Vaidya Walk: A sampling algorithm based on the volumetric barrier

Yuansi Chen*, Raaz Dwivedi†, Martin J. Wainwright*† and Bin Yu*†
*Department of Statistics, †Department of Electrical Engineering and Computer Sciences,
University of California, Berkeley
{yuansi.chen, raaz.rsk, wainwrig, binyu}@berkeley.edu

***Abstract*—The problem of sampling from the uniform distribution over a polytope arises in various contexts. We propose a new random walk for this purpose, which we refer to as the Vaidya walk, since it is based on the volumetric-logarithmic barrier introduced by Vaidya in the context of interior point methods for optimization. We show that the Vaidya walk mixes in significantly fewer steps compared to the Dikin walk, a random walk previously studied by Kannan and Narayanan. In particular, we prove that for a polytope in $\mathbb{R}^d$ defined by n constraints, the Vaidya walk mixes in $\mathcal{O}\left(\sqrt{n/d}\right)$ fewer steps than the Dikin walk. The per iteration cost for our method is at most twice that of the Dikin walk, and hence the speed up is significant for polytopes with $n \gg d$. Furthermore, the algorithm is also faster than the Ball walk and Hit-and-Run for a large family of polytopes. We illustrate the speed-up of the Vaidya walk over the Dikin walk via several numerical examples and discuss possible new and faster algorithms for sampling from polytopes.**

I. Introduction

Sampling from distributions is a core problem that arises in statistics, probability, operations research, and other areas involving stochastic models [1], [2], [3], [4]. Sampling algorithms are a prerequisite for applying Monte Carlo methods to order to approximate expectations and other integrals. Recent decades have witnessed great success of Markov Chain Monte Carlo (MCMC) algorithms; for instance, see the handbook [5] and references therein. These methods are based on constructing a Markov chain whose stationary distribution is equal to the target distribution, and then drawing samples by simulating the chain for a certain number of steps. An advantage of MCMC algorithms is that they only require knowledge of the target density up to a proportionality constant. However, the theoretical understanding of MCMC algorithms used in practice is far from complete. In particular, a general challenge is to bound the *mixing time* of a given MCMC algorithm, meaning the number of iterations—as a function of the error tolerance δ, problem dimension d and other parameters—for the chain to arrive at a distribution within distance δ of the target.

In this paper, we study a certain class of MCMC algorithms designed for the problem of drawing samples from the uniform distribution over a polytope. The polytope is specified in the form $\mathcal{K} := \{x \in \mathbb{R}^d \mid Ax \leq b\}$, parameterized by the matrix-vector pair $(A, b) \in \mathbb{R}^{n\times d} \times \mathbb{R}^n$. Our goal is to understand the computational complexity required to obtain δ-accurate samples, and how it grows as a function of the pair (n, d). The problem of sampling uniformly from a polytope is important in various applications and methodologies. For instance, it underlies various methods for computing randomized approximations to polytope volumes. There is a long line of work on sampling methods being used to obtain randomized approximations to the volumes of polytopes and other convex bodies (e.g., [6], [7], [8], [9], [10]). Polytope sampling is also useful in developing faster randomized algorithms for linear programming and sampling contingency tables [11], as well as in randomized methods for approximately solving mixed integer convex programs [12], [13]. Sampling from polytopes is also related to simlations of the hard-disk model in statistical physics [14], as well as to simulations of error events for linear programming in communication [15].

Many algorithms have been studied for sampling from polytopes, and more generally, from convex bodies. Some early examples include the Ball Walk [6] and the Hit-and-Run algorithm [8], [9], which apply to sampling from general convex bodies. Although these algorithms can be applied to polytopes, they do not exploit any special structure of the problem. In contrast, the Dikin walk introduced by Kannan and Narayanan [11] is specialized to polytopes, and thus can achieve faster convergence rates than generic algorithms. The Dikin walk was the first sampling algorithm based on a connection to interior point methods for solving linear programs. More specifically, as we discuss in detail below, it constructs proposal distributions based on the standard logarithmic barrier for a polytope. In a later paper, Narayanan [16] extended the Dikin walk to general convex sets equipped with self-concordant barriers.

For a polytope defined by n constraints, Kannan and Narayanan [11] proved an upper bound on the mixing time of the Dikin walk that scales linearly with n. In many applications, the number of constraints n can be much larger than the number of variables d. It is also possible that for a given problem, various constraints are redundant or repeated. For such problems, linear dependence on the number of constraints is not desirable. Consequently, it is natural to ask if it is possible to design a sampling algorithm whose mixing time scales in a sub-linear manner with the number of constraints. Our main contribution is to investigate and answer

*†-The first two authors contributed equally to this work.

this question in this affirmative—in particular, by designing and analyzing a new sampling algorithm with provably faster convergence rate than the the Dikin walk while retaining its advantages over the ball walk and the hit-and-run methods.

Our contributions: We introduce and analyze a new random walk, which we refer to as the *Vaidya walk* since it is based on the *volumetric-logarithmic barrier* introduced by Vaidya [17]. We show that for a polytope in $\mathbb{R}^d$ defined by n-constraints, the Dikin walk [11] has mixing time bounded as $\mathcal{O}(nd)$, whereas the Vaidya walk mixes in $\mathcal{O}\left(n^{1/2}d^{3/2}\right)$ steps, and so is better in the regime $n \gg d$. We show that when compared to the Dikin walk, the per-iteration computational complexities of the Vaidya walk is within a constant factor. Thus, in the regime $n \gg d$ the overall upper bound on the complexity of generating an approximately uniform sample is much smaller for the Vaidya walk compared to the Dikin walk.

Organization: The remainder of the paper is organized as follows. In Section II, we discuss some polynomial-time random walks on convex sets and polytopes, and motivate the starting point for the new random walk. In Section III, we formally introduce Vaidya walk and provide its rate of convergence. We demonstrate the speed-up of our method over Dikin walk for some illustrative examples in Section III-D. In Section IV we provide the road map to analyze the random walk with some details of the proof of our main result. We conclude with possible extensions of our work in Section V.

Notation: For two sequences a_δ and b_δ indexed by $\delta \in I \subseteq \mathbb{R}$, we say that $a_\delta = \mathcal{O}(b_\delta)$ if there exists a universal constant $C > 0$ such that $a_\delta \leq C b_\delta$ for all $\delta \in I$. For a set $\mathcal{K} \subset \mathbb{R}^d$, the sets $\text{int}(\mathcal{K})$ and $\mathcal{K}^c$ denote the interior and complement of $\mathcal{K}$ respectively. We denote the boundary of the set $\mathcal{K}$ by $\partial\mathcal{K}$. We use $\gamma_{\mathcal{K}}$ to denote the condition number of the set $\mathcal{K}$. In particular, if the set $\mathcal{K}$ contains a ball of radius $R_{\min}$ and is contained in a ball of radius $R_{\max}$, then $\gamma_{\mathcal{K}} \leq R_{\max}/R_{\min}$. The Euclidean norm for any vector $x \in \mathbb{R}^d$ is denoted by $\|x\|_2$. For two distributions $\mathcal{P}_1$ and $\mathcal{P}_2$ defined on the same probability space $(\mathcal{X}, \mathbb{B}(\mathcal{X}))$, we denote the total-variation (TV) distance between the two by $\|\mathcal{P}_1 - \mathcal{P}_2\|_{\text{TV}}$.

II. Related work and problem setting

In this section, we discuss mixing rate of existing random walks on convex sets and investigate the interior point method literature to motivate our Vaidya walk. We end the section by setting up the sampling from polytope problem and essential notations for Vaidya walk.

We begin with a few definitions. For a Markov chain $\{X_0, X_1, \ldots,\}$ on some probability space $\mathcal{X}$ with transition kernel $\mathbb{T}$ and initial distribution μ_0, we use $\mu_0\mathbb{T}^k$ to denote the probability distribution of the iterate X_k. Note that $\mathbb{T}$ is a map $\mathbb{T} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ such that for any $x \in \mathcal{X}$, $\mathbb{T}(x, \cdot)$ is a valid probability distribution on $(\mathcal{X}, \mathbb{B}(\mathcal{X}))$ where $\mathbb{B}(\mathcal{X}))$ denotes a valid sigma-field on $\mathcal{X}$. If this chain has a unique invariant distribution π^*, we denote the Markov chain by the triplet $(\mu_0, \mathbb{T}, \pi^*)$.

Definition 1 (Mixing-time). *For a Markov chain $(\mu_0, \mathbb{T}, \pi^*)$, the δ-mixing time is defined as*

$$k_{mix}(\delta) := \min\left\{k \,\middle|\, \left\|\mu_0\mathbb{T}^k - \pi^*\right\|_{TV} \leq \delta\right\}. \tag{1}$$

Definition 2 (Warm-start). *A distribution $\mathcal{P}_1$ (with density p_1) is said to be M-warm with respect to the distribution $\mathcal{P}_2$ (with density p_2), if*

$$\sup_{A \in \mathbb{B}(\mathcal{X})} \left(\frac{\mathcal{P}_1(A)}{\mathcal{P}_2(A)}\right) = \sup_{x \in \mathcal{X}} \left(\frac{p_1(x)}{p_2(x)}\right) \leq M.$$

We say that the Markov chain $(\mu_0, \mathbb{T}, \pi^*)$ has a warm start, if the distribution μ_0 is M-warm with respect to the distribution π^* for some $M < \infty$.

A. Related work

We first describe a couple of random algorithms that are tailored to generating samples from approximate uniform distribution on bounded convex sets $\mathcal{K} \subset \mathbb{R}^d$. We say that a Markov chain mixes in $\mathcal{O}(f(\delta))$ steps to mean that for any $\delta \in (0, 1)$, we have $k_{\text{mix}}(\delta) = \mathcal{O}(f(\delta))$. The general problem of sampling from convex body given by a membership oracle has witnessed significant progress recently [6], [18], [9], [19], [20], [21], [22]. In such a problem, for each point $x \in \mathbb{R}^d$, one can query an oracle that answers Yes/No depending on whether x is in the convex body $\mathcal{K}$ or not. The mixing-time of these algorithms is measured in terms of the number of oracle calls to obtain an approximate sample from the target distribution on $\mathcal{K}$. In Ball walk [6], when at point x one generates a uniform point u from a ball of radius r centered at x, where r denotes the step size of the algorithm. If $u \in \mathcal{K}$, the walk moves to u, else it remains at x. This walk mixes in $\mathcal{O}\left(d^2\gamma_{\mathcal{K}}^2(M/\delta)^2\log(M/\delta)\right)$ steps from an M-warm start. In Hit-and-Run random walk [9], when at point x, we draw a uniform line ℓ and sample a point uniformly from the intersection $\ell \cap \mathcal{K}$. From an M-warm start, Hit-and-Run mixes in $\mathcal{O}\left(d^2\gamma_{\mathcal{K}}^2\log^3(M/\delta)\right)$ steps. We remark that the convergence rate of both of these algorithms depends on the condition number $\gamma_{\mathcal{K}}$ of the set which can be arbitrary large and which render the algorithms ineffective for high dimensional problems.

Sampling uniformly from polytopes is a special case of sampling from convex sets. Kannan and Narayanan [11] introduced Dikin walk for sampling from polytopes defined as $\mathcal{K} = \{x \in \mathbb{R}^d \mid Ax \leq b, A \in \mathbb{R}^{n \times d}, b \in \mathbb{R}^d\}$, where A and b are known. Evidently, the construction of Dikin walk relied on more information about the set $\mathcal{K}$ beyond the membership oracle. This random walk is an instance of a randomized interior point method and is based on the log-barrier. Their method enjoys affine invariance and thereby its convergence rate is independent of the condition number $\gamma_{\mathcal{K}}$ of the set. Its mixing time was proven to be $\mathcal{O}(nd\log(M/\delta))$ from an M-warm start. Consequently, Dikin walk is faster compared to Ball walk and Hit-and-run for certain class of polytopes, e.g., polytopes with polynomially many faces in d. Dikin walk [11] proceeds by proposing a uniform point in a suitable *state-dependent ellipsoid* followed by an accept-reject step. This algorithm is similar to ball walk except that state-dependent ellipsoids are used in place of fixed-radius Euclidean balls to generate proposals. Dikin walk successfully adapts to the boundary and the proposal ellipsoid remains inside $\mathcal{K}$ for all $x \in \text{int}(\mathcal{K})$, unlike ball walk where radius has to be reduced to avoid high rejection rate near boundary of $\mathcal{K}$.

The idea of randomized interior point method was further extended by Narayanan [16] to introduce a polynomial-time random walk for arbitrary convex set equipped with a self-concordant barrier. In particular, for the general convex sets, he designed a random walk on the Riemannian manifold associated with the Hessian of the self-concordant barrier of the set, and proved convergence rates with polynomial dependence on the dimension of the state space and the self-concordance parameter. The extended Dikin walk when adapted to polytopes simplifies to Gaussian proposals with state-dependent covariance. We remark that for high dimensional problems, the two Dikin walks are almost similar because of the two well-known concentration phenomena: (1) concentration of volume on the boundary for a sphere, and (2) concentration of the normalized Gaussian random vectors on the unit sphere.

More recently, a random walk on Riemannian manifolds—geodesic walk [23]—was introduced to sample from uniform distribution on polytopes. The geodesic paths bend away from the boundary which allows the walk to take large steps while still staying inside the polytope. From a warm start, geodesic walk has an $\mathcal{O}\left(nd^{3/4}\right)$ mixing time, thereby breaking the quadratic barrier on mixing times. We discuss possible extensions in Section V.

B. Problem setting and background on Dikin walk

Given $\mathcal{K} = \{x \in \mathbb{R}^d \mid Ax \leq b, A \in \mathbb{R}^{n\times d}, b \in \mathbb{R}^d\}$, a bounded polytope with a non-empty interior described by n inequalities, the goal is to sample uniformly from the polytope. Note that the set $\mathcal{K}$ is bounded and hence the matrix A has full-rank $d \leq n$.

The Dikin walk is closely related to the interior point methods for solving linear programs. In order to understand the Vaidya walk, it is useful to understand this connection in more detail. Let a_i denote the i-th row vector of matrix A. Consider the *logarithmic barrier* for the polytope $\mathcal{K}$ given by

$$\mathcal{F}_x := \mathcal{F}(x) = -\sum_{i=1}^{n} \log(b_i - a_i^\top x). \tag{2}$$

Each step of an interior point algorithm [24] involves (approximately) solving a linear system involving the Hessian of the barrier function, which is given by

$$\nabla^2 \mathcal{F}_x := \nabla^2 \mathcal{F}(x) \ = \sum_{i=1}^{n} \frac{a_i a_i^\top}{s_{x,i}^2}. \tag{3}$$

Narayanan et al. [11] designed the random walk with proposal ellipsoid at point x (or scaled inverse covariance for Gaussian proposal of extended Dikin walk [16]) given by $D_x := \nabla^2 \mathcal{F}_x$. For all $x \in \mathcal{K}$, D_x is a positive definite matrix. We can define the Dikin local norm as the Mahalanobis distance [25] using D_x as the covariance matrix. Note that in contrast to the ball walk, the proposal distribution now depends on the current state.

We first note that the behavior of the logarithmic barrier relies heavily on the representation of the polytope. For instance, a polytope is unchanged if we duplicate any of its constraints or add any superficial constraint. Nevertheless, unnecessary constraints have significant effect on the Hessian of the barrier and provably reduce the size of the Dikin ellipsoid thereby adversely affecting the convergence rate of Dikin walk. Consequently, to improve the mixing time of the random walk one might consider using a barrier with unequal weights for the logarithmic terms in equation (2) or equivalently covariance with unequal weights for the rank one matrices in equation (3). In fact, such a modification does lead to significant improvements in interior point methods for optimization [17], [26]. Inspired by this line of work, we introduce weighted logarithmic barrier to define the Vaidya walk. In our method, the Gaussian proposals at x are generated using the following inverse covariance matrix (up to scaling)

$$V_x := \sum_{i=1}^{n} \left(\sigma_{x,i} + \beta_{\mathrm{v}}\right) \frac{a_i a_i^\top}{s_{x,i}^2}, \tag{4}$$

where $\beta_{\mathrm{v}} := d/n$ and the scores σ_x are defined as

$$\sigma_x := \left(\frac{a_1^\top (\nabla^2 \mathcal{F}_x)^{-1} a_1}{s_{x,1}^2}, \ldots, \frac{a_n^\top (\nabla^2 \mathcal{F}_x)^{-1} a_n}{s_{x,n}^2}\right)^\top, \tag{5}$$

for $x \in \text{int}(\mathcal{K})$. The covariance matrix V_x is related to the Hessian of the following combination of *volumetric barrier* and the logarithmic barrier introduced by Vaidya et al. [17], [27],

$$\mathcal{V}_x := \log\det \nabla^2 \mathcal{F}_x + \beta_{\mathrm{v}} \mathcal{F}_x. \tag{6}$$

In particular, the quadratic forms $v^\top V_x v$ and $v^\top \nabla^2 \mathcal{V}_x v$ satisfy the condition [17]

$$\forall v \in \mathbb{R}^d, \quad 5 v^\top \nabla^2 \mathcal{V}_x v \geq v^\top V_x v \geq v^\top \nabla^2 \mathcal{V}_x v.$$

To solve a linear program in $\mathbb{R}^d$ with n constraints, the log-barrier method requires $\mathcal{O}\left(\sqrt{n}\right)$ iterations and Vaidya's barrier method requires $\mathcal{O}\left((nd)^{1/4}\right)$ iterations where the per-iteration complexity remains the same order of solving a linear system of equations. The speed-up obtained is primarily because of the larger Newton steps with the Vaidya barrier in comparison to the Newton steps with the logarithmic barrier. In this work, we show that the improvement enjoyed by the use of weighted barrier in linear program solver can be transferred to a sampling problem as well.

C. Comparison of different random walks

To provide an overview of the rates of the random walks so far, we summarize the rates and per step complexity for different random walks in Table I. Except for ball walk and Hit-and-run, all the random walks have a per iteration complexity of the order of linear-system solver. In the rightmost column of the Table I, we mention the overall complexity of different random walks for sampling from convex sets with condition number of $\mathcal{O}\left(d^2\right)$. We remark that the condition

Algorithm 1: Vaidya Walk with parameter r (VW(r))

Input: Parameter r and $x_0 \in \text{int}(\mathcal{K})$
Output: Sequence $x_1, x_2, \ldots$
for $i = 0, 1, \ldots$ **do**
  $C_i \sim$ Fair Coin
  if $C_i = $ *Heads* **then** $x_{i+1} \leftarrow x_i$ // lazy step
  else
    $\xi_{i+1} \sim \mathcal{N}(0, \mathbb{I}_d)$
    $z_{i+1} = x_{i+1} + \frac{r}{(nd)^{1/4}} V_{x_i}^{-1/2} \xi_{i+1}$ // propose a new state
    if $z_{i+1} \notin \mathcal{K}$ **then** $x_{i+1} \leftarrow x_i$ // reject an infeasible proposal
    else
      $\alpha_{i+1} = \min\left\{1, \frac{p^{\mathrm{V}}_{z_{i+1}}(x_{i+1})}{p^{\mathrm{V}}_{x_{i+1}}(z_{i+1})}\right\}$
      $U_{i+1} \sim U[0, 1]$
      if $U_{i+1} \geq \alpha_{i+1}$ **then** $x_{i+1} \leftarrow x_i$ // reject even a valid proposal
      else $x_{i+1} \leftarrow z_{i+1}$ // accept the proposal
    end
  end
end

number $\gamma_{\mathcal{K}}$ of polytopes with polynomially many faces can not be $\mathcal{O}\left(d^{\frac{1}{2}-\epsilon}\right)$ for any $\epsilon > 0$, but can be arbitrarily larger, even exponential in dimension d.

III. Vaidya walk and convergence

In this section, we provide the details of Vaidya walk, provide its rate of convergence and illustrate its performance via simulated examples.

A. Vaidya walk

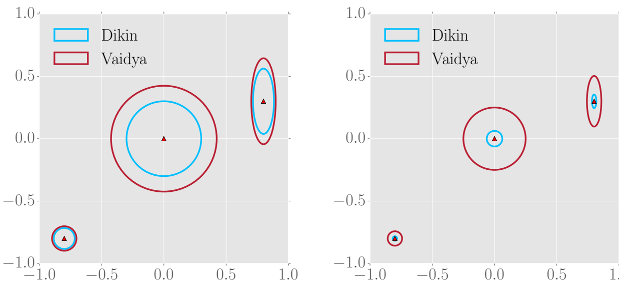
The Vaidya walk with radius parameter $r > 0$, denoted by VW(r) for short, is defined by a Gaussian proposal distribution: given a current state $x \in \text{int}(\mathcal{K})$, it proposes a new point by sampling from the multivariate Gaussian distribution $\mathcal{N}\left(x, \frac{r^2}{\sqrt{nd}} V_x^{-1}\right)$. In analytic terms, the proposal density at x is given by

$$p^{\mathrm{V}}_x(z) := p_{\text{Vaidya}}(x, z)$$
$$= \sqrt{\det V_x} \left(\frac{nd}{2\pi r^2}\right)^{d/2} \exp\left(-\frac{\sqrt{nd}}{2r^2} (z - x)^\top V_x (z - x)\right). \tag{7}$$

As the target distribution for our walk is the uniform distribution on $\mathcal{K}$, the proposal step is followed by an accept-reject step. Thus the overall transition distribution for the walk at state x is defined by a density given by

$$q_{\text{Vaidya}}(x, z) = \begin{cases} \min\left\{p^{\mathrm{V}}_x(z), p^{\mathrm{V}}_z(x)\right\}, & z \in \mathcal{K} \text{ and } z \neq x, \\ 0, & z \notin \mathcal{K}, \end{cases} \tag{8}$$

and a probability mass at x, given by $1 - \int_{z \in \mathcal{K}} \min\left\{p_x(z), p_z(x)\right\} dz$. In Algorithm 1, we summarize the different steps of the Vaidya walk. As pointed out before, Dikin and Vaidya walk differ in the proposal step. The proposal distribution for (the Gaussian proposal) Dikin walk is given by $\mathcal{N}\left(x, \frac{r'^2}{d} D_x^{-1}\right)$ for a suitable universal constant r'.



(a) $n = 64$ (b) $n = 2048$

Fig. 1. High probability regions for proposal distributions of Dikin and Vaidya walks at different points inside $[-1, 1]^2$ for different values of n. The regions become smaller when the chain moves near the boundary. Clearly, Vaidya ellipsoids are larger than Dikin ellipsoids and are affected less severely on increasing n.

B. Main results

Now we state our main theorem which provides a bound for the mixing time of the walk. We use π^* to denote the uniform distribution on $\mathcal{K}$ from now on.

Theorem 1. *Let $(\mu_0, \mathbb{T}_{\text{Vaidya}}, \pi^*)$ denote the Markov chain associated with the random walk VW(10^{-4}) (Algorithm 1) and let the chain have an M-warm start. Then for any $\delta \in (0, 1]$, the mixing time of the Markov chain is bounded as $k_{mix}(\delta) \leq C n^{1/2} d^{3/2} \log(\sqrt{M}/\delta)$, where $C > 0$ is a universal constant.*

A brief outline of the formal proof is provided in Section IV-C. We now provide a high level proof of the mixing time of Markov chain. The volumetric-logarithmic barrier for polytopes has a smaller self-concordance parameter than the log-barrier [17], as the barrier puts unequal weight on each constraint. This fact is the primary source of the speed up in convergence rate of Vaidya walk over Dikin walk on a polytope. In simple words, a Markov chain's mixing time depends on how fast it explores its state space. One way to ensure that the exploration happens fast is by designing a chain such it jumps across far-off regions with non-vanishing probability. For both Dikin and Vaidya walks, the size of the high probability proposal regions depend on how far the chain is from the boundary and the number of constraints used to define the polytope. However, for the new method the effect of number of constraints is less severe. We demonstrate these two facts in Figure 1 for the polytope $[-1, 1]^2$. The set is defined exactly by 4 constraints $\{x = \pm 1, y = \pm 1\}$, however repeating the constraints multiple times changes the proposal ellipsoids (high probability regions for the proposal distribution).

Theorem 1 assumes an M-warm start for the random walk. Instead, we can also have a deterministic start for the walk from a point $x_0 \in \text{int}(\mathcal{K})$ that is not too close to the boundary $\partial\mathcal{K}$. Such a point can be found using standard optimization methods, e.g., using a Phase-I method for Newton's algorithm. (See Section 11.5.4 in the book [24] for further discussion on Phase-I method.) As expected, the mixing times now depend on the distance of the starting point from the boundary. A point $x \in \text{int}(\mathcal{K})$ is called s-central if for any

Random walk	$\mathbf{k_{mix}}(\delta)$	Iteration cost	Complexity for $\gamma_{\mathcal{K}} = d^2$
Ball walk [6]	$d^2\gamma_{\mathcal{K}}^2 \frac{M^2}{\delta^2} \log \frac{M}{\delta}$	nd	$nd^7 \cdot \frac{M^2}{\delta^2} \log \frac{M}{\delta}$
Hit-and-Run [19]	$d^2\gamma_{\mathcal{K}}^2 \log^3 \frac{M}{\delta}$	$nd \log \gamma_{\mathcal{K}}$	$nd^7 \log d \cdot \log^3 \frac{M}{\delta}$
Dikin walk [11]	$nd \log \frac{M}{\delta}$	nd^2	$n^2 d^3 \cdot \log \frac{M}{\delta}$
Geodesic walk [23]	$nd^{3/4} \log \frac{M}{\delta}$	nd^2	$n^2 d^{11/4} \cdot \log \frac{M}{\delta}$
Vaidya walk	$n^{1/2} d^{3/2} \log \frac{M}{\delta}$	nd^2	$n^{3/2} d^{7/2} \cdot \log \frac{M}{\delta}$

TABLE I. Order of computational complexity of random walks from M-warm start on polytope $\mathcal{K} = \{x \in \mathbb{R}^d | Ax \leq b, A \in \mathbb{R}^{n\times d}, b \in \mathbb{R}^n\}$ with condition number $\gamma_{\mathcal{K}}$. Note that nd^2 denotes the complexity of linear system solving, using standard and numerically stable algorithms, for n equations in d dimensions.

chord $\overline{ef}$ passing through x such that $e, f \in \partial\mathcal{K}$, we have $\|e - x\|_2 / \|f - x\|_2 \leq s$. For a start at an s-central point x_0, Dikin walk with proposals uniformly generated from the Dikin ellipsoid of radius $r' = 3/40$ has a polynomial bound on mixing time. (See Algorithm 1 in the paper [11].) The authors proved that when the walk moves to a new state for the first time (i.e., $\min\{i \mid x_{i+1} \neq x_i\}$), the distribution of the iterate is $\left(\sqrt{2ns}/r'\right)^d$-warm with respect to the distribution π^*. It was also shown that the number of steps needed to move to a new state follows a geometric distribution with a bounded mean. These two facts motivate us to consider a hybrid random walk where we use Dikin walk for a few steps in the beginning to provide a warm start to Vaidya walk.

Given an s-central point x_0, we simulate Dikin walk till we observe a new state. Let $\mathbf{k}_1$ denote the (random) number of steps taken to make the first non-trivial move. After $\mathbf{k}_1$ steps, we run the walk $\mathrm{VW}(r)$ with $x_{\mathbf{k}_1}$ as the initial point. We call such a walk as "s-central-Dikin-start-Vaidya-walk with parameter r". Let $\mathbb{T}_{\text{Dikin}}$ denote the transition kernel of the Dikin walk stated above. Then, we have the following mixing time bound for this hybrid walk.

Corollary 1. *There exist positive universal constants c, C, C' such that for any positive s, an s-central-Dikin-start-Vaidya-walk with parameter $r = c$ satisfies $\left\|\delta_{x_0}\mathbb{T}^{\mathbf{k_1}}_{\textit{Dikin}}\mathbb{T}^{k}_{\textit{Vaidya}} - \pi^*\right\|_{TV} \leq \delta$, for all $k \geq Cn^{1/2}d^{5/2}\log(ns/\delta)$, where $\mathbf{k_1}$ is a geometric random variable with $\mathbb{E}[\mathbf{k_1}] \leq C'$.*

The proof follows immediately from Theorem 1 by Kannan et al. [11] and our Theorem 1 and is thereby omitted. Once again we observe that the mixing time bounds are improved by a factor of $\mathcal{O}\left(\sqrt{n/d}\right)$ when compared to Dikin walk from an s-central start [11], [16].

C. Per Iteration Cost

We show that the per-iteration costs of the Dikin and Vaidya walks are of the same order. The proposal step of Vaidya walk requires matrix operations like matrix inversion, matrix multiplication and singular value decomposition (SVD). The accept-reject step requires computation of matrix determinants, besides a few matrix inverses and matrix-vector products. The complexity of all aforementioned operations is $\mathcal{O}\left(nd^2\right)$. Thus, per iteration computational complexity for the Vaidya walk is $\mathcal{O}\left(nd^2\right)$.

Both the Dikin and Vaidya walks requires an SVD computation for inverting the Hessian of Dikin barrier $\nabla^2\mathcal{F}_x$. In addition for the Vaidya walk, we have to invert the matrix V_x, which leads to almost twice the computation time of the Dikin walk per step. This difference can be observed in practice.

We now compare the performance of the two random walks for some simulated examples.

D. Numerical experiments

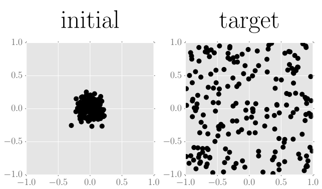
In this section, we demonstrate the speed-up gained by Vaidya walk over Dikin walk for a warm start on different polytopes. We also provide an efficient implementation of both random walks and all experiments presented in this subsection at https://github.com/rzrsk/vaidya-walk. In particular, we simulate the random walks in $\mathbb{R}^2$ with initial distribution $\mu_0 = \mathcal{N}(0, 0.04\,\mathbb{I}_2)$, on the following three different types of polytopes:

1) The set $[-1, 1]^2$,
2) symmetric polytopes with n-constraints generated randomly, and
3) the interior of regular n-polygons on the unit circle.

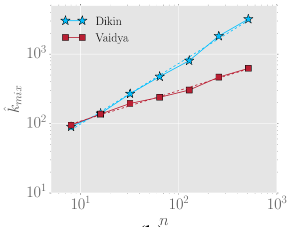
We remark that the warmness-M in all cases is bounded by 500.

For Case 1, we can represent the set exactly by 4 linear constraints. Repeating the constraints increases n for the matrix A associated with $\mathcal{K}$ and hence affects the mixing times of the random walks. We plot the empirical distribution for the iterates from the random walks for $n = 64$ and 512 in Figure 2 from which we observe significant difference in the effect of n on their rates. Note that the warmness $M \leq 8$ for this case. Further, we also plot the approximate mixing time for the set $\mathcal{S} = ([-1, -1/2] \cup [1/2, 1]) \times [-1, 1]$. For a fixed value of n,

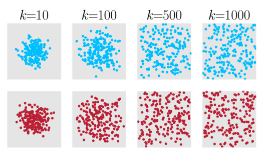
In theory, the matrix computations for the Dikin walk can be carried out in time nd^ν for an exponent $\nu < 2$, but such algorithms are not stable enough for practical use.



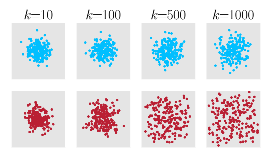
(a)



(b)



(c) $n = 64$



(d) $n = 2048$

Fig. 2. Comparison of Dikin and Vaidya walks (200 runs) on the polytope $\mathcal{K} = [-1, 1]^2$. (a) Samples from the initial distribution $\mu_0 = \mathcal{N}(0, 0.04\,\mathbb{I}_2)$ and the uniform distribution (π^*) on $[-1, 1]^2$. (b) Plot of k_{mix} (9) versus the number of constraints (n) for $\mathcal{S} = ([-1, -1/2] \cup [1/2, 1]) \times [-1, 1]$. Dotted lines show the best-fit lines which have slopes 0.88 and 0.45 or Dikin and Vaidya walks respectively. (c, d) Empirical distribution of samples of Dikin (blue/top rows) and Vaidya (red/bottom rows) walks for different values of n at iteration $k = 10, 100, 500$ and 1000. From the figures we can observe: (1) As n increases, Dikin and Vaidya walks take more number of iterations to mix. (2) The effect of increasing n on mixing time of Vaidya walk is significantly lesser compared to that on the mixing time of Dikin walk.

let $\mu_0\hat{\mathbb{T}}^k$ denote the empirical measure after k-iterations across 1000 experiments. In Figure 2b, we plot k_{mix} as a function of n where

$$\hat{k}_{\text{mix}} := \min \left\{ k \left| \frac{\pi^*(\mathcal{S}) - \mu_0\hat{\mathbb{T}}^k(\mathcal{S})}{\pi^*(\mathcal{S})} \leq \frac{1}{4} \right. \right\}, \tag{9}$$

and observe that the slope of the best fit lines in the log-log plot is approximately 0.88, 0.45 for Dikin and Vaidya walks respectively, which is in accordance with Theorem 1.

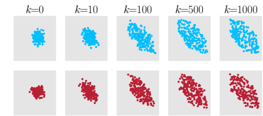
In Case 2, for each constraint i, we fix $b_i = 1$. To generate a_i, first we draw two uniform random variables and then flip the sign of both of them with probability $1/2$ and assign these values to the vector a_i. From Figure 3a-3b we observe that the effect of n on the mixing time of the two walks is different for this case, and Vaidya walk seems to mix faster than Dikin walk. A similar observation can be made even from Figure 3c-3d for Case 3.

IV. Analysis of Vaidya walk

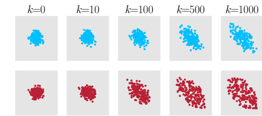
In this section, we first provide an outline of a general method of bounding the rate of convergence of geometric random walks on convex sets in Section IV-A, followed by some auxiliary results in Section IV-B that are used to invoke the general method in our case. We then prove our main result in Section IV-C. For the proofs of the auxiliary results, we refer the reader to the arxiv technical report.

A. A general method to bound mixing time

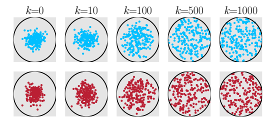
For a discrete-space discrete time Markov chain, a bound on mixing time is obtained via bounds on the *spectral gap* of the



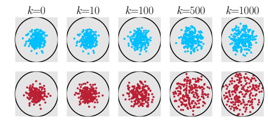
(a) $n = 64$



(b) $n = 2048$



(c) $n = 64$



(d) $n = 2048$

Fig. 3. Empirical distribution of samples from 200 runs of Dikin (blue/top rows) and Vaidya (red/bottom rows) on different polytopes. k denotes the iteration number. (a, b) We simulate the two random walks on random polytopes with 64 and 2048 constraints respectively (for details refer to Section III-D). (c, d) We simulate the walks on regular n-polygons inscribed in the unit circle, for $n = 64$ and 2048. For both cases, we observe that higher n slows down the walks, with visibly more effect on mixing time of Dikin walk compared to Vaidya walk.

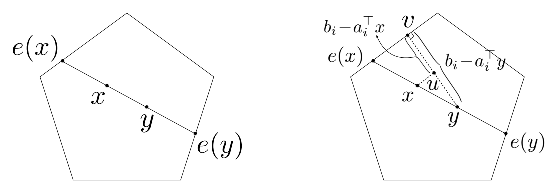
transition matrix associated with the chain. Often, an indirect bound on the spectral gap is obtained via Cheeger's inequality that bounds the spectral gap in terms of the *conductance* of the chain. Lovász and Simonovits [18] proved a similar connection between conductance and convergence rate for continuous-space Markov chains. Thus proving an upper bound for the mixing time of a geometric random walk on convex sets often boils down to showing a good lower bound on the conductance of the chain—these arguments have been used for ball walk [6], Hit-and-run [9], [19] and Dikin walk [16], [11], [28] on convex sets. We refer the reader to the survey by Vempala [29] for a more thorough discussion on geometric random walks.

For a convex and bounded state space, using some powerful isoperimetric inequalities, Lovász showed that to bound the conductance, it suffices to establish that the chain satisfies the following good-neighborhood-property: "if two points are close, then their one-step transition distribution are also close." For quantifying closeness in the property, the distributions are contrasted with the total-variation distance, while for distance between points we use the cross ratio, that we define below. We formally state the result by Lovász in Lemma 1. Much of our technical work focuses on establishing this property for our method where we prove that for Vaidya walk when compared to Dikin walk, the points can be much far apart, with their one-step transition distributions still being close.

For a given pair of points $x, y \in \mathcal{K}$, let $e(x), e(y) \in \partial\mathcal{K}$ denote the intersection of the chord joining x and y with $\mathcal{K}$ such that $e(x), x, y, e(y)$ are in order (see Figure 4a). The cross-ratio $d_{\mathcal{K}}(x, y)$ is given by

$$d_{\mathcal{K}}(x, y) = \frac{\|e(x) - e(y)\|_2 \, \|x - y\|_2}{\|e(x) - x\|_2 \, \|e(y) - y\|_2}. \tag{10}$$

The ratio $d_{\mathcal{K}}(x, y)$ is related to the Hilbert metric on $\mathcal{K}$, which is given by $\log(1 + d_{\mathcal{K}}(x, y))$ (see the paper by Bushell [30] for more details). Let $X_0, X_1, \ldots$ denote a *lazy* reversible



(a) (b)

Fig. 4. Polytope $\mathcal{K} = \{x \in \mathbb{R}^d | Ax \le b\}$. (a) The points $e(x)$ and $e(y)$ denote the intersection points of the chord joining x and y with $\mathcal{K}$ such that $e(x), x, y, e(y)$ are in order. (b) A geometric illustration of the argument (13). It is straightforward to observe that $\|x-y\|_2 / \|e(x)-x\|_2 = \|u-y\|_2/\|u-v\|_2 = \left|a_i^\top(y-x)\right| / \left(b_i - a_i^\top x\right)$.

random walk on a bounded convex set $\mathcal{K}$ with transition kernel $\mathcal{T}(x,\cdot) = \delta_x(\cdot)/2 + \tilde{\mathcal{T}}(x,\cdot)/2$ where δ_x denotes the dirac-delta distribution with unit probability mass at x and $\tilde{\mathcal{T}}(\cdot,\cdot)$ denotes an arbitrary valid transition kernel on $\mathcal{X}$. Let the chain be stationary with respect to the uniform distribution on $\mathcal{K}$ (denoted by π^*). We use the shorthands $\mathcal{T}_x = \mathcal{T}(x,\cdot), \tilde{\mathcal{T}}_x = \tilde{\mathcal{T}}(x,\cdot)$. The following lemma gives a bound on the mixing-time of the chain $\{X_t, t \ge 0\}$.

Lemma 1. *Suppose that* $\left\|\tilde{\mathcal{T}}_x - \tilde{\mathcal{T}}_y\right\|_{TV} \le 1-\rho$, *for all* $x, y \in \operatorname{int}(\mathcal{K})$ *such that* $d_{\mathcal{K}}(x,y) < \Delta$, *for some* $\rho \in (0,1)$ *and* $\Delta > 0$. *Then for every distribution* μ_0 *that is* M*-warm with respect to* π^*, *we have*

$$\left\|\mu_0 \mathcal{T}^k - \pi^*\right\|_{TV} \le \sqrt{M} \exp\left(-k\frac{\Delta^2\rho^2}{4096}\right).$$

To prove Theorem 1, we show that the random walk VW(10^{-4}) satisfies the assumptions of Lemma 1 with suitable Δ and ρ. Besides Lemma 1, our proof techniques are inspired by the proofs of convergence rate of Dikin walk on polytopes presented by Kannan et al. [11] and the simple proof of Dikin walk provided by Sachdeva et al. [28].

B. Some auxiliary results

We now introduce some notation and auxiliary results that are useful for the proof. For all $x \in \operatorname{int}(\mathcal{K})$, define the "Vaidya local norm at x" as

$$\|.\|_x : v \mapsto \left\|V_x^{1/2} v\right\|_2 = \sqrt{\sum_{i=1}^n (\sigma_{x,i} + \beta_{\mathrm{v}}) \frac{(a_i^\top v)^2}{s_{x,i}^2}}. \quad (11)$$

The following lemma provides some properties of the leverage scores σ_x (5) for all $x \in \operatorname{int}(\mathcal{K})$.

Lemma 2. *For any* $x \in \operatorname{int}(\mathcal{K})$, *the following properties hold:*

(a) $\sigma_{x,i} \in [0,1]$ *for all* $i \in [n]$, *and*
(b) $\sum_{i=1}^n \sigma_{x,i} = d$.

Next, we state a lemma that shows that (1) for suitable choice of parameter, the transition and proposal distributions are close for all $x \in \operatorname{int}(\mathcal{K})$, and (2) if two points $x, y \in \operatorname{int}(\mathcal{K})$ are close in Vaidya-local norm, then the one-step proposal distributions are also close in TV-distance. We use $\mathcal{P}_x$ to denote the proposal distribution (7) and $\mathbb{T}_x$ to denote the transition distribution (8) at $x \in \operatorname{int}(\mathcal{K})$ for the Vaidya walk with parameter 10^{-4}.

Lemma 3. *There exists a continuous non-decreasing function* $f : [0, 1/12] \to \mathbb{R}_+$ *with* $f(1/12) \ge 10^{-4}$, *such that for any* $\epsilon \in (0, 1/4]$, *the random walk VW(r) with* $r \in [0, f(\epsilon)]$ *satisfies for all* $x \in \operatorname{int}(\mathcal{K})$, $\|\mathbb{T}_x - \mathcal{P}_x\|_{TV} \le 5\epsilon$ *and* $\|\mathcal{P}_x - \mathcal{P}_y\|_{TV} \le \epsilon$, $\forall x, y \in \operatorname{int}(\mathcal{K})$ *such that* $\|x-y\|_x \le \epsilon r/(2(nd)^{1/4})$.

We are now well equipped to provide a formal proof of our main result. The proof of the lemmas stated above are provided in the arXiv technical report.

C. Proof of Theorem 1

To invoke Lemma 1 for VW(10^{-4}), we need to show that for any two points $x, y \in \operatorname{int}(\mathcal{K})$ such that $d_{\mathcal{K}}(x,y)$ is small, we have that $\|\mathbb{T}_x - \mathbb{T}_y\|_{\mathrm{TV}}$ is small. Along the outline discussed in previous subsection, we break our analysis in two steps—(A) We first relate the cross-ratio $d_{\mathcal{K}}(x,y)$ to the local norm (11) at x, and (B) then use Lemma 3 to show that if $x, y \in \operatorname{int}(\mathcal{K})$ are close in local-norm, the transition kernels $\mathbb{T}_x$ and $\mathbb{T}_y$ are close in TV-distance.

Step (A): We claim that for all $x, y \in \operatorname{int}(\mathcal{K})$, the cross-ratio can be lower bounded as $d_{\mathcal{K}}(x,y) \ge \frac{1}{\sqrt{2d}} \|x-y\|_x$. Note that we have

$$\begin{aligned} d_{\mathcal{K}}(x,y) &\overset{(i)}{\ge} \max\left\{\frac{\|x-y\|_2}{\|e(x)-x\|_2}, \frac{\|x-y\|_2}{\|e(y)-y\|_2}\right\} \\ &\overset{(ii)}{\ge} \max\left\{\frac{\|x-y\|_2}{\|e(x)-x\|_2}, \frac{\|x-y\|_2}{\|e(y)-x\|_2}\right\} \end{aligned}$$

where step (i) follows from the inequality $\|e(x)-e(y)\|_2 \ge \max\left\{\|e(y)-y\|_2, \|e(x)-x\|_2\right\}$ and step (ii) from the inequality $\|e(x)-x\|_2 \le \|e(y)-x\|_2$. Furthermore, from Figure 4b, we observe that

$$\max\left\{\frac{\|x-y\|_2}{\|e(x)-x\|_2}, \frac{\|x-y\|_2}{\|e(y)-x\|_2}\right\} = \max_{i \in [n]} \left|\frac{a_i^\top (x-y)}{s_{x,i}}\right|. \quad (13)$$

Note that maximum of a set of non-negative numbers is greater than the weighted mean of the numbers. Using this fact with weights $\{\sigma_{x,i} + \beta_{\mathrm{v}}\}_{i=1}^n$ and using properties (a) and (b) from Lemma 2 yields the claim.

Step (B): By the triangle inequality, we have

$$\begin{aligned} \|\mathbb{T}_x - \mathbb{T}_y\|_{\mathrm{TV}} \le \|\mathbb{T}_x - \mathcal{P}_x\|_{\mathrm{TV}} \\ + \|\mathcal{P}_x - \mathcal{P}_y\|_{\mathrm{TV}} + \|\mathcal{P}_y - \mathbb{T}_y\|_{\mathrm{TV}}. \end{aligned}$$

Thus, for any (r, ϵ) such that $\epsilon \in [0, 1/4]$ and $r \le f(\epsilon)$, Lemma 3 implies that $\forall x, y \in \operatorname{int}(\mathcal{K})$ such that $\|x-y\|_x \le \frac{r\epsilon}{2(nd)^{1/4}}$, we have $\|\mathbb{T}_x - \mathbb{T}_y\|_{\mathrm{TV}} \le 11\epsilon$. Consequently, the walk VW(r) satisfies the assumptions of Lemma 1 with $\Delta = \frac{1}{\sqrt{2d}} \cdot \frac{r\epsilon}{2(nd)^{1/4}}$ and $\rho = 1 - 11\epsilon$. Since $f(1/12) \ge 10^{-4}$, we can set $\epsilon = 1/12$ and $r = 10^{-4}$ and doing some algebra yields the claimed upper bound for the mixing time of Vaidya Walk.

V. Discussion

In this paper, we focused on improving MCMC sampling algorithms for convex sets by building on the advancements in the field of interior point methods. We specialized our discussion to the polytopes. The better self-concordance property and several other key properties exhibited by the volumetric-logarithmic barrier for polytopes were extended by Anstreicher [31] to more general convex sets defined by semidefinite-constraints, namely, linear matrix inequality (LMI) constraints. Moreover, Narayanan [16] showed that for a convex set defined by LMI constraints and equipped with the log-det barrier, Dikin walk mixes in polynomial time. It is possible that an appropriate Vaidya walk on such sets would have a speedup over Dikin walk.

Narayanan et al. [32] use Dikin walk to generate samples from time varying log-concave distributions with appropriate scaling of the radius for difference class of distributions. It would be interesting to see if a suitable adaptation of Vaidya walk for such cases would provide a significant gain.

Another possible extension of our work can be a new random walk on Riemannian manifolds based on the matrix V_x, in contrast to the Geodesic walk [23] where the manifold is based on the Hessian $\nabla^2 \mathcal{F}_x$. In contrast to Dikin walk's $\mathcal{O}\left(nd\right)$ mixing time, the Geodesic walk has an $\mathcal{O}\left(nd^{3/4}\right)$ dependence on mixing time. It would be interesting to see whether a geodesic version of Vaidya walk has a convergence rate of $\mathcal{O}\left(n^{1/2}d^{5/4}\right)$.

Acknowledgements

This research was supported by Office of Naval Research grant DOD ONR-N00014 to MJW and in part by ARO grant W911NF1710005, NSF-DMS 1613002, the Center for Science of Information (CSoI), a US NSF Science and Technology Center, under grant agreement CCF-0939370 and the Miller Professorship (2016-2017) at UC Berkeley to BY. In addition, MJW was partially supported by National Science Foundation grant NSF-DMS-1612948 and RD was partially supported by the Berkeley Fellowship.

References

[1] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. PAMI*, vol. 6, pp. 721–741, 1984.

[2] P. Brémaud, *Markov chains, Gibbs fields, Monte Carlo simulation, and queues*. Springer, 1991.

[3] B. D. Ripley, *Stochastic simulation*. New York: Wiley, 1987.

[4] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[5] S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, 2011.

[6] L. Lovász and M. Simonovits, "The mixing rate of Markov chains, an isoperimetric inequality, and computing the volume," in *Proceedings of 31st Annual Symposium on Foundations of Computer Science, 1990*, pp. 346–354, IEEE, 1990.

[7] J. Lawrence, "Polytope volume computation," *Mathematics of Computation*, vol. 57, no. 195, pp. 259–271, 1991.

[8] C. J. Bélisle, H. E. Romeijn, and R. L. Smith, "Hit-and-run algorithms for generating multivariate distributions," *Mathematics of Operations Research*, vol. 18, no. 2, pp. 255–266, 1993.

[9] L. Lovász, "Hit-and-run mixes fast," *Mathematical Programming*, vol. 86, no. 3, pp. 443–461, 1999.

[10] B. Cousins and S. Vempala, "A cubic algorithm for computing Gaussian volume," in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1215–1228, Society for Industrial and Applied Mathematics, 2014.

[11] R. Kannan and H. Narayanan, "Random walks on polytopes and an affine interior point method for linear programming," *Mathematics of Operations Research*, vol. 37, no. 1, pp. 1–20, 2012.

[12] K.-L. Huang and S. Mehrotra, "An empirical evaluation of walk-and-round heuristics for mixed integer linear programs," *Computational Optimization and Applications*, vol. 55, no. 3, pp. 545–570, 2013.

[13] K.-L. Huang and S. Mehrotra, "An empirical evaluation of a walk-relax-round heuristic for mixed integer convex programs," *Computational Optimization and Applications*, vol. 60, no. 3, pp. 559–585, 2015.

[14] S. C. Kapfer and W. Krauth, "Sampling from a polytope and hard-disk Monte Carlo," 2013.

[15] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using linear programming to decode binary linear codes," vol. 51, pp. 954–972, March 2005.

[16] H. Narayanan, "Randomized interior point methods for sampling and optimization," *The Annals of Applied Probability*, vol. 26, no. 1, pp. 597–641, 2016.

[17] P. M. Vaidya, "A new algorithm for minimizing convex functions over convex sets," in *30th Annual Symposium on Foundations of Computer Science, 1989*, pp. 338–343, IEEE, 1989.

[18] L. Lovász and M. Simonovits, "Random walks in a convex body and an improved volume algorithm," *Random Structures & Algorithms*, vol. 4, no. 4, pp. 359–412, 1993.

[19] L. Lovász and S. Vempala, "Hit-and-run from a corner," *SIAM Journal on Computing*, vol. 35, no. 4, pp. 985–1005, 2006.

[20] L. Lovász and S. Vempala, "Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization," in *47th Annual IEEE Symposium on Foundations of Computer Science, 2006*, pp. 57–68, IEEE, 2006.

[21] L. Lovász and S. Vempala, "Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm," *Journal of Computer and System Sciences*, vol. 72, no. 2, pp. 392–417, 2006.

[22] L. Lovász and S. Vempala, "The geometry of logconcave functions and sampling algorithms," *Random Structures & Algorithms*, vol. 30, no. 3, pp. 307–358, 2007.

[23] Y. T. Lee and S. S. Vempala, "Geodesic walks in polytopes," *arXiv preprint arXiv:1606.04696*, 2016.

[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[25] P. C. Mahalanobis, "On the generalised distance in statistics," *Proceedings of the National Institute of Sciences of India, 1936*, pp. 49–55, 1936.

[26] Y. T. Lee and A. Sidford, "Path finding methods for linear programming: Solving linear programs in o (vrank) iterations and faster algorithms for maximum flow," in *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pp. 424–433, IEEE, 2014.

[27] P. M. Vaidya and D. S. Atkinson, "A technique for bounding the number of iterations in path following algorithms," in *Complexity in Numerical Optimization*, pp. 462–489, World Scientific, 1993.

[28] S. Sachdeva and N. K. Vishnoi, "The mixing time of the Dikin walk in a polytope—a simple proof," *Operations Research Letters*, vol. 44, no. 5, pp. 630–634, 2016.

[29] S. Vempala, "Geometric random walks: a survey," *Combinatorial and Computational Geometry*, vol. 52, no. 573-612, p. 2, 2005.

[30] P. J. Bushell, "Hilbert's metric and positive contraction mappings in a Banach space," *Archive for Rational Mechanics and Analysis*, vol. 52, no. 4, pp. 330–338, 1973.

[31] K. M. Anstreicher, "The volumetric barrier for semidefinite programming," *Mathematics of Operations Research*, vol. 25, no. 3, pp. 365–380, 2000.

[32] H. Narayanan and A. Rakhlin, "Random walk approach to regret minimization," in *Advances in Neural Information Processing Systems*, pp. 1777–1785, 2010.