BAIR
BERKELEY ARTIFICIAL INTELLIGENCE RESEARCH

Berkeley | EECS
Electrical Engineering and Computer Sciences

Theoretical insights for MCMC algorithms

Raaz Dwivedi, UC Berkeley

Talk at BIDS Statistics and Machine Learning Discussion Group, Mar 18

Yuansi Chen

Joint work with

Martin Wainwright

Bin Yu

Random Sampling

- We consider the problem of drawing random samples from a given density (known up-to proportionality)

$$X_1, X_2, \ldots, X_m \sim \pi$$

Sampling: A fundamental task

Sampling → Integration

$$\mathbb{E}\left[g(X)\right] = \int g(x)\pi(x)dx \approx \frac{1}{m}\sum_{i=1}^{m} g(X_i)$$

$$X_1, X_2, \ldots, X_m \sim \pi$$

Monte Carlo Approximations

Rare event simulations

Bayesian inference

Sampling → Optimization

$$\min_x g(x) \quad \Longleftrightarrow \quad \text{sample from } e^{-g(x)/T}$$

Zeroth order optimization

Escaping saddle points

Simulated annealing

Popular recipes for sampling

- Rejection sampling
- Gibbs sampling
- Markov Chain Monte Carlo (MCMC) methods

Popular recipes for sampling

In high dimensions
too many rejections

- Rejection sampling

Requires tractable
conditional distributions

- Gibbs sampling

- Markov Chain Monte Carlo (MCMC) methods

Require
knowledge of density up to
proportionality

MCMC 101

- *Design of Markov Chain*
  - Starting point: random or deterministic
  - Transition distribution: given a point, how to make a transition
  - Target distribution
- *Mixing Time*
  - Number of steps steps after which the distribution of the chain is **close to** the target distribution

MCMC 102:
Metropolis-Hastings Recipe

- Typical two step design for $\pi(x) \propto e^{-f(x)}$
- **Proposal step**:

$$z \sim \mathbb{P}(x, \cdot)$$

- **Accept-reject step**: Accept z with probability

$$\min\left\{1, \frac{e^{-f(z)}}{e^{-f(x)}} \frac{P(z \to x)}{P(x \to z)}\right\}$$

Also called ``Metropolis-Hastings step/correction".

Our work

- Typical two step design for $\pi(x) \propto e^{-f(x)}$
- **Proposal step:**

$$z \sim \mathbb{P}(x, \cdot)$$

How to select the proposal distribution?

- **Accept-reject step**: Accept z with probability

$$\min\left\{1, \frac{e^{-f(z)}}{e^{-f(x)}} \frac{P(z \to x)}{P(x \to z)}\right\}$$

Should I do this step or not?

Outline

- Power of accept-reject (Langevin algorithms)
- Power of gradients for sampling

Three popular algorithms

Algorithm	Proposal Step
Random Walk	$z = x + \sqrt{2h}\xi$
Langevin algorithm	$z = x - h\nabla f(x) + \sqrt{2h}\xi$
Hamiltonian Monte Carlo	Multi step version of Langevin algorithm

$\xi \sim \mathcal{N}(0, \mathbb{I}_d)$

Three popular algorithms

Algorithm	Proposal Step
Random Walk	$z = x + \sqrt{2h}\xi$
Langevin algorithm	$z = x - h\nabla f(x) + \sqrt{2h}\xi$
Hamiltonian Monte Carlo	Multi step version of Langevin algorithm

$\xi \sim \mathcal{N}(0, \mathbb{I}_d)$

From optimization to sampling

Optimization

- Find the global minimum (or a stationary point)

$$\min_{x \in \mathbb{R}^d} f(x)$$

- Gradient descent:

$$x_{k+1} = x_k - h\nabla f(x_k)$$

- Stochastic Gradient Algorithm:

$$X_{k+1} = X_k - h\nabla f(X_k) + h\xi_{k+1}$$

From optimization to sampling

Optimization

- Find the global minimum (or a stationary point)

$$\min_{x\in\mathbb{R}^d} f(x)$$

- Gradient descent:

$$x_{k+1} = x_k - h\nabla f(x_k)$$

- Stochastic Gradient Algorithm:

$$X_{k+1} = X_k - h\nabla f(X_k) + h\xi_{k+1}$$

Sampling

- Draw samples from the density

$$\pi(x) \propto e^{-f(x)}$$

- Unadjusted Langevin algorithm (ULA):

$$X_{k+1} = X_k - h\nabla f(X_k) + \sqrt{2h}\xi_{k+1}$$

$$\xi_k \overset{i.i.d.}{\sim} \mathcal{N}(0, I_{d\times d})$$

[Parisi 1981, Grenander & Miller 1994, Roberts & Tweedie 1996]

Langevin algorithms: Origins?

- Classical Langevin stochastic differential equation

$$dX_t = -\nabla f(X_t)dt + \sqrt{2}dB_t \quad \text{where } B_t \text{ is standard Brownian motion}$$

Langevin algorithms: Origins?

- Classical Langevin stochastic differential equation

$$dX_t = -\nabla f(X_t)dt + \sqrt{2}dB_t \quad \text{where } B_t \text{ is standard Brownian motion}$$

- It has the right limiting distribution $\pi(x) \propto e^{-f(x)}$

$$\|P(X_t) - \pi\|_{\mathrm{TV}} \xrightarrow{t\uparrow\infty} 0$$

Langevin algorithms: Origins?

- Classical Langevin stochastic differential equation

$$dX_t = -\nabla f(X_t)dt + \sqrt{2}dB_t \quad \text{where } B_t \text{ is standard Brownian motion}$$

- It has the right limiting distribution $\pi(x) \propto e^{-f(x)}$

$$\|P(X_t) - \pi\|_{\mathrm{TV}} \xrightarrow{t\uparrow\infty} 0$$

- ULA updates: forward discretization of the Langevin SDE

$$X_{k+1} - X_k = -h\nabla f(X_k) + \sqrt{2h}\xi_{k+1}$$

(no accept-reject step)

Langevin algorithms: Origins?

- Classical Langevin stochastic differential equation

$$dX_t = -\nabla f(X_t)dt + \sqrt{2}dB_t \quad \text{where } B_t \text{ is standard Brownian motion}$$

- It has the right limiting distribution $\pi(x) \propto e^{-f(x)}$

$$\|P(X_t) - \pi\|_{\mathrm{TV}} \xrightarrow{t\uparrow\infty} 0$$

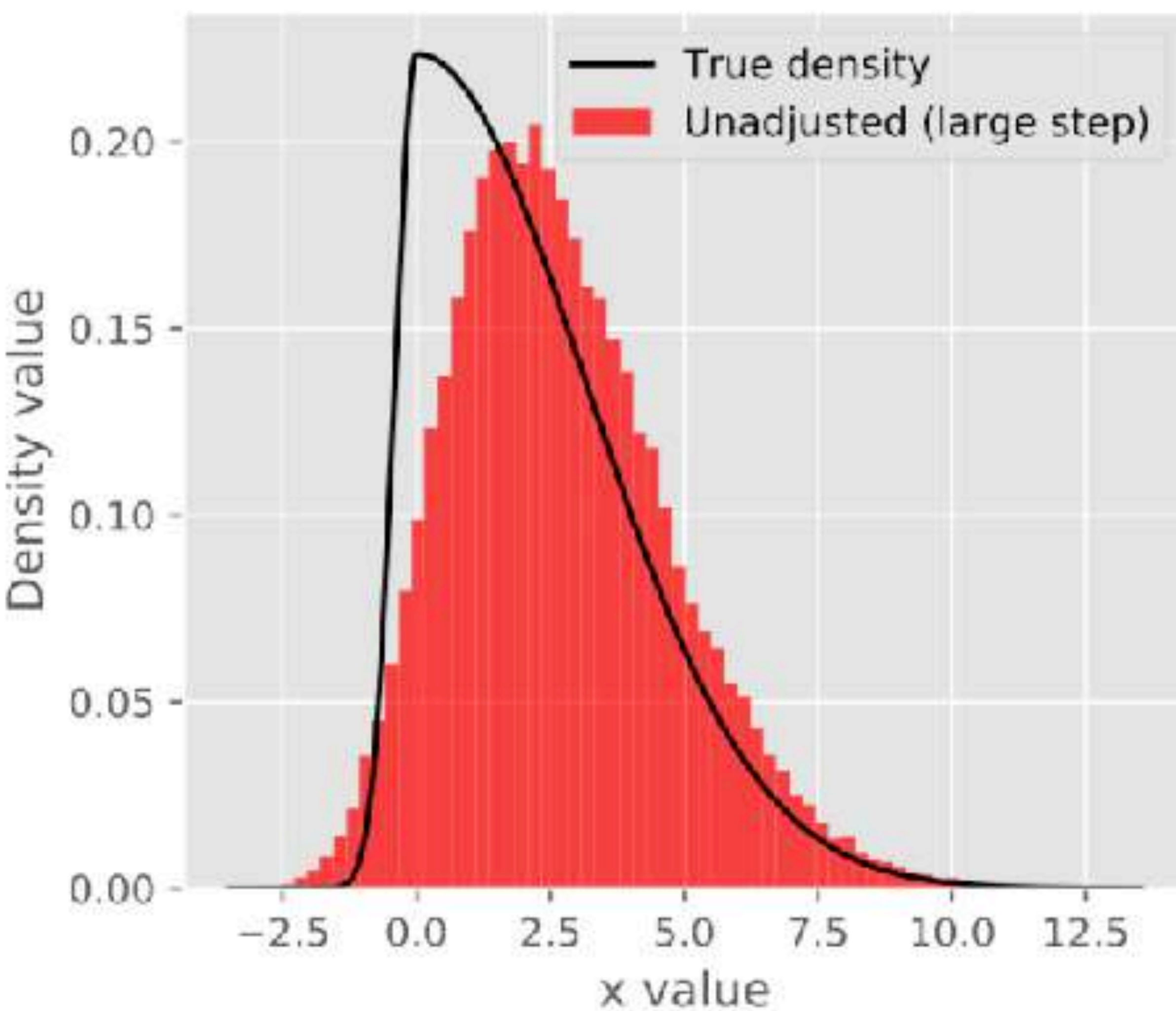
- ULA updates: forward discretization of the Langevin SDE

$$X_{k+1} - X_k = -h\nabla f(X_k) + \sqrt{2h}\xi_{k+1}$$

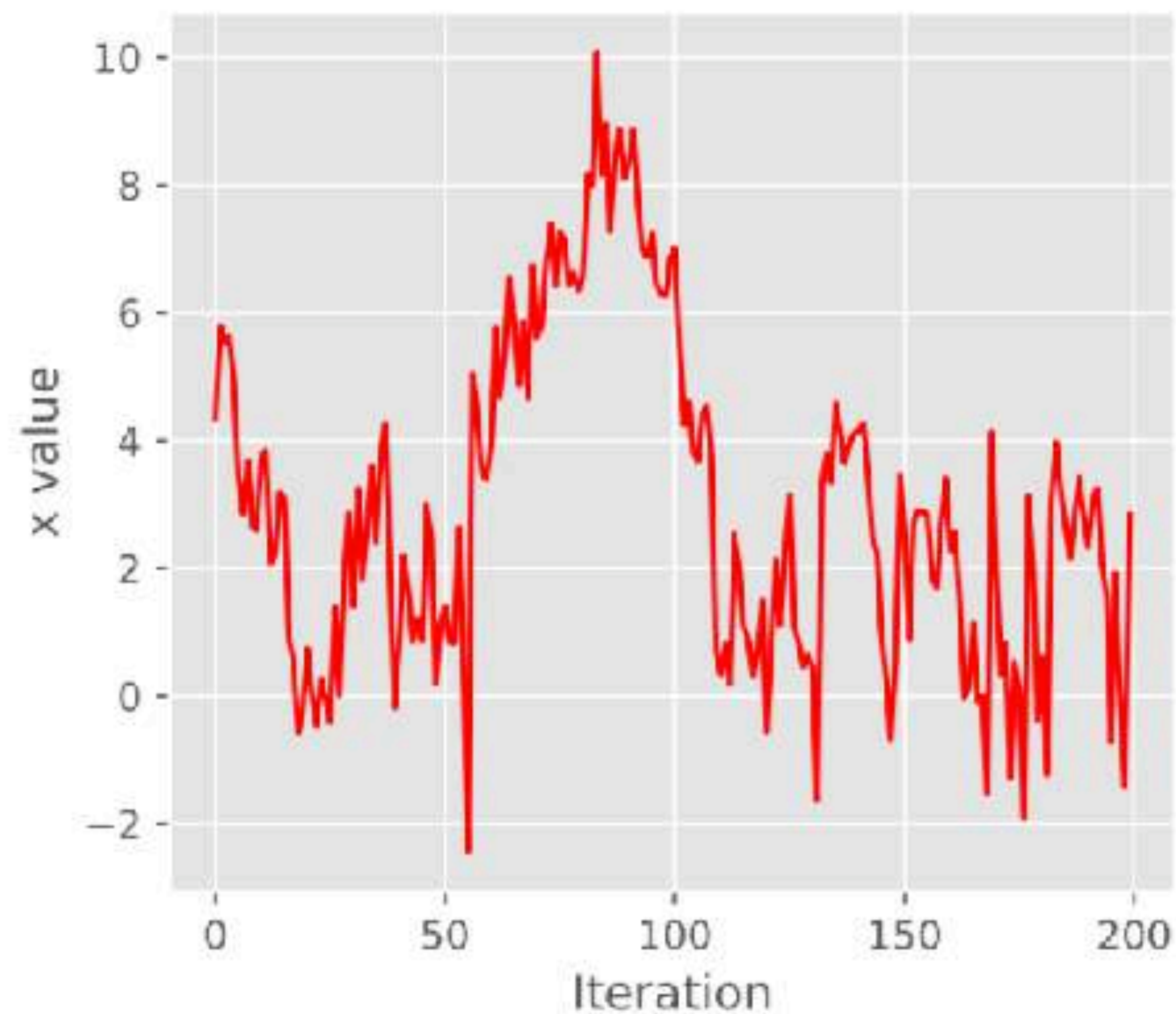
(no accept-reject step)

Effect of step size h?

ULA performance:
Large step = large bias + fast mixing

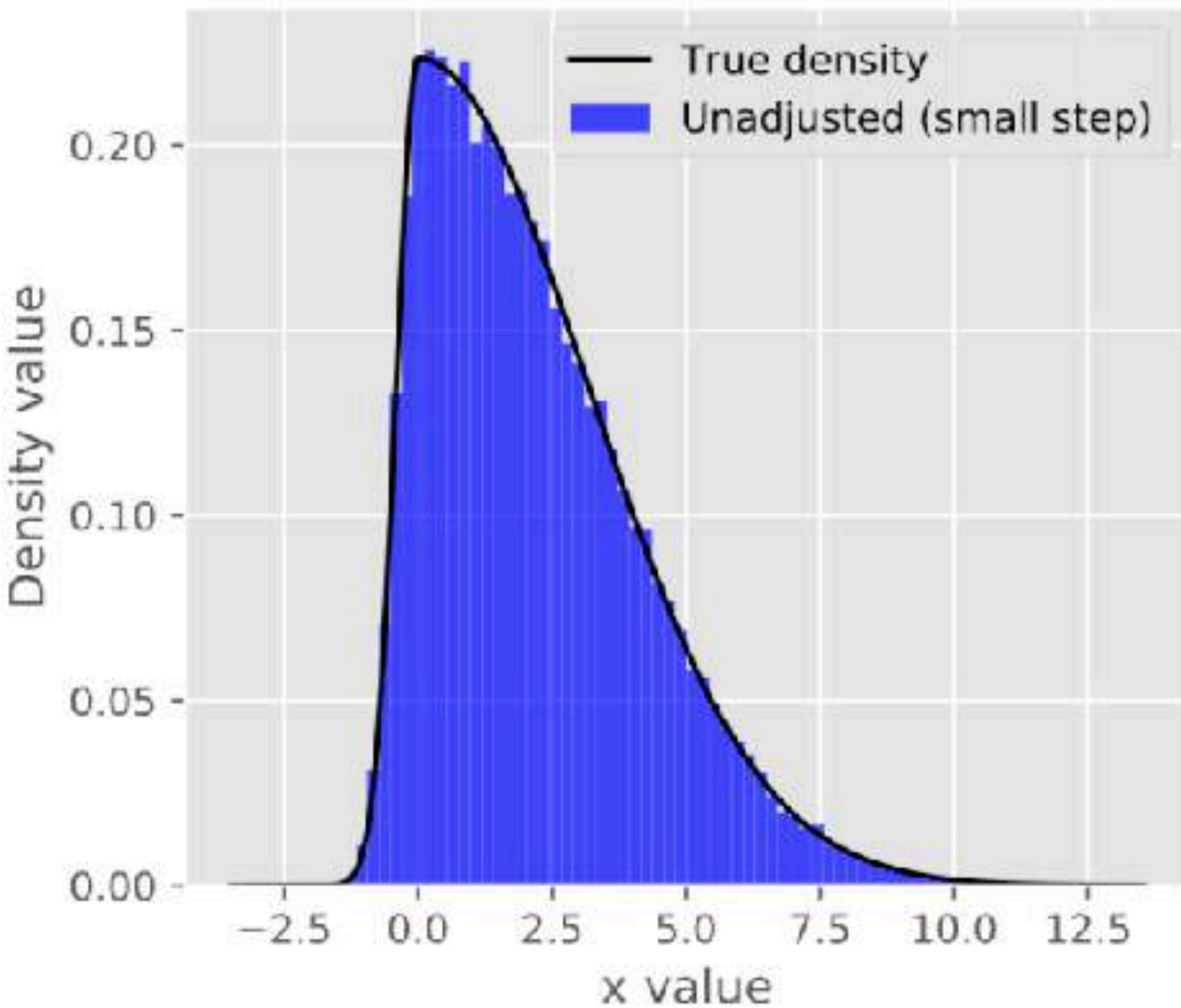


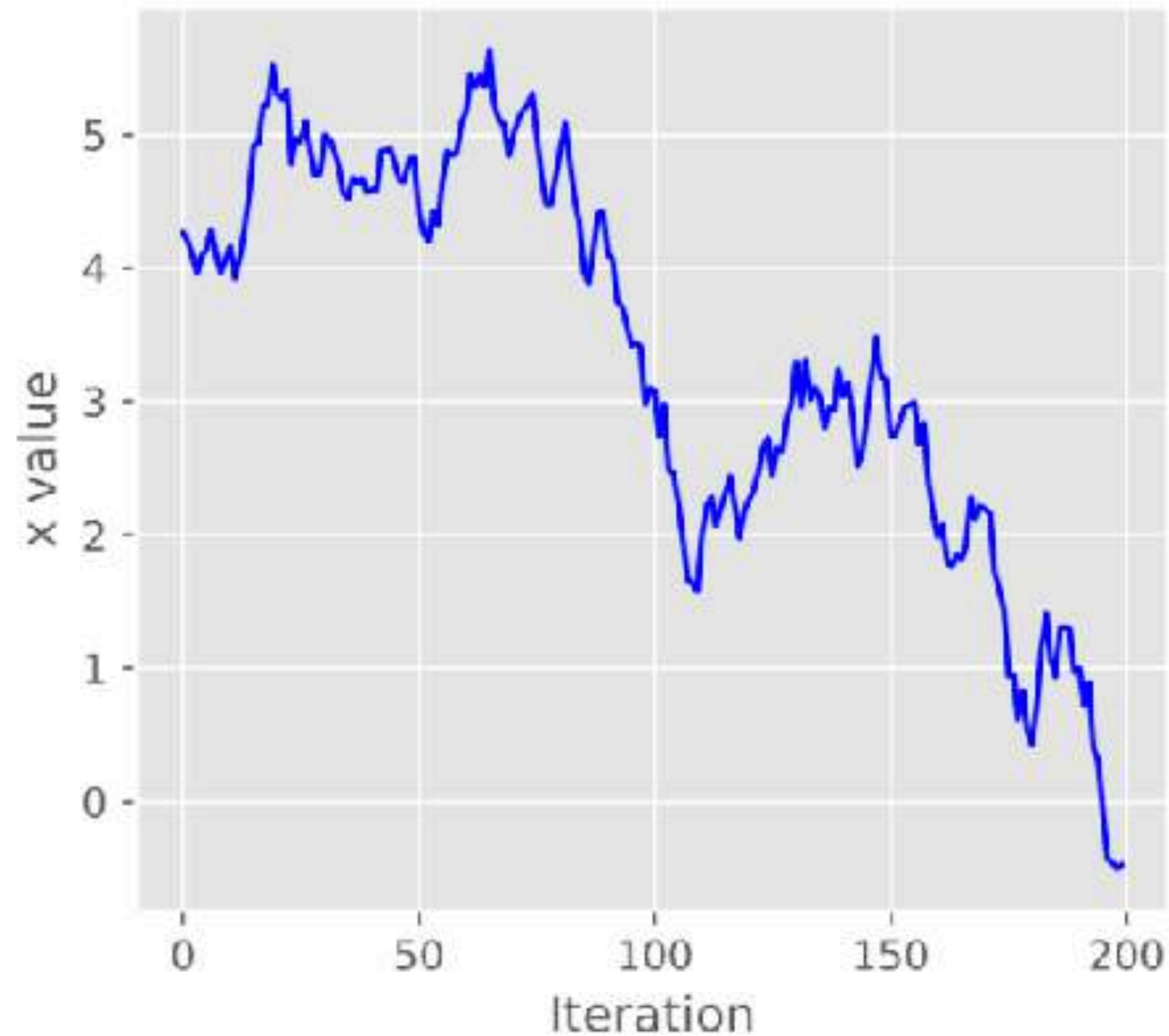
Upon convergence:
Histogram across multiple runs is biased



The iterates for one run are diverse

ULA performance:
Small step = small bias + slow mixing



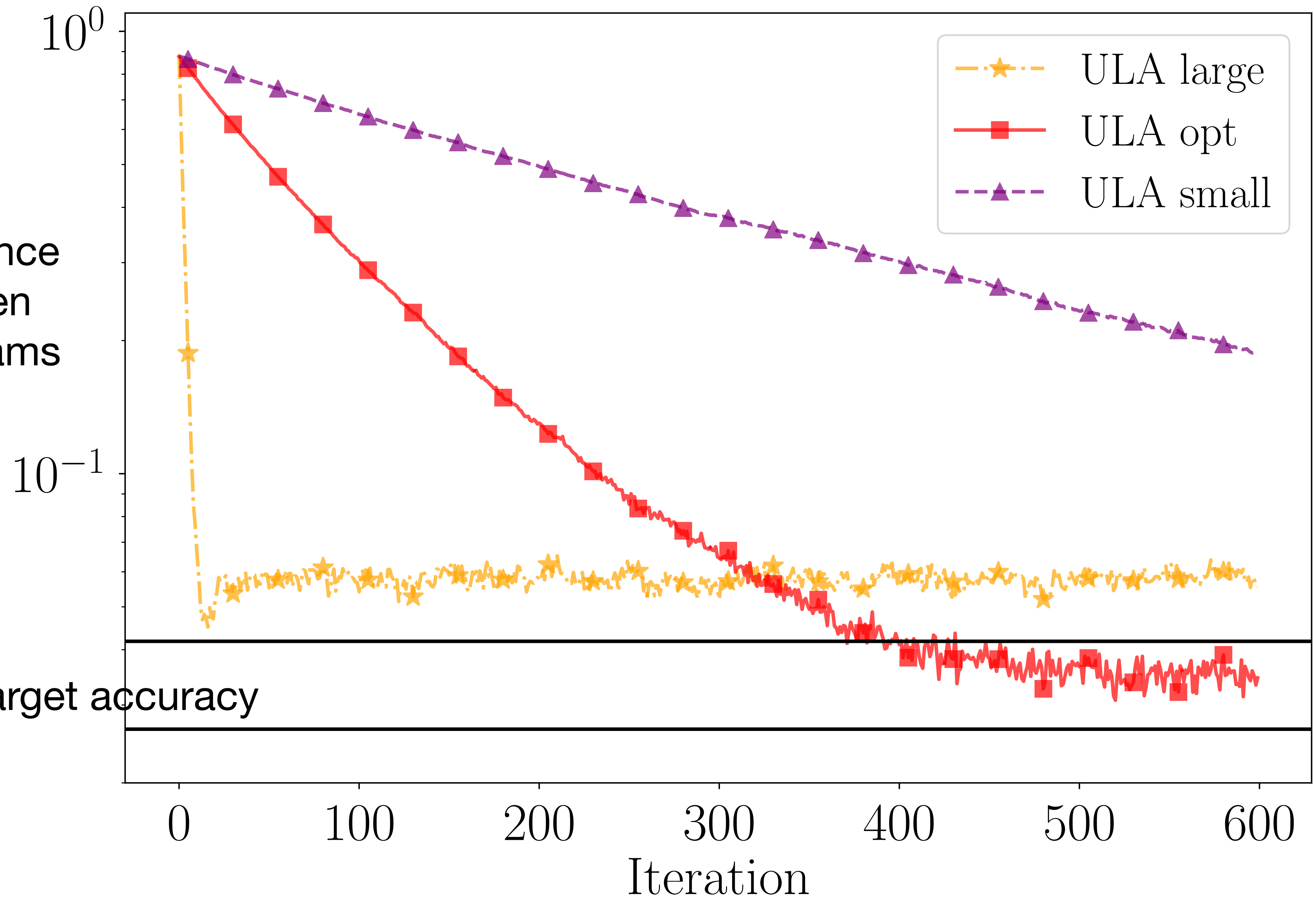


Upon convergence:
Histogram across multiple runs is *almost* unbiased

The iterates for one run are highly correlated

ULA: Step-size and speed/bias tradeoff

L1 distance
between
histograms



target accuracy

How do we remove the asymptotic bias?

- Via the **classical** Metropolis-Hastings correction step
- Metropolis adjusted Langevin algorithm (MALA):

1. Use ULA updates as proposals

$$z = x - h\nabla f(x) + \sqrt{2h}\xi$$

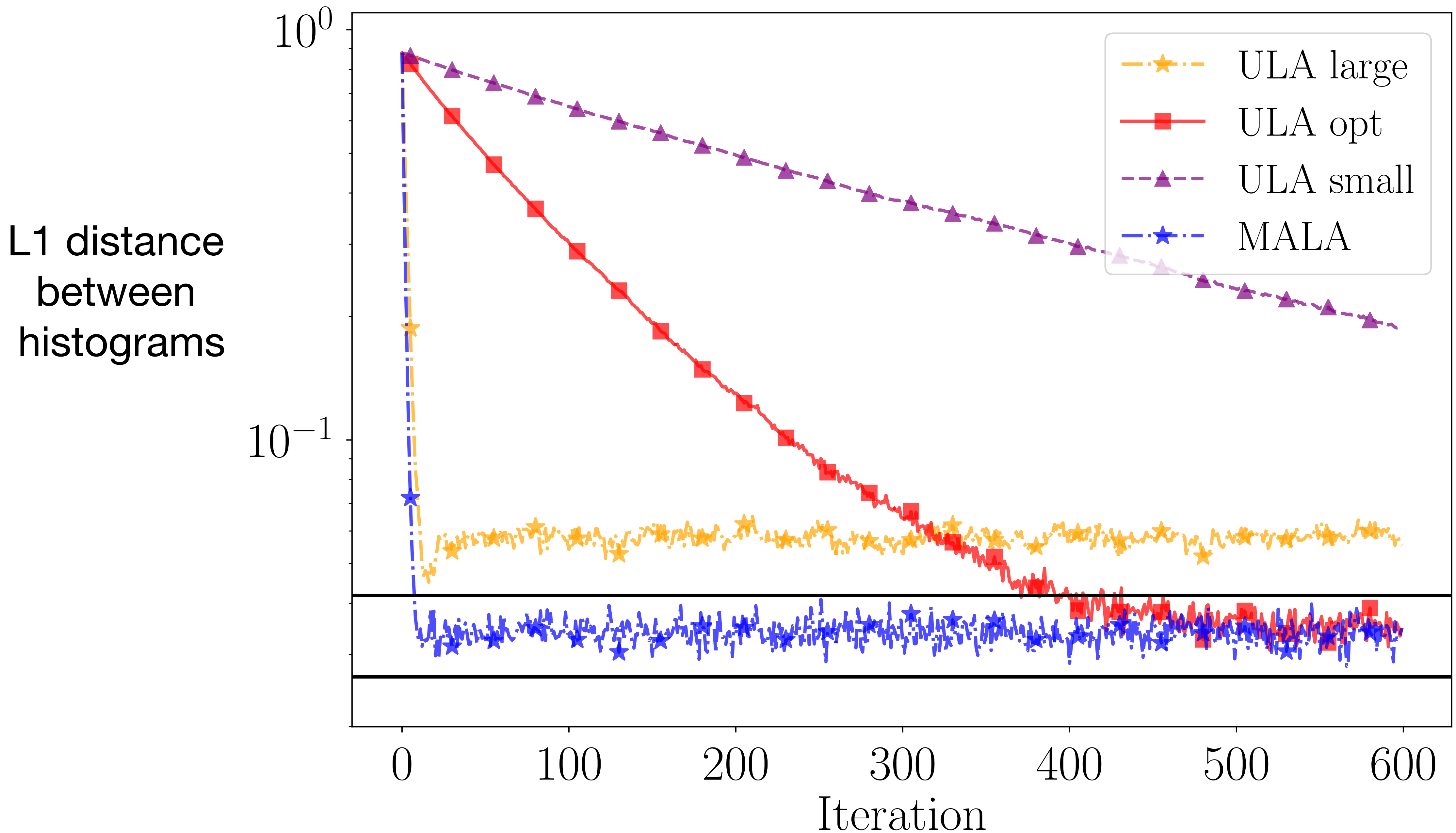
2. Accept z with probability

$$\min\left\{1, \frac{e^{-f(z)}}{e^{-f(x)}} \frac{P(z \to x)}{P(x \to z)}\right\}$$

3. In case of rejection, stay at x

Accept-reject makes the chain unbiased due to detailed balance condition

MALA: Fast convergence with no bias



Proof techniques for convergence of Markov Chains

✳Discrete state
Markov chains

- Coupling construction

- Conductance method

✳Continuous state
Markov chains

- Coupling construction
  - Coupling + Lyapunov
  - Coupling + SDE

- Conductance method

Mixing time bounds: Strongly log-concave

$$\|P(X_k) - \pi\|_{\mathrm{TV}} \le \delta$$

$\pi(x) \propto e^{-f(x)}$

Algorithm	ULA [Dalalyan 2016]	
f is L-smooth and m -strongly-convex	$d\left(\frac{L}{m}\right)^2 \frac{1}{\delta^2}$	

Mixing time bounds: Strongly log-concave

$$\|P(X_k) - \pi\|_{\mathrm{TV}} \leq \delta$$

$\pi(x) \propto e^{-f(x)}$

Algorithm	**ULA** [Dalalyan 2016]	**MALA** [Our work]
f is L-smooth and m -strongly-convex	$d\left(\frac{L}{m}\right)^2 \frac{1}{\delta^2}$	$d\left(\frac{L}{m}\right) \log \frac{1}{\delta}$

Mixing time bounds: Strongly log-concave

$$\|P(X_k) - \pi\|_{\mathrm{TV}} \leq \delta$$

$\pi(x) \propto e^{-f(x)}$

Algorithm	ULA [Dalalyan 2016]	MALA [Our work]
f is L-smooth and m -strongly-convex	$d\left(\frac{L}{m}\right)^2 \frac{1}{\delta^2}$	$d\left(\frac{L}{m}\right) \log \frac{1}{\delta}$
	Mixing time of MALA has • exponentially better dependence on accuracy δ • better dependence on conditioning L/m	

Mixing time bounds: Strongly and weakly log-concave

$$\|P(X_k) - \pi\|_{\mathrm{TV}} \leq \delta$$

$\pi(x) \propto e^{-f(x)}$

Algorithm	ULA [Dalalyan 2016]	MALA [Our work]
f is L-smooth and m -strongly-convex	$d\left(\frac{L}{m}\right)^2 \frac{1}{\delta^2}$	$d\left(\frac{L}{m}\right)\log\frac{1}{\delta}$
f is convex and L-smooth	$d^3 L^2 \frac{1}{\delta^4}$	$d^2 L^{1.5} \frac{1}{\delta^{1.5}}$

Mixing time bounds:
Strongly and weakly log-concave

$$\|P(X_k) - \pi\|_{\mathrm{TV}} \leq \delta$$

$\pi(x) \propto e^{-f(x)}$

Algorithm	ULA [Dalalyan 2016]	MALA [Our work]
f is L-smooth and m -strongly-convex	$d\left(\frac{L}{m}\right)^2 \frac{1}{\delta^2}$	$d\left(\frac{L}{m}\right) \log \frac{1}{\delta}$
f is convex and L-smooth	$d^3 L^2 \frac{1}{\delta^4}$	$d^2 L^{1.5} \frac{1}{\delta^{1.5}}$

Faster!

The difference between MALA and ULA: An informal proof

- Both algorithms converge quickly to their stationary distributions

The difference between MALA and ULA: An informal proof

- Both algorithms converge quickly to their stationary distributions

- ULA has a biased stationary distribution

Bias

$$\|P(x_k) - \pi\|_{\mathrm{TV}} \leq \|P(x_k) - \pi_{\mathrm{ULA}}\|_{\mathrm{TV}} + \|\pi_{\mathrm{ULA}} - \pi\|_{\mathrm{TV}}$$

The difference between MALA and ULA: An informal proof

- Both algorithms converge quickly to their stationary distributions

- ULA has a biased stationary distribution

Bias

$$\|P(x_k) - \pi\|_{\mathrm{TV}} \leq \underbrace{\|P(x_k) - \pi_{\mathrm{ULA}}\|_{\mathrm{TV}}}_{\mathcal{O}(e^{-kh})} + \underbrace{\|\pi_{\mathrm{ULA}} - \pi\|_{\mathrm{TV}}}_{\mathcal{O}(\sqrt{h})}$$

The difference between MALA and ULA: An informal proof

- Both algorithms converge quickly to their stationary distributions

- ULA has a biased stationary distribution

Bias

$$\|P(x_k) - \pi\|_{\mathrm{TV}} \leq \|P(x_k) - \pi_{\mathrm{ULA}}\|_{\mathrm{TV}} + \|\pi_{\mathrm{ULA}} - \pi\|_{\mathrm{TV}}$$

$$\mathcal{O}(e^{-kh}) \leq \delta/2 \qquad \mathcal{O}(\sqrt{h}) \leq \delta/2$$

$$k \geq \mathcal{O}\left(\frac{1}{h}\log\frac{1}{\delta}\right) = \mathcal{O}\left(\frac{1}{\delta^2}\right)$$

The difference between MALA and ULA: An informal proof

- Both algorithms converge quickly to their stationary distributions

- ULA has a biased stationary distribution

Bias

$$\|P(x_k) - \pi\|_{\mathrm{TV}} \leq \|P(x_k) - \pi_{\mathrm{ULA}}\|_{\mathrm{TV}} + \|\pi_{\mathrm{ULA}} - \pi\|_{\mathrm{TV}}$$

$$\mathcal{O}(e^{-kh}) \leq \delta/2 \qquad \mathcal{O}(\sqrt{h}) \leq \delta/2$$

$$k \geq \mathcal{O}\left(\frac{1}{h}\log\frac{1}{\delta}\right) = \mathcal{O}\left(\frac{1}{\delta^2}\right)$$

- MALA is unbiased: larger step size implies faster mixing

The difference between MALA and ULA: An informal proof

- Both algorithms converge quickly to their stationary distributions

- ULA has a biased stationary distribution

Bias

$$\|P(x_k) - \pi\|_{\mathrm{TV}} \leq \|P(x_k) - \pi_{\mathrm{ULA}}\|_{\mathrm{TV}} + \|\pi_{\mathrm{ULA}} - \pi\|_{\mathrm{TV}}$$

$$\mathcal{O}(e^{-kh}) \leq \delta/2 \qquad \mathcal{O}(\sqrt{h}) \leq \delta/2$$

$$k \geq \mathcal{O}\left(\frac{1}{h}\log\frac{1}{\delta}\right) = \mathcal{O}\left(\frac{1}{\delta^2}\right)$$

- MALA is unbiased: larger step size implies faster mixing

Step size limited by the rejection rate

Power of gradients

- What if we do not have gradient info?
- What if we take multiple gradient steps for each proposal step?

Three popular algorithms

Algorithm	Proposal Step
Random Walk (zeroth order)	$z = x + \sqrt{2h}\xi$
Langevin algorithm (first order)	$z = x - h\nabla f(x) + \sqrt{2h}\xi$
Hamiltonian Monte Carlo (first-second order)	Multi step version of Langevin algorithm

$\xi \sim \mathcal{N}(0, \mathbb{I}_d)$

Three popular algorithms

$$\pi(x) \propto e^{-f(x)}, \quad m\mathbb{I}_d \preceq \nabla^2 f(x) \preceq L\mathbb{I}_d, \quad \kappa = L/m$$

Algorithm	Mixing time
Metropolized Random Walk (MRW)	
Metropolis Adjusted Langevin Algorithm (MALA)	
Metropolized Hamiltonian Monte Carlo (HMC)	

Three popular algorithms

$$\pi(x) \propto e^{-f(x)}, \quad m\mathbb{I}_d \preceq \nabla^2 f(x) \preceq L\mathbb{I}_d, \quad \kappa = L/m$$

Algorithm	Mixing time
Metropolized Random Walk (MRW)	$d\kappa^2$
Metropolis Adjusted Langevin Algorithm (MALA)	$d\kappa$
Metropolized Hamiltonian Monte Carlo (HMC)	$d\kappa^{\frac{3}{4}}$

Three popular algorithms

$$\pi(x) \propto e^{-f(x)}, \quad m\mathbb{I}_d \preceq \nabla^2 f(x) \preceq L\mathbb{I}_d, \quad \kappa = L/m$$

Algorithm	Mixing time
Metropolized Random Walk (MRW)	$d\kappa^2$
Metropolis Adjusted Langevin Algorithm (MALA)	$d\kappa$
Metropolized Hamiltonian Monte Carlo (HMC)	$d\kappa^{\frac{3}{4}}$

More gradient information

Faster

Proof techniques for convergence of Markov Chains

✴Discrete state Markov chains

- Coupling construction
- Conductance method

✴Continuous state Markov chains

- Coupling construction
- Conductance method

Langevin algorithms: Prior work

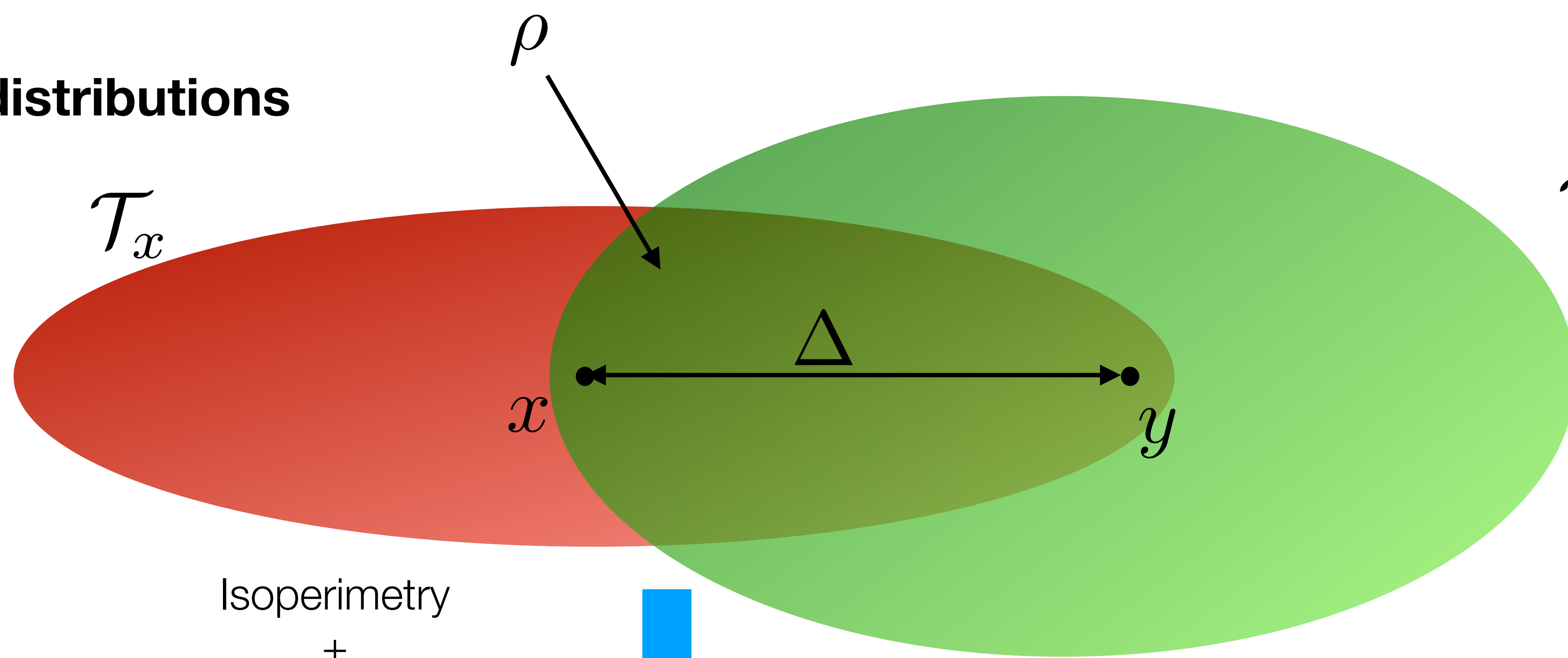
Type of results	Existing Literature	Techniques
Asymptotic convergence	[Talay & Tubaro ‘90], [Meyn & Tweedie ‘95], [Roberts & Rosenthal ‘96, ‘01, ‘02]	Lyapunov arguments
First non-asymptotic results	[Bou-Rabee & Hairer ‘09], [Roberts & Rosenthal ‘14], [Eberle ‘14]	Coupling + Lyapunov arguments
Explicit non-asymptotic bounds	[Dalalyan ‘15, ‘17], [Durmus & Moulines ‘15, ‘16], [Cheng & Bartlett ‘17]	Coupling + SDE errors

Langevin algorithms: Non-asymptotic bounds

Conductance method

Proof Outline

Transition distributions



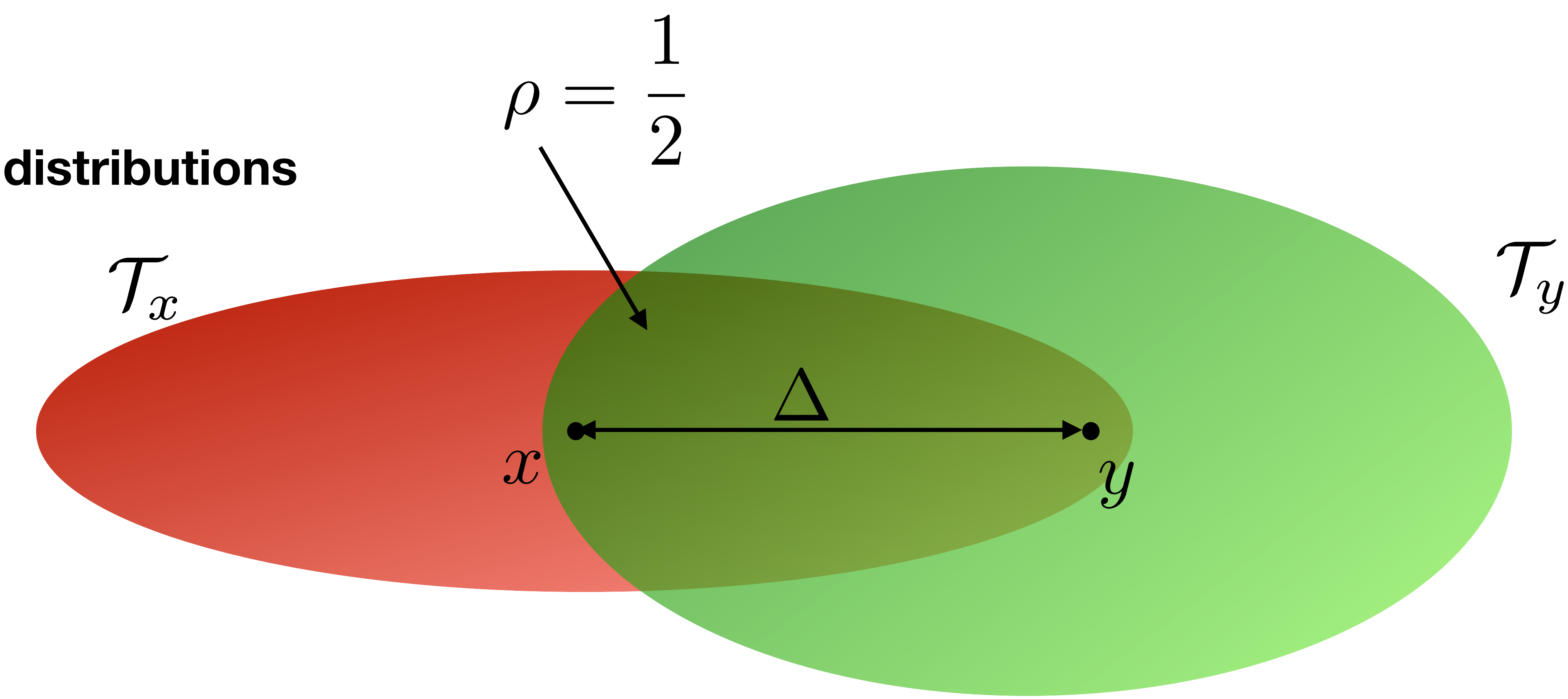
Isoperimetry
+
Conductance bounds for
spectral gap

$$\text{spectral gap} \geq 1 - \frac{\rho^2 \Delta^2}{2}$$

$$\|P(x_k) - \pi\|_{\mathrm{TV}} \leq \delta \text{ for } k \geq \mathcal{O}\left(\frac{\log(1/\delta)}{\Delta^2 \rho^2}\right)$$

Proof Outline

Transition distributions

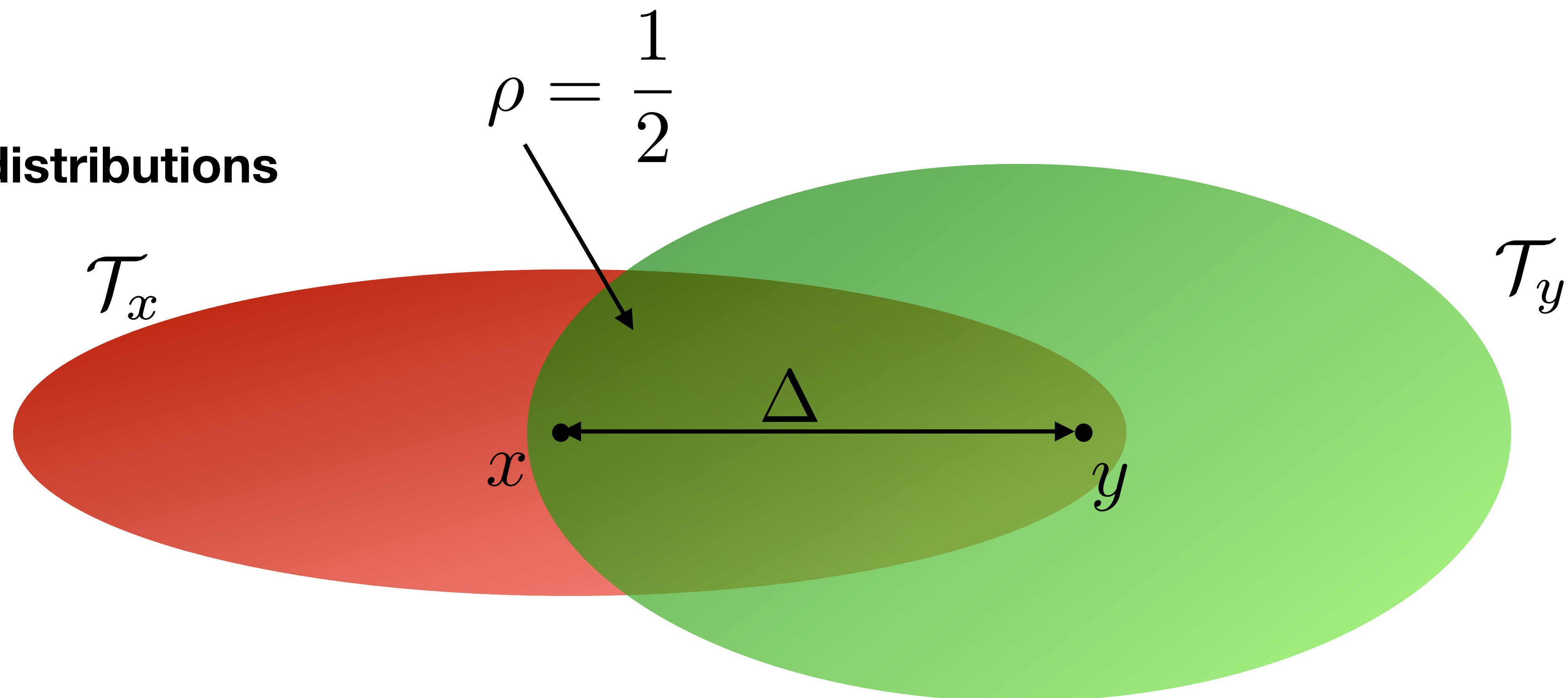


$$\|\mathcal{T}_x - \mathcal{T}_y\|_{\mathrm{TV}} \leq \tfrac{1}{2} \text{ whenever } d(x,y) \leq \Delta$$

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

Proof Outline

Transition distributions

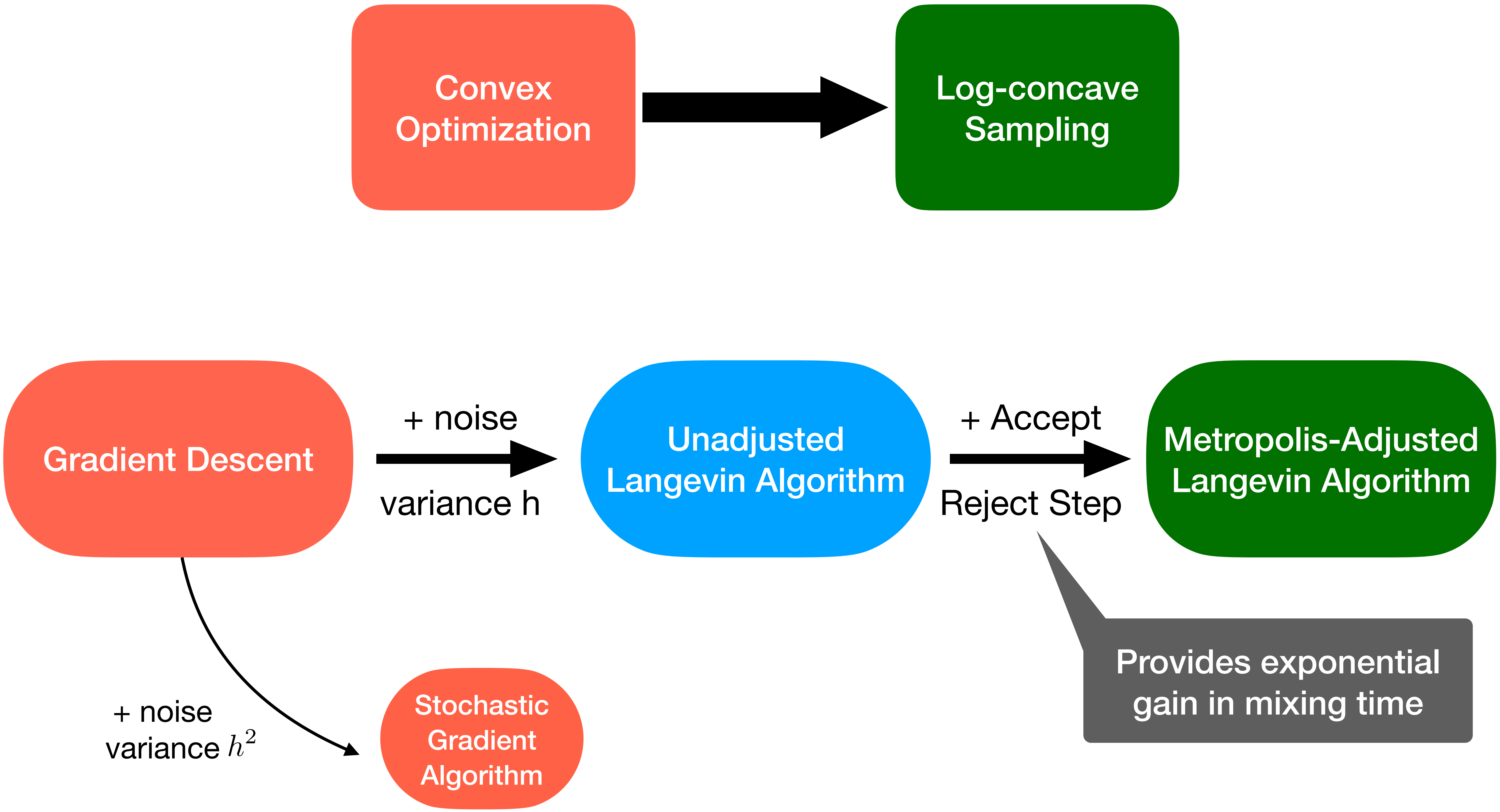


Difference in proposal and transition distribution due to accept-reject step

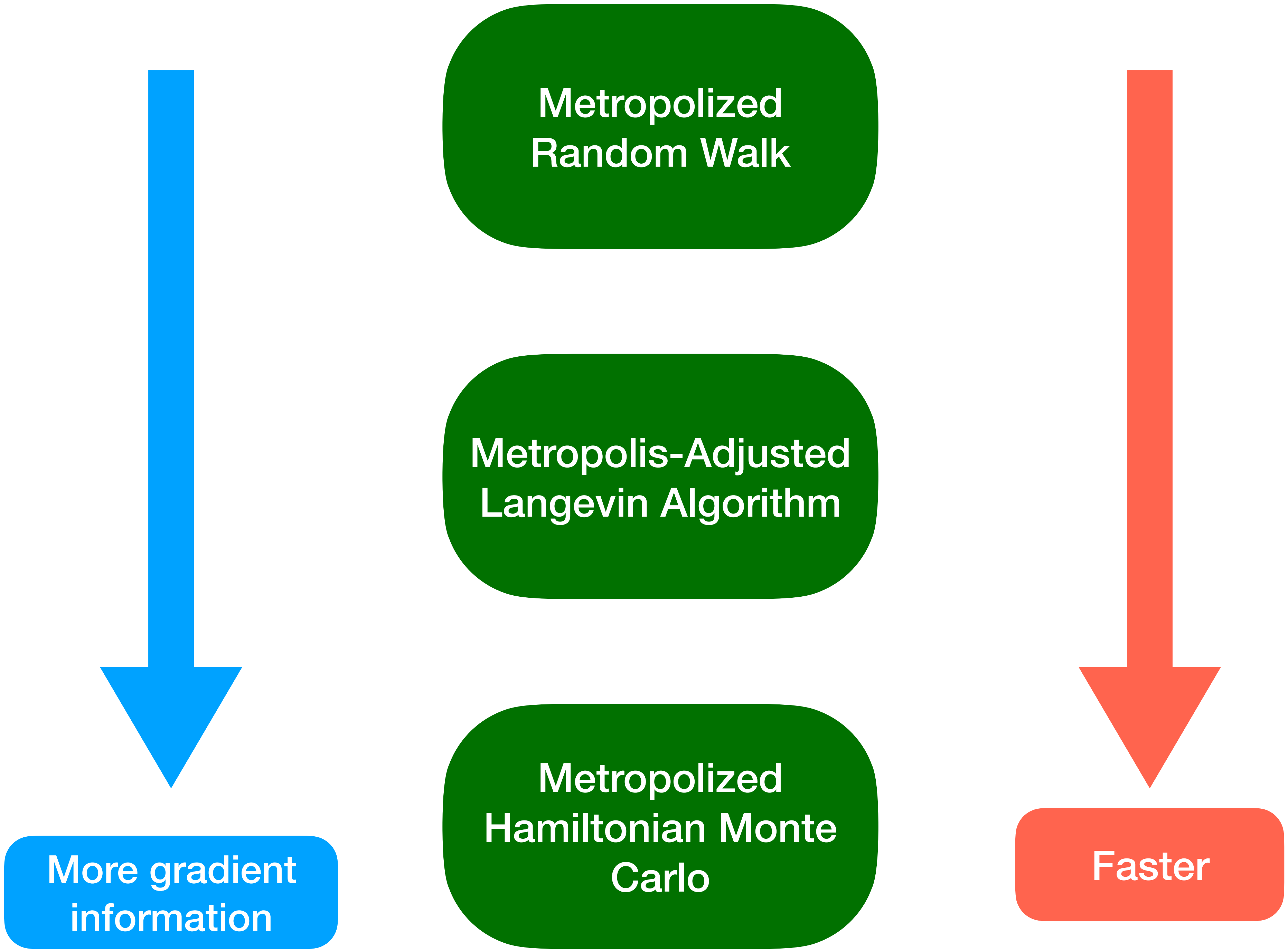
$$\|\mathcal{T}_x - \mathcal{T}_y\|_{\mathrm{TV}} \leq \|\mathcal{T}_x - \mathcal{P}_x\|_{\mathrm{TV}} + \|\mathcal{T}_y - \mathcal{P}_y\|_{\mathrm{TV}}$$
$$+ \|\mathcal{P}_x - \mathcal{P}_y\|_{\mathrm{TV}}$$

Difference in proposal distributions at two points

Power of accept-reject



Power of gradient information



http://people.eecs.berkeley.edu/~raaz.rsk/publications.html

- Log-concave sampling: Metropolis Hastings Algorithms are fast
- Fast Mixing of Metropolized Hamiltonian Monte Carlo: Benefits of Multi-Step Gradients