Efficient sampling of constraint spaces in practice

Vissarion Fisikopoulos

joint with Apostolos Chalkis, Elias Tsigaridas

PGMODAYS 2024

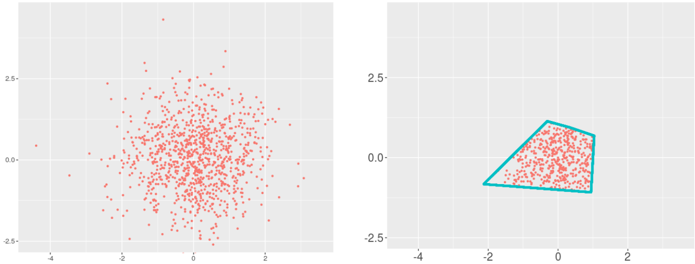
National and Kapodistrian UNIVERSITY OF ATHENS

GeomScale

(Truncated) distributions

- Multivariate probability distribution with density function $\pi(x)$
- Truncate π to a polytope $P := \{Ax \leq b\}$ we obtain p.d.f. π_P

$$\pi_P(x) = \frac{f(x)\pi(x)}{\int_P \pi(x)dx}, \quad f(x) = \begin{cases} 1, & \text{if } x \in P \\ 0, & \text{if } x \notin P \end{cases}$$



The support is the polytope P and π_P is the uniform ditribution over P.

In general the support is a convex body $K \subset \mathbb{R}^n$

Sampling from (truncated) distributions

Problem

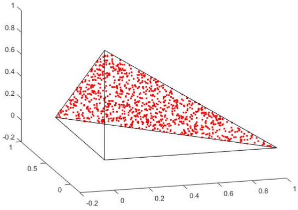
Sample (efficiently?) from a (truncated) distribution with density π_K ?

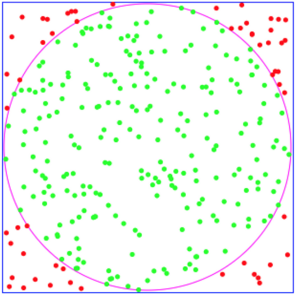
Interesting directions

- Algorithms and (efficient) implementations
- Complexity bounds (gaps between theoretical bounds and practical performance, provide (statistical guarantees))
- Take advantage of the geometry of support $K \subset \mathbb{R}^n$ (polytope, non-linear, e.g., spectrahedron, basic semi-algebraic set)
- Applications (Volume, integration, Bayesian inference, optimization, ...)

Simple cases and simplistic approaches

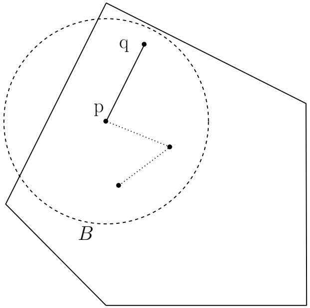
- Fundamental shapes (hypercube, hypershpere, simplex) admit efficient methods
- Acception/rejection sampling does not scale to high dimensions



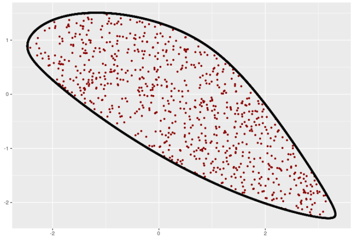


Geometric Random Walks

- A Geometric Random Walk starts at some interior point and at each step moves to a "neighboring" point, chosen according to some distribution depending only on the current point.
- Goal: analyse geometric random walks as Marcov Chains with some target distribution



Steps of a ball walk.



Uniform target distribution

Useful questions and terminology

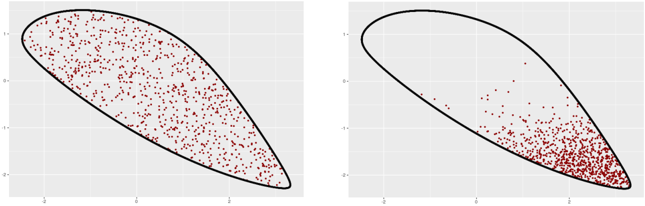
- Does the random walk converges asymptotically to the target distribution? (Correctness)
- How fast does it converge?
  (Equivalently) How many steps do we have to perform until we get a point that is ϵ-close to a point draw from the target distribution? (mixing time)
- Does the initial point of the walk affects the efficiency? (warm start)
- What is the cost per step of the random walk?
- Do we assume anything about K? (isotropic position, well rounded)

Target probability distributions

Definition

Let $\pi(\boldsymbol{x}) \propto e^{-f(\boldsymbol{x})}$, where $f : \mathbb{R}^d \to \mathbb{R}$ is a convex function.
$\pi(\boldsymbol{x})$ is called *log-concave (LC) probability density*.

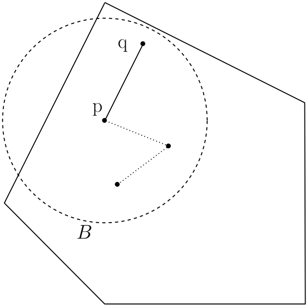
- Let $\pi(\boldsymbol{x})$ be restricted to **convex body** $K \subset \mathbb{R}^d$.
- Special cases: Uniform, Gaussian, Exponential/Boltzmann.



Ball walk

Ball Walk(K, p, δ, f): convex $K \subset \mathbb{R}^d$, $p \in P$, radius δ, $f : \mathbb{R}^d \to \mathbb{R}_+$

1. Pick a uniform random point x in $B(p, \delta)$.
2. **return** x with probability $\min\left\{1, \frac{f(x)}{f(p)}\right\}$;
   return p with the remaining probability.

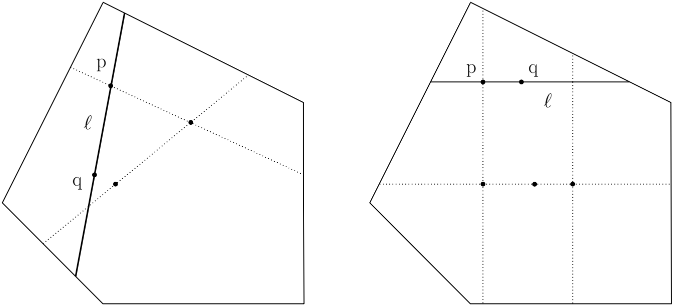


If the density is not restricted in K, then it is the **Metropolis-Hastings** *algorithm*

Hit-and-Run

Hit and Run(K, p, f): convex $K \subset \mathbb{R}^d$, point $p \in P$, $f : \mathbb{R}^d \to \mathbb{R}_+$

1. Pick uniformly a line ℓ through p.
2. **return** a random point on the chord $\ell \cap K$ chosen from the distribution $\pi_{\ell,f}$ restricted in $K \cap \ell$.

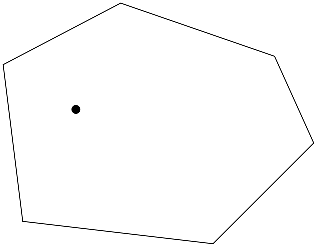


- **Q**: How do we compute $\ell \cap K$? Can we do it *exactly*?

Billiard walk - Uniform case

BW(K, p_i, τ, R) [Polyak'14]

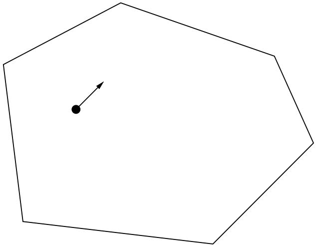
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0,1)$.
2. Pick a uniform direction v to define the trajectory. then the direction becomes $v \leftarrow v - 2\langle v, s\rangle$.
3. If the trajectory meets a boundary with internal normal s, $||s|| = 1$,
4. **return** the end of the trajectory as p_{i+1}.
   If the number of reflections exceeds R, then **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

BW(K, p_i, τ, R) [Polyak'14]

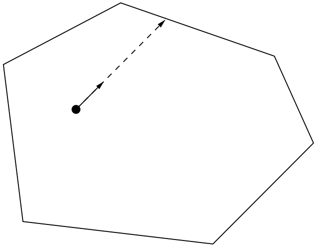
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0, 1)$.
2. Pick a uniform direction v to define the trajectory. then the direction becomes $v \leftarrow v - 2\langle v, s \rangle$.
3. If the trajectory meets a boundary with internal normal s, $||s|| = 1$,
4. **return** the end of the trajectory as p_{i+1}.
   If the number of reflections exceeds R, then **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

BW(K, p_i, τ, R) [Polyak'14]

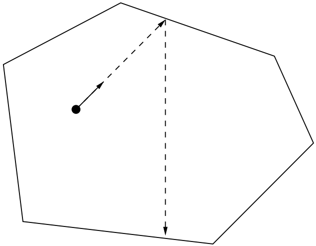
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0,1)$.
2. Pick a uniform direction v to define the trajectory. then the direction becomes $v \leftarrow v - 2\langle v, s \rangle$.
3. If the trajectory meets a boundary with internal normal $s,\ ||s|| = 1$,
4. **return** the end of the trajectory as p_{i+1}.
   If the number of reflections exceeds R, then **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

BW(K, p_i, τ, R) [Polyak'14]

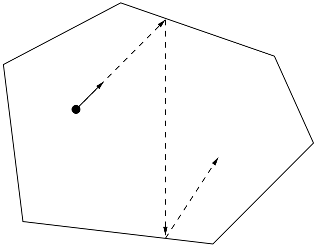
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0,1)$.
2. Pick a uniform direction v to define the trajectory. then the direction becomes $v \leftarrow v - 2\langle v, s\rangle$.
3. If the trajectory meets a boundary with internal normal $s,\ ||s|| = 1$,
4. **return** the end of the trajectory as p_{i+1}.
   If the number of reflections exceeds R, then **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

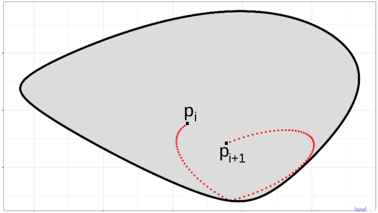
BW(K, p_i, τ, R) [Polyak'14]

1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0, 1)$.
2. Pick a uniform direction v to define the trajectory. then the direction becomes $v \leftarrow v - 2\langle v, s \rangle$.
3. If the trajectory meets a boundary with internal normal s, $||s|| = 1$,
4. **return** the end of the trajectory as p_{i+1}.
   If the number of reflections exceeds R, then **return** $p_{i+1} = p_i$.

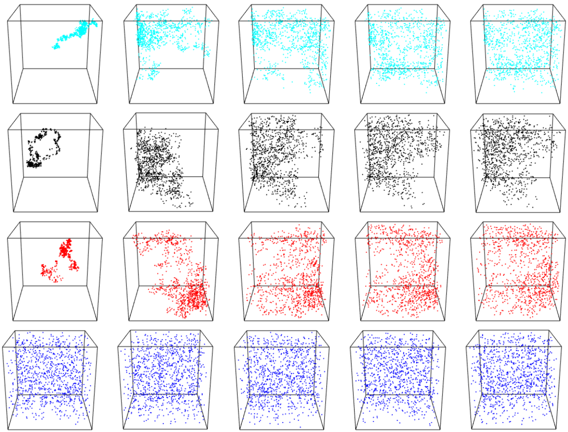


Hamiltonian Monte Carlo

- Similar to billiard walk but with non-linear trajectory
- Trajectory is defined by Hamiltonian dynamics simulated using a time-reversible and volume-preserving numerical integrator (typically the leapfrog integrator)
- Reflected: The trajectory stays inside K by using boundary reflections.
- Riemannian: Using the barrier of K the trajectory is always inside K.



Mixing time experiment (uniform case)



- Uniform sampling from the hypercube $[-1,1]^{200}$ and projection to $\mathbb{R}^3$.
- Rows: Ball Walk, Coordinate Directions Hit and Run, Random Directions Hit and Run, Billiard Walk.
- Columns: walk length, $\{1, 50, 100, 150, 200\}$

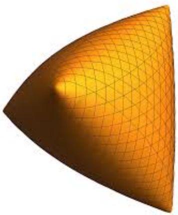
Complexity bounds

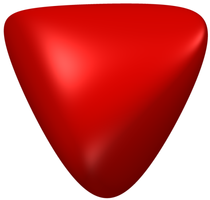
Year & Authors	Random walk	Mixing time*	Distribution
[Smith: 1986]	Hit-and-Run	$\widetilde{O}(d^3)$	any LC
[Berbee, Smith: 1987]	Coordinate Hit-and-Run	$\widetilde{O}(d^{10})$	any LC
[Lovasz,Simonovits'90]	Ball walk	$\widetilde{O}(d^3)$	any LC
[Kannan,Narayanan'12]	Dikin walk	$\widetilde{O}(d^2)$	uniform (H-polytope)
[Polyak,Dabbene'14]	**Billiard walk**	??	uniform
[Afshar,Domke'15]	**Reflective HMC**	??	any LC (polytopes)
[Lee,Vempala'16]	Geodesic walk	$O(md^{3/4})$	uniform (H-polytope)
[Lee,Vempala'17]	Remannian HMC	$\widetilde{O}(md^{2/3})$	any LC (H-polytopes)
[Chen,Dwivedi,Wainwright,Yu'19]	John walk	$\widetilde{O}(d^{5/2})$	uniform (H-polytope)
[Chen,Dwivedi,Wainwright,Yu'19]	Vaidya walk	$O(m^{1/2}d^{3/2})$	uniform (H-polytope)

- Cost per sample: *cost per step* $\times$ *mixing time (#steps)*.
- The *cost per step* depends on the convex body.
- Hit-and-Run (HR): widely used & well studied.
- Coordinate Hit-and-Run (CDHR): seems more efficient than HR in practice.
- Most existing software uses either CDHR or HR (H-polytopes).

Convex bodies

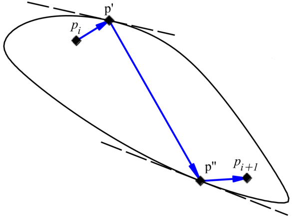
- H-polytopes: system of linear inequalities
- V-polytopes: convex hull of point sets
- Minkowski sums of polytopes
- Spectrahedra: feasible sets of linear matrix inequalities





Geometric and algebraic oracles

- Membership oracle (Ball walk)
- Boundary (intersection) oracle (HnR)
- Reflection oracle (Billiard, ReHMC)
- Optimization oracle (Minkowski sums, Secondary polytopes)



Volume Computation

Computing the exact volume of P,

- is #P-hard for all the representations [DyerFrieze'88]
- is open if both H- and V- representations available
- is APX-hard (oracle model) [Elekes'86]
- Randomized approximation algorithms

Theorem

[Dyer, Frieze, Kannan'91] For any convex body P and any $0 \leq \epsilon,\ \delta \leq 1$, there is a randomized algorithm which computes an estimate V s.t. with probability $1 - \delta$ we have $(1 - \epsilon)vol(P) \leq V \leq (1 + \epsilon)vol(P)$, and the number of oracle calls is $poly(d, 1/\epsilon, log(1/\delta))$.

Multiphase Monte Carlo

Let a sequence of functions $\{f_0, \ldots, f_m\}$, $f_i : \mathbb{R}^d \to \mathbb{R}$. Then,

$$\mathrm{vol}(P) = \int_P dx = \int_P f_m(x)dx \; \frac{\int_P f_{m-1}(x)dx}{\int_P f_m(x)dx} \cdots \frac{\int_P f_0(x)dx}{\int_P f_1(x)dx} \; \frac{\int_P dx}{\int_P f_0(x)dx}$$

Then select f_i s.t.,

- The number of phases, m, is as small as possible.
- Each integral ratio can be efficiently estimated by sampling from $\pi \propto f_i$ restricted to P (using geometric random walks).
- There is a closed formula for $\int_P f_m(x)dx$.

complexity = #phases × #points per phase × cost per point

State-of-the-art

Authors-Year	Complexity (oracle calls)	f_i	random walk
[Dyer, Frieze, Kannan'91]	$\widetilde{O}(d^{23})$	Indicator function of a ball	grid walk
[Kannan, Lovasz, Simonovits'97]	$\widetilde{O}(d^{5})$	Indicator function of a ball	ball walk
[Lovasz, Vempala'03]	$\widetilde{O}(d^{4})$	Exponential	hit-and-run
[Cousins, Vempala'15]	$\widetilde{O}(d^{3})$	Spherical Gaussians	ball walk

- Cannot be implemented as they are due to large constants in the complexity and pessimistic theoretical bounds.

Practical algorithms: Follow the theory but make practical adjustments (experimental)

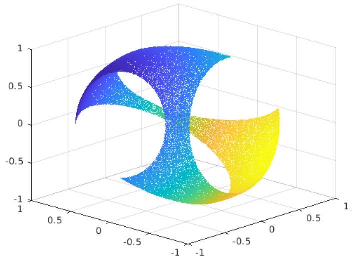
- [Emiris, Fisikopoulos'14] Sequence of balls + coordinate hit-and-run.
- [Cousins, Vempala'16] Spherical Gaussians + hit-and-run
- [Chevallier, Cazals, Fearnhead'22] Piecewise deterministic Markov processes
- [Emiris, Chalkis, F:'23] Cooling of balls + billiard walk

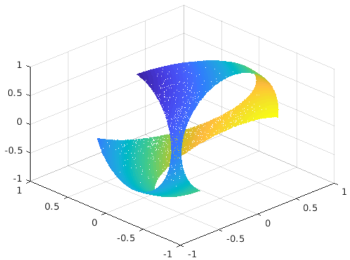
latter: most efficient, scales to 1000s dims, supports: V-polytopes, Zonotopes, extends to spectahedra [Chalkis, F, Tsigaridas]

Applications in finance

Portfolio analysis

- The set of portfolios (investments in a collection of stocks) is a simplex.
- Constraints on investments yield a general polytope.
- Portfolios with same volatility (the degree of variation of a trading price series over time) lie on an ellipsoid.



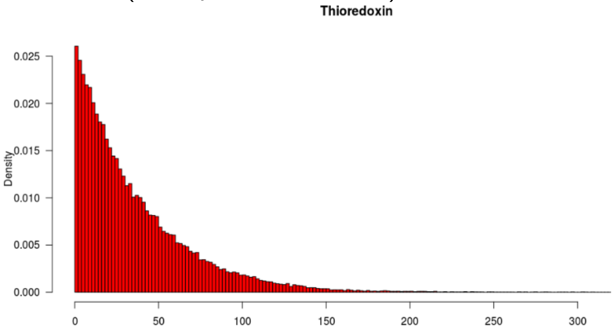


Randomized geometric tools for anomaly detection in stock markets [Bachelard,Chalkis,F,Tsigaridas'23]

Applications in structural biology

[Chalkis,F, Tsigaridas, Zafeiropoulos]

- Metabolic networks model the reactions of metabolites in an organim or system.
- Each reaction has a flow or rate called flux.
- The set of states of the network where fluxes are in balance (rate of production = rate of consumption) is a convex polytope.
- Sampling from polytope yield probability densities for reaction fluxes (example: thioredoxin)



GeomScale org

https://geomscale.github.io

C++ volume approximation & sampling from convex bodies

Python interface with extra tools for metabolic network analysis (FBA, copulas, visualization)

R interface with extra tools for finance (portfolio analysis)

NumFOCUS Affiliated Project.

- Support from an open community.
- More than 15 000 lines of code.

Thank you!