Importance Tempering of MCMC Schemes

Presenter: Quan Zhou

Department of Statistics, Texas A&M University

Acknowledgment

This talk is based on my recent works co-authored with

- Guanxun Li (PhD student/postdoc), Texas A&M University
- Hyunwoong (Woody) Chang (PhD student), Texas A&M University
- Aaron Smith, University of Ottawa

The research presented in this talk is supported by NSF DMS-2245591, DMS-2311307.

Markov chain Monte Carlo sampling

Markov chain Monte Carlo (MCMC) algorithms can be used to generate samples from a target distribution π. The main idea is to simulate a Markov chain with stationary distribution π.

Examples: Metropolis-Hastings (MH) algorithms, Gibbs samplers, etc.

Widely used in Bayesian statistics, since posterior distributions often involve intractable normalizing constants.

Example 1: variable selection

Consider the linear regression model

$$\boldsymbol{y} = \boldsymbol{Z}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

where $\boldsymbol{y}, \boldsymbol{\epsilon} \in \mathbb{R}^n$, $\boldsymbol{\beta} \in \mathbb{R}^p$ and $\boldsymbol{Z}$ is an $n \times p$ design matrix. We assume most entries of $\boldsymbol{\beta}$ are zero, and our goal is to identify

$$\gamma = \{1 \leq k \leq p \colon \beta_k \neq 0\}.$$

Search space

The search space is $2^{\{1,\ldots,p\}}$, which has cardinality 2^p.

Example 1: variable selection

In high-dimensional settings, sparsity constraints need to be imposed, but usually the search space still grows *super-polynomially* in p.

Local algorithms

Most sampling algorithms for variable selection are "local": the next move is selected from a "small" set of neighboring states which has cardinality *polynomial* in p.

Example: a typical search path in variable selection.

$$\{1,2\} \xrightarrow{\text{add covariate 4}} \{1,2,4\} \xrightarrow{\text{swap covariate 2 with 3}} \{1,3,4\}$$

$$\xrightarrow{\text{delete covariate 4}} \{1,3\} \xrightarrow{\text{delete covariate 1}} \{3\}$$

Example 1: variable selection

Example 2: structure learning

DAG model

A p-variate directed acyclic graph (DAG) encodes the conditional independence (CI) relations among p node variables.

Structure learning

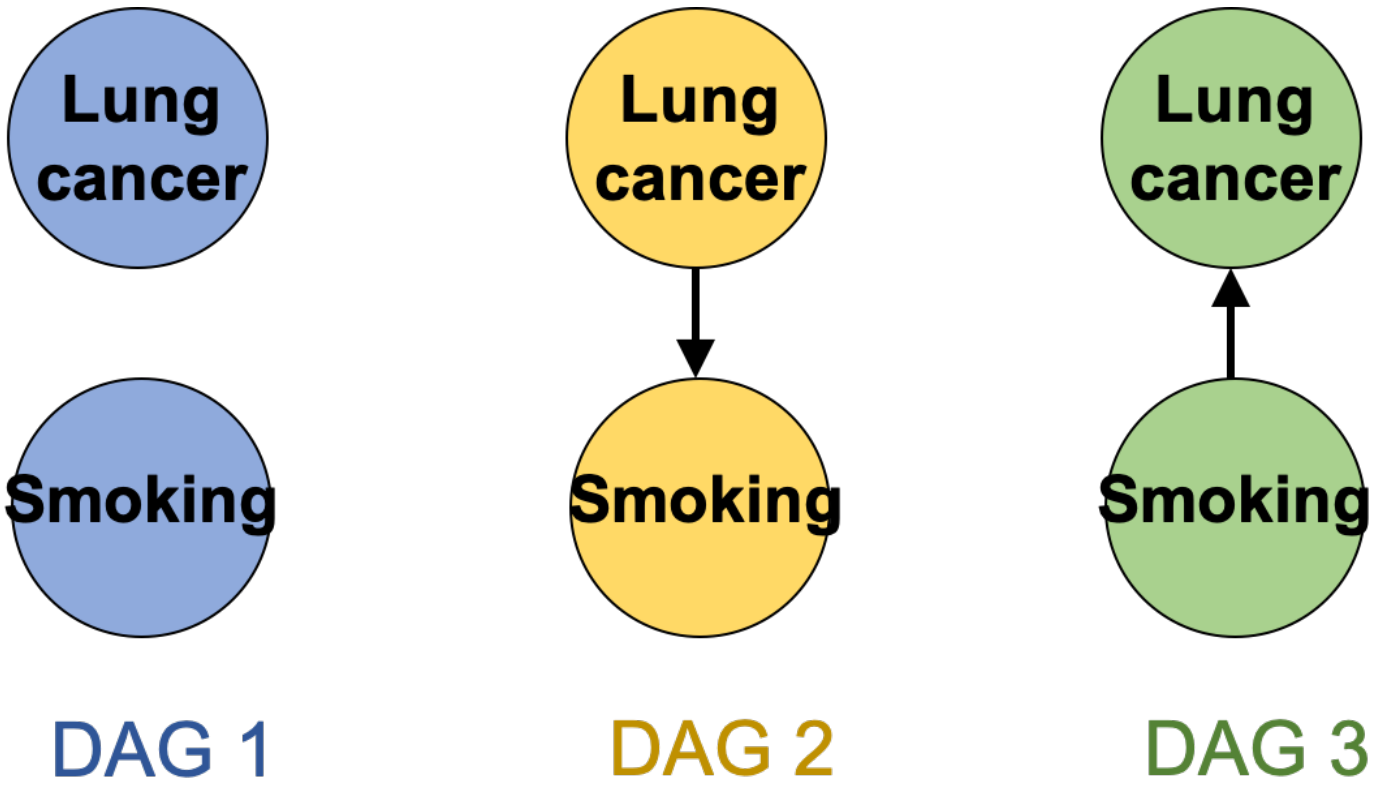
Learn the underlying DAG model of a p-variate probability distribution from n i.i.d. observations, $\boldsymbol{Z}_1, \boldsymbol{Z}_2, \ldots, \boldsymbol{Z}_n$; each $\boldsymbol{Z}_i \in \mathbb{R}^p$.

Search space

The collection of all p-vertex labeled DAGs; cardinality is super-exponential in p.

Example 2: structure learning

For two variables, there are 3 possible DAGs.



Example 3: estimation of PDE parameters

Suppose that we have i.i.d. observations $(z_1, y_1), (z_2, y_2), \ldots, (z_n, y_n)$ generated from

$$y_i = f(z_i; \boldsymbol{\theta}) + \epsilon_i,$$

where f is the solution to a partial differential equation parameterized by $\boldsymbol{\theta}$. Our goal is to estimate $\boldsymbol{\theta} \in \mathbb{R}^p$.

Search space

The parameter space is $\mathbb{R}^p$. Though it is continuous, gradient-based sampling methods cannot be applied.

Metropolis-Hastings (MH) algorithms

Let the state space be discrete. Assume that at every x, the *proposal* distribution $Q(x, \cdot)$ is a uniform distribution on N neighboring states. So

$$Q(x, y) = 1/N, \text{ if } y \sim x \text{ (i.e., } y \text{ is a neighbor of } x).$$

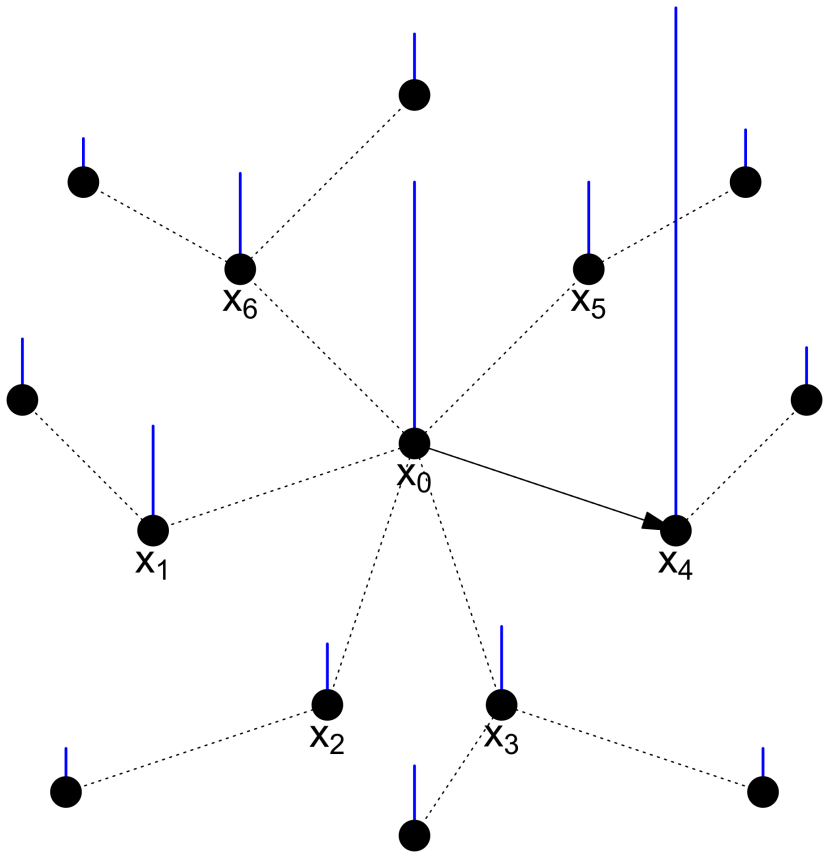
MH algorithm with proposal Q targeting π

An iteration at state x:

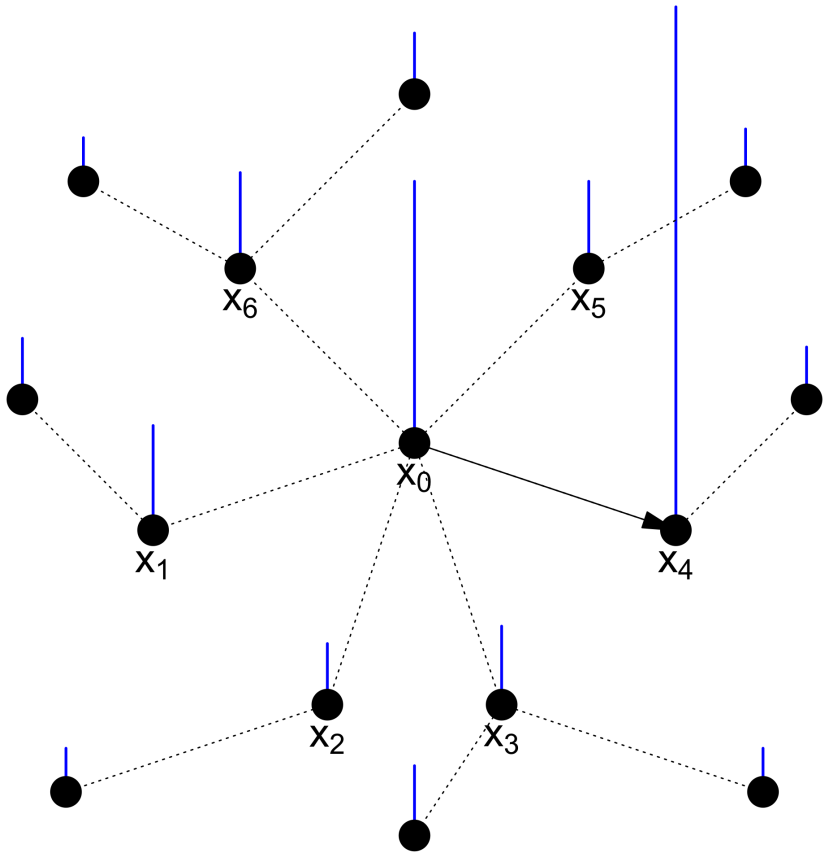
1. Draw y from $Q(x, \cdot)$.
2. Accept y with probability

$$\alpha(x, y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}.$$

3. If y is accepted, we move to y; otherwise, stay at x.



Each dot represents a state, and the height of the blue bar indicates $\pi(\cdot)$.
At point x_0, the best move is $x_0 \to x_4$.



At point x_0, a random walk proposal proposes x_4 with probability $1/6$. We may use a *locally informed* proposal to increase this probability; e.g. we propose x_i with probability $\propto \pi(x_i)$.

Remarks on informed proposals

- Similar ideas are used in many MCMC methods [Titsias and Yau, 2017, Zanella and Roberts, 2019, Zanella, 2020, Griffin et al., 2021] and some non-MCMC methods [Hans et al., 2007, Shin et al., 2018].
- To implement an informed proposal at x, we need to evaluate $\pi(y)$ for each $y \sim x$; this can be parallelized.
- Difficult to control the acceptance probability.
- Informed MH algorithms can mix even more slowly than RWMH.

Questions addressed in my works

1. Q: Do we have theoretical guarantees for informed MH algorithms?

   A: Yes. See Zhou et al. [2022], Zhou and Chang [2023].

2. Q: How to choose the informed proposal scheme?

   A: Quite complicated, but some guidance is provided in Zhou et al. [2022], Zhou and Smith [2022].

3. Q: Do we have to use MH schemes?

   A: No, we can use importance tempering [Zhou and Smith, 2022].

4. Q: How general is this importance tempering technique?

   A: Almost every MH scheme can be converted to an importance tempering scheme [Li et al., 2023].

Q3 and Q4 are the focus of today's talk.

Metropolis-Hastings (MH) algorithms

Assume that $Q(x, y) = 1/N$, if $y \sim x$ (i.e., y is a neighbor of x).

MH algorithm with proposal Q targeting π

An iteration at state x:

1. Draw y from $Q(x, \cdot)$.
2. Accept y with probability

$$\alpha(x, y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}.$$

3. If y is accepted, we move to y; otherwise, stay at x.

Metropolis-Hastings (MH) algorithms

Let P denote the transition matrix of the MH sampler:

$$P(x,y) = Q(x,y)\alpha(x,y) = N^{-1}\alpha(x,y)\mathbb{1}(y \sim x), \quad \forall\, x \neq y.$$

And $P(x,x) = 1 - \sum_{y:\ y\sim x} P(x,y)$.

- P is *reversible* w.r..t. π (detailed balance condition):

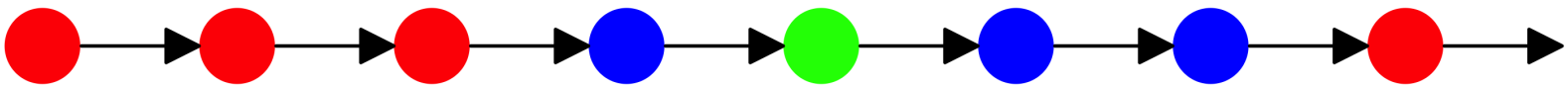
$$\pi(x)P(x,y) = \pi(y)P(y,x),$$
$$\text{i.e., } \pi(x)\alpha(x,y) = \pi(y)\alpha(y,x).$$

- To simulate P, we only need to know an *un-normalized* version of π.

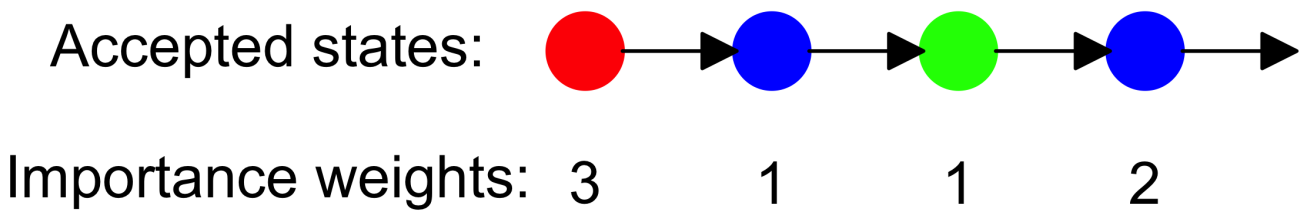
Another perspective on MH algorithms

Accepted moves of an MH algorithm also form a Markov chain.

Example: Let the state space be {red, blue, green} and consider the following path.



An equivalent representation of the above path is



Another perspective on MH algorithms

Denote the transition matrix of accepted moves by $\tilde{P}$. Observe that

$\tilde{P}(x, \cdot)$ is the conditional distribution of $Y \sim P(x, \cdot)$ given that $Y \neq x$.

For any $x \neq y$, we have

$$\tilde{P}(x, y) = \frac{P(x, y)}{Z(x)}, \text{ where } Z(x) = \sum_{x' :\, x' \sim x} P(x, x').$$

Stationary distribution of $\tilde{P}$

$\tilde{P}$ is reversible with respect to the distribution $\tilde{\pi}$ such that

$$\tilde{\pi}(x) \propto \pi(x) Z(x).$$

That is, $Z(x)^{-1} \propto \pi(x)/\tilde{\pi}(x)$ is the *un-normalized* importance weight!

Another perspective on MH algorithms

P	R	B	G
R	$1/4$	$1/2$	$1/4$
B	$1/2$	$1/4$	$1/4$
G	$1/2$	$1/2$	0

$\Rightarrow$

$\tilde{P}$	R	B	G
R	0	$2/3$	$1/3$
B	$2/3$	0	$1/3$
G	$1/2$	$1/2$	0

	π	Z	πZ
R	$2/5$	$3/4$	$3/10$
B	$2/5$	$3/4$	$3/10$
G	$1/5$	1	$2/10$

$\Rightarrow$

	$\tilde{\pi}$	$\pi/\tilde{\pi}$
R	$3/8$	$16/15$
B	$3/8$	$16/15$
G	$2/8$	$12/15$

Another perspective on MH algorithms

Interpretation of $Z(x)^{-1}$

The number of iterations the MH algorithm stays at x is a geometric random variable with mean $Z(x)^{-1}$.

Hence, we can think of MH as an algorithm which

- targets the posterior distribution π,
- simulates the Markov chain $\tilde{P}$, and
- uses acceptance-rejection to *unbiasedly* estimate importance weights.

See, e.g., Douc and Robert [2011].

A trivial but surprisingly useful observation

Recall that

$$Z(x) = \sum_{y\colon y\sim x} P(x,y) = \sum_{y\colon y\sim x} \alpha(x,y)/N.$$

For discrete-space problems, we can often calculate $Z(x)$ exactly. All we need is to evaluate $\pi(y)/\pi(x)$ for every y such that $y \sim x$.

Why is it useful?

Imagine that the MH algorithm is at some x^* such that $\pi(x^*) \gg \pi(y)$ for every $y \sim x$. Exact calculation of $Z(x^*)$ guarantees that we always leave x^* after N posterior evaluations.

Revisit the detailed balance condition

Consider the detailed balance condition of the MH algorithm:

$$\pi(x)\alpha(x,y) = \pi(y)\alpha(y,x).$$

There are many choices of α such that the above condition holds. The choice $\min\{1, \pi(y)/\pi(x)\}$ is most popular since *Peskun's ordering* guarantees that this is the optimal choice for MH schemes.

But if we calculate $Z(x) = \sum_{y\sim x} \alpha(x,y)/N$ exactly,

- the optimal choice of α is no longer clear, and
- we can simulate $\tilde{P}$ even if α is *unbounded*.

How to choose α

Balancing function (Zanella, JASA 2020)

We say $h\colon (0,\infty) \to (0,\infty)$ is a *balancing* function if

$$h(u) = u\, h(1/u), \qquad \text{for all } u > 0.$$

Examples: $h(u) = 1 + u, h(u) = \min\{1, u\}, h(u) = \sqrt{u}$.

Main idea: pick a balancing function h and replace α with

$$\alpha_h(x, y) = h\left(\frac{\pi(y)}{\pi(x)}\right)$$

in the previous sampling schemes.

Informed importance tempering (IIT)

Informed importance tempering

An iteration at state x:

1. Calculate $\alpha_h(x, y)$ for every $y \sim x$.
2. Calculate $Z_h(x) = \sum_{y \sim x} \alpha_h(x, y)/N$.
3. Assign to x importance weight $1/Z_h(x)$.
4. Move to x_{new} with probability proportional to $\alpha_h(x, x_{\text{new}})$.

This generalizes the tempered Gibbs sampler of Zanella and Roberts [2019], an MCMC scheme for variable selection that can be seen as IIT with balancing function $h(u) = 1 + u$.

The term "importance tempering" comes from Gramacy et al. [2010], Zanella and Roberts [2019].

Rapid convergence of IIT

An obvious advantage of IIT is that the chain is always moving.

In Zhou and Smith [2022], we show that:

- If π is "strongly unimodal," IIT with $h(u) = 1 + u$ converges "extremely fast."
- However, $h(u) = 1 + u$ is too aggresive and can be very inefficient for multimodal targets.
- $h(u) = \sqrt{u}$ performs well in a wide range of settings.

Extensions of IIT

This importance tempering trick turns out to be widely applicable. Most existing Metropolis-Hastings schemes can be converted to IIT versions.

In Li et al. [2023], we propose the following IIT variants:

- IIT schemes that do not require posterior evaluation of all neighbors;
- integration of IIT and simulated tempering algorithm;
- integration of IIT and pseudo-marginal methods;
- importance-tempered multiple-try algorithm, which is applicable to general state spaces.

IIT schemes appear to always converge faster than their MH counterparts in our numerical studies.

Example: combining MH and IIT

Consider the naive IIT with $h(u) = \min\{1, u\}$. The difference between it and MH is that, for each accepted state x,

- IIT *exactly* calculates the importance weight $1/Z_h(x)$,
- MH estimates the importance weight using a *geometric* random variable with mean $1/Z_h(x)$.

We can combine the two approaches. In each iteration,

- with probability ϵ, we exactly calculate $Z_h(x)$;
- with probability $1 - \epsilon$, we perform the acceptance-rejection as in MH.

Why is this useful?

Example: importance tempering of MTM

Multiple-try Metropolis (MTM) algorithms are widely used but known to suffer from low acceptance rates [Yang and Liu, 2023]. Chang et al. [2022], Gagnon et al. [2022] proposed to use balancing functions to construct locally balanced MTM schemes, which tend to have high acceptance rates.

But importance tempering can be used to further enforce the sampler to always accept the proposal. There is *no* extra computational cost for obtaining the importance weight; it is just the term used in the acceptance ratio of MTM.

Example: importance tempering of MTM

Locally balanced MTM on general state spaces

Let $Q(x, \cdot)$ denote a symmetric proposal with density q such that $q(x, y) = q(y, x)$. Let h be a balancing function.

An iteration of MTM at state x with m tries:

1. Draw $y_1, \ldots, y_m$ from $Q(x, \cdot)$.
2. Select y from $y_1, \ldots, y_m$ with probability $\propto h(\pi(y)/\pi(x))$.
3. Draw $x_1, \ldots, x_{m-1}$ from $Q(y, \cdot)$. Set $x_m = x$.
4. Accept y with probability

$$\min\left\{1, \frac{Z_h(x, y_1, \ldots, y_m)}{Z_h(y, x_1, \ldots, x_m)}\right\},$$

where $Z_h(x, y_1, \ldots, y_m) = \sum_{k=1}^{m} h(\pi(y_k)/\pi(x))$.

Example: importance tempering of MTM

Multiple-try importance tempering

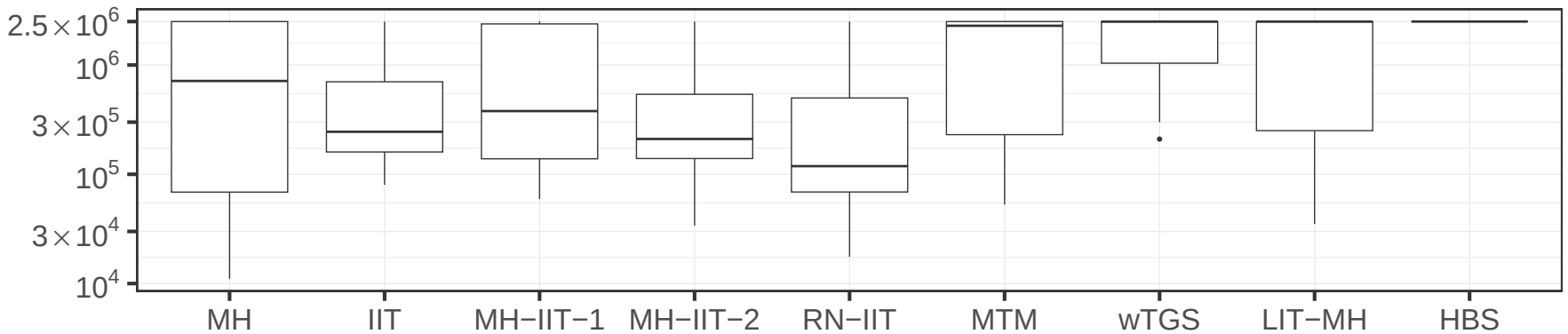
In Step 4, we can actually just accept y and assign to the previous state x importance weight $1/Z_h(x, y_1, \ldots, y_m)$.

Caveat: in the next iteration, the m candidate neighboring states of y are NOT resampled (we have already generated them in Step 3).

Why is it correct? One can show that this algorithm is just a standard IIT algorithm on an *augmented* space with auxiliary variables being the m candidate neighboring states.

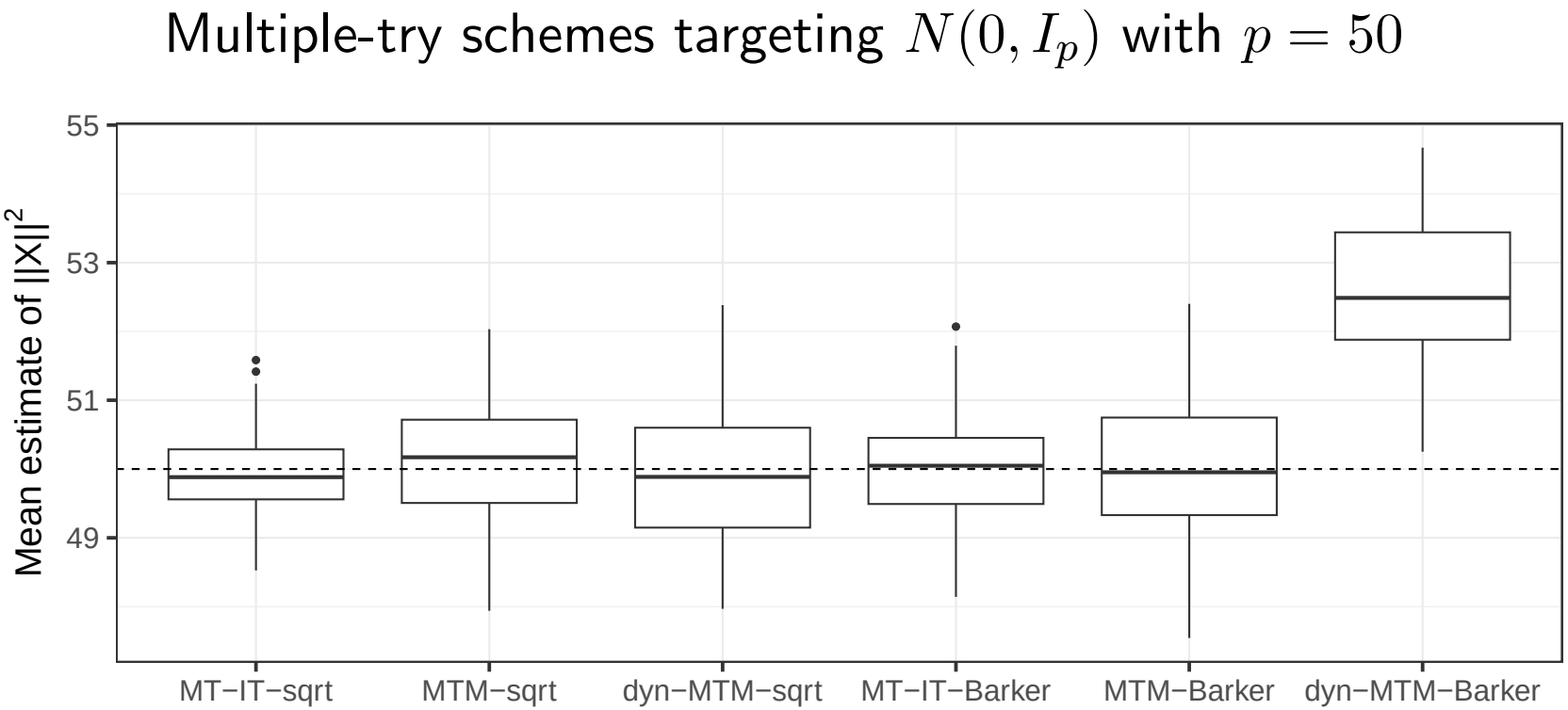
Numerical examples

A variable selection problem with $n = 1,000$ and $p = 5,000$



Box plot for the number of posterior calls (truncated at 2.5M) needed to find the best model. We consider a setting described in [Yang et al., 2016], where the design matrix has high collinearity, and the signal-to-noise ratio is intermediate. RN-IIT is a variant of the multiple-try importance tempering on discrete spaces. MTM: [Chang et al., 2022]; wTGS: [Zanella and Roberts, 2019]; LIT-MH: [Zhou et al., 2022]; HBS: [Titsias and Yau, 2017].

Numerical examples



Box plot of the estimate of $E||X||_2^2$ where $X \sim N(0, I_p)$. All algorithms are run for 5×10^5 posterior calls, and the estimate is calculated using the second half of MCMC samples. MT-IT denotes our method; MTM: [Gagnon et al., 2022]; sqrt: $h(u) = \sqrt{u}$; Barker: $h(u) = u/(1+u)$.

More IIT schemes

Simulated tempering

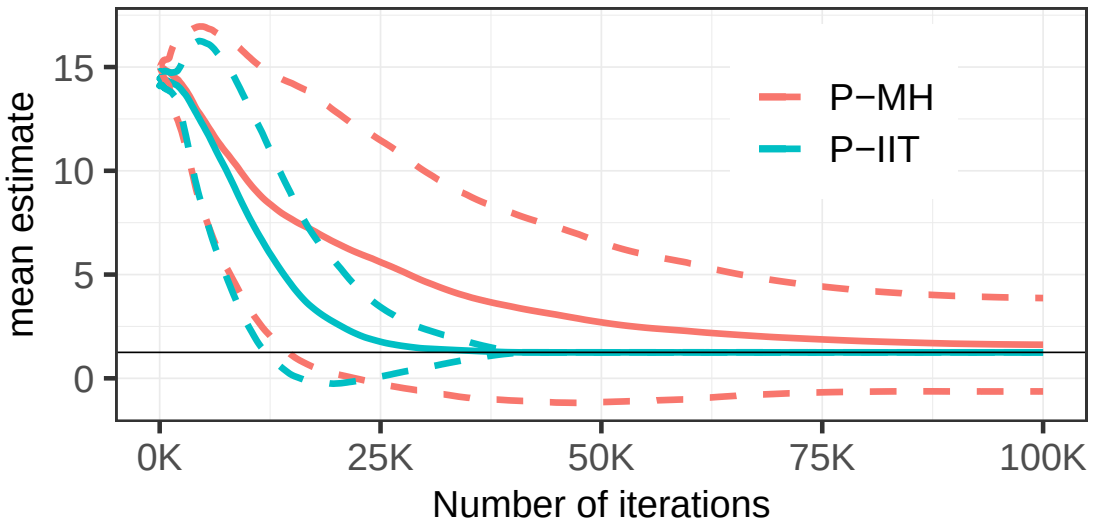
For simulated/parallel tempering, we can replace the MH chain at each temperature with an IIT chain. One important advantage is:
we can use aggressive h at lower temperatures to encourage *exploitation*, and conservative h at higher temperatures to encourage *exploration*.

Pseudo-marginal MCMC

As in pseudo-marginal MH algorithms, we can also use unbiased estimates for π in an IIT scheme. The pseudo-marginal IIT never gets stuck at any state, which is very different from pseudo-marginal MH.

Numerical example

A geometric distribution example studied in Lee and Łatuszyński [2014]



Trace plot for the posterior mean estimate with half of samples discarded as burn-in. π is a geometric distribution, and $\pi(x)$ is estimated by using the mean of K Bernoulli random variables ($K = 100$) such that the success probability $\to 0$ as $x \to \infty$. Dashed lines: ± 1 standard deviation from 50 runs. Both samplers are initialized at $x = 15$.

Concluding remarks

- Informed MCMC methods can be much more efficient than random walk MH for model selection problems. Local evaluation of π can be parallelized. For more results on informed MCMC, see my slides [link].
- For mixing time analysis, see Zhou et al. [2022], Zhou and Smith [2022], Zhou and Chang [2023].
- The balancing function h needs to be chosen with caution.
- Importance tempering seems always better than MH for utilizing informed proposals. See Li et al. [2023] for more examples.
- Importance tempering perspective opens doors to devising new MCMC schemes that are more efficient than existing ones.

Thank you!

Slides available at `https://web.stat.tamu.edu/~quan`

- QZ and A. Smith. "Rapid convergence of informed importance tempering." *AISTATS* (oral presentation), `arXiv:2107.10827`.
- G. Li, A. Smith and QZ. "Importance is Important: A Guide to Informed Importance Tempering Methods." `arXiv:2304.06251`.

References I

Hyunwoong Chang, Changwoo Lee, Zhao Tang Luo, Huiyan Sang, and Quan Zhou. Rapidly mixing multiple-try Metropolis algorithms for model selection problems. *Advances in Neural Information Processing Systems*, 35: 25842–25855, 2022.

Randal Douc and Christian P Robert. A vanilla Rao–Blackwellization of Metropolis–Hastings algorithms. *The Annals of Statistics*, 39(1):261–277, 2011.

Philippe Gagnon, Florian Maire, and Giacomo Zanella. Improving multiple-try Metropolis with local balancing. *arXiv preprint arXiv:2211.11613*, 2022.

Robert Gramacy, Richard Samworth, and Ruth King. Importance tempering. *Statistics and Computing*, 20:1–7, 2010.

JE Griffin, KG Łatuszyński, and MFJ Steel. In search of lost mixing time: adaptive Markov chain Monte Carlo schemes for Bayesian variable selection with very large p. *Biometrika*, 108(1):53–69, 2021.

Chris Hans, Adrian Dobra, and Mike West. Shotgun stochastic search for "large p" regression. *Journal of the American Statistical Association*, 102(478): 507–516, 2007.

References II

Anthony Lee and Krzysztof Łatuszyński. Variance bounding and geometric ergodicity of Markov chain Monte Carlo kernels for approximate Bayesian computation. *Biometrika*, 101(3):655–671, 2014.

Guanxun Li, Aaron Smith, and Quan Zhou. Importance is important: A guide to informed importance tempering methods. *arXiv preprint arXiv:2304.06251*, 2023.

Minsuk Shin, Anirban Bhattacharya, and Valen E Johnson. Scalable Bayesian variable selection using nonlocal prior densities in ultrahigh-dimensional settings. *Statistica Sinica*, 28(2):1053, 2018.

Michalis K Titsias and Christopher Yau. The Hamming ball sampler. *Journal of the American Statistical Association*, 112(520):1598–1611, 2017.

Xiaodong Yang and Jun S Liu. Convergence rate of multiple-try Metropolis independent sampler. *Statistics and Computing*, 33(4):79, 2023.

Yun Yang, Martin J Wainwright, and Michael I Jordan. On the computational complexity of high-dimensional Bayesian variable selection. *The Annals of Statistics*, 44(6):2497–2532, 2016.

Giacomo Zanella. Informed proposals for local MCMC in discrete spaces. *Journal of the American Statistical Association*, 115(530):852–865, 2020.

References III

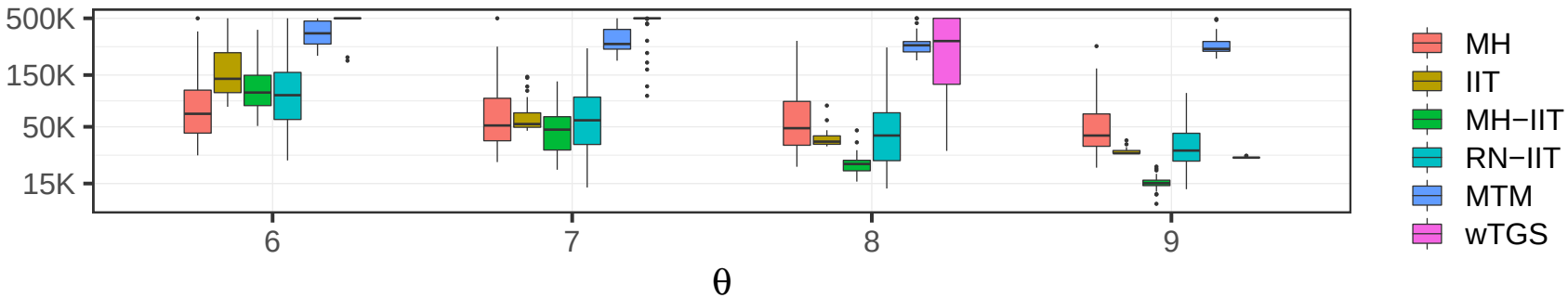
Giacomo Zanella and Gareth Roberts. Scalable importance tempering and Bayesian variable selection. *Journal of the Royal Statistical Society Series B*, 81(3):489–517, 2019.

Quan Zhou and Hyunwoong Chang. Complexity analysis of Bayesian learning of high-dimensional DAG models and their equivalence classes. *Annals of Statistics*, to appear, 2023.

Quan Zhou and Aaron Smith. Rapid convergence of informed importance tempering. pages 10939–10965, 2022.

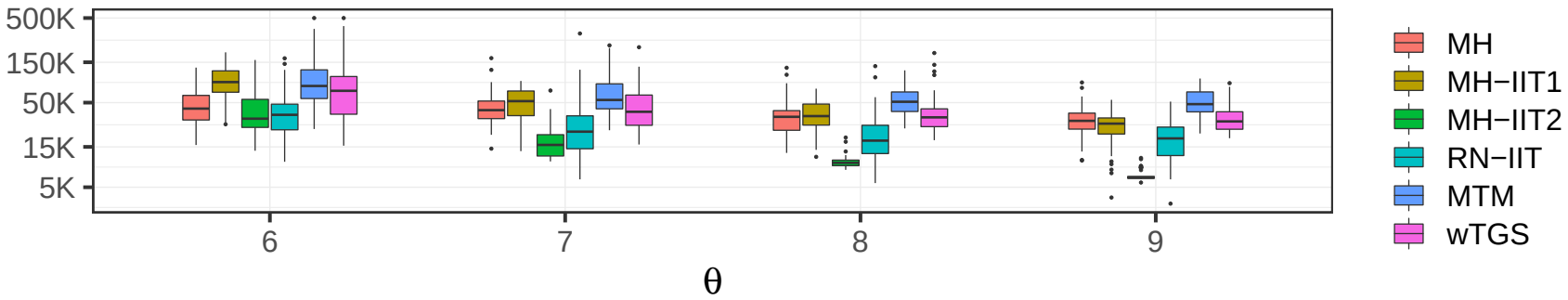
Quan Zhou, Jun Yang, Dootika Vats, Gareth O Roberts, and Jeffrey S Rosenthal. Dimension-free mixing for high-dimensional bayesian variable selection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(5): 1751–1784, 2022.

Numerical examples



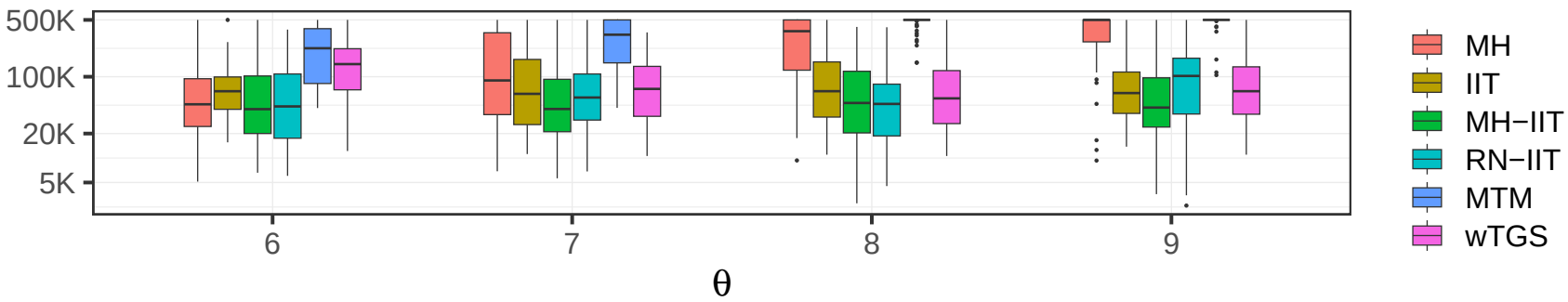
Box plot for the number of posterior calls (truncated at 500K) needed to accurately approximate π, which is a unimodal distribution on $\{0,1\}^p$ with independent coordinates ($p = 50$). A larger θ corresponds to faster tail decay. RN-IIT is a variant of the multiple-try importance tempering on discrete spaces.

Numerical examples



π is a unimodal distribution on $\{0,1\}^p$ with dependent coordinates ($p = 10$).

Numerical examples



π is a mixture of two unimodal distributions on $\{0,1\}^p$ with independent coordinates ($p = 50$).