volume," *Discrete & Computational Geometry*, no., pp. 289–292, 1986.

[3] I. Bárány and Z. Füredi, "Computing the volume is difficult," *Discrete Comput. Geom.*, 2, no., pp. 319–326, 1987.

[4] L. Lovász and M. Simonovits, "Random walks in a convex body and an improved volume algorithm," in *Random Structures and Alg.*, vol. 4, 1993, pp. 359–412.

[5] R. Kannan, L. Lovász, and M. Simonovits, "Isoperimetric problems for convex bodies and a localization lemama," *Discrete & Computational Geometry*, 13, no., pp. 541–559, 1995.

[6] R. Eldan, "Thin shell implies spectral gap up to polylog via a stochastic localization scheme," *Geometric and Functional Analysis*, 23, no., pp. 532–569, 2013.

[7] R. Kannan, L. Lovász, and M. Simonovits, "Random walks and an $O^*(n^5)$ volume algorithm for convex bodies," *Random Structures and Algorithms*, 11, no., pp. 1–50, 1997.

[8] L. Lovász and S. Vempala, "Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm," *J. Comput. Syst. Sci.*, 72, no., pp. 392–417, 2006.

[9] L. Lovász and S. Vempala, "The geometry of logconcave functions and sampling algorithms," *Random Struct. Algorithms*, 30, no., pp. 307–358, 2007.

[10] L. Lovász and S. Vempala, "Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization," in *FOCS*, 2006, pp. 57–68.

[11] D. Revuz, *Markov chains*, Second, ser. North-Holland Mathematical Library. North-Holland Publishing Co., Amsterdam, 1984, vol. 11, pp. xi+374, ISBN: 0-444-86400-8.

[12] J. R. Norris, *Markov chains*, ser. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge, 1998, vol. 2, pp. xvi+237, Reprint of 1997 original, ISBN: 0-521-48181-3.

[13] M. E. Dyer, A. M. Frieze, and R. Kannan, "A random polynomial time algorithm for approximating the volume of convex bodies," in *STOC*, 1989, pp. 375–381.

[14] H. Narayanan, "Randomized interior point methods for sampling and optimization," *Ann. Appl. Probab.*, 26, no., pp. 597–641, 2016.

[15] Y. Lee and S. Vempala, "Geodesic walks on polytopes," no., 2017.

[16] ——, "Eldan's stochastic localization and the kls hyperplane conjecture," no., 2016.

[17] F. Le Gall, "Powers of tensors and fast matrix multiplication," in *ISSAC 2014—Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, ACM, New York, 2014, pp. 296–303.

[18] M. E. Dyer and A. M. Frieze, "Computing the volume of a convex body: A case where randomness provably helps," in *Proc. of AMS Symposium on Probabilistic Combinatorics and Its Applications*, 1991, pp. 123–170.

[19] L. Lovász and S. Vempala, "Hit-and-run from a corner," *SIAM J. Computing*, 35, no., pp. 985–1005, 4 2006.

[20] R. Kannan and H. Narayanan, "Random walks on polytopes and an affine interior point method for linear programming," in *STOC*, 2009, pp. 561–570.

[21] A. Dinghas, "Uber eine klasse superadditiver mengenfunktionale vonbrunn-minkowski-lusternikschem typus," *Math. Zeitschr.*, 68, no., pp. 111–125, 1957.

[22] L. Leindler, "On a certain converse of Hölder's inequality ii," *Acta Sci. Math. Szeged*, 33, no., pp. 217–223, 1972.

[23] A. Prekopa, "Logarithmic concave measures and functions," *Acta Sci. Math. Szeged*, 34, no., pp. 335–343, 1973.

[24] ——, "On logarithmic concave measures with applications to stochastic programming," *Acta Sci. Math. Szeged*, 32, no., pp. 301–316, 1973.

[25] F. John, "Extremum problems with inequalities as subsidiary conditions," in *Studies and Essays Presented to R. Courant on his 60th Birthday, January 8, 1948*, Interscience Publishers, Inc., New York, N. Y., 1948, pp. 187–204.

[26] L. Lovász and M. Simonovits, "On the randomized complexity of volume and diameter," in *Proc. 33rd IEEE Annual Symp. on Found. of Comp. Sci.*, 1992, pp. 482–491.

[27] L. Lovász and S. Vempala, "Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm," in *FOCS*, 2003, pp. 650–659.

[28] K. Chandrasekaran, A. Deshpande, and S. Vempala, "Sampling s-concave functions: The limit of convexity based isoperimetry," in *APPROX-RANDOM*, 2009, pp. 420–433.

[29] K. Chandrasekaran, D. Dadush, and S. Vempala, "Thin partitions: Isoperimetric inequalities and a sampling algorithm for star shaped bodies," in *SODA*, 2010, pp. 1630–1645.

[30] H. J. Brascamp and E. H. Lieb, "On extensions of the brunn-minkowski and prekopa-liendler theorems, including inequalities for log-concave functions, and with application to the diffusion equationslogarithmic concave measures and functions," *J. Functional Anal.*, 22, no., pp. 366–389, 1976.

[31] L. Lovász and S. Vempala, "The geometry of logconcave functions and sampling algorithms," *Random Structures and Algorithms*, 30, no., pp. 307–358, 2007.

[32] B. Cousins and S. Vempala, "A cubic algorithm for computing Gaussian volume," in *SODA*, 2014, pp. 1215–1228.

[33] L. Lovász and S. Vempala, "The geometry of logconcave functions and sampling algorithms," *Random Struct. Algorithms*, 30, no., pp. 307–358, 2007.

[34] M. E. Dyer and A. M. Frieze, "On the complexity of computing the volume of a polyhedron," *SIAM J. Comput.*, 17, no., pp. 967–974, 1988.

[35] Z. Furedi and I. Barany, "Approximation of the sphere by polytopes having few vertices," *Proceedings of the AMS*, 102, no., 1988.

[36] D. Applegate and R. Kannan, "Sampling and integration of near log-concave functions," in *STOC*, 1991, pp. 156–163.

[37] L. Lovász and M. Simonovits, "Mixing rate of Markov chains, an isoperimetric inequality, and computing the volume," in *ROCS*, 1990, pp. 482–491.

[38] R. Bubley, M. Dyer, and M. Jerrum, "An elementary analysis of a procedure for sampling points in a convex body," *Random Structures Algorithms*, 12, no., pp. 213–235, 1998.

[39] R. Eldan and B. Klartag, "Approximately gaussian marginals and the hyperplane conjecture," *Contermporary Mathematics*, 545, no., 2011.

[40] K. M. Ball, "Logarithmically concave functions and sections of convex sets in rn," *Studia Mathematica*, 88, no., pp. 69–84, 1988.

[41] B. Fleury, "Concentration in a thin euclidean shell for log-concave measures," *J. Funct. Anal.*, 259(4), no., pp. 832–841, 2010.

[42] L. Rademacher and S. Vempala, "Dispersion of mass and the complexity of randomized geometric algorithms," *Advances in Mathematics*, 219, no., pp. 1037–1069, 2008.

[43] M. Dyer, P. Gritzmann, and A. Hufnagel, "On the complexity of computing mixed volumes," *SIAM J. Comput.*, 27, no., pp. 356–400, 1998.

[44] G. Martynov, "Evaluation of the normal distribution function," *J. Soviet Math*, 77, no., pp. 1857–1875, 1980.

[45] I. S., "Evaluation of normal probabilities of symmetric regions," *SIAM J. Sci. Stat. Comput.*, 9, no., pp. 812–837, 1988.

[46] R. Kannan and G. Li, "Sampling according to the multivariate normal density," in *FOCS '96: Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, Washington, DC, USA: IEEE Computer Society, 1996, p. 204.

[47] P. N. Somerville, "Numerical computation of multivariate normal and multivariate-t probabilities over convex regions," *J. Comput. Graph. Stat.*, 7, no., pp. 529–545, 1998.

[48] A. Genz and F. Bretz, *Computation of Multivariate Normal and t Probabilities*. Springer, 2009, ISBN: 978-3-642-01689-9.

[49] J. Schellenberger and B. Palsson, "Use of randomized sampling for analysis of metabolic networks," *Journal of Biological Chemistry*, 284, no., pp. 5457–5461, 2009.

[50] B. Cousins and S. Vempala, "Bypassing KLS: Gaussian cooling and an $O^*(n^3)$ volume algorithm," in *STOC*, 2015, pp. 539–548.

[51] L. Lovász and I. Deák, "Computational results of an $O^*(n^4)$ volume algorithm," *European J. of Oper. Res.*, 216, no., 2012.

[52] B. Cousins and S. Vempala, "Volume computation of convex bodies," *MATLAB File Exchange*, no., 2013, http://www.mathworks.com/matlabcentral/fileexchange/43596-volume-computation-of-convex-bodies.

[53] I. Emiris and V. Fisikopoulos, "Efficient random-walk methods for approximating polytope volume," in *Proceedings of the 30th Annual Symposium on Computational Geometry*, ACM, 2014, p. 318.

[54] D. Stefankovic, S. Vempala, and E. Vigoda, "Adaptive simulated annealing: A near-optimal connection between sampling and counting," in *FOCS*, Washington, DC, USA: IEEE Computer Society, 2007, pp. 183–193, ISBN: 0-7695-3010-9.

[55] J. Bourgain, "Random points in isotropic convex sets," *Convex geometric analysis*, Math. Sci. Res. Inst. Publ. 34, no., pp. 53–58, 1996.

[56] M. Rudelson, "Random vectors in the isotropic position," *Journal of Functional Analysis*, 164, no., pp. 60–72, 1999.

[57] R. Adamczak, A. Litvak, A. Pajor, and N. Tomczak-Jaegermann, "Quantitative estimates of the convergence of the empirical covariance matrix in log-concave ensembles," *J. Amer. Math. Soc.*, 23, no., pp. 535–561, 2010.

[58] D. Gillman, "A Chernoff bound for random walks on expander graphs," in *FOCS*, Los Alamitos, CA: IEEE Comput. Soc. Press, 1993, pp. 680–691.

[59] C. Chan, D. Robbins, and D. Yuen, "On the volume of a certain polytope," *Experiment. Math.*, 9, no., pp. 91–99, 2000.

[60] D. Zeilberger, "Proof of a conjecture of Chan, Robbins, and Yuen," *Electron. Trans. Numer. Anal.*, 9, no., 147–148 (electronic), 1999, Orthogonal polynomials: numerical and symbolic algorithms (Leganés, 1998).

[61] I. Pak, "Four questions on Birkhoff polytope," *Ann. Comb.*, 4, no., pp. 83–90, 2000.

[62] E. R. Canfield and B. McKay, "The asymptotic volume of the Birkhoff polytope," *Online J. Anal. Comb.*, no., p. 4, 2009.

[63] J. A. De Loera, F. Liu, and R. Yoshida, "A generating function for all semi-magic squares and the volume of the Birkhoff polytope," *J. Algebraic Combin.*, 30, no., pp. 113–139, 2009.

[64] C. Chan and D. Robbins, "On the volume of the polytope of doubly stochastic matrices," *Experiment. Math.*, 8, no., pp. 291–300, 1999.

[65] M. Beck and D. Pixton, "The Ehrhart polynomial of the Birkhoff polytope," *Discrete Comput. Geom.*, 30, no., pp. 623–637, 2003.

[66] B. Cousins and S. Vempala, *Volume computation and sampling*, `http://www.cc.gatech.edu/~bcousins/volume.html`, 2013.

[67] N. Price, J. Schellenberger, and B. Palsson, "Uniform sampling of steady-state flux spaces: Means to design experiments and to interpret enzymopathies," *Biophysical journal*, 87, no., pp. 2172–2186, 2004.

[68] J. Schellenberger, R. Que, R. Fleming, I. Thiele, J. Orth, A. Feist, D. Zielinski, A. Bordbar, N. Lewis, S. Rahmanian, *et al.*, "Quantitative prediction of cellular metabolism with constraint-based models: The cobra toolbox v2. 0," *Nature protocols*, 6, no., pp. 1290–1307, 2011.

[69] S. Wiback, I. Famili, H. Greenberg, and B. Palsson, "Monte carlo sampling can be used to determine the size and shape of the steady-state flux space," *Journal of theoretical biology*, 228, no., pp. 437–447, 2004.

[70] N. Lewis, G. Schramm, A. Bordbar, J. Schellenberger, M. Andersen, J. Cheng, N. Patel, A. Yee, R. Lewis, R. Eils, *et al.*, "Large-scale in silico modeling of metabolic interactions between cell types in the human brain," *Nature biotechnology*, 28, no., pp. 1279–1285, 2010.

[71] E. Almaas, B. Kovacs, T. Vicsek, Z. Oltvai, and A.-L. Barabási, "Global organization of metabolic fluxes in the bacterium escherichia coli," *Nature*, 427, no., pp. 839–843, 2004.

[72] D. Kaufman and R. Smith, "Direction choice for accelerated convergence in hit-and-run sampling," *Operations Research*, 46, no., pp. 84–95, 1998.

[73] I. Thiele, N. Price, T. Vo, and B. Palsson, "Candidate metabolic network states in human mitochondria impact of diabetes, ischemia, and diet," *Journal of Biological Chemistry*, 280, no., pp. 11 683–11 695, 2005.

[74] W. Megchelenbrink, M. Huynen, and E. Marchiori, "Optgpsampler: An improved tool for uniformly sampling the solution-space of genome-scale metabolic networks," *PloS one*, 9, no., e86587, 2014.

[75] S. Sahoo, H. Haraldsdóttir, R. Fleming, and I. Thiele, "Modeling the effects of commonly used drugs on human metabolism," *FEBS Journal*, 282, no., pp. 297–317, 2015.

[76] P. Saa and L. Nielson, "Ll-achrb: A scalable algorithm for sampling the feasible solution space of metabolic networks," 32, no., pp. 2330–2337,

[77] D. De Martino, M Mori, and V Parisi, "Uniform sampling of steady states in metabolic networks: Heterogeneous scales and rounding," 10, no., e0122670,

[78] Y. Zhang and L. Gao, "On numerical solution of the maximum volume ellipsoid problem," *SIAM J. Optim.*, 14, no., pp. 53–76, 2003.

[79] J. D. Orth, B. Ø. Palsson, and R. M. T. Fleming, "Reconstruction and Use of Microbial Metabolic Networks: The Core Escherichia coli Metabolic Model as an Educational Guide," *EcoSal Plus*, 1, no., Feb. 2010.

[80] Y. Zhang, I. Thiele, D. Weekes, Z. Li, L. Jaroszewski, K. Ginalski, A. M. Deacon, J. Wooley, S. a Lesley, I. a. Wilson, B. Palsson, A. Osterman, and A. Godzik, "Three-dimensional structural view of the central metabolic network of Thermotoga maritima.," *Science*, 325, no., pp. 1544–1549, 2009.

[81] I. Thiele, T. D. Vo, N. D. Price, and B. Ø. Palsson, "Expanded metabolic reconstruction of Helicobacter pylori (iIT341 GSM/GPR): An in silico genome-scale characterization of single- and double-deletion mutants," *J. Bacteriol.*, 187, no., pp. 5818–5830, Aug. 2005.

[82] A. M. Feist, J. C. M. Scholten, B. Ø. Palsson, F. J. Brockman, and T. Ideker, "Modeling methanogenesis with a genome-scale metabolic reconstruction of Methanosarcina barkeri," *Mol. Syst. Biol.*, 2, no., p. 2006.0004, 2006.

[83] S. A. Becker and B. Ø. Palsson, "Genome-scale reconstruction of the metabolic network in Staphylococcus aureus N315: An initial draft to the two-dimensional annotation," *BMC microbiol.*, 5, no., p. 8, 2005.

[84] H. Nagarajan, M. Sahin, J. Nogales, H. Latif, D. R. Lovley, A. Ebrahim, and K. Zengler, "Characterizing acetogenic metabolism using a genome-scale metabolic reconstruction of Clostridium ljungdahlii," *Microb. Cell. Fact.*, 12, no., p. 118, 2013.

[85] J. Nogales, S. Gudmundsson, E. M. Knight, B. O. Palsson, and I. Thiele, "Detailing the optimality of photosynthesis in cyanobacteria through systems biology analysis," *PNAS*, 109, no., pp. 2678–2683, Feb. 2012.

[86] J. Nogales, B. Ø. Palsson, and I. Thiele, "A genome-scale metabolic reconstruction of Pseudomonas putida KT2440: iJN746 as a cell factory.," *BMC Syst. Biol.*, 2, no., p. 79, 2008.

[87] A. Bordbar, N. Jamshidi, and B. Ø. Palsson, "iAB-RBC-283: A proteomically derived knowledge-base of erythrocyte metabolism that can be used to simulate its physiological and patho-physiological states.," *BMC Syst. Biol.*, 5, no., p. 110, 2011.

[88] A. M. Feist, H. Nagarajan, A.-E. Rotaru, P.-L. Tremblay, T. Zhang, K. P. Nevin, D. R. Lovley, and K. Zengler, "Constraint-based modeling of carbon fixation and the energetics

of electron transfer in Geobacter metallireducens," *PLoS comput. biol.*, 10, no., e1003575, Apr. 2014.

[89] Y.-K. Oh, B. O. Palsson, S. M. Park, C. H. Schilling, and R. Mahadevan, "Genome-scale reconstruction of metabolic network in Bacillus subtilis based on high-throughput phenotyping and gene essentiality data," *J. Biol. Chem.*, 282, no., pp. 28 791–28 799, Sep. 2007.

[90] A. Thomas, S. Rahmanian, A. Bordbar, B. Ø. Palsson, and N. Jamshidi, "Network reconstruction of platelet metabolism identifies metabolic signature for aspirin resistance," *Sci. Rep.*, 4, no., p. 3925, 2014.

[91] J. M. Monk, P. Charusanti, R. K. Aziz, J. A. Lerman, N. Premyodhin, J. D. Orth, A. M. Feist, and B. Ø. Palsson, "Genome-scale metabolic reconstructions of multiple Escherichia coli strains highlight strain-specific adaptations to nutritional environments," *PNAS*, 110, no., pp. 20 338–20 343, Dec. 2013.

[92] J. D. Orth, T. M. Conrad, J. Na, J. A. Lerman, H. Nam, A. M. Feist, and B. Ø. Palsson, "A comprehensive genome-scale reconstruction of Escherichia coli metabolism–2011," *Mol. Syst. Biol.*, 7, no., p. 535, 2011.

[93] I. Thiele, N. Swainston, R. M. T. Fleming, A. Hoppe, S. Sahoo, M. K. Aurich, H. Haraldsdóttir, M. L. Mo, O. Rolfsson, M. D. Stobbe, S. G. Thorleifsson, R. Agren, C. Bölling, S. Bordel, A. K. Chavali, P. Dobson, W. B. Dunn, L. Endler, D. Hala, M. Hucka, D. Hull, D. Jameson, N. Jamshidi, J. J. Jonsson, N. Juty, S. Keating, I. Nookaew, N. Le Novère, N. Malys, A. Mazein, J. A. Papin, N. D. Price, E. Selkov Sr, M. I. Sigurdsson, E. Simeonidis, N. Sonnenschein, K. Smallbone, A. Sorokin, J. H.G. M. van Beek, D. Weichart, I. Goryanin, J. Nielsen, H. V. Westerhoff, D. B. Kell, P. Mendes, and B. Ø. Palsson, "A community-driven global reconstruction of human metabolism," *Nat. Biotechnol.*, 31, no., pp. 419–425, May 2013.

[94] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian data analysis*, Third, ser. Texts in Statistical Science Series. CRC Press, Boca Raton, FL, 2014, pp. xiv+661, ISBN: 978-1-4398-4095-5.

[95] S. Särkkä and A. Vehtari, no., 2014.