Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Billiard walk - a new sampling algorithm for control and optimization

B. T. Polyak * **E. N. Gryazina** **

* *Institute for Control Sciences RAS and Moscow Institute of Physics and Technology, Moscow, Russia (e-mail: boris@ipu.ru)*
** *Institute for Control Science, Russian Academy of Sciences, Moscow, Russia (e-mail: gryazina@gmail.com)*

Abstract: Hit-and-Run is known to be one of the best versions of Markov Chain Monte Carlo sampler. Nevertheless, in practice the number of iterations required to achieve uniformly distributed samples is rather high. We propose new random walk algorithm based on billiard trajectories and prove its asymptotic uniformity. Numerical experiments demonstrate much faster convergence to uniform distribution for Billiard Walk algorithm compared to Hit-and-Run. We discuss a class of global optimization problems that can be efficiently solved with Monte Carlo sampler.

Keywords: Randomized methods, Hit-and-Run, global optimization, linear systems, stabilization.

1. INTRODUCTION

Generating points uniformly distributed in an arbitrary bounded region $Q \subset \mathbb{R}^n$ (sampling) finds applications in many computational problems (Tempo et al. [2004], Rubinstein and Kroese [2008]). Straightforward sampling techniques are usually based on one of three approaches: rejection, transformation and composition. Rejection implies enclosure of Q within the region with available uniform sampler B (usually a box or a ball). At the next step, samples that do not belong to Q are rejected. Transformation fits the case when we can map region B onto Q via smooth deterministic function with constant Jacobian. Composition works well when Q can be partitioned for finite number of sets that can be efficiently sampled.

Other sampling procedures use modern versions of Monte Carlo technique, based on Markov Chain Monte Carlo (MCMC) approach (Gilks et al. [1996], Diaconis [2009]). *Hit-and-Run* (HR) is known as one of the most famous and effective algorithms of MCMC type, it is originally proposed by Turchin [1971] and independently by Smith [1984]. We used to apply HR to various control and optimization problems: Polyak and Gryazina [2008, 2011]. Unfortunately, even for simple bad-shaped regions, such as level sets of ill-posed functions, HR techniques fail or turn out computationally inefficient. HR algorithm originated numerous extensions. For instance, Barrier Monte Carlo method (Polyak and Gryazina [2010]) exploits the approach developed for interior-point methods of convex optimization (Nesterov and Nemirovsky [1994]). It generates random points that are preferable in comparison with standard Hit-and-Run. But the complexity of each iteration in general is high enough (the calculation of $\left(\nabla^2 F(x)\right)^{-1/2}$, where $F(x)$ is a barrier function for Q, is needed). Moreover such approach can not accelerate convergence for sets like simplices.

We propose a new random walk algorithm motivated by physical phenomena of a gas filling uniformly a vessel. A gas particle moves with constant speed reflecting from a boundary of the vessel (the angle of incidence equals the angle of reflection) and colliding with other particles. The mean free path can be simulated as $\ell \sim -\log\eta$, η being uniform random in $[0, 1]$. In our simplified model we assume that after collision the direction changes as uniform random on the unit sphere while speed remains the same. Thus we combine ideas of Hit-and-Run technique with use of billiard trajectories. Traditional mathematical billiards theory (Tabachnikov [1995], Sinai [1970]) addresses the behavior of one particular billiard trajectory, its ergodic properties and the conditions for existence of periodic orbits. We extend billiard trajectories with random change of directions and use it to sample for interior of Q.

Besides, we distinguish a class of global optimization problems that can be efficiently solved with Monte Carlo approach. In particular, search for a most distant point from a given one in the polytope.

The paper is organized as follows. In Section 2 we present novel sampling algorithm *Billiard Walk* (BW) and prove that it produces asymptotically uniformly distributed samples in Q. Simulation of BW for particular test domains is presented in Section 3. Much attention is devoted to ability of BW to escape from the corner in comparison to HR. In Section 4 we briefly discuss possible applications of the algorithm and propose some optimistic estimates on the probability to obtain suboptimal solution via multi-start technique. Section 5 contains conclusive remarks.

⋆ The study was supported by Laboratory of Structural Methods of Data Analysis in Predictive Modeling in Moscow Institute of Physics and Technology (“mega-grant” of Russian Government) and by RFFI grant 13-07-12111

Copyright © 2014 IFAC

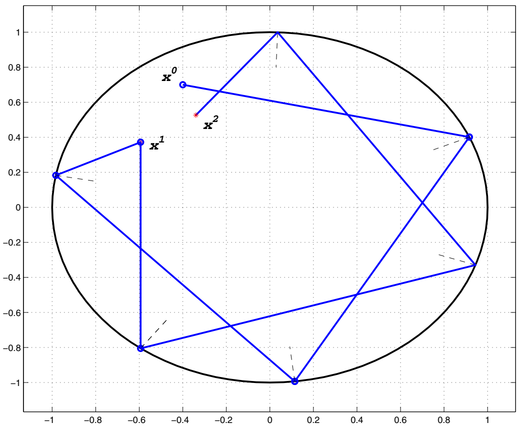

Fig. 1. The scheme of the Billiard Walk algorithm.

2. ALGORITHM

Let $Q \subset \mathbb{R}^n$ be an open, bounded and connected region and a point $x^0 \in Q$. Our aim is to generate asymptotically uniform samples $x^i \in Q$, $i = 1, \dots, N$.

The brief description of Hit-and-Run algorithm is as follows. At every step HR generates a random direction uniformly over the unit sphere and chooses next point uniformly from the segment of the line in given direction in Q.

The new BW algorithm generates a random direction uniformly as Hit-and-Run. But the next point is chosen as the end of the billiard trajectory of length ℓ. The scheme of the method is given in Fig. 1 while the precise routine is as follows.

Billiard Walk algorithm.

1. Starting point $x^0 \in \text{Int}\, Q$ is given; $i = 0$, $x = x^0$.
2. Generate the length of the trajectory $\ell = -\tau \log \eta$, η being uniform random in $[0, 1]$, τ is a specified parameter of the algorithm.
3. Pick random direction $d \in \mathbb{R}^n$ uniformly distributed on the unit sphere (i.e., $d^i = \xi/\|\xi\|$, where ξ is a standard Gaussian vector in n dimensions. Construct a billiard trajectory starting at x^i with initial direction $d = d^i$. When the trajectory meets a boundary with internal normal s, $||s|| = 1$, the direction is changed as
$$d \to d - 2(d, s)s.$$
4. Calculate the end of the trajectory of length ℓ. If a point with nonsmooth boundary is met or the number of reflections exceeds R go to step 2.
5. $i = i + 1$, take the end point as x^{i+1} and go to step 2.

We prove asymptotical uniformity of the samples produced by BW for convex and nonconvex cases separately. The requirements for Q are different for these two cases, while the sampling algorithm remains the same. Consider the Markov Chain induced by the BW algorithm $x^0, x^1, \dots$. For an arbitrary measurable set $A \subseteq Q$, denote by $\mathbf{P}(A|x)$ the probability of obtaining $x^{i+1} \in A$ for $x^i = x$ by the BW algorithm. Then $\mathbf{P}_N(A|x)$ is the probability to get $x^{i+N} \in A$ for $x^i = x$. We also denote by $p(y|x)$ the probability density function for $\mathbf{P}(A|x)$, i.e. $\mathbf{P}(A|x) = \int_A p(y|x)dy$.

Theorem 1. Assume Q is an open bounded convex set in $\mathbb{R}^n$, the boundary of Q is piecewise smooth. Then the distribution of points x^i sampled by the BW algorithm tends to the uniform one over Q, i.e.

$$\lim_{N\to\infty} \mathbf{P}_N(A|x) = \lambda(A)$$

for any measurable $A \subseteq Q$, $\lambda(A) = \text{Vol}(A)/\text{Vol}(Q)$ and any starting point x.

Proof. First, the algorithm is well-defined: at step 4 with zero probability the algorithm sticks at a point with nonsmooth boundary. On the other hand ℓ and d are chosen such that with positive probability x^{i+1} is obtained by less than R reflections.

In view of Theorem 2 in Smith [1984] based on the asymptotic properties of Markov Chains, the two assumptions on $p(y|x)$ imply that the uniform distribution over Q is a unique stationary distribution, and it is achieved for any starting point $x \in Q$. The first assumption requires the existence of $p(y|x)$ and its symmetry; the second assumption claims its positivity $p(y|x) > 0$ for all $x, y \in Q$.

The existence of a probability density means that for any $x, y \in Q$, the transition probability from x to a small neighborhood δy of y is proportional to the volume of δy. Among the trajectories proceeding from x to δy, there are a conic bundles of trajectories with no reflections and with $1, 2, \dots, R$ reflections. These bundles of trajectories are cones with small spatial angle $\delta\theta$. The area of reflection with a smooth boundary can be approximated as plain region. Then a reflection does not change the geometry of the bundle and the reasonings for these bundles remain the same as for the bundle of trajectories with no reflections. $\mathbf{P}(\delta y|x) \sim \mathbf{P}(\delta\theta)\mathbf{P}(\delta\ell)$, where $\mathbf{P}(\delta\theta) \sim S$ is the probability of choosing the spatial angle (proportional to the volume of the base of the cone) and $\mathbf{P}(\delta\ell) \sim \delta\ell$ is the probability of choosing a certain trajectory length $\ell \in \delta\ell$. Thus $\mathbf{P}(\delta y|x) \sim \text{vol}(\delta y)$ and $p(y|x)$ exists for all $x, y \in Q$.

For convex bodies, the positivity of $p(y|x)$ is obvious, all the points are reachable by the trajectory with no reflections. The symmetry of the probability density function follows from the uniformity of the distribution of the directions and reversibility of a billiard trajectory due to the reflection law: the angle of incidence equals the angle of reflection. Thus all the assumptions on $p(y|x)$ are satisfied and the distribution of points x^i generated by the BW algorithm tends to uniform distribution on Q. $\square$

Theorem 2. Assume Q is connected, bounded and open set, the boundary of Q is piecewise smooth and for all $x, y \in Q$ there exists a piecewise-linear path such that it connects x and y, lies inside Q and has no more than $B+1$ linear parts. Then the distribution of points x^i generated by the BW algorithm tends to the uniform distribution on Q in the same sense as in Theorem 1.

Proof. Again, the algorithm is well defined: with probability one $x^{i+1} \neq x^i$ is found for arbitrary $x^i \in Q$.

All the constraints on Q are important. Connectedness guarantees that starting from any point, we can reach a measurable neighborhood of any other point of Q. Boundedness is necessary to define the uniform distribution on Q and to avoid the trajectories going to infinity. Openness allows us to connect any two points with a tube of nonzero measure. Thus, there exists a piecewise linear trajectory connecting two arbitrary points.

Consider $p_N(y|x)$, the probability density function of $\mathbf{P}_N(A|x)$. The inequality $p_N(y|x) > 0$ holds for all integer $N > B$. The equality $p(x^{i+1}|x^i) = p(x^i|x^{i+1})$ (reversibility) holds for every pair of consecutive points due to the reflection law: the angle of incidence is equal to the angle of reflection. Therefore, $p_N(y|x) = p_N(x|y)$.

The subsequences $x^i, x^{N+i}, x^{2N+i}, \ldots,\ N > B$ have asymptotically uniform distribution, thus the distribution of points x^i generated by the BW algorithm tends to the uniform distribution on Q. $\square$

There exist plenty of nonconvex domains that satisfy the conditions of Theorem 2. For instance, an estimate of B for the toroid is given in Subsection 4.8. Note that the constant B characterizes the geometry of Q.

2.1 Choice of τ and R.

We need to specify parameter τ to run the algorithm. The value of τ strongly influences the behavior of the method. When τ is small enough BW becomes slower that HR, it behaves as a ball walk with radius τ. Empirical observations show that fast convergence to uniform distribution is achieved for $\tau \approx \mathrm{diam}Q$, where $\mathrm{diam}Q$ is the diameter of the set Q.

We restrict the number of reflections by R for every trajectory (step 4 of the Algorithm). The goal is to avoid situations when the trajectory length remains less than ℓ after a large number of reflections (a typical example is addressed in Subsection 3.3). The choice of R is mostly focused on eliminating computationally hard trajectories. We usually take $R = 10n$ to make it dimension dependent.

2.2 Preliminary transformation of Q.

Some "ill-shaped" domains Q can be improved with linear transformation. For instance, a simple scaling transforms a stretched box $Q = \{x \in \mathbb{R}^n : |x_i| \le a_i, i = 1, \ldots, n\}$ with $\min a_i / \max a_i \ll 1$ into a cube. In general case for convex domains the linear mapping $d' = Td$ can be helpfull, where $T = \left(\nabla^2 F(x^*)\right)^{-1/2}$, $F(x)$ is a barrier function for Q. However sometimes none of transformations can improve the shape of the set, a simplex is known to be the worst-case example.

2.3 Boundary oracle and normals

Both algorithms HR and BW require a calculation of an intersection of a straight line (defined by the point x^k and the direction d of the trajectory) with the set Q. We call the procedure for a segment bounds computing $[\underline{t}, \bar{t}]$ *Boundary Oracle* (BO), where

$$\underline{t} = \max_{t<0}\{t : x^k + td \in \partial Q\}, \quad \bar{t} = \min_{t>0}\{t : x^k + td \in \partial Q\}$$

(here we suppose that Q is convex, otherwise the points of the first intersection of a straight line and ∂Q are taken). In most applications finding BO is an easy task. For instance, if Q is a polytope defined by m linear inequalities

$$Q = \{x \in R^n : (a^i, x) \le b_i,\ i = 1, \ldots, m\}$$

then $[\underline{t}, \bar{t}]$ can be written explicitly. Calculate $t_i = \frac{b_i - (a^i, x^k)}{(a^i, d)}, i = 1, \ldots, m$, and take

$$\underline{t} = \max_{i:\ t_i<0} t_i, \quad \bar{t} = \min_{i:\ t_i>0} t_i. \tag{1}$$

Numerous examples of BO for other sets Q can be found in Polyak and Gryazina [2008, 2011]. Up to our knowledge the first attempt to apply HR for uniform sampling in the interior of an LMI feasible domain is performed in Calafiore [2004]. Consider a typical set described by linear matrix inequalities with matrix variable:

$$Q = \{X \in \mathbb{S}^{n\times n} : \quad X \succ 0, \quad AX + XA^T \preceq 0\},$$

where $\mathbb{S}^{n\times n}$ is the space of symmetric $n \times n$ matrices, A is a stable matrix and $H \prec 0$. A random direction in $\mathbb{S}^{n\times n}$ is a matrix $D = D^T$, $||D||_F = 1$ uniformly distributed on the unit sphere in Frobenius norm. BO provides $L = \{t \in R : X^0 + tD \in Q\}$ and it can be found explicitly. Indeed, we reach the boundary of Q at such t that either matrix $X^0 + tD$ or matrix $AX^0 + X^0A^T - H + t\left(AD + DA^T\right)$ becomes singular and $L = (\underline{t}, \bar{t})$ with $\bar{t} = \min \lambda_i$, $\underline{t} = \max \mu_i$, λ_i are positive real eigenvalues and μ_i are negative real eigenvalues of matrix pencils $(X^0, -D)$ and $\left(AX^0 + X^0A^T - H, -(AD + DA^T)\right)$.

BW walks requires also calculation of normals s for boundary points. In most applications it is not hard. For instance, for a polytope $s = a_i$, i being an index where maximum or minimum in (1) is achieved. For boundary point X^0 of linear matrix inequality $X \succeq 0$ the normal is

$$S = ee^T,$$

e being the eigenvector corresponding to the zero eigenvalue of X^0 (X^0 is singular since it is a boundary of the domain described by LMI). For matrix case the direction is changed as $D \to D - 2\ \mathrm{trace}(DS)S$, trace serves as an inner product.

3. TEST SETS AND SIMULATION

Our goal in test examples below is to compare HR and BW. We estimate the number of iterations to escape from the corner, demonstrate strong serial correlation in samples, use parametric partition of Q and compare the number of samples in every part with the theoretical number for uniform distribution. To make final conclusions on comparison we keep in mind that every sample of BW is computationally harder than of HR. We characterize the computational complexity by the number of calls to the BO and compare the outcomes of HR and BW obtained from the same number of BO calls (the number of samples is different in this case). Every HR sample needs two BO, for a BW sample number of BP calls equals to the number of reflections.

3.1 Plane angle and polyhedral cone

Let $Q \subset R^2$ be the plane angle $\alpha < \pi$. We say that a billiard trajectory quits the corner if it goes to infinity, for HR it means to choose the direction such that BO gives the unbounded segment. Then billiard trajectory independently of initial point and initial direction quits Q after no more than $N^* = \lceil \pi/\alpha \rceil$ reflections, here $\lceil a \rceil$ stands for smallest integer $\geq a$. The hint for the proof: if we reflect our angle N times around its side, billiard trajectory will become a straight line. It can not intersect any straight line (noncoinciding with itself) twice.

For HR we quit Q with probability $1 - (1 - \alpha/\pi)^N$ after N iterations. For $N = N^*$ large enough we quit Q with probability $1 - 1/e = 0.63$ after N^* iterations, while for BW we do it w.p.1.

It is of interest to estimate an average number of reflections (over random initial directions). Consider the triangle $Q = \{x \in \mathbb{R}^2 : |x_1| \leq \operatorname{atan}\frac{\alpha}{2}, x_2 \leq 1\}$. Let BW trajectories start at $[0; 0.1]$ and calculate the number of reflections until the trajectory reaches the line $x_2 = 1$. The results for 5000 runs and various α are given in Table 1. For HR we calculate the number of iterations until BO reaches the line $x_2 = 1$.

α	BW	HR
$\pi/2$	2.28 (0.87)	2.37 (1.74)
$\pi/4$	3.08 (1.3)	3.75 (2.98)
$\pi/10$	5.94 (2.93)	8.23 (7.1)
$\pi/50$	25.08 (14.46)	39.25 (34.54)

Table 1. The mean and standard deviation (in parentheses) for the number of BW reflections and the number of HR iterations required to quit the angle α.

For polyhedral cone there exists M independent of initial data such that billiard trajectory escapes from Q after no more than M reflections (see Tabachnikov [1995], Theorem 7.17). If M is large ($M > R$) BW algorithm sometimes returns to the initial point but it remains well defined w.p.1.

3.2 Orthant $Q = \{x \in R^n : x > 0\}$

It is easy to note that billiard trajectory independently of initial point and initial direction quits Q after no more than n reflections. Indeed, if d is direction of trajectory, $I = \{i : d_i < 0\}$ then at each reflection I decreases, and after $\leq n$ reflections $I = \emptyset$.

HR trajectory quits Q with probability $2^{-(n-1)}$ after a single iteration, thus it requires approximately 2^{n-1} iterations to quit Q with probability $1 - 1/e = 0.63$. Hence BW is much more effective than HR for this case.

3.3 Concave corner

In concave corners some directions produce a billiard trajectory with a large number of reflections. Consider a typical domain

$$Q = \{x \in R^2 : -x_1^4 < x_2 < x_1^4, \quad x_1 \geq 1\}. \tag{2}$$

Start a trajectory at the point $x^0 = [0.9; \varepsilon]$, ε being small enough, fix $\ell = 1$, $d = (-1; 0)$ and compute the number of reflections needed to calculate the end of the trajectory. Table 2 demonstrates the results.

As one can notice, the number of reflections increases dramatically as the second coordinate of x^0 tends to zero and even for $x_1^0 = 10^{-4}$ the trajectory becomes

ε	Number of reflections
1e-3	746
5e-4	1851
4e-4	2480
3e-4	3617
2e-4	6158
1.1e-4	13496
1.01e-4	>5e+6

Table 2. The number of reflections for the billiard trajectory for domain (2).

unrealizable. So to be on the safe side of situations like this we restrict the number of reflections in the algorithm.

Nevertheless these "bad" directions are rare. Fig. 2 depicts 200 points for domain (2), the average number of reflections per point is 5.2.

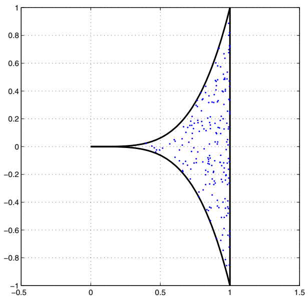

Fig. 2. BW samples for domain (2), $N_{BW} = 200$.

3.4 Cube

For the unit cube $Q = \{x \in \mathbb{R}^n : 0 < x < 1\}$ (inequality is understood component-wise) we can derive the next point of the BW algorithm explicitly.

At the current point x for the given ℓ and d calculate $k_i = \lfloor x_i + \ell d_i \rfloor$ ($\lfloor x \rfloor$ is maximal integer less than or equal to x) and walk to y:

$$y_i = \begin{cases} x_i + \ell d_i - k_i, & k_i \text{ is even} \\ 1 - (x_i + \ell d_i - k_i), & k_i \text{ is odd} \end{cases}, \quad i = 1, \dots, n.$$

In fact, there is no need to apply MCMC algorithms for random sampling in a cube, one can generate a vector of n independent uniform random variables over $[0, 1]$. Moreover, the shape of a cube is so nice that distribution of HR points converges to uniform fast enough. Nevertheless Hit-and-Run demonstrates strong serial correlation. We compare $r_k = \mathbf{E}||x^k - x^0||_\infty$ for $n = 50$ averaged over 500 runs, r_k tends to $r^* = \mathbf{E}_{x \in Q}||x - x^0||_\infty$. For the "warm-start" initial point $x^0 = [1/2, \dots, 1/2]^T$ we obtain that it requires 3 iterations for BW and around 30 iterations for HR to converge to r^*. For the "cold-start" initial point $x^0 = [1/n, \dots, 1/n]^T$ BW demonstrates convergence in 3-5 iterations while HR requires thousands iterations to converge to r^*.

Then we make χ^2 frequency test for 10000 HR points and 2148 BW point both requiring 20000 BO procedures in $\mathbb{R}^{10}$. We take 10 equal volume slabs in the ith coordinate direction for $i = 1, \dots, 10$, and make 10 χ^2 tests all together. HR fails all 10 χ^2 tests while BW fails just 2 out of 10 tests.

3.5 Simplex

The next test set is a standard n-dimensional simplex

$$Q = \{x_i > 0, \sum x_i = 1, i = 0, 1, \dots, n\}.$$

The simplex is a set with many corners and the geometry of the simplex can't be improved by any affine transformation. We know that for HR walk it takes a lot of iterations to get out of a corner, thus it is interesting to compare HR and BW.

For $n = 2$ samples look uniformly distributed for both algorithms. To judge about uniformity more rigorously in multidimensional case we consider the sequence of enclosed simplices $S_\alpha = \{x \in \mathbb{R}^{n+1} : x_i \geq \alpha, \sum x_i = 1\}$, $0 \leq \alpha \leq \frac{1}{n+1}$. For $\alpha = 0$, S_0 is the initial simplex Q, for $\alpha = \frac{1}{n+1}$, simplex S_α becomes empty. Let $\widehat{f}(\alpha)$ be the portion of points contained in S_α, and denote $f(\alpha) = \mathrm{vol}S_\alpha / \mathrm{vol}S_0 = (1-(n+1)\alpha)^n$. Fig. 3 shows $\widehat{f}(\alpha)$ for $n = 50$, $N = 300$, $x^0 = \{1/(n+1), \dots 1/(n+1)\}$. Red line corresponds to uniformly distributed points, black line describes the distribution for HR points and blue line for BW points. We conclude that for BW samples empirical values of $\widehat{f}(\alpha)$ are much closer to mean value $f(\alpha)$ than for HR samples.

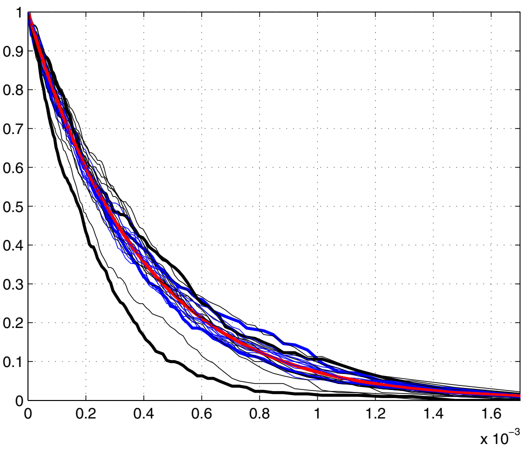

Fig. 3. Portion of points contained in S_α for uniformly distributed points (red), HR (black) and BW (blue). $n = 50$, 300 points. Horizontal line corresponds to parameter α.

3.6 Toroid

Both the HR and BW algorithms are applicable to nonconvex sets. Consider a toroid formed by an n-dimensional ball of radius r with its center rotating over a circle in the (x_1, x_2)-plane:

$$Q = \{x \in \mathbb{R}^n : ||x - c_x|| \leq r\}, \tag{3}$$

where $c_{xi} = \frac{x_i}{\sqrt{x_1^2 + x_2^2}}$, $i = 1, 2$, $c_{xi} = 0$, $i > 2$.

The conditions of Theorem 2 are satisfied with $B = \left\lceil \frac{\pi}{2 \arccos \frac{1-r}{1+r}} \right\rceil + 1$, i.e. for all $x, y \in Q$ there exists a piecewise-linear path such that it connects x and y, lies inside Q, and has no more than B linear parts. Fig. 4 demonstrates $N = 1000$ samples (projected onto (x_1, x_2)-plane) for the set (3) with $r = 1/3$ of dimension 10. HR points are plotted with black dots, BW points with blue ones.

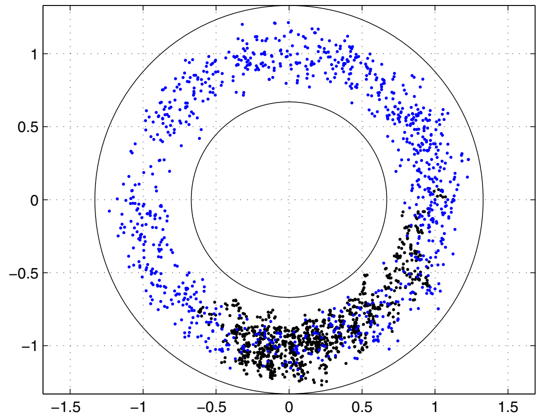

Fig. 4. (x_1, x_2)-projection for HR (black) and BW (blue) points for (3). $n = 10$, $N_{BW} = 500$, $N_{HR} = 1764$.

Note that angle distribution of BW points is close to uniform. HR points mostly remain in the neighborhood of the initial point. Radial distribution looks far from uniform both for HR and BW; this is visual effect (we have 2D projection of 10D points).

4. APPLICATIONS IN CONTROL AND OPTIMIZATION

Here we briefly mention possible applications of new version of random sampling: control problems and global optimization.

Control systems design discover many situations when sampling of systems with prescribed properties is of interest. Static output feedback, robust fixed order control as well as other design problems converted to Bilinear Matrix Inequalities (BMI) (VanAntwerp and Braatz [2000]), or recasted as concave optimization problems (Apkarian and Tuan [1999]), — all these problems admit efficient randomized solutions. Besides, numerous problems of robust stability analysis and design can be solved by generation of admissible uncertainties. The examples can be found in Polyak and Gryazina [2011].

In our previous works we developed cutting plane methods for convex optimization, exploiting Monte Carlo techniques. They are based on Hit-and-Run algorithm for generating samples in Q, and definitely all previous techniques can be strongly improved by replacing standard Hit-and-Run with Billiard Walk algorithm.

Consider a general global optimization problem

$$\min f(x), \quad x \in Q,$$

where $f(x)$ is a nonconvex function while Q is a convex domain. We apply multi-start framework containing two steps: generate uniform samples in Q and implement local descent procedure for every sample as a starting point. Every single starting point gives a local optimum.

Sampling in Q serves as diversification to overcome local optimality since various local optima have various basins of attraction and uniform samplings tends to capture mostly all of them. These approach is well tailored for concave $f(x)$ and polyhedral Q. The local search problems are linear programming ones:

$$\min(\nabla f(x_i), x) \quad x \in Q$$

and can be solved fast. There exist the huge literature on deterministic methods for concave optimization; see, e.g., Horst and Tuy [1996]; we conjecture that the proposed randomized methods could be good competitors to them for some classes of f.

General estimates for the number of function calculations (calls of oracle) are quite disappointing for global optimization. For $x \in \mathbb{R}^{10}$ and $f(x)$ being Lipschitz continuous on the box with $L = 2$ it requires no less that 10^{20} calls of oracle to guarantee 1% accuracy Nesterov [2004]. It is the worst case example where global optimum has very small basin of attraction.

In contrast, we propose a class of problems with a promising trade-off between the basins of attraction and the value difference for local and global optima. For instance, consider the search of the most distant point in a cube:

$$\max \ ||x - a||^2 \qquad (4)$$
$$-1 \le x_i \le 1, \quad i = 1, \dots, n.$$

It may look as a toy problem − the exact solution is straightforward − but we treat it as a representative problem of concave optimization. In general it is a hard combinatorial problem, every vertex is a local optimum and the number of vertices grows exponentially with dimension n.

For problem (4) we observe large basin of attraction for local optima close to global one. Thus we have an opportunity to obtain rather accurate suboptimal value with high probability. We take $N = 1000$ uniform initial samples and estimate the probability to obtain $\hat{f}$ such that

$$\frac{f_{\max} - \hat{f}}{f_{\max}} \le 0.01 \frac{f_{\max} - f_{\min}}{f_{\max}},$$

where $f_{\max} = \sum_{i=1}^{n} (1 + |a_i|)^2$, $f_{\min} = \sum_{i=1}^{n} (1 - |a_i|)^2$ are the largest and the smallest local solutions to (4). Table 3 shows the numerical results averaged over 100 random a.

n	Probability
20	0.9995
25	0.95
30	0.7

Table 3. The probability to guarantee 1% relative accuracy by multi-start algorithm with 1000 starting samples for problem (4).

The data show that problem (4) can be efficiently solved up to $n = 30$.

5. CONCLUSION

We introduce the new uniform sampling algorithm – Billiard Walk and provide its theoretical validation. Test simulations show that random points produced by BW are preferable in comparison with standard Hit-and-Run. Applications of BW to control problems and global optimization are briefly discussed. We demonstrate that for a certain class of problems suboptimal solution can be obtained with high probability via multi-start technique.

REFERENCES

P. Apkarian and H. Tuan. Concave programming in control theory. *Journal of Global Optimization*, 15:343–370, 1999.

G. Calafiore. Random walks for probabilistic robustness. In *IEEE Conference on Decision and Control*, volume 5, pages 5316–5321, 2004.

P. Diaconis. The markov chain monte carlo revolution. *Bull. of the AMS*, 46(2):175–205, 2009.

W. Gilks, S. Richardson, and D. Spiegelhalter. *Markov Chain Monte Carlo*. Chapmen and Hall, 1996.

R. Horst and H. Tuy. *Global Optimization: Deterministic Approach*. Springer, Berlin, 1996.

Y. Nesterov. *Introductory Lectures on Convex Optimization: a Basic Course*. Springer, 2004.

Y. Nesterov and A. Nemirovsky. *Interior Point Polynomial Methods in Convex Programming*. SIAM, Philadelphia, 1994.

B. Polyak and E. Gryazina. Hit-and-Run: New design technique for stabilization, robustness and optimization of linear systems. In *Proceedings of the IFAC World Congress*, pages 376–380, Seoul, South Korea, 2008.

B. Polyak and E. Gryazina. Markov Chain Monte Carlo method exploiting barrier functions with applications to control and optimization. In *IEEE Multi-Conference on Systems and Control*, pages 1553–1557, 2010.

B. Polyak and E. Gryazina. Randomized methods based on new Monte Carlo schemes for control and optimization. *Annals of Operational Research*, 189(1):343–356, 2011.

R.Y. Rubinstein and D.P. Kroese. *Simulation and the Monte Carlo Method*. Wiley, NJ, 2008.

Ya. G. Sinai. Dynamical systems with elastic reflections. *Russian Mathematical Surveys*, 25(2):137–189, 1970.

R.L. Smith. Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, 32(6):1296–1308, 1984.

S. Tabachnikov. *Geometry and Billiards*. RI: Amer. Math. Soc., 1995.

R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems*. Communications and Control Engineering Series. Springer-Verlag, London, 2004.

V. Turchin. On the computation of multidimensional integrals by the Monte Carlo method. *Theory of Probability and its Applications*, 16(4):720–724, 1971.

J. VanAntwerp and R. Braatz. A tutorial on linear and bylinear matrix inequalities. *Journal on Process Control*, 10(4):363–385, 2000.