Discrete Hit-and-Run for Sampling Points from Arbitrary Distributions over Subsets of Integer Hyper-rectangles *

Stephen Baumert
Department of Operational Sciences
Air Force Institute of Technology
Wright Patterson AFB, Ohio 45433

Archis Ghate
Industrial Engineering Program
University of Washington
Seattle, Washington 98195

Seksan Kiatsupaibul
Department of Statistics
Chulalongkorn University
Bangkok 10330, Thailand

Yanfang Shen
Clearsight Systems, Inc.
Bellevue, Washington

Robert L. Smith
Department of Industrial and Operations Engineering
University of Michigan
Ann Arbor, Michigan 48105

Zelda B. Zabinsky
Industrial Engineering Program
University of Washington
Seattle, WA 98195-2650

January 21, 2008

*This work has been funded in part by an NSF collaborative grant, numbers DMI-9820744, DMI-9820878, DMI-0244286 and DMI-0244291.

Abstract

We consider the problem of sampling a point from an arbitrary distribution π over an arbitrary subset S of an integer hyper-rectangle. Neither the distribution π nor the support set S are assumed to be available as explicit mathematical equations but may only be defined through oracles and in particular computer programs. This problem commonly occurs in black-box discrete optimization as well as counting and estimation problems. The generality of this setting and high-dimensionality of S precludes the application of conventional random variable generation methods. As a result, we turn to Markov Chain Monte Carlo (MCMC) sampling, where we execute an ergodic Markov chain that converges to π so that the distribution of the point delivered after sufficiently many steps can be made arbitrarily close to π. Unfortunately, classical Markov chains such as the nearest neighbor random walk or the co-ordinate direction random walk fail to converge to π as they can get trapped in isolated regions of the support set. To surmount this difficulty, we propose Discrete Hit-and-Run (DHR), a Markov chain motivated by the Hit-and-Run algorithm known to be the most efficient method for sampling from log-concave distributions over convex bodies in R^n. We prove that the limiting distribution of DHR is π as desired, thus enabling us to sample approximately from π by delivering the last iterate of a sufficiently large number of iterations of DHR. In addition to this asymptotic analysis, we investigate finite-time behavior of DHR and present a variety of examples where DHR exhibits polynomial performance.

1 Introduction

We are interested in the Monte Carlo problem of sampling a point distributed according to a general distribution over an arbitrary subset of an integer hyper-rectangle [2, 4, 9, 10, 27, 35]. Given H, a bounded hyper-rectangular subset of the n dimensional integer lattice Z^n, a membership oracle $\mathcal{O}$ [14] for a finite set $S \subseteq H$, and a point $x_0 \in S$, our aim is to sample a point from S distributed approximately according to a distribution $\pi > 0$ on S. The oracle $\mathcal{O}$ receives a point $y \in H$ as input and returns YES if $y \in S$ and NO otherwise. It is standard practice (see [14]) to assume knowledge of an initial feasible point x_0 in complexity analyses of algorithms that use membership oracles. In fact, a membership oracle is typically defined to 'come with' one feasible point. In practice, an initial feasible point is usually available through domain-specific information about the physical system that is under consideration. The target distribution π is assumed to be defined by an evaluation oracle, which receives a point $y \in S$ as input and returns $\pi(y)$ (or a number proportional to $\pi(y)$) as output.

One important situation where this problem arises is black-box discrete optimization. For example, Simulated Annealing (SA) [2, 4, 17] attempts to sample points from the feasible region S of the opti-

mization problem according to Boltzmann distributions parameterized by the so-called 'temperature' T. For SA, the target distribution $\pi(x)$ is proportional to $\exp(-f(x)/T)$, where f is the objective function of the optimization problem. The idea is to decrease the temperature parameter T gradually to a very small value since a point sampled from a low-temperature Boltzmann distribution is highly likely to be an optimal solution to the problem. In several science, engineering and economic optimization models, S and f (and hence π) are only available through computer programs rather than explicit formulas or equations [24, 35], making it crucial to allow for the generality of both membership and evaluation oracles. For example, (i) in structural engineering, the performance of a bridge or a building subject to earthquake shocks may be available only through computer code that performs extensive finite element calculations, (ii) in biology, the potential energy of a protein may be available through computer code running molecular computations, (iii) in manufacturing, the performance of an assembly line is typically obtained by running a computer simulation, and finally, (iv) in cancer treatment by radiation therapy, radiation dose delivered to various organs by a specific intensity profile is available through the so-called dose-calculation programs. A compilation of several case studies attempting to optimize a wide range of challenging models including cellular mobile network design, robot design, composite structure design, assembly line design, chemical process optimization, water resource systems, radiation therapy planning, and virus structure determination is presented in [24].

Beside optimization, modified versions of the sampling problem occur in counting and estimation [15, 27]. For example, several derivative pricing techniques in computational finance increasingly depend on discrete lattice approximations of stochastic processes representing fluctuations of the underlying security. When a large lattice is used for high accuracy, enumeration of nodes becomes impractical and sophisticated techniques are required to sample from nodes to estimate the price of the derivative instrument [10]. Similarly, several canonical problems in statistical mechanics involve estimation of the so-called 'partition function' of a system with a large number of interacting particles, often achieved by sampling from a distribution proportional to this function [30].

Unfortunately, the Alias method [32, 33] cannot be used to sample points from π because S is known only through the oracle $\mathcal{O}$. Even if it were applicable, its complexity would be $O(|S|)$ [28], making it highly inefficient when S is a large set. Inverse transformation methods are also not applicable since π is known only through an evaluation oracle. Our approach is to design an ergodic Markov chain with state space S and stationary distribution π. Since the k-step distribution of an ergodic chain converges to π pointwise as $k \to \infty$, the distribution of the 'last' state delivered by the chain is arbitrarily close to π in

variation distance if we simulate the chain long enough. More interestingly, the probability that the last state is distributed *exactly* according to π can be made arbitrarily close to 1 (see Coupling Lemma [23], page 275). The reader is referred to [3, 4, 9, 23, 31] for detailed descriptions of Markov chain sampling methods.

Designing an ergodic Markov chain that converges quickly to a general distribution over a finite, high-dimensional integer set given by a membership oracle is a challenging problem for a variety of reasons. First and foremost, no a priori information is available about S other than an initial point $x_0 \in S$ and a bounded hyper-rectangle H with $S \subseteq H \subset Z^n$. If the volume of S is exponentially smaller than that of H, an n dimensional rejection technique is extremely inefficient even if π is a relatively simple distribution known explicitly. For example, consider the case when S is a hyper-rectangle (given by a membership oracle) whose sides are half those of H (see Figure 1 (a) for example), a point $x_0 \in S$ is given and π is the uniform distribution on S. Of course, if the coordinates of S were known, a uniform point could be sampled by direct methods, but since they are not (as S is given by a membership oracle), a rejection technique on H is a straightforward way to generate a uniformly distributed point in S. In particular, we would generate a uniformly distributed point x in H and output it as a uniformly distributed point in S if the membership oracle returns YES when queried by sending x as input. However, the expected number of uniformly distributed points in H that need to be generated until the membership oracle returns YES is 2^n. Interestingly, we show (see Corollary 4.5) that the approach proposed in this paper samples an approximately uniformly distributed point from a box within a box in polynomial time. Similar arguments hold for a wedge inside a box (see Figure 1 (b) and Corollary 4.6). More generally, S might have isolated regions (see Figure 1 (c)) preventing use of traditional Markov chains such as the nearest neighbor walk, which moves from a feasible point to one of its feasible nearest neighbors, i.e., a point that differs by exactly one in exactly one co-ordinate, chosen randomly. In particular, for sets S such as Figure 1 (c), the nearest neighbor random walk gets trapped in an isolated region, resulting in a reducible Markov chain. Even worse, for sets such as Figure 1 (d), the nearest neighbor random walk cannot leave its initial point $x_0 \in S$. On the other hand, we show (see Corollaries 4.7 and 4.8) that our approach samples an approximately uniform point from n-dimensional versions of sets in Figure 1 (c) and (d) in polynomial time. Moreover, both the nearest neighbor random walk and the co-ordinate direction random walk (that moves to a feasible point a random distance away along a direction chosen randomly from the co-ordinate directions) get trapped in the initial point when applied to sets as in Figure 1 (e). However, our approach can in fact sample an approximately uniform point from n-dimensional versions of the set in Figure 1 (e).

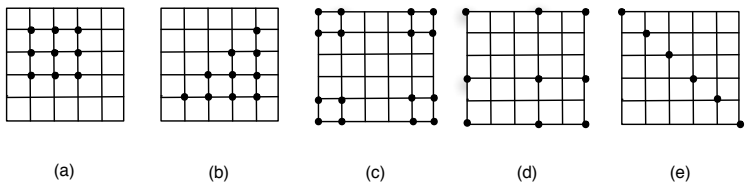

Figure 1: H is shown as a square grid. S is shown by black dots. (a) S is a box within a box. An n dimensional rejection applied to uniformly distributed points on H is exponentially inefficient. (b) A wedge within a box. An n dimensional rejection applied to uniformly distributed points on H is exponentially inefficient. (c) S includes four isolated regions. A nearest neighbor random walk that starts in any one of these regions cannot leave it and hence cannot be applied to sample points from S. (d) Every point in S is isolated. A nearest neighbor random walk will be trapped in the point it starts out at and hence cannot be applied to sample points from S. (e) Again, every point in S is isolated. Both the nearest neighbor random walk and the co-ordinate direction random walk get trapped in the point they start out at, and hence cannot be applied.

These examples show that if one employs the nearest neighbor random walk or the coordinate direction random walk to the problem at hand, it must be modified so that it can leave isolated regions or points of S and explore 'remote corners' of H. However, a naive modification amounts to an n-dimensional rejection technique on H, which is exponentially inefficient even for the simplest cases as described above. In light of this discussion, our goal is two-fold: (i) to design a Markov chain sampler that converges asymptotically to any *arbitrary* distribution π given by an evaluation oracle over *any* $S \subseteq H$ given by a membership oracle, and hence can be employed for approximate sampling from π over S, and (ii) compute finite-time performance bounds for this Markov chain sampler for some special cases of S and π.

The approach proposed in this paper is motivated by the success of the Hit-and-Run algorithm introduced by Smith [29] to generate a sequence of points that asymptotically converges in total variation to a uniform distribution on an arbitrary bounded open subset S of R^n. It was later shown that Hit-and-Run with a Metropolis filter [25], or alternatively using a conditional version [6] can be used to generate arbitrary continuous multivariate distributions. Hit-and-Run is a Markov chain sampler whose simplest version makes a transition from a point $x \in S$ to another point $y \in S$ by generating a direction vector uniformly on the surface of an n-dimensional hypersphere from x, followed by generating a point y uniformly distributed on the line segments created by the intersection of a line along this direction and S (this is accomplished by employing a one-dimensional rejection method on the line segment intersected by the line with an enclosing hyper-rectangle). This version of Hit-and-Run was shown to be the fastest method for generating an asymptotically uniform point from a convex body in R^n [18] assuming that the initial distribution of the Hit-and-Run Markov chain is not far from uniform, i.e., a 'warm start'.

This assumption was later relaxed [21] making Hit-and-Run the only known random walk that converges efficiently to a uniform distribution starting from any point inside a convex body. These results were later extended to log-concave distributions over convex bodies [19]. The Hit-and-Run sampler has found many applications such as identifying non-redundant constraints [7], global optimization [25, 34], convex optimization [8, 16], and computing the volume of convex bodies [20]. The algorithm described in this paper is in essence an analogous version of Hit-and-Run that works for finite sets $S \subseteq H \subset Z^n$, and hence the name Discrete Hit-and-Run (DHR).

Every one-step transition of DHR has three components. The first step involves running an independent pair of nearest neighbor random walks on H that start at the current state of the Markov chain and stop when they step out of H. The term *Biwalk* will be used to refer to this pair of random walks. Moreover, one of these two random walks will be called the *forward walk* and the other the *backward walk* (the choice of which walk to call 'forward' is arbitrary without loss of generality). The ordered sequence of points visited by the Biwalk will be called the *list* in this paper. In particular, the list is formed by listing the points visited by the backward walk in the reverse order followed by the points visited by the forward walk in their actual order (the initial state of the Biwalk is either listed in the forward walk or the backward walk but not both). In the second step, a candidate point is chosen uniformly from the multiset of points in the list that are also in S. This is implemented in a manner similar to Hit-and-Run where a point is generated uniformly from the list and sent to the oracle to determine whether the point is in S. These two steps are referred to as the candidate generator Markov chain. In the third step, the candidate point is then accepted or rejected by a Metropolis filter with respect to the target distribution π to complete the state transition of DHR. Figure 2 illustrates a Biwalk on a square where S is given by black dots. The reason for employing two independent nearest neighbor walks instead of one and for working with the ordered sequence of points as opposed to the set of points visited is to ensure symmetry of the candidate generator Markov chain. See Theorem 2.1 for a proof of symmetry. It is easy to construct examples where symmetry fails if we employ one nearest neighbor random walk and/or use the set of points visited. In addition, note that the candidate generator chain is globally reaching, i.e., for any two points $x, y \in S$, there is a positive probability that y will be chosen as a candidate point from x. Hence, the DHR candidate generator is irreducible and aperiodic. This together with symmetry implies that its limiting distribution is uniform over S. The limiting distribution of DHR with the Metropolis filter is the target distribution π as desired (see Theorem 2.3).

At this point, it is instructive to notice an analogy between continuous Hit-and-Run and DHR. DHR

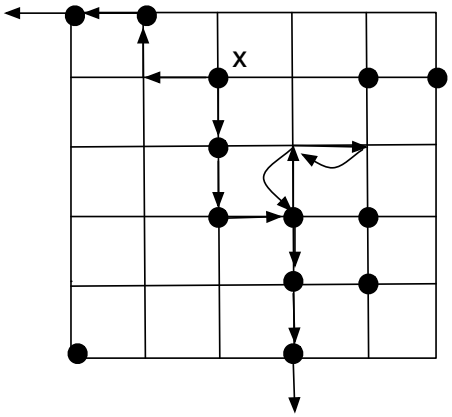

Figure 2: Illustration of a Biwalk starting from x in two dimensions. Set S is shown with black dots, and is a subset of an integer square shown as a grid.

generates a list (instead of a line) followed by a point randomly chosen from the intersection of this list with S (instead of from the line segments associated with the line). Moreover, just like continuous Hit-and-Run, the candidate generator of DHR is globally reaching, which in part distinguishes it from the more standard random walks. See [5, 13, 16] for benefits of employing globally reaching candidate generators for optimization.

The remainder of this paper is organized as follows. In the next section, we formally introduce DHR and state two basic properties of its candidate generator chain. Then, we review well-known ideas regarding mixing times of ergodic Markov chains in the third section. These concepts are then used in the fourth and fifth sections to derive finite-time performance bounds for DHR for special cases of S and π. We conclude in the sixth section.

2 Discrete Hit-and-Run

In this section, we formally present the DHR algorithm introduced in Section 1 and state two results that will be useful in analyzing its finite-time performance in Sections 4 and 5.

Discrete Hit-and-Run (DHR) Algorithm

To make a transition from x to y for any $x, y \in S$

1. Generate a **Biwalk** by running two independent, nearest neighbor random walks in Z^n that start at x and end when they step out of H. One of these two random walks is called the **forward walk**

and the other one is called the **backward walk**. The Biwalk may have loops but has finite length with probability one. The sequence of points visited by the Biwalk is stored in an ordered **list**.

2. Generate a **candidate point** z by choosing a point uniformly distributed from the multiset of points in the list that are also in S. This multiset is called the **segment**.

3. Apply the Metropolis filter to complete the transition to y where,

$$y = \begin{cases} z & \text{with probability } \min(1, \pi(z)/\pi(x)) \\ x & \text{otherwise.} \end{cases}$$

Note that the nearest neighbor random walks in step 1 are implemented on Z^n and hence the probability that any such random walk jumps from its current state i to its neighbor j is $1/(2n)$. In particular, the boundary of hyper-rectangle H does not affect this probability. If point i is on the boundary, and if a nearest neighbor random walk jumps to a neighbor $j \notin H$, then the random walk is terminated at i. Let $Q = \{q_{ij}\}$ denote the transition matrix of the candidate point generator Markov chain in steps 1 and 2 of DHR. As noted in the introduction, the candidate generator of DHR is irreducible, and aperiodic. Theorem 2.1 concludes that it is also symmetric.

Theorem 2.1. *The transition matrix Q of the DHR candidate generator Markov chain is symmetric, i.e., the probability q_{ij} that the Biwalk starting at $i \in S$ generates $j \in S$ as the candidate is the same as the probability q_{ji} that the Biwalk starting at $j \in S$ generates $i \in S$ as the candidate.*

The proof requires some notation and a lemma. We use $B = \{i_1, i_2, \ldots, i_{m_B}\}$ to denote an ordered list as generated by the Biwalk in step 1 of the DHR algorithm, where m_B is the total number of points in list B. Note that a list generated by a Biwalk is characterized by the following two properties; (i) i_1 and i_{m_B} are on the boundary of H (since the backward walk must terminate at i_1 whereas the forward walk must terminate at i_{m_B}) and (ii) i_{k+1} and i_k are nearest neighbors for every k (a point $j \in Z^n$ is a nearest neighbor of, or adjacent to, $i \in Z^n$ if i and j differ in exactly one coordinate and exactly by one). The set of all possible lists with these two properties is denoted by $\mathcal{B}$, $\mathcal{B}_i$ denotes the set of all lists that include $i \in H$, and $\mathcal{B}_{ij}$ denotes the set of all lists that include $i, j \in H$. Note that $\mathcal{B}_{ij}$ is the same as $\mathcal{B}_{ji}$ and is a subset of both $\mathcal{B}_i$ and $\mathcal{B}_j$. Let $P_B(r)$ be the probability that Biwalk B is generated if we start a Biwalk at the point occupying the rth position of B, i.e., i_r. More precisely, $P_B(r)$ is the probability that forward walk $i_r, i_{r+1}, \ldots, i_{m_B}$ in Z^n and backward walk $i_r, i_{r-1}, \ldots, i_1$ in Z^n are generated if we start a Biwalk at i_r.

Lemma 2.2. *$P_B(r)$ is invariant over r, that is,*

$$P_B(r) = P_B(s), \ \ \forall r, s \in \{1, 2, \ldots, m_B\}.$$

Proof. Let $p(i)$ be the probability that a nearest neighbor random walk steps out of H in one step given it is at $i \in H$. If we start a Biwalk at i_r, the probability that forward walk $i_r, i_{r+1}, \ldots, i_{m_B}$ in Z^n is generated is $(1/2n)^{m_B - r}\, p(i_{m_B})$ since steps of the forward walk are implemented independently of each other and the probability of jumping from any point in a forward walk in Z^n to one of its nearest neighbors in Z^n is $1/2n$. Similarly, if we start a Biwalk at i_r, the probability that backward walk $i_r, i_{r-1}, \ldots, i_1$ in Z^n is generated is $(1/2n)^{r-1}\, p(i_1)$. Since the forward and the backward walks are generated independently, we have

$$\begin{aligned} P_B(r) = p(i_1) \left(\frac{1}{2n}\right)^{r-1} \left(\frac{1}{2n}\right)^{m_B - r} p(i_{m_B}) &= p(i_1) \left(\frac{1}{2n}\right)^{m_B - 1} p(i_{m_B}) \\ &= p(i_1) \left(\frac{1}{2n}\right)^{s-1} \left(\frac{1}{2n}\right)^{m_B - s} p(i_{m_B}) = P_B(s). \end{aligned}$$

□

In view of this lemma, we use P_B to denote the probability that list B was generated by a Biwalk starting at the point in any position in B.

Proof of Theorem 2.1 Let $i, j \in S \subset H$. For $B \in \mathcal{B}_i$, let $P_i(B)$ be the probability that the list B is generated if we start our Biwalk at point i. Let $m_B(i)$ be the number of occurrences of point i in B. Then, we have $P_i(B) = m_B(i) P_B$ by Lemma 2.2. For $B \in \mathcal{B}_j$, let $P(j|B)$ be the probability of generating candidate point j uniformly from the segment of B in S as in step 2 of the DHR algorithm given that list B was generated in step 1. We use m_{B_S} to denote the number of feasible points in B, i.e., points also in S. Then, $P(j|B) = m_B(j)/m_{B_S}$. We have

$$q_{ij} = \sum_{B \in \mathcal{B}_{ij}} P_i(B) P(j|B) = \sum_{B \in \mathcal{B}_{ij}} m_B(i) P_B \frac{m_B(j)}{m_{B_S}} = \sum_{B \in \mathcal{B}_{ji}} m_B(j) P_B \frac{m_B(i)}{m_{B_S}} = q_{ji},$$

because $\mathcal{B}_{ij} = \mathcal{B}_{ji}$. This completes the proof. □

The DHR candidate generator Markov chain is symmetric (from Theorem 2.1), irreducible and aperiodic. Thus, its limiting distribution is uniform over S. Moreover, the transition matrix of DHR, denoted

$P = \{p_{ij}\}$, is obtained by applying the Metropolis filter to the candidate generator, and hence, is reversible with respect to π (see [4]). As a result, the limiting distribution of DHR is π (see [4] and the beginning of Section 3). This is precisely stated as

Theorem 2.3. *The DHR Markov chain is ergodic with limiting distribution π. In particular, the k-step distribution of its state converges pointwise to π regardless of the initial state as $k \to \infty$.*

More specifically, if we simulate the DHR Markov chain 'long enough', the distribution of its state arbitrarily well-approximates π. It is crucial to realize the importance of this result in view of the fact that random walks such as the nearest neighbor walk or the coordinate direction random walk fail to converge to a uniform distribution over some of the sets illustrated in Figure 1. The above theorem shows that DHR meets the first goal outlined in Section 1.

The rest of this article focuses on the second goal, i.e., obtaining finite-time performance bounds for DHR for special cases of S and π. In particular, we are interested in deciding how long we have to simulate DHR in order to be 'close' to π. We need the following notation. Let $a = (a_1, \ldots, a_n) \in Z^n$ and $b = (b_1, \ldots, b_n) \in Z^n$ be the lower and upper bounds of the hyper-rectangle H, that is,

$$H = \{x = (x_1, \ldots, x_n) \in Z^n : a_i \leq x_i \leq b_i \text{ for all } i = 1, \ldots, n\}. \tag{1}$$

Let $L_i = b_i - a_i + 1$ for $i = 1, \ldots, n$ be the number of discrete points along the ith coordinate of the hyper-rectangle. Let $L = \max\{L_i : i = 1, \ldots, n\}$ be the length of the longest side of H. We only consider non-trivial cases where $b_i > a_i$ for at least one i implying $L \geq 2$.

The proposition below provides a (uniform) upper bound on the expected length of the forward (or the backward) walk generated in step 1 of the DHR algorithm starting at any point $x \in S$.

Proposition 2.4. *Let W_x be the random variable for the number of steps of the forward walk (or backward walk) that starts at $x \in H$ and stops when it steps out of H. Then*

$$E[W_x] \leq n(L+2)^2/4, \tag{2}$$

which is in turn bounded above by nL^2 for all non-trivial cases $L \geq 2$.

Proof. To prove the proposition, we first recall the symmetric gambler's ruin problem. In the gambler's ruin problem there are two players with initial non-negative fortunes c and d dollars respectively. They each wage a dollar to repeatedly play a competitive game where they each have an equal probability of

winning. Let $N(c,d)$ denote the random variable representing the number of plays of this game until one of the players has lost all of his money. It is easy to verify that $E[N(c,d)] = cd$.

Now consider the forward walk of the Biwalk starting at $x \in H$. As there are $2n$ faces of the hyper-rectangle, there are $2n$ ways that the random walk can step out of H. For any coordinate $j \in \{1,\ldots,n\}$, the walk exits H if either $x_j - a_j + 1$ net decrementing steps or $b_j - x_j + 1$ net incrementing steps are made in the jth coordinate. Thus, if each coordinate, j, is considered as a gambler's ruin with initial fortunes $x_j - a_j + 1$ and $b_j - x_j + 1$ respectively, then the problem of bounding the number of steps until the walk exits H amounts to finding the coordinate j gambler's ruin that terminates first. However, each step of the random walk amounts to independently and uniformly choosing one of the n gambler's ruin problems. By the pigeonhole principal, after kn steps of the random walk, there is a gambler's ruin problem, label it j_k^*, that has been chosen at least k times. Thus, if the coordinate j_k^* gambler's ruin problem has terminated within k iterates, then the random walk on the integer lattice has terminated within kn iterates. Hence, for the coordinate j_k^* gambler's ruin problem at iteration nk, if $N(x_{j_k^*} - a_{j_k^*} + 1, b_{j_k^*} - x_{j_k^*} + 1) \leq k$, then $W_x \leq kn$. Hence,

$$P(W_x > kn) \leq P\Big(N(x_{j_k^*} - a_{j_k^*} + 1, b_{j_k^*} - x_{j_k^*} + 1) > k\Big). \tag{3}$$

The right hand side examines a gambler's ruin problem in which the total fortune of the two players is $b_{j_k^*} - a_{j_k^*} + 2$. Suppose that this fortune is distributed evenly between the players at the beginning of the game. Let J be the number of plays before one of the players has a fortune of $x_{j_k^*} - a_{j_k^*} + 1$. Note that J is a random variable with positive integer values and is finite almost surely. Hence,

$$\begin{aligned} &P\Big(N\Big(\Big\lfloor \frac{b_{j_k^*} - a_{j_k^*} + 2}{2} \Big\rfloor, \Big\lceil \frac{b_{j_k^*} - a_{j_k^*} + 2}{2} \Big\rceil\Big) > k \text{ for some } j_k^* \in \{1,2,\cdots,n\}\Big) \\ &\quad = P\Big(N(x_{j_k^*} - a_{j_k^*} + 1, b_{j_k^*} - x_{j_k^*} + 1) > k - J\Big) \\ &\quad \geq P\Big(N(x_{j_k^*} - a_{j_k^*} + 1, b_{j_k^*} - x_{j_k^*} + 1) > k\Big). \end{aligned} \tag{4}$$

Finally, it is noted that by endowing both players with additional initial fortunes, $L \geq b_{j_k^*} - a_{j_k^*} + 1$, the left side of Equation (4) is bounded above by

$$P\Big(N\Big(\Big\lfloor \frac{L+1}{2} \Big\rfloor, \Big\lceil \frac{L+1}{2} \Big\rceil\Big) > k\Big). \tag{5}$$

Combining Equation (3), Equation (4), and Equation (5),

$$P(W_x > kn) \leq P\Big(N\Big(\Big\lfloor \frac{L+1}{2} \Big\rfloor, \Big\lceil \frac{L+1}{2} \Big\rceil\Big) > k\Big). \tag{6}$$

Thus,

$$\begin{aligned} E[W_x] &= \sum_{j\geq 0} P(W_x > j) = \sum_{k\geq 0}\sum_{i=0}^{n-1} P(W_x > kn+i) \leq \sum_{k\geq 0} nP(W_x > kn) \\ &= n\sum_{k\geq 0} P(W_x > kn) \leq n\sum_{k\geq 0} P\Big(N\Big(\Big\lfloor \frac{L+1}{2} \Big\rfloor, \Big\lceil \frac{L+1}{2} \Big\rceil\Big) > k\Big) \qquad (7) \\ &= nE\Big[N\Big(\Big\lfloor \frac{L+1}{2} \Big\rfloor, \Big\lceil \frac{L+1}{2} \Big\rceil\Big)\Big] = n\Big(\Big\lfloor \frac{L+1}{2} \Big\rfloor \Big\lceil \frac{L+1}{2} \Big\rceil\Big) \leq n(L+2)^2/4, \qquad (8) \end{aligned}$$

where Equation (7) follows from Equation (6) and the last inequality in Equation (8) follows from the mean time of a gambler's ruin and the mean geometric inequality. The right hand side in (8) is in turn bounded by nL^2 for all non-trivial cases $L \geq 2$. □

The main utility of the above proposition is in proving the following crucial result.

Proposition 2.5. *Let $i = (i_1, \ldots, i_n)$ and $j = (j_1, \ldots, j_n)$ be two points in S. Let $d(i,j)$ denote the Manhattan or l_1 distance between i and j; $d(i,j) = \sum_{t=1}^{n} |i_t - j_t|$. Then, for all non-trivial cases $L \geq 2$, the transition matrix Q of the DHR candidate generator satisfies*

$$q_{ij} \geq \frac{1}{2^{d(i,j)+2} n^{d(i,j)+1} L^2}.$$

Proof. Recall that the Biwalk is comprised of a forward walk and a backward walk on H starting at a point $i \in S$ and ending when the walk steps out of H. A transition is made to j, if both j is in the Biwalk list and it is the point chosen as the uniform sample from the segment of the list contained in S. Let B_1 be the event that the forward walk takes a specific shortest route from i to j. It is clear that $P(B_1) = 1/(2n)^{d(i,j)}$. Let B_2 be the event that j is chosen as the uniform sample from the Biwalk. Then q_{ij} is equal to $P(B_2|B_1)P(B_1) + P(B_2|\text{not } B_1)P(\text{not } B_1)$, which is at least $P(B_2|B_1)P(B_1) = \frac{1}{(2n)^{d(i,j)}} P(B_2|B_1)$.

Now denote the lengths of the forward and the backward random walks by X_1 and X_2 respectively. As i is the starting point in each of these walks, the Biwalk has length $X = X_1 + X_2 - 1$. If the Biwalk goes through j and has length k, then the probability that j is chosen is bounded below by $1/k$, and then $P(B_2|X = k, B_1) \geq 1/k$. Also denote X_1' as the remaining length of X_1 starting at j conditional

on event B_1. Thus, $X_1' = X_1 - d(i,j)$. Hence, $q_{ij} \geq \frac{1}{(2n)^{d(i,j)}} P(B_2|B_1) = \frac{1}{(2n)^{d(i,j)}} \sum_{k\geq 1} P(B_2|X = k, B_1)P(X = k|B_1)$, which is bounded below by $\frac{1}{(2n)^{d(i,j)}} \sum_{k\geq 1} \frac{1}{k} P(X = k|B_1) = \frac{1}{(2n)^{d(i,j)}} E\left[\frac{1}{X}\middle| B_1\right] = \frac{1}{(2n)^{d(i,j)}} E\left[\frac{1}{X_1+X_2-1}\middle| B_1\right]$. The last term is simply equal to $\frac{1}{(2n)^{d(i,j)}} E\left[\frac{1}{X_1'+X_2+d(i,j)-1}\middle| B_1\right]$. Since X_1' is the length of the forward walk starting from j and X_2 is the length of the backward walk starting at i, both are independent of B_1. Then the above term is equal to $\frac{1}{(2n)^{d(i,j)}} E\left[\frac{1}{X_1'+X_2+d(i,j)-1}\right]$, which is at least $\frac{1}{(2n)^{d(i,j)}} E\left[\frac{1}{X_1'+X_2+nL}\right]$. Jensen' inequality then yields the lower bound $\frac{1}{(2n)^{d(i,j)}} \frac{1}{E(X_1')+E(X_2)+nL}$. The last term is at least $\frac{1}{(2n)^{d(i,j)}} \left(\frac{1}{2nL^2+nL}\right)$ by applying Proposition 2.4 for non-trivial cases $L \geq 2$. The final expression is bounded below by $\frac{1}{(2n)^{d(i,j)}} \left(\frac{1}{3nL^2}\right)$, which in turn is at least $\frac{1}{2^{d(i,j)+2} n^{d(i,j)+1} L^2}$. □

For any two points $i = (i_1, \ldots, i_n)$ and $j = (j_1, \ldots, j_n)$ in S, we define $\partial(i,j)$ as the number of co-ordinates in which points i and j differ. Mathematically, $\partial(i,j) = |\{r \in \{1, \ldots, n\} : i_r \neq j_r\}|$. Note that $d(i,j)$ is independent of n if and only if $\partial(i,j)$ is independent of n. This leads to a Corollary that provides sufficient conditions under which q_{ij} is bounded below by an inverse polynomial, and in particular, is not exponentially small.

Corollary 2.6. *Let $i, j \in S$.*

1. *Suppose i and j are nearest neighbors, i.e., $d(i,j) = 1$. Then $q_{ij} \geq \frac{1}{(8n^2L^2)}$, i.e., q_{ij} is at least inverse polynomial.*

2. *More generally, suppose points i and j are along one co-ordinate axis, i.e., $\partial(i,j) = 1$, and let $d(i,j) = d \leq L$ for some constant d. Then $q_{ij} \geq \frac{1}{(2^{d+2} n^{d+1} L^2)}$, i.e., q_{ij} is at least inverse polynomial.*

3. *Even more generally, suppose i and j are such that $\partial(i,j) = c$ where c is independent of n. Then $q_{ij} \geq \frac{1}{(2^{cL+2} n^{cL+1} L^2)}$, i.e., q_{ij} is at least inverse polynomial.*

Further analysis of DHR is based on well-known results on rates of convergence of Markov chains that we briefly review in Section 3. The reader is referred to [4, 11, 12, 26, 27] for details.

3 Review of Rapid Mixing Markov Chains

We digress from our problem setting and review relevant material for general finite state Markov chains. Let S be any finite set, and $\mathcal{M} = \mathcal{M}(x,y)$ be the transition matrix of a discrete-time Markov chain on state space S. Suppose that $\mathcal{M}$ is irreducible and reversible with respect to a probability distribution μ

on S, i.e., it satisfies the detailed balance equations

$$R(x,y) \equiv \mu(x)\mathcal{M}(x,y) = \mu(y)\mathcal{M}(y,x) \ \ \forall x,y \in S.$$

This then implies that μ satisfies $\mu\mathcal{M} = \mu$, i.e., μ is a stationary distribution for $\mathcal{M}$. If $\mathcal{M}$ is also aperiodic, the distribution of the state at time t converges to μ as $t \to \infty$, and the chain is said to be *ergodic*. If we simulate $\mathcal{M}$ for a sufficiently long time and observe the final state, we have an algorithm for sampling elements of S from a distribution arbitrarily close to μ.

We can identify a reversible Markov chain $\mathcal{M}$ with its *transition graph*, which is a weighted, undirected graph G with vertex set S. The edge set E is given by edges of weight $R(x,y)$ connecting vertices $x, y \in S$ if and only if $R(x,y) > 0$.

It is well-known [26] that $\mathcal{M}$ has real eigenvalues $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \ldots \lambda_{N-1} \geq -1$, where N is the cardinality of S and $\lambda_{N-1} > -1$ since $\mathcal{M}$ is ergodic. For an ergodic chain, the second largest eigenvalue in absolute value, denoted λ^*, governs the rate of convergence to μ. Let x be the state of the chain at time $k = 0$. Let $\mathcal{M}^k(x,\cdot)$ denote the state distribution at time k. The variation distance at time k with initial state x is given by $\Delta_x(k) = \frac{1}{2}\sum_{y \in X} |\mathcal{M}^k(x,y) - \mu(y)|$. The rate of convergence of $\mathcal{M}$ to μ is characterized using the function $\tau_x(\epsilon)$ defined for any $\epsilon > 0$ by $\tau_x(\epsilon) = \min\{k : \Delta_x(k') \leq \epsilon \ \text{ for all } \ k' \geq k\}$. This function is called the mixing time of the Markov chain $\mathcal{M}$, and it is the *smallest* time k such that the variation distance is less than ϵ for all time steps including and after k. The mixing time is well defined because the variation distance is nonincreasing in k (see [23], Theorem 11.4, page 280). The variation distance has a nice interpretation that follows from the Coupling Lemma ([23] page 275): if we observe the state of a Markov chain at a time step when the variation distance is at most ϵ, the probability that the sampled state is distributed *exactly* according to μ is at least $(1-\epsilon)$. The mixing time is related to λ^* as follows (see [26], Proposition 1).

Proposition 3.1. *The mixing time* $\tau_x(\epsilon)$ *satisfies* $\tau_x(\epsilon) \leq \frac{1}{1-\lambda^*}\log\frac{1}{\mu(x)\epsilon}$ *and* $\max_{x \in S} \tau_x(\epsilon)$ *is at least* $\frac{\lambda^*}{2(1-\lambda^*)}\log\frac{1}{2\epsilon}$.

Note that the first inequality gives an upper bound on the time required to reach near stationarity from a given initial state x. The second inequality asserts that convergence cannot be rapid unless λ^* is bounded away from 1. In particular, rapid mixing can be identified with a large value of the spectral gap $(1-\lambda^*)$, i.e., a small value of $1/(1-\lambda^*)$. It is customary (see [4], [26], [27]) to ignore the smallest eigenvalue λ_{N-1} by following a simple approach: add a stalling probability of $\frac{1}{2}$ to every state, i.e., modify $\mathcal{M}$ to $\frac{1}{2}(I+\mathcal{M})$,

where I is the $N \times N$ identity matrix. In that case, all eigenvalues are non-negative and the gap $(1 - \lambda_1)$ is decreased only by a factor of 2. We can then focus on finding an upper bound on $1/(1 - \lambda_1)$. This modified Markov chain is called the *lazy* chain. We follow this approach when analyzing DHR.

Various methods have been proposed in the literature for estimating λ_1. These include the conductance approach [26, 27], Poincaré inequality approach [11], canonical path approach [26], and combinatorial approach [22]. In this paper, we mainly employ the canonical path approach as in [26]. The idea in this approach is to construct a canonical path γ_{xy} in the transition graph G between each ordered pair of distinct states x and y. The collection of these paths is denoted by $\Gamma = \{\gamma_{xy}\}$. Then one employs the quantity $\bar{\rho}(\Gamma)$ defined below to get a bound on λ_1. Mathematically,

$$\bar{\rho}(\Gamma) = \max_{e \in E} \frac{1}{R(e)} \sum_{\gamma_{xy} \ni e} \mu(x)\mu(y)|\gamma_{xy}|, \tag{9}$$

where $|\gamma_{xy}|$ is the length (number of edges) of path γ_{xy}, and $R(e) = R(u, v)$ for edge $e = (u, v)$. The following relation between $\bar{\rho}(\Gamma)$ and λ_1 is well-known (see [26] Theorem 5).

Theorem 3.2. *For any reversible Markov chain, and any choice of canonical paths* Γ*, the second eigenvalue* λ_1 *satisfies* $\frac{1}{1-\lambda_1} \leq \bar{\rho}(\Gamma)$.

We say that the canonical path $\gamma_{xy} \in \Gamma$ *meets* edge $e = (i, j) \in E$ if e lies on γ_{xy}. We use $S_{ij}(\Gamma)$ to denote the set of pairs $(x, y) \in S \times S$ such that the canonical path $\gamma_{xy} \in \Gamma$ meets edge $(i, j) \in E$.
We now employ these ideas to analyze the mixing time of DHR. We start with the uniform distribution.

4 DHR Mixing Time When π is Uniform

Suppose the target distribution π is uniform over S. In that case, every candidate point generated by the DHR candidate generator Q in step 2 is accepted by the Metropolis filter in step 3. In other words, step 3 has no effect on the evolution of the DHR Markov chain. As mentioned earlier, the limiting distribution of the candidate generator Markov chain Q is uniform over S. Thus it suffices to analyze finite time performance of the candidate generator. Let $G \equiv G(S, E)$ denote the Markov transition graph for the DHR candidate generator Markov chain with state space S and transition matrix Q. Suppose Γ is any prescription of canonical paths and let $S_{ij}(\Gamma)$ be the set of pairs $(x, y) \in S \times S$ such that the canonical path $\gamma_{xy} \in \Gamma$ meets edge $(i, j) \in E$. For given $u \in S$, we define $E_u(S, \Gamma)$ as the set of points $v \in S$ for which there exist distinct nodes x, y in S such that the path $\gamma_{xy} \in \Gamma$ meets edge (u, v) in E. Mathematically,

$E_u(S,\Gamma) = \{v \in S : \exists x, y \in S \text{ such that } \gamma_{xy} \in \Gamma \text{ meets } (u,v)\}$. Moreover, let $I(S,\Gamma)$ be the maximum l_1 distance between any points u and v in S for which there exists a path prescribed by Γ that meets edge $e = (u,v)$, i.e.,

$$I(S,\Gamma) = \max_{u \in S} \max_{v \in E_u(S,\Gamma)} d(u,v). \tag{10}$$

Theorem 4.1. *Let $\Gamma = \{\gamma_{xy}\}$ be any prescription of canonical paths in the transition graph $G(S,E)$ of the DHR candidate generator Markov chain Q over S as described above. Let $\gamma(S,\Gamma) = \max_{x,y \in S} |\gamma_{xy}|$ be the length of the longest path prescribed by Γ in $G(S,E)$, $S(\Gamma) = \max_{(u,v) \in E} |S_{uv}(\Gamma)|$ be the cardinality of set $S_{uv}(\Gamma)$ for the edge (u,v) that meets the most number of paths in Γ. Then for any initial state $x \in S$ and any $\epsilon > 0$, the ϵ mixing time of the DHR candidate generator Markov chain Q is bounded as follows:*

$$\tau_x(\epsilon) \leq \frac{2^{I(S,\Gamma)+2} n^{I(S,\Gamma)+1} L^2 \gamma(S,\Gamma) S(\Gamma)}{|S|} \log \frac{|S|}{\epsilon}. \tag{11}$$

Proof. Let $\bar{\rho}_Q(\Gamma)$ denote the quantity defined in Equation (9) for the special case of transition matrix Q. Proposition 3.1, Theorem 3.2 and $\pi(x) = 1/|S|$ since π is uniform imply that $\tau_x(\epsilon) \leq \bar{\rho}_Q(\Gamma) \log \frac{|S|}{\epsilon}$. We show that $\bar{\rho}_Q(\Gamma) \leq \frac{2^{I(S,\Gamma)+2} n^{I(S,\Gamma)+1} L^2 \gamma(S,\Gamma) S(\Gamma)}{|S|}$ to complete the proof. Equation (9) implies that $\bar{\rho}_Q(\Gamma)$ is

$$= \max_{(u,v) \in E} \frac{1}{\pi(u) q_{uv}} \sum_{\gamma_{xy} \ni (u,v)} \pi(x)\pi(y)|\gamma_{xy}| = \max_{(u,v) \in E} \frac{1}{|S| q_{uv}} \sum_{\gamma_{xy} \ni (u,v)} |\gamma_{xy}| \leq \max_{(u,v) \in E} \frac{\gamma(S,\Gamma)}{|S| q_{uv}} \sum_{\gamma_{xy} \ni (u,v)} 1$$

$$= \max_{(u,v) \in E} \frac{\gamma(S,\Gamma)|S_{uv}(\Gamma)|}{|S| q_{uv}} \leq \frac{\gamma(S,\Gamma) S(\Gamma)}{|S|} \max_{(u,v) \in E} \frac{1}{q_{uv}} = \frac{\gamma(S,\Gamma) S(\Gamma)}{|S|} \max_{u \in S} \max_{v \in E_u(S,\Gamma)} \frac{1}{q_{uv}}$$

$$\leq \frac{\gamma(S,\Gamma) S(\Gamma)}{|S|} \max_{u \in S} \max_{v \in E_u(S,\Gamma)} 2^{d(u,v)+2} n^{d(u,v)+1} L^2 = \frac{2^{I(S,\Gamma)+2} n^{I(S,\Gamma)+1} L^2 \gamma(S,\Gamma) S(\Gamma)}{|S|},$$

where the last inequality follows from Proposition 2.5 and the last equality from the definition of $I(S,\Gamma)$. This completes the proof. □

In order to use this theorem, we need to (i) estimate $|S|$, (ii) choose a prescription of canonical paths Γ that works for S, and (iii) compute (upper bounds on) $\gamma(S,\Gamma)$, $S(\Gamma)$ and $I(S,\Gamma)$. Since $|S|$ is known to be at most L^n, the logarithm on the right hand side of Equation (11) is at most $n \log L + \log(1/\epsilon)$ and hence grows at most linearly with n. In our experience (see, for example, the sets analyzed in Section 4), for most prescriptions of canonical paths Γ on subsets S of n-dimensional integer hyper-rectangles H of sides at most L, $\gamma(S,\Gamma)$ is at most Ln. When that is true, the polynomiality of the mixing time of the

DHR candidate generator crucially depends on the ratio $S(\Gamma)/|S|$ and the number $I(S,\Gamma)$. In particular, when the former is polynomial in n and the latter is independent of n, the DHR candidate generator mixes in polynomial time. Although estimating the above quantities can be challenging in general, we successfully compute them for the five cases illustrated in Figure 1. We derive polynomial upper bounds on the mixing time of the DHR candidate generator for the first four examples but are able to obtain only exponential bounds for the fifth. We introduce some simple terminology that will be used in the sequel.

Definition 4.2. *We say that*

1. *A point $i \in S$ is an* **isolated** *point if it has no nearest neighbors in S.*

2. *Hence S* **has no isolated points** *if every point in S has at least one nearest neighbor in S.*

It is useful to contrast a set without any isolated points with a path connected set defined below.

Definition 4.3. *A staircase path between two points x and y of the integer lattice Z^n is a finite sequence of points $(x = i_1, i_2, ..., i_m = y) \in Z^n$ such that i_k and i_{k+1} are nearest neighbors on Z^n for $k = 1, 2, \ldots, m-1$. We say that S is* **path connected** *if there exists a staircase path between any two points $x, y \in S$ that lies in S.*

Notice that a path connected set has no isolated points but the converse is not true. The sets in Figures 1 (a) and (b) have no isolated points and are path connected, the set in Figure 1 (c) has no isolated points but is not path connected, and finally, every point in the sets in Figures 1 (d) and (e) is isolated and hence they are not path connected.

4.1 Box Inside a Box

Suppose $R \subseteq H$ is a hyper-rectangular subset of H whose sides (oriented along sides of H) have lengths $r_1, r_2, \ldots, r_n$ respectively, and let $r = \max\{r_1, \ldots, r_n\} \leq L$ be the length of the longest side of R. Refer to Figure 3 (a) for an example. We choose the *coordinate direction prescription* of canonical paths [4], denoted Γ_0. This prescription constructs a path between any two points x, y in a bounded, hyper-rectangular subset of Z^n by increasing or decreasing the first component of x each time by one until it matches the first component of y, then repeats this for the second component of x, and so on. Thus, it follows a 'lowest index first' rule. We begin with a lemma.

Lemma 4.4. *For any $(i,j) \in E$ with $S = T$, the cardinality $|S_{ij}(\Gamma_0)|$ satisfies*

$$|S_{ij}(\Gamma_0)| \leq \frac{(r_1 \times r_2 \times \cdots \times r_n) \times r}{4}. \tag{12}$$

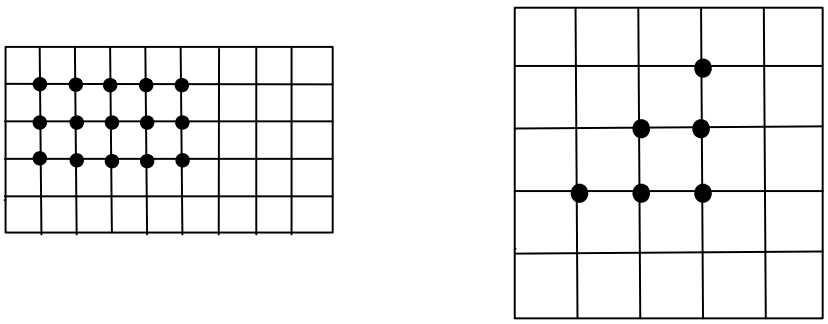

Figure 3: Illustration of a box inside a box and a wedge inside a cube. The set H consists of all the grid points. Set S is shown with black dots. (a) Box inside a box: $S = R$ with $L_1 = 10$, $L_2 = 6$, $r_1 = 5$, $r_2 = 3$. (b) Wedge inside a box: $S = W(2, 3, 6)$.

As a result, $S(\Gamma_0)$ *is at most* $\frac{(r_1 \times r_2 \times \cdots \times r_n) \times r}{4}$.

Proof. Let R be such that its side along the kth coordinate is given by integer lattice points $\{x_1^k, \ldots, x_{r_k}^k\}$. When i and j are not adjacent nodes in the integer lattice, $|S_{ij}(\Gamma_0)| = 0$ since no path prescribed by Γ_0 meets edge (i, j). In that case, the upper bound in Equation (12) holds trivially. Now suppose that i and j are adjacent nodes in the integer lattice. Without loss of generality, suppose that $j_\beta = i_\beta + 1$ for some $\beta \in \{1, \ldots, n\}$ and $i_k = j_k$ for $k \neq \beta$. Then a canonical path γ_{st} passes through the directed edge (i, j) if and only if (s, t) is of the form,

$$((\alpha_1, \ldots, \alpha_{\beta-1}, i_\beta^-, i_{\beta+1}, \ldots, i_n), (i_1, \ldots, i_{\beta-1}, j_\beta^+, \alpha_{\beta+1}, \ldots, \alpha_n)), \tag{13}$$

where $\alpha_k \in \{x_1^k, \ldots, x_{r_k}^k\}$ for $k \notin \beta$ and $i_\beta^- \leq i_\beta < i_\beta + 1 = j_\beta \leq j_\beta^+$. Hence there are at most

$$\begin{aligned}
&(r_1 \times \cdots \times r_{\beta-1})(i_\beta - x_1^\beta + 1)(x_{r_\beta}^\beta - i_\beta)(r_{\beta+1} \times \cdots \times r_n) = \\
&\quad = (r_1 \times \cdots \times r_{\beta-1})(i_\beta + 1 - x_1^\beta)((x_{r_\beta}^\beta + 1) - (i_\beta + 1))(r_{\beta+1} \times \cdots \times r_n) \\
&\quad \leq (r_1 \times \cdots \times r_{\beta-1})\Big(\frac{x_{r_\beta}^\beta - x_1^\beta + 1}{2}\Big)^2 (r_{\beta+1} \times \cdots \times r_n)
\end{aligned} \tag{14}$$

possible pairs (s, t) whose γ_{st} meet (i, j). Equation (14) follows from the mean-geometric inequality. Hence the number of canonical paths, γ_{st}, that meet the edge (i, j) is bounded above by $(r_1 \times \cdots \times r_n) \times r/4$, which does not depend on (i, j). □

In addition, it is easy to see that $|\gamma_{xy}| \leq rn$ for any path γ_{xy} prescribed by Γ_0, implying $\gamma(R, \Gamma_0) \leq rn$.

Moreover, a path prescribed by Γ_0 passes through an edge $e = (u, v)$ only if u and v are nearest neighbors in S, meaning $d(u, v) = 1$. As a result, $I(R, \Gamma_0) = 1$. Finally, $|R| = (r_1 \times \cdots \times r_n) \leq r^n$. In view of Theorem 4.1, this discussion leads to the following corollary.

Corollary 4.5. *The ϵ mixing time for the DHR candidate generator on R is at most $(2n^3r^2L^2(n \log r + \log\left(\frac{1}{\epsilon}\right)))$ regardless of the initial state.*

Notice for example that when $(L/r) = 2$, implying R has exponentially smaller volume than H, the Biwalk is **exponentially faster** than an n dimensional rejection technique. The intuitive reason for this is that the Biwalk performs a *one dimensional rejection* on the list to compute the segment.

4.2 Wedge Inside a Cube

In this section we assume that H is a hyper-cube of side L, denoted $C(n, L)$. Let $C(n, M) \subseteq C(n, L)$ be another hyper-cube of side $M \leq L$, whose sides are oriented along those of $C(n, L)$. Moreover, let S be the *wedge* formed by removing the lattice points of $C(n, M)$ that lie *strictly above* one of its symmetric hyperplanes (see Figure 1 (b) and [1] for a classification of hyperplanes in hyper-cubes). We denote such a wedge by $W(n, M, L)$. Figure 3 (b) shows the wedge $W(2, 3, 6)$. Note that the coordinate direction prescription of canonical paths Γ_0 described above also works for the wedge. Specifically, the upper bound in Lemma 4.4 holds implying that $|S_{ij}(\Gamma_0)| \leq M^{n+1}/4$. It is easy to see that $|\gamma_{xy}| \leq Mn$ for any path γ_{xy} prescribed by Γ_0 implying $\gamma(W(n, M, L), \Gamma_0) \leq Mn$. In addition, recall that no Γ_0 paths meet edge $e = (u, v)$ if $d(u, v) \neq 1$ implying $I(W(n, M, L), \Gamma_0) = 1$. Finally, observe that $M^n \geq |W(n, M, L)| \geq M^n/2$. Theorem 4.1 then implies

Corollary 4.6. *The ϵ mixing time for the DHR candidate generator on $W(n, M, L)$ is at most*

$$4n^3M^2L^2\left(n \log M + \log\left(\frac{1}{\epsilon}\right)\right)$$

regardless of the initial state.

Note that conventional random walks such as the nearest neighbor random walk and the co-ordinate direction random walk also mix in polynomial time on the two sets analyzed above. Therefore we now present more interesting examples where one or both of these walks get stuck in isolated regions of S and fail to converge to a uniform distribution.

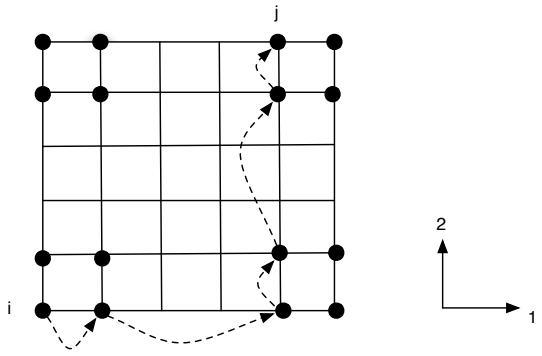

Figure 4: Co-ordinate direction prescription with hops shown with dotted arrows from point i to point j in $C'(2, 2, 6)$. The co-ordinate directions are shown with arrows next to the picture.

4.3 Multiple Cubes Inside a Cube

Again, let H be the n-dimensional hyper-cube of side L, denoted $C(n, L)$. Consider the case where S consists of 2^n n-dimensional cubes each of side $M < L/2$ embedded in the 2^n corners of $C(n, L)$. We denote such an S by $C'(n, M, L)$. The set shown in Figure 1 (c) is a special case of this situation where $n = 2$, $L = 6$ and $M = 2$. Note that $C'(n, M, L)$ is not path connected but has no isolated points (unless $M = 1$, in which case every point is isolated). The prescription Γ_0 does not work here. We introduce the *co-ordinate direction prescription with hops*, denoted Γ_1. As the name suggests, this prescription is identical to Γ_0 except that it allows jumps of size more than one if and only if they are needed (see Figure 4). It is easy to see that the result of Lemma 4.4 for an n-dimensional hyper-cube of side $2M$ continues to hold for prescription Γ_1. In other words, we have $|S_{ij}(\Gamma_1)| \leq (2M)^{n+1}/4$ when $S = C'(n, M, L)$. In addition, $|\gamma_{xy}| \leq 2Mn$ for any path γ_{xy} prescribed by Γ_1 implying that $\gamma(C'(n, M, L), \Gamma_2) \leq 2Mn$. Moreover, if a path prescribed by Γ_1 meets an edge $e = (u, v)$ then u and v differ in exactly one co-ordinate and $d(u, v) \leq (L-2M)+1$ implying that $I(C'(n, M, L), \Gamma_2) = (L-2M)+1$. Finally, $|C'(n, M, L)| = (2^n)(M^n)$. Then Theorem 4.1 yields

Corollary 4.7. *The ϵ mixing time for the DHR candidate generator on $C'(n, M, L)$ is at most*

$$2^{L-2M+3} n^{L-2M+3} M^2 L^2 \left(n \log(2M) + \log\left(\frac{1}{\epsilon}\right)\right)$$

regardless of the initial state.

4.4 A Box with Isolated Points

Now consider the case where S is an n-dimensional box-shaped subset of H oriented along its sides however this time every point of S is isolated (an n-dimensional generalization of Figure 1 (d)). Let $r_1, r_2, \ldots, r_n$ be the numbers of lattice points along the n sides of S with $r = \max_{i=1,\ldots,n} \{r_i\}$. We denote such an S by $\tilde{R}$. For any point $x \in \tilde{R}$, let $N_x(\tilde{R})$ be the set of points that differ from x in exactly one co-ordinate and $\delta(\tilde{R}) = \max_{x \in \tilde{R}} \max_{y \in N_x(\tilde{R})} d(x, y)$, i.e., the biggest Manhattan distance between any two points in $\tilde{R}$ that are located on the same co-ordinate axis. The set illustrated in Figure 1 (d) with $n = 2$, has $r_1 = r_2 = r = 3$ and $\delta(\tilde{R}) = 3$. We again use the co-ordinate direction prescription with hops denoted Γ_1 to analyze the DHR candidate generator. It is easy to see that the bound developed in Lemma 4.4 holds here. Therefore, $\tilde{R}(\Gamma_1) \leq (r_1 \times \ldots \times r_n) \times r/4$. In addition $|\gamma_{xy}| \leq rn$ for any path γ_{xy} prescribed by Γ_1 implying $\gamma(\tilde{R}, \Gamma_1) \leq rn$. Finally, $I(\tilde{R}, \Gamma_1) = \delta(\tilde{R})$, and $|\tilde{R}| = (r_1 \times \ldots \times r_n) \leq r^n$. Then Theorem 4.1 implies

Corollary 4.8. *The ϵ mixing time for the DHR candidate generator on $\tilde{R}$ is at most*

$$2^{\delta(\tilde{R})} n^{\delta(\tilde{R})+2} r^2 L^2 \left(n \log r + \log \left(\frac{1}{\epsilon} \right) \right)$$

regardless of the initial state.

4.5 Diagonal of a Cube

Again, let $C(n, L)$ be the n-dimensional hyper-cube of side L and S be the set of points along $C(n, L)$'s diagonal. This S is denoted $D(C(n, L))$ and $|D(C(n, L))| = L$. The set shown in Figure 1 (e) is a special case of this situation. First notice that none of the prescriptions of canonical paths discussed thus far work for this case. We introduce the *diagonal prescription* of canonical paths Γ_2. According to this prescription, a path is constructed from node i to node j in the transition graph of the DHR candidate generator Markov chain on $D(C(n, L))$ by moving along the diagonal step-by-step. Refer to Figure 5. In order to estimate $S_{ij}(\Gamma_2)$ when $S = D(C(n, L))$, we denote the points in $D(C(n, L))$ as a linearly ordered set $\mathcal{L} = \{1, \ldots, L\}$. With this nomenclature, note that a diagonal path γ_{xy} from $x \in \mathcal{L}$ to $y \in \mathcal{L}$ for $x > y$ meets an edge $e = (u, v)$ in the transition graph for $u, v \in \mathcal{L}$ if and only if $u > v$, $u - v = 1$, $x \geq u$, and

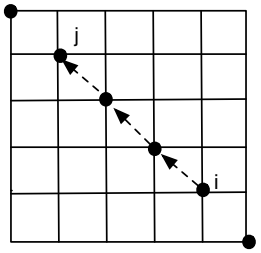

Figure 5: Diagonal prescription of canonical paths.

$v \geq y$. As a result, for $u - v = 1$, we have

$$|S_{uv}(\Gamma_2)| = \sum_{x=u}^{L} \sum_{y=1}^{v} 1 \leq \sum_{x=1}^{L} \sum_{y=1}^{L} 1 = L^2.$$

Moreover, for any $x, y \in \mathcal{L}$, $|\gamma_{xy}|$ is at most L implying $\gamma(D(C(n, L)), \Gamma_2) \leq L$. In addition, if a path prescribed by Γ_2 meets an edge $e = (u, v)$ then $d(u, v) = n$. Therefore, $I(D(C(n, L)), \Gamma_2) = n$. Then Theorem 4.1 implies

Corollary 4.9. *The ϵ mixing time for the DHR candidate generator on $D(C(n, L))$ is at most*

$$2^{n+2} n^{n+1} L^4 \log\left(\frac{L}{\epsilon}\right)$$

regardless of the initial state.

Unlike earlier examples, our upper bound on mixing time of DHR on the diagonal of a cube is exponential. As a result, this example hints at a rule of thumb for the practitioner - the DHR candidate generator is likely to exhibit poor finite-time performance when points in S 'are highly isolated'. We quantify this notion through a concept we call *isolation index*.

Definition 4.10. *The* **isolation index of a point** $x \in S$*, denoted $\mathcal{I}_x(S)$, is defined as its l_1 distance from a point $y \in S \setminus \{x\}$ that is closest to x. That is, $\mathcal{I}_x(S) = \min_{y \in S \setminus \{x\}} d(x, y)$. The* **isolation index of set** *S, denoted $\mathcal{I}(S)$, is the isolation index of a point with the highest isolation index, i.e., $\mathcal{I}(S) = \max_{x \in S} \mathcal{I}_x(S)$.*

Clearly, the higher the isolation index of a set, the more isolated its points are. The importance of isolation index $\mathcal{I}(S)$ stems from its relation to $I(S, \Gamma)$ defined in Equation (10). In particular, for any prescription of canonical paths Γ,

$$\mathcal{I}(S) = \max_{x \in S} \min_{y \in S \setminus x} d(x, y) \leq \max_{x \in S} \min_{y \in E_x(S, \Gamma)} d(x, y) \leq \max_{x \in S} \max_{y \in E_x(S, \Gamma)} d(x, y) = I(S, \Gamma), \tag{15}$$

since $E_x(S,\Gamma) \subseteq S \setminus x$. Specifically, when $\mathcal{I}(S)$ is increasing in n, so is $I(S,\Gamma)$, which is likely to make the right hand side in Equation (11) exponential in n. For example, when $S = D(C(n,L))$, the diagonal of a cube analyzed above, $\mathcal{I}(S) = I(S,\Gamma_2) = n$ leading to a poor performance bound in Corollary 4.9. Equation (15) is especially critical for the practitioner since $I(S,\Gamma)$ depends on the choice of canonical paths (which is a somewhat abstract notion) whereas $\mathcal{I}(S)$ is solely a geometric property of set S.

As another example where the DHR candidate generator performs poorly, consider the case where S is constructed 'randomly' by choosing $k+1$ i.i.d. uniformly distributed points in H, the n dimensional cube of side L. Suppose $k << L^n$ so that the points of S are spread out over H and highly isolated. The expected length of the Biwalk that starts at any point $i \in S$ is at most nL^2 according to Proposition 2.4 for $L \geq 2$. Thus, the probability that this Biwalk does not visit any other point of S is roughly $\left(1 - \frac{nL^2}{L^n}\right)^k \approx \left(1 - k\frac{nL^2}{L^n}\right)$. In particular, the probability that this Biwalk visits at least one point of S is roughly $k\frac{nL^2}{L^n}$, which is exponentially small when $k = (L/2)^n$, i.e., when S has as many points as an n dimensional cube of side $L/2$. This shows that the performance of the DHR candidate generator depends not only on the cardinality of S but also on the isolation index of S. We now extend the ideas presented in this section to arbitrary target distributions.

5 DHR Mixing Time When π is Arbitrary

Recall that Q denotes the transition matrix of the DHR candidate generator and P denotes the transition matrix of the DHR Markov chain after employing the Metropolis filter in step 3 of the DHR algorithm. Moreover, the limiting distribution of Q is uniform over S, whereas the limiting distribution of P is π. Let a_π denote $\frac{\min_{i\in S}\pi(i)}{\max_{i\in S}\pi(i)}$ for brevity and $\lambda_1(P)$ be the second eigenvalue of the transition matrix P. Notice that the transition graphs for Q and P are structurally identical but with different edge weights. Therefore, any prescription of canonical paths Γ for the transition graph $G(S,E)$ of Q also works for the transition graph of P. Let $\bar{\rho}_Q(\Gamma)$ and $\bar{\rho}_P(\Gamma)$ be the quantities defined by Equation (9) for these two transition graphs. Let $\gamma(S,\Gamma)$ and $S(\Gamma)$ be as defined in Theorem 4.1. Then using this notation, we have

Theorem 5.1. *For any initial state $x \in S$ and any $\epsilon > 0$, the ϵ mixing time of the DHR Markov chain P is at most $\frac{1}{a_\pi^2}\bar{\rho}_Q(\Gamma)\log\left(\frac{1}{\pi(x)\epsilon}\right)$, which is in turn bounded by*

$$\frac{1}{a_\pi^2}\frac{2^{I(S,\Gamma)+2}n^{I(S,\Gamma)+1}L^2\gamma(S,\Gamma)S(\Gamma)}{|S|}\log\left(\frac{1}{\pi(x)\epsilon}\right).$$

Proof. Equation (9) implies that $\bar{\rho}_P(\Gamma)$ is equal to

$$\max_{(u,v)\in E} \frac{\sum_{\gamma_{xy}\ni(u,v)} \pi(x)\pi(y)|\gamma_{xy}|}{\pi(u)p_{uv}} = \max_{(u,v)\in E} \frac{\sum_{\gamma_{xy}\ni(u,v)} \pi(x)\pi(y)|\gamma_{xy}|}{\pi(u)q_{uv}\min\{1,\pi(v)/\pi(u)\}} \leq \max_{(u,v)\in E} \frac{\sum_{\gamma_{xy}\ni(u,v)} \pi(x)\pi(y)|\gamma_{xy}|}{\left(\min_{x\in S}\pi(x)\right) q_{uv}}$$

$$\leq \max_{(u,v)\in E} \frac{\left(\max_{x\in S}\pi(x)\right)^2}{\left(\min_{x\in S}\pi(x)\right) q_{uv}} \sum_{\gamma_{xy}\ni(u,v)} |\gamma_{xy}| = \max_{(u,v)\in E} \frac{\left(\max_{x\in S}\pi(x)\right)}{a_\pi q_{uv}} \sum_{\gamma_{xy}\ni(u,v)} |\gamma_{xy}|$$

$$= \max_{(u,v)\in E} \frac{|S||S|\left(\max_{x\in S}\pi(x)\right)}{a_\pi q_{uv}} \sum_{\gamma_{xy}\ni(u,v)} \frac{|\gamma_{xy}|}{|S||S|} = \frac{|S|\left(\max_{x\in S}\pi(x)\right)}{a_\pi} \max_{(u,v)\in E} \frac{|S|}{q_{uv}} \sum_{\gamma_{xy}\ni(u,v)} \frac{|\gamma_{xy}|}{|S||S|}$$

$$= \frac{|S|\left(\max_{x\in S}\pi(x)\right)}{a_\pi}\bar{\rho}_Q(\Gamma) \leq \frac{\left(\max_{x\in S}\pi(x)\right)}{\left(\min_{x\in S}\pi(x)\right) a_\pi}\bar{\rho}_Q(\Gamma) \leq \frac{\bar{\rho}_Q(\Gamma)}{a_\pi^2} \leq \frac{1}{a_\pi^2}\frac{2^{I(S,\Gamma)+2}n^{I(S,\Gamma)+1}L^2\gamma(S,\Gamma)S(\Gamma)}{|S|}$$

where the last inequality follows from the bound on $\bar{\rho}_Q(\Gamma)$ derived in the proof of Theorem 4.1. The claim then follows from Proposition 3.1 and Theorem 3.2. □

We now specialize this result to Boltzmann distributions as they are an important component of optimization algorithms akin to SA as discussed in Section 1. Let f be a real valued function defined over S. Let $T > 0$ be a 'temperature' parameter. The Boltzmann (T) distribution on S is given by

$$\pi_T(i) = \frac{e^{-f(i)/T}}{\sum_{k\in S} e^{-f(k)/T}} \quad \forall i \in S. \tag{16}$$

Corollary 5.2. *Suppose* $\max_{j\in S} f(j) - \min_{i\in S} f(i) \leq A(\log n)$ *for some positive constant* A *independent of* n*, i.e., the depth of the function is at most order* $\log n$*, and* $\bar{\rho}_Q(\Gamma) \leq Bn^C$ *for some prescription of canonical paths* Γ*, and constants* $B > 0$*,* $C > 0$ *that are independent of* n*. Then the mixing time of DHR Markov chain* P *for sampling from a Boltzmann* (T) *distribution over* S *is at most* $Bn^{(C+2A/T)} \log\left(\frac{n^{(A/T)}|S|}{\epsilon}\right)$.

Proof. Theorem 5.1 implies that starting at any $x \in S$ and for any $\epsilon > 0$,

$$\tau_x(\epsilon) \leq (n^{A/T})^2 Bn^C \log\left(\frac{\sum_{y\in S}\exp(-f(y)/T)}{\exp(-f(x)/T)\epsilon}\right) \leq (n^{A/T})^2 Bn^C \log\left(\frac{n^{(A/T)}|S|}{\epsilon}\right).$$

This proves the claim. □

Note the condition that the depth of the function be at most order $\log n$ is restrictive since it essentially requires the variation in function f to be very small relative to the cardinality of the hyper-rectangle H. On the other hand, we do not expect polynomial bounds on DHR mixing time for general functions since that includes problems from class NP.

6 Conclusions

In this paper, we introduced Discrete Hit-and-Run, a Markov chain for sampling approximately from arbitrary distributions over arbitrary subsets of integer hyper-rectangles. Unlike other traditional Markov chains such as the nearest neighbor random walk and the co-ordinate direction random walk, DHR does not get trapped in isolated regions or points of the support set. It has a positive probability of moving from one point to any other point in one step, i.e., it is globally reaching and hence irreducible and aperiodic. Moreover, it is a reversible Markov chain on S, ensuring asymptotic convergence to any arbitrary target distribution. We investigated finite-time performance of DHR and discussed several examples where it mixes in polynomial time.

References

[1] O. Aichholzer, and F. Aurenhammer. Classifying hyperplanes in hyper-cubes. *SIAM J. Discrete Math.*, 9(2): 225-232, May 1996.

[2] D. Aldous. On the Markov chain simulation method for uniform combinatorial distributions and simulated annealing. *Probability in the Engineering and Informational Sciences*, 1:33–46, 1987.

[3] D. Aldous and J. Fill. Reversible Markov chains and random walks on graphs. Draft available at http://www.stat.berkeley.edu/users/aldous/book.html.

[4] E. Behrends. Introduction to Markov Chains. Vieweg-Verlag, Braunschweig, 2000.

[5] C. J. P. Bélisle. Convergence theorems for a class of simulated annealing algorithms on R^d. *Journal of Applied Probability*, 29:885–895, 1992.

[6] C.J.P. Bélisle, H.E. Romeijn and R.L. Smith. Hit-and-run algorithms for generating multivariate distributions. *Mathematics of Operations Research*, 18:255–266, 1993.

[7] H.C.P. Berbee, C.G.E. Boender, A.H.G. Rinnooy Kan, C.L. Scheffer, R.L. Smith and J. Telgen. Hit-and-run algorithm for the identification of nonredundant linear inequalities. *Mathematical Programming*, 37:184–270, 1987.

[8] D. Bertsimas and S. Vempala. Solving convex programs by random walks. *Journal of the ACM*, 51(4): 540–556, 2004.

[9] P. Bremaud. Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues. Springer, New York, 1999.

[10] S. R. Das and A. Sinclair. A Markov Chain Monte Carlo Method for Derivative Pricing and Risk Assessment. *Journal of Investment Management*, 3 (1), 2005.

[11] P. Diaconis and D. Stroock. Geometric bounds for eigenvalues of Markov chains. *The Annals of Applied Probability*, 1(1):36–61, 1991.

[12] J. A. Fill. Eigenvalue bounds on convergence to stationarity for nonreversible Markov chains, with an application to the exclusion process. *The Annals of Applied Probability*, 1(1):62–87, 1991.

[13] A. Ghate and R. L. Smith. A Markov chain Monte Carlo method for global optimization using non-reversible and stochastic acceptance probabilities. Technical Report 05-02, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, 2005.

[14] M. Grotschel, L. Lovász, and A. Schrijver, Geometric algorithms and combinatorial optimization, Springer-Verlag, 1993.

[15] M. Jerrum and A. Sinclair. Approximating the permanent. *SIAM Journal of Computing*, 18:1149–1178, 1989.

[16] A. Kalai and S. Vempala. Convex optimization by simulated annealing. *Mathematics of Operations Research*, 31 (2), 253-266, 2006.

[17] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671-680, May 1983.

[18] L. Lovász. Hit-and-run mixes fast. *Mathematical Programming, Series A*, 86:443–461, 1999.

[19] L. Lovász and S. Vempala. Hit-and-run is fast and fun. *Microsoft Research Tech. Rep. MSR-TR-2003-05*, 2003.

[20] L. Lovász and S. Vempala. Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm. Proc. of the 44th IEEE Foundations of Computer Science, Boston, 2003.

[21] L. Lovász and S. Vempala. Hit-and-run from a corner. *SIAM J. Computing*, 35(4), 985–1005, 2006.

[22] M. Mihail. Conductance and convergence of Markov chains: a combinatorial treatment of expanders. *Proceedings of the 30th IEEE Symposium on Foundations of Computer Science*, 526–531, 1989.

[23] M. Mitzenmacher and E. Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, Cambridge, UK, 2005.

[24] J.D. Pinter, editor, Global optimization: scientific and engineering case studies, Springer, 2006.

[25] H.E. Romeijn and R.L. Smith. Simulated annealing for constrained global optimization. *Journal of Global Optimization*, 5:101–126, 1994.

[26] A. Sinclair. Improved bounds for mixing rates of Markov chains and multicommodity flow. *Combinatorics, Probability & Computing*, 1:351–370, 1992.

[27] A. Sinclair. Algorithms for random generation and counting. Birkhauser, Boston, 1993.

[28] J. C. Smith and S. H. Jacobson. An analysis of the Alias method for discrete random variable generation. *Informs Journal on Computing*, 17(3):321–327, 2005.

[29] R.L. Smith. Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, 32:1296–1308, 1984.

[30] D. Stefankovic, S. Vempala, and E. Vigoda. Adaptive simulated annealing: a near-optimal connection between sampling and counting. Available at http://arxiv.org/abs/cs/0612058, 2006.

[31] S. Vempala. Geometric random walks: A Survey. 52nd MSRI volume on Combinatorial and Computational Geometry, 2005.

[32] A. J. Walker. New fast method for generating discrete random numbers with arbitrary frequency distributions. *Electronics Letters*, 10:127-128, 1974.

[33] A. J. Walker. An efficient method for generating discrete random variable with general distributions. *ACM Trans. Math. Software*, 3:253-256, 1977.

[34] Z.B. Zabinsky, R.L. Smith, J.F. McDonald, H.E. Romeijn and D.E. Kaufman. Improving hit-and-run for global optimization. *Journal of Global Optimization*, 3:171–192, 1993.

[35] Z.B. Zabinsky, Stochastic adaptive search for global optimization, Springer, 2003.