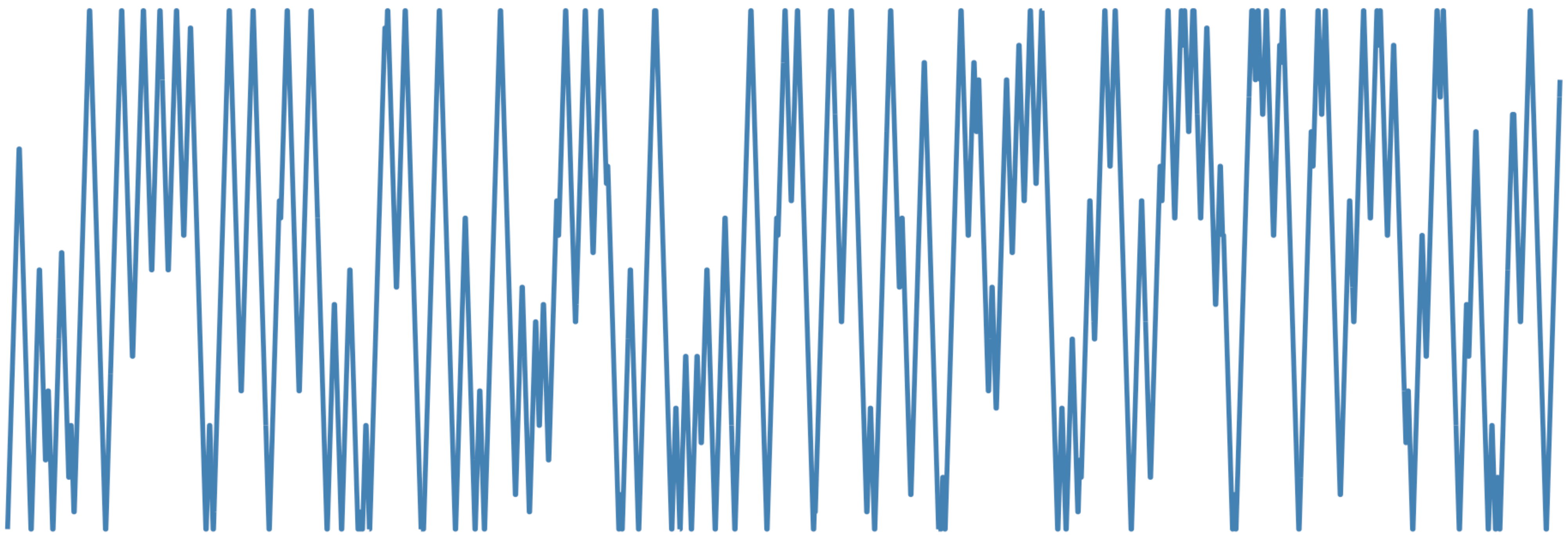
NON-REVERSIBLE PARALLEL TEMPERING ON OPTIMIZED PATHS



Saifuddin Syed

UNIVERSITY OF OXFORD

MOTIVATION

MOTIVATION

- **Problem:** Want to compute expectations

$$\mathbb{E}[f] = \int_{\mathcal{X}} f(x)\pi_1(x)\mathrm{d}x$$

- π_1 is a general probability distribution
- $\mathcal{X}$ is a general state space
- $f(x)$ is a general function

MOTIVATION

- **Problem:** Want to compute expectations

$$\mathbb{E}[f] = \int_{\mathcal{X}} f(x)\pi_1(x)\mathrm{d}x$$

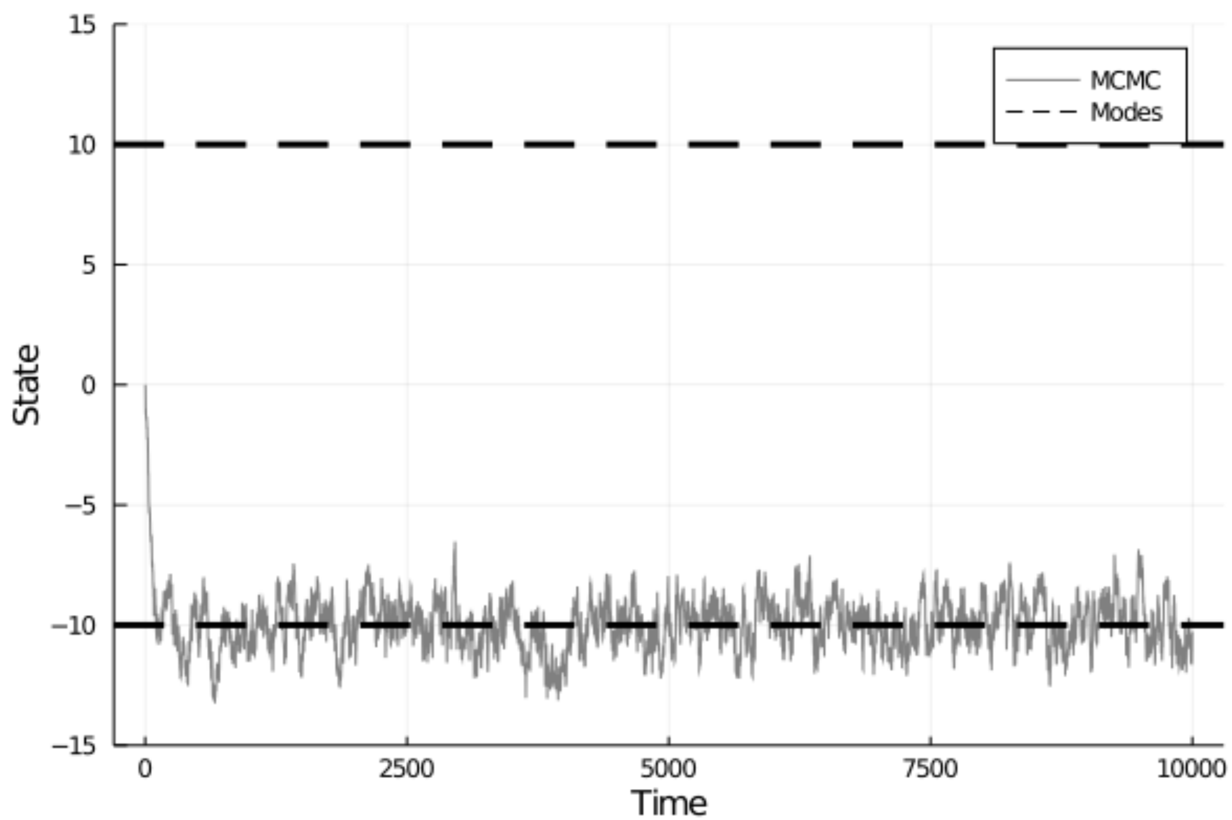
- π_1 is a general probability distribution
- $\mathcal{X}$ is a general state space
- $f(x)$ is a general function

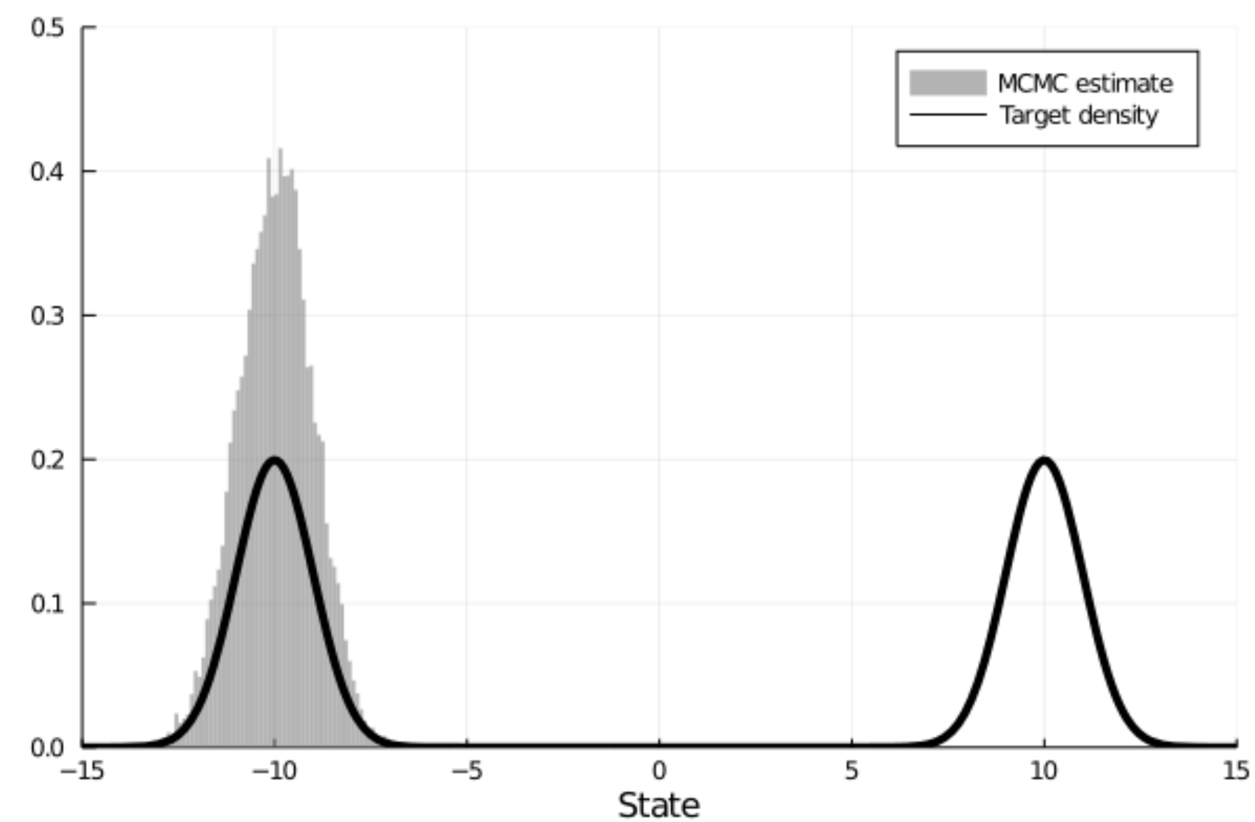
- **Solution:** Numerically approximate expectation using MCMC
- Run Markov chain stationary with respect to target and use ergodic theorem.

$$\mathbb{E}[f] = \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} f(X_t)$$

MCMC

- **Problem:** MCMC can get stuck :(
  - Chains get trapped in local regions of high probability and mix poorly
  - Worse with high dimensions, discrete spaces, constrained topologies





PROBLEMS WITH SERIAL MCMC

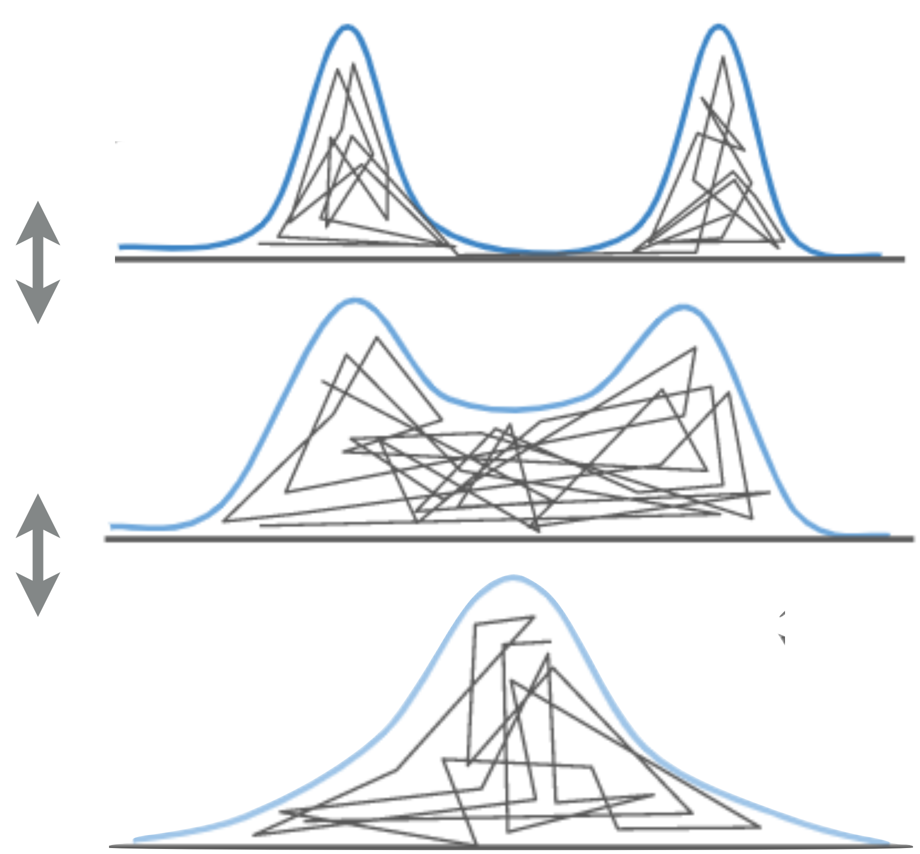
PROBLEMS WITH SERIAL MCMC

- **Hardware**: Clock speeds stagnant and modern computation is distributed
- **Algorithmic**: Single chain tries to simultaneously efficiently explore & be accurate

PROBLEMS WITH SERIAL MCMC

- **Hardware**: Clock speeds stagnant and modern computation is distributed
- **Algorithmic**: Single chain tries to simultaneously efficiently explore & be accurate

- **A Solution:** Run N additional chains in parallel
  - Delegate exploration to tractable **reference** chain
  - Communicate to **target** through the remaining $N-1$ chains



Chain N
Complex **Target**

Chains $1, \ldots, N-1$

Chain 0
Simple **Reference**
(often i.i.d. samples)

ANNEALING

ANNEALING

- π_1 **Target** density (eg. Posterior), can evaluate up to normalizing constant

ANNEALING

- π_1 **Target** density (eg. Posterior), can evaluate up to normalizing constant
- π_0 **Reference** density (eg. Prior), which is easy to sample from

ANNEALING

- π_1 **Target** density (eg. Posterior), can evaluate up to normalizing constant
- π_0 **Reference** density (eg. Prior), which is easy to sample from
- π_β **Annealing path** continuously interpolate between the reference and target

$$\pi_\beta(x) \propto \pi_0^{1-\beta}(x)\pi_1(x)^\beta$$

"Linear path"

ANNEALING

- π_1 **Target** density (eg. Posterior), can evaluate up to normalizing constant
- π_0 **Reference** density (eg. Prior), which is easy to sample from
- π_β **Annealing path** continuously interpolate between the reference and target

$$\pi_\beta(x) \propto \pi_0^{1-\beta}(x)\pi_1(x)^\beta$$

"Linear path"

- $\mathcal{B}_N$ **Annealing schedule** interpolate between the two

$$\mathcal{B}_N = (\beta_n)_{n=0}^N$$

$$0 = \beta_0 < \beta_1 < \cdots < \beta_N = 1$$

ANNEALING

- π_1 **Target** density (eg. Posterior), can evaluate up to normalizing constant
- π_0 **Reference** density (eg. Prior), which is easy to sample from
- π_β **Annealing path** continuously interpolate between the reference and target

$$\pi_\beta(x) \propto \pi_0^{1-\beta}(x)\pi_1(x)^\beta$$

"Linear path"

- $\mathcal{B}_N$ **Annealing schedule** interpolate between the two

$$\mathcal{B}_N = (\beta_n)_{n=0}^N$$

$$0 = \beta_0 < \beta_1 < \cdots < \beta_N = 1$$

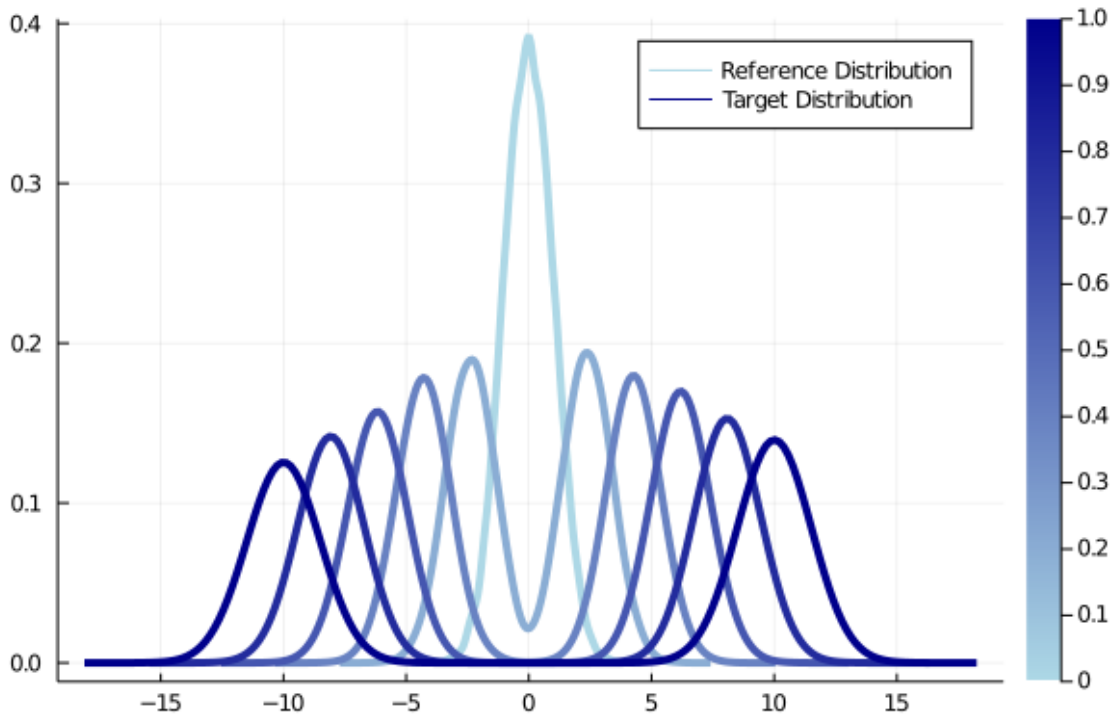
- π_n **Annealing distribution** interpolate between the two

$$\pi_n = \pi_{\beta_n}$$

ANNEALING

ANNEALING

$$\pi_{\beta_0}, \ldots, \pi_{\beta_N}$$



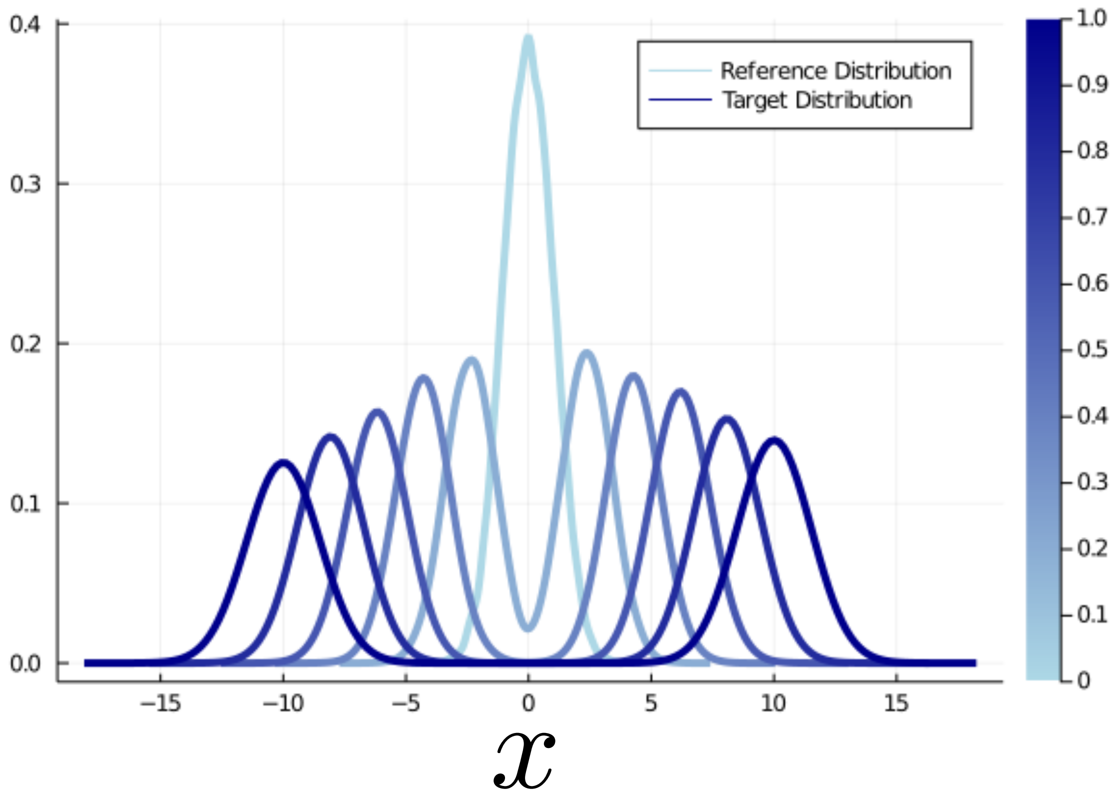
$$x$$

$$N = 5$$

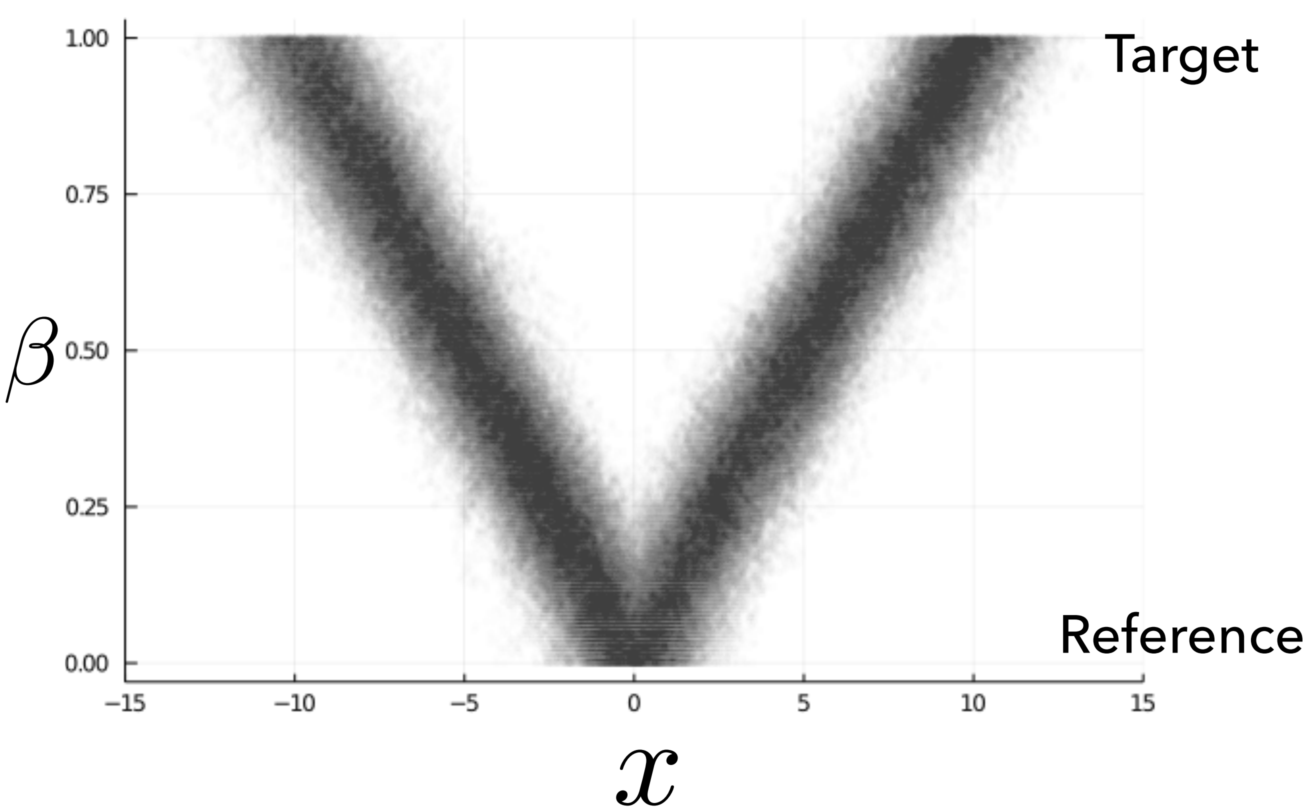
ANNEALING

$$\pi_{\beta_0}, \ldots, \pi_{\beta_N}$$

$$\pi_\beta(x) \propto \pi_0^{1-\beta}(x)\pi_1(x)^\beta$$



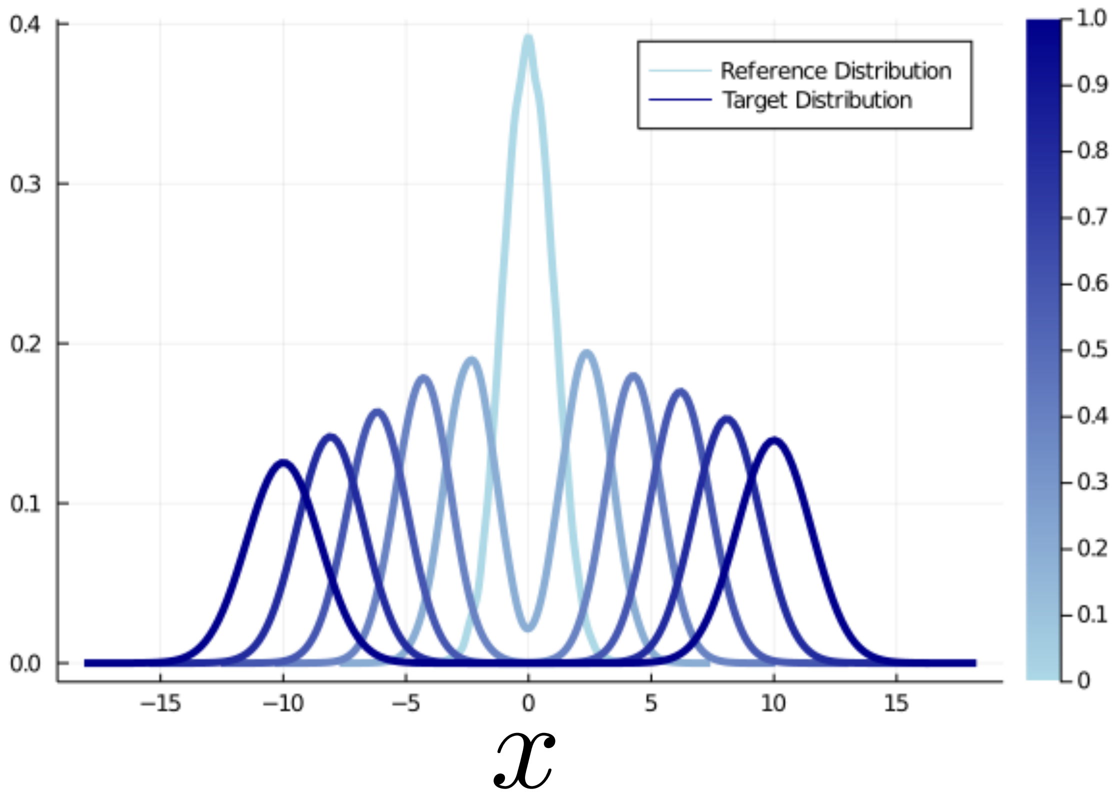
$N = 5$



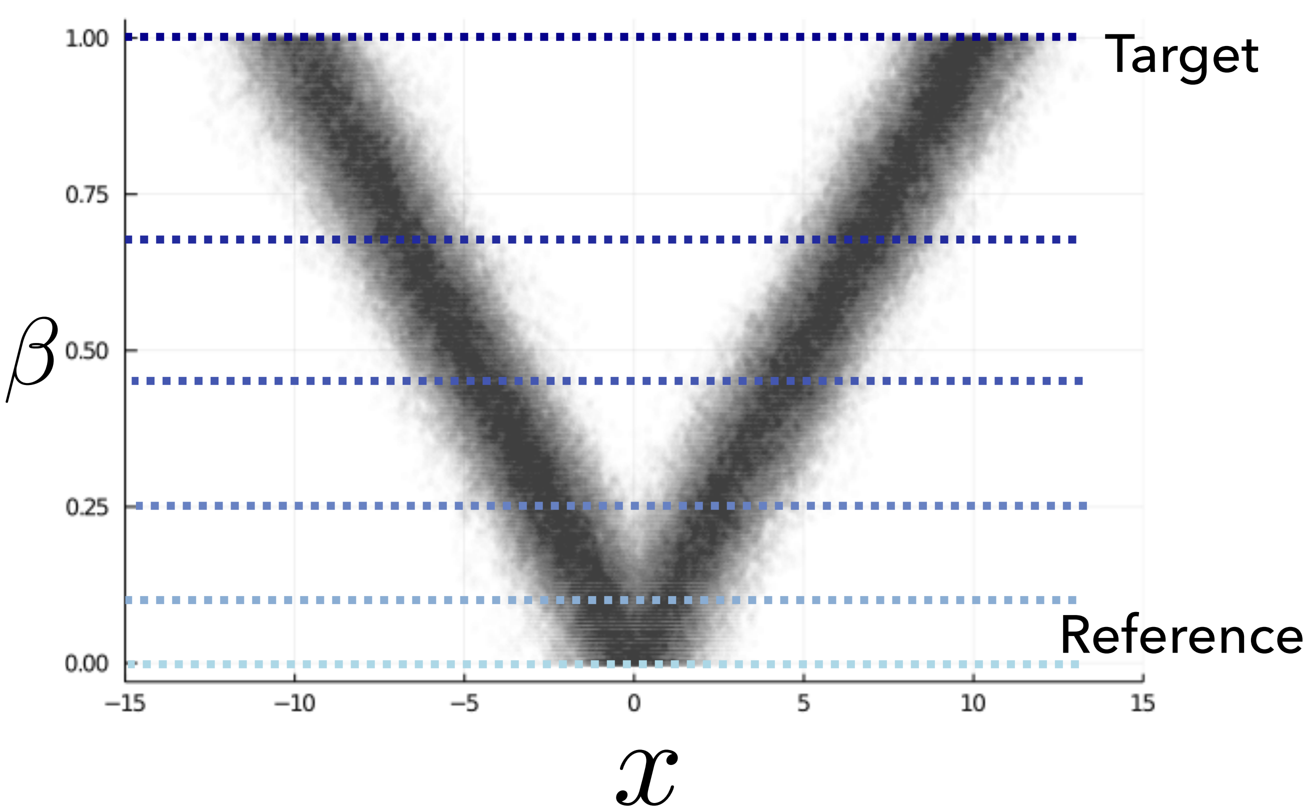
ANNEALING

$$\pi_{\beta_0}, \ldots, \pi_{\beta_N}$$

$$\pi_\beta(x) \propto \pi_0^{1-\beta}(x)\pi_1(x)^\beta$$



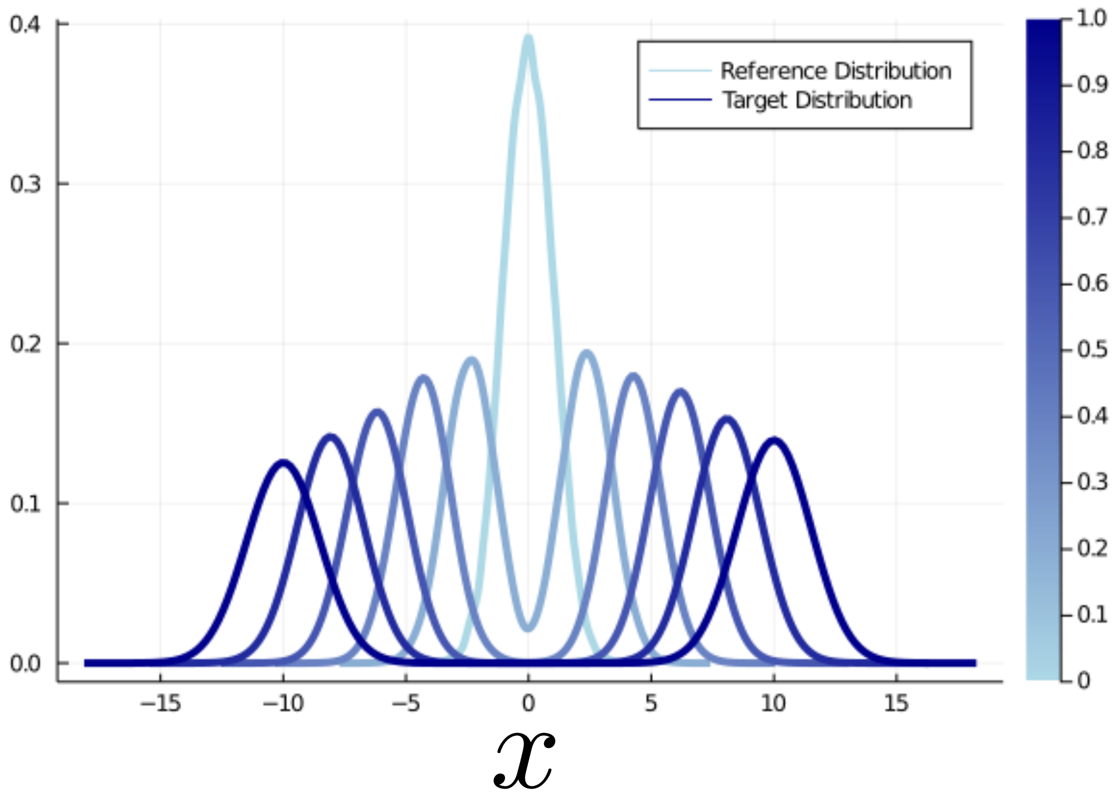
$N = 5$



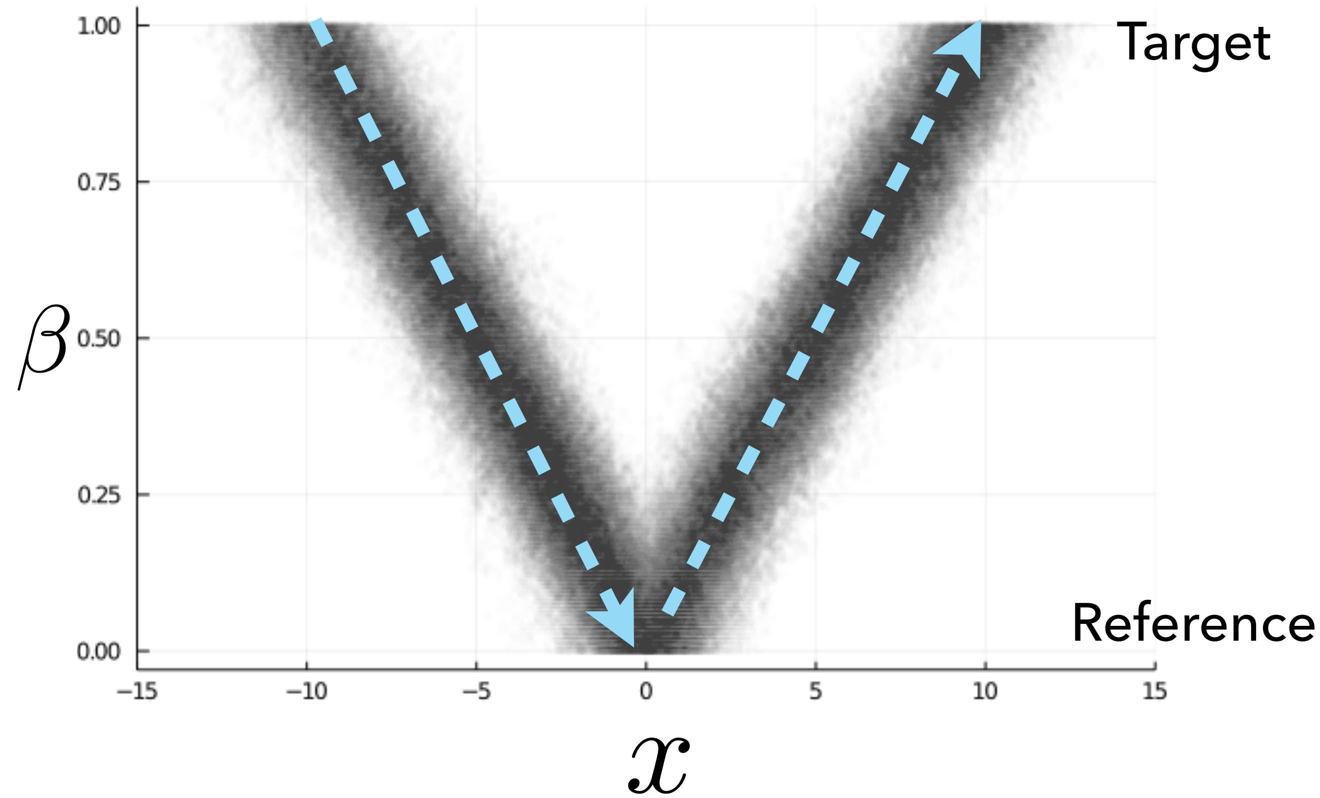
ANNEALING

$$\pi_{\beta_0}, \ldots, \pi_{\beta_N}$$

$$\pi_\beta(x) \propto \pi_0^{1-\beta}(x)\pi_1(x)^\beta$$



$N = 5$



PARALLEL TEMPERING

PARALLEL TEMPERING

- Extend state space $\mathcal{X}^{N+1}$ with join distribution

$$\boldsymbol{\pi}(\mathbf{x}) = \prod_{n=0}^{N} \pi_{\beta_n}(x^n)$$

PARALLEL TEMPERING

- Extend state space $\mathcal{X}^{N+1}$ with join distribution

$$\boldsymbol{\pi}(\mathbf{x}) = \prod_{n=0}^{N} \pi_{\beta_n}(x^n)$$

- Construct Markov Chain $\mathbf{X}_t$ targeting $\boldsymbol{\pi}$:

$$\mathbf{X}_t = (X_t^0, \ldots, X_t^N) \in \mathcal{X}^{N+1}$$

$$\mathbb{E}[f] = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} f(X_t^N)$$

PARALLEL TEMPERING

- Extend state space $\mathcal{X}^{N+1}$ with join distribution

$$\boldsymbol{\pi}(\mathbf{x}) = \prod_{n=0}^{N} \pi_{\beta_n}(x^n)$$

- Construct Markov Chain $\mathbf{X}_t$ targeting $\boldsymbol{\pi}$:

$$\mathbf{X}_t = (X_t^0, \ldots, X_t^N) \in \mathcal{X}^{N+1}$$

$$\mathbb{E}[f] = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} f(X_t^N)$$

1. **Local exploration:** Update each component with MCMC move
    - I.e. For component n apply MCMC kernel targeting π_{β_n}

PARALLEL TEMPERING

- Extend state space $\mathcal{X}^{N+1}$ with join distribution

$$\boldsymbol{\pi}(\mathbf{x}) = \prod_{n=0}^{N} \pi_{\beta_n}(x^n)$$

- Construct Markov Chain $\mathbf{X}_t$ targeting $\boldsymbol{\pi}$:

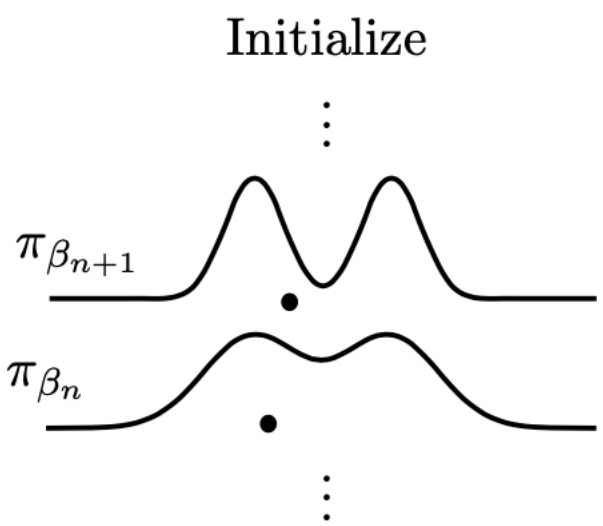
$$\mathbf{X}_t = (X_t^0, \ldots, X_t^N) \in \mathcal{X}^{N+1}$$

$$\mathbb{E}[f] = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} f(X_t^N)$$

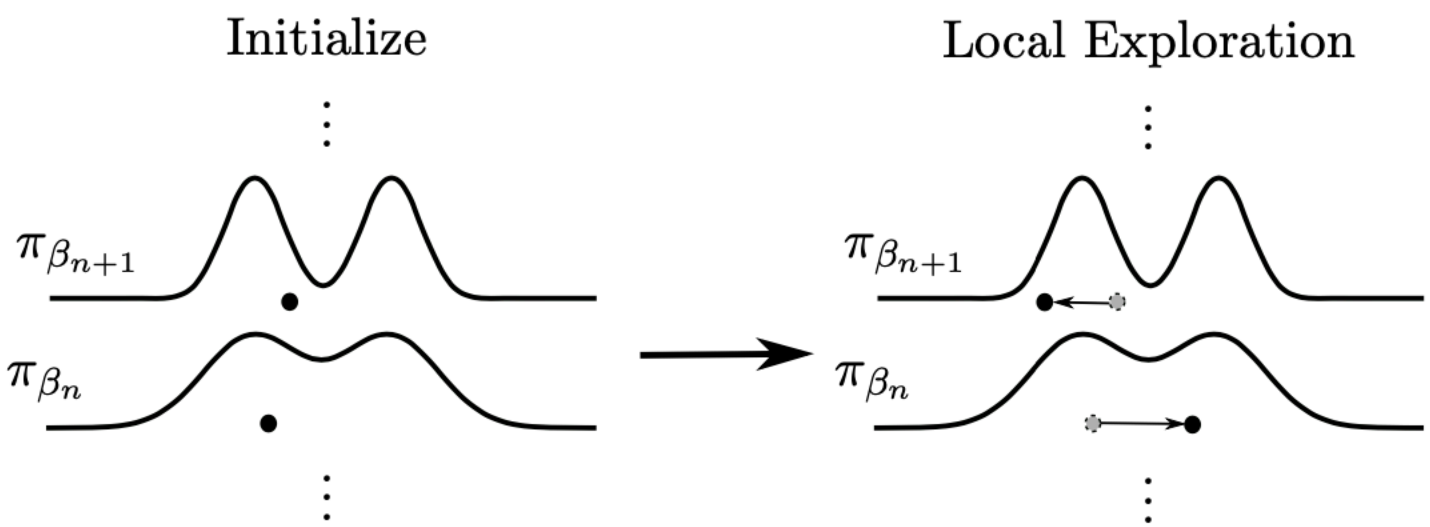
1. **Local exploration:** Update each component with MCMC move
    - I.e. For component n apply MCMC kernel targeting π_{β_n}
2. **Communication:** propose swaps for adjacent component
    - Accept/reject according to Metropolis-Hastings:

PARALLEL TEMPERING

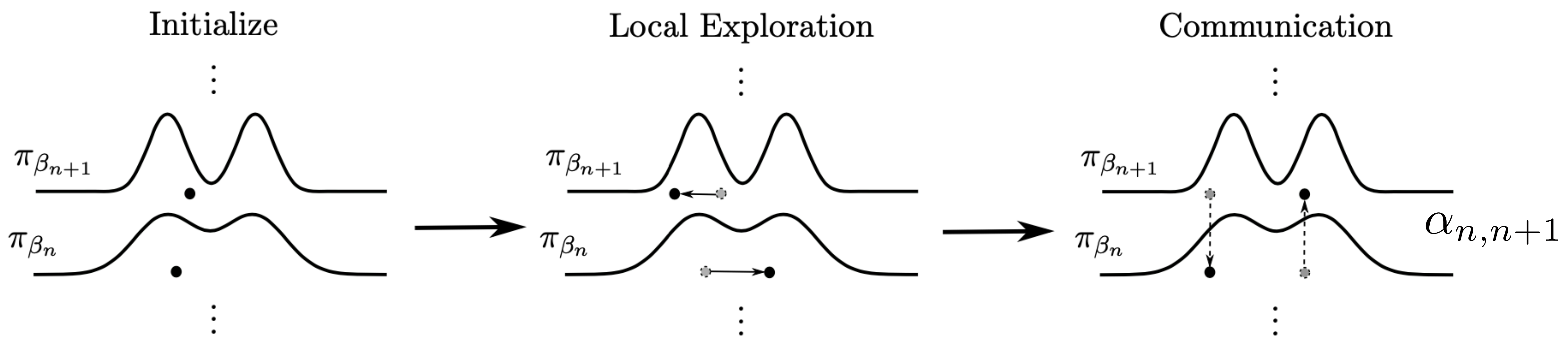
PARALLEL TEMPERING



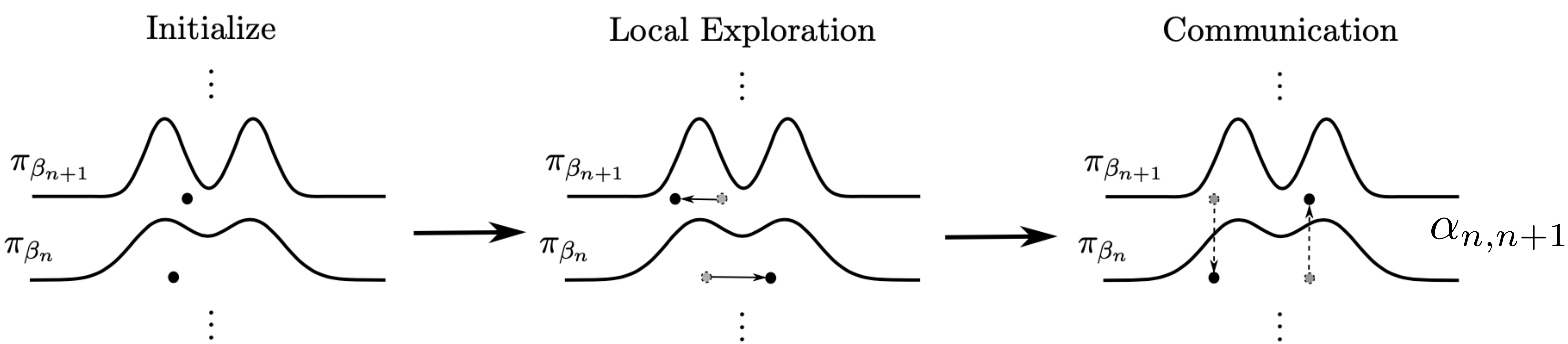
PARALLEL TEMPERING



PARALLEL TEMPERING



PARALLEL TEMPERING



- Acceptance probability:

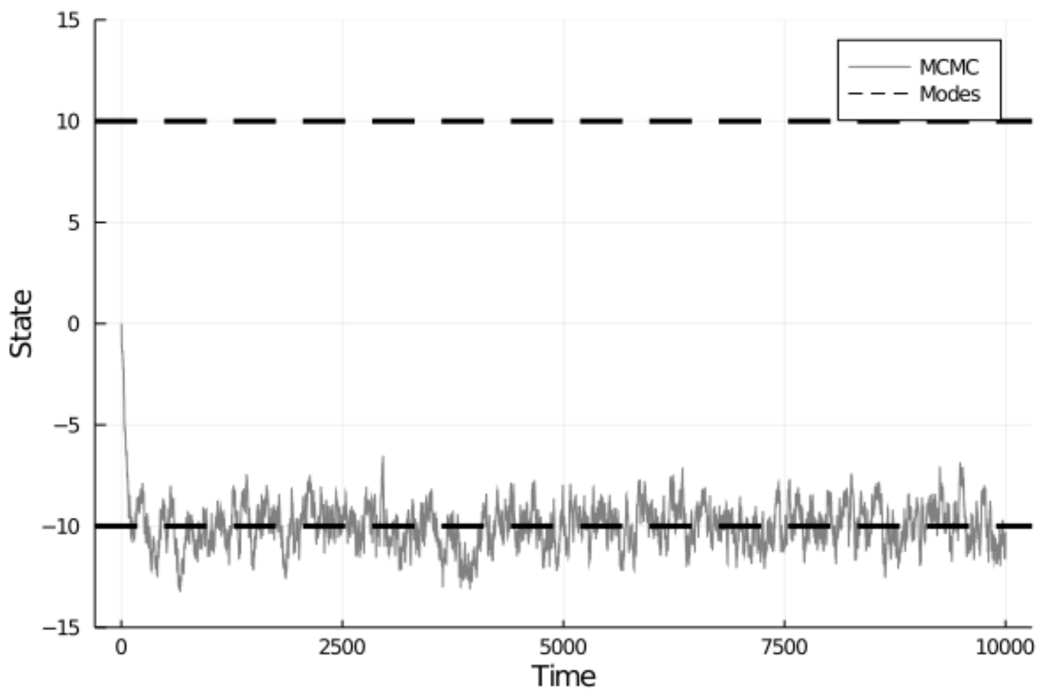
$$\alpha_{n,n+1} = 1 \wedge \frac{\pi_n(x^{n+1})\pi_{n+1}(x^n)}{\pi_n(x^n)\pi_{n+1}(x^{n+1})}$$

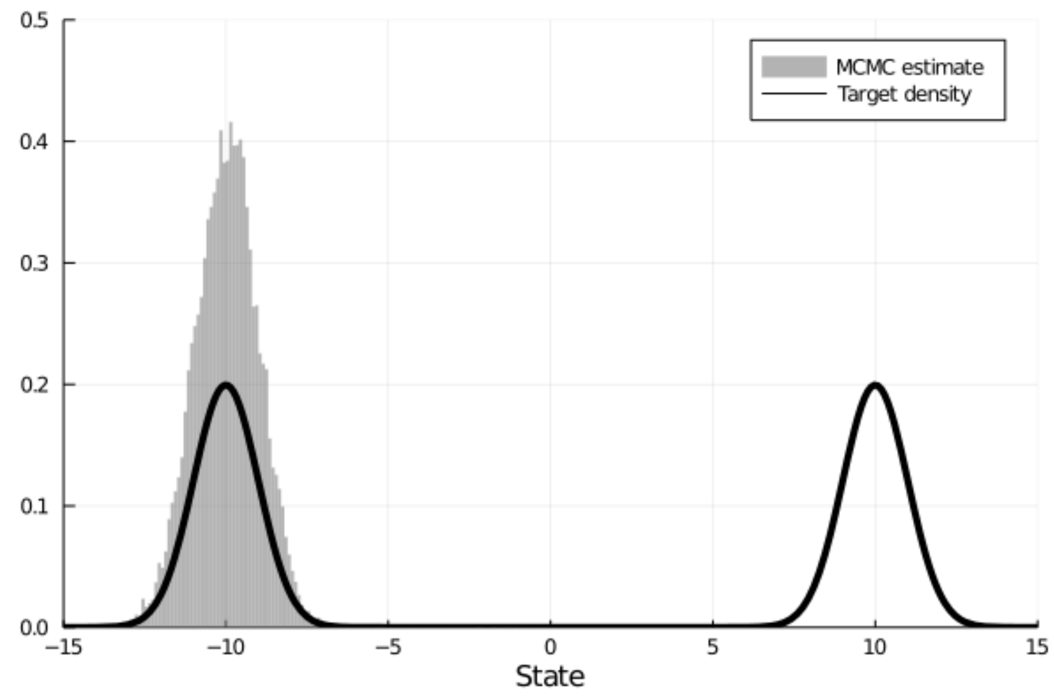
  - Notice swap probability only depends on likelihood ratio of states
  - Agnostic to location of modes

MCMC VS PARALLEL TEMPERING

MCMC VS PARALLEL TEMPERING

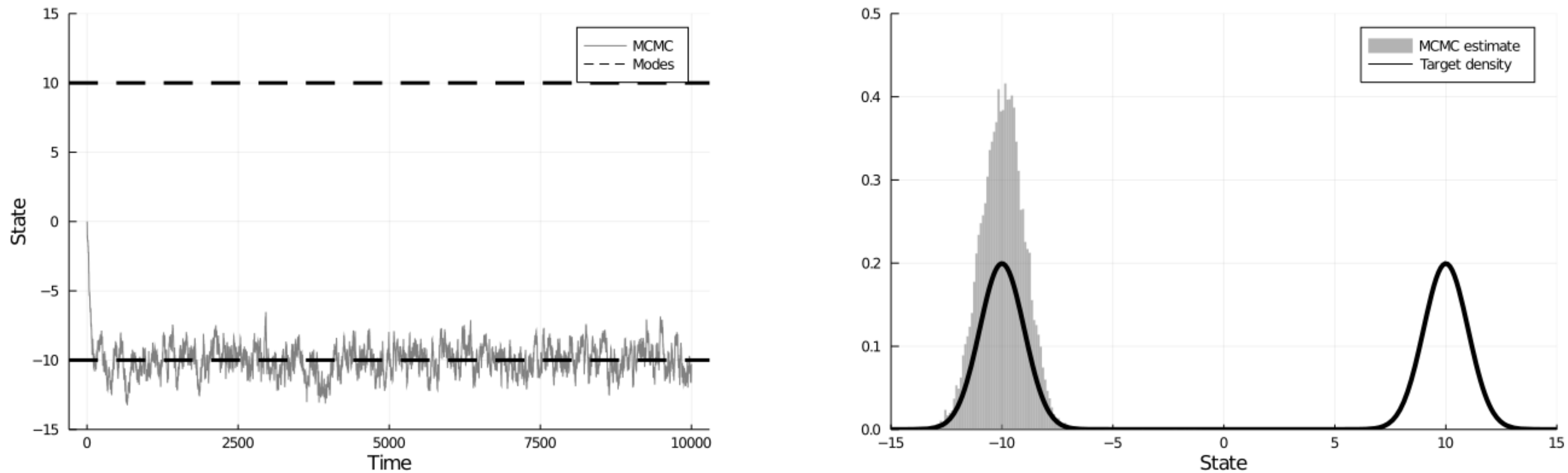
Single chain MCMC



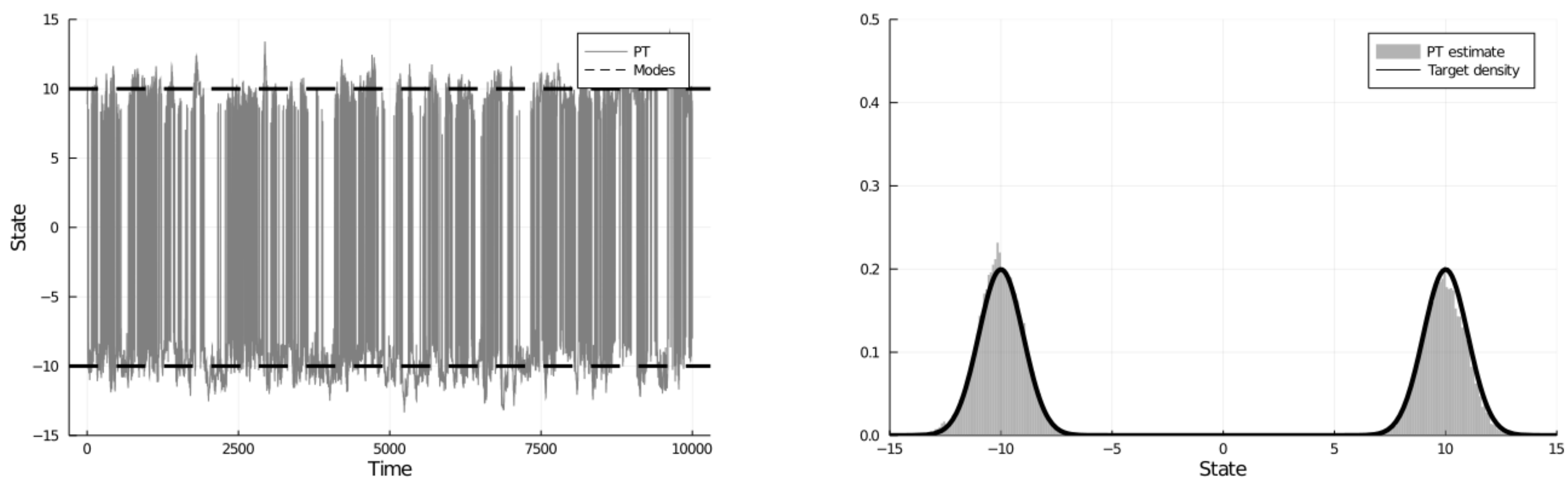


MCMC VS PARALLEL TEMPERING

Single chain MCMC



Parallel Tempering (N = 10)



TUNING PT

TUNING PT

- How to propose swaps?
- What is an optimal schedule?
- How to tune the schedule?
- How to pick number of chains?
- How to pick path?
- Hard problem or bad implementation?

TUNING PT

- How to propose swaps? At random (reversible)
- What is an optimal schedule?
- How to tune the schedule?
- How to pick number of chains?
- How to pick path?
- Hard problem or bad implementation?

TUNING PT

- How to propose swaps? At random (reversible)
- What is an optimal schedule? Reject 77% of swaps (e.g. Atchadé, 2011)
- How to tune the schedule?
- How to pick number of chains?
- How to pick path?
- Hard problem or bad implementation?

TUNING PT

- How to propose swaps? At random (reversible)
- What is an optimal schedule? Reject 77% of swaps (e.g. Atchadé, 2011)
- How to tune the schedule? Stochastic optimization (e.g. Miasojedow, 2013)
- How to pick number of chains? $N \approx 10$ or adaptively reduce
- How to pick path?
- Hard problem or bad implementation?

TUNING PT

- How to propose swaps? At random (reversible)
- What is an optimal schedule? Reject 77% of swaps (e.g. Atchadé, 2011)
- How to tune the schedule? Stochastic optimization (e.g. Miasojedow, 2013)
- How to pick number of chains? $N \approx 10$ or adaptively reduce
- How to pick path? Linear path : $\pi_\beta \propto \pi_0^{1-\beta} \pi_1^{\beta}$
- Hard problem or bad implementation?

TUNING PT

- How to propose swaps? At random (reversible)
- What is an optimal schedule? Reject 77% of swaps (e.g. Atchadé, 2011)
- How to tune the schedule? Stochastic optimization (e.g. Miasojedow, 2013)
- How to pick number of chains? $N \approx 10$ or adaptively reduce
- How to pick path? Linear path : $\pi_\beta \propto \pi_0^{1-\beta}\pi_1^{\beta}$
- Hard problem or bad implementation?

OUTLINE

OUTLINE

- **Non-reversible parallel tempering (JRSS-B, 2021)***
  - Introduce non-reversible PT + scaling + tuning + diagnostics

Alexandre Bouchard-Côté*†

George Deligiannidis*

Arnaud Doucet*

OUTLINE

- **Non-reversible parallel tempering (JRSS-B, 2021)***
  - Introduce non-reversible PT + scaling + tuning + diagnostics
- **Parallel tempering on optimized paths (ICML, 2021)†**
  - Introduce geometry of PT + path tuning.

Alexandre Bouchard-Côté*†

George Deligiannidis*

Arnaud Doucet*

Vittorio Romaniello†

Trevor Campbell†

DISTRIBUTED PT

DISTRIBUTED PT

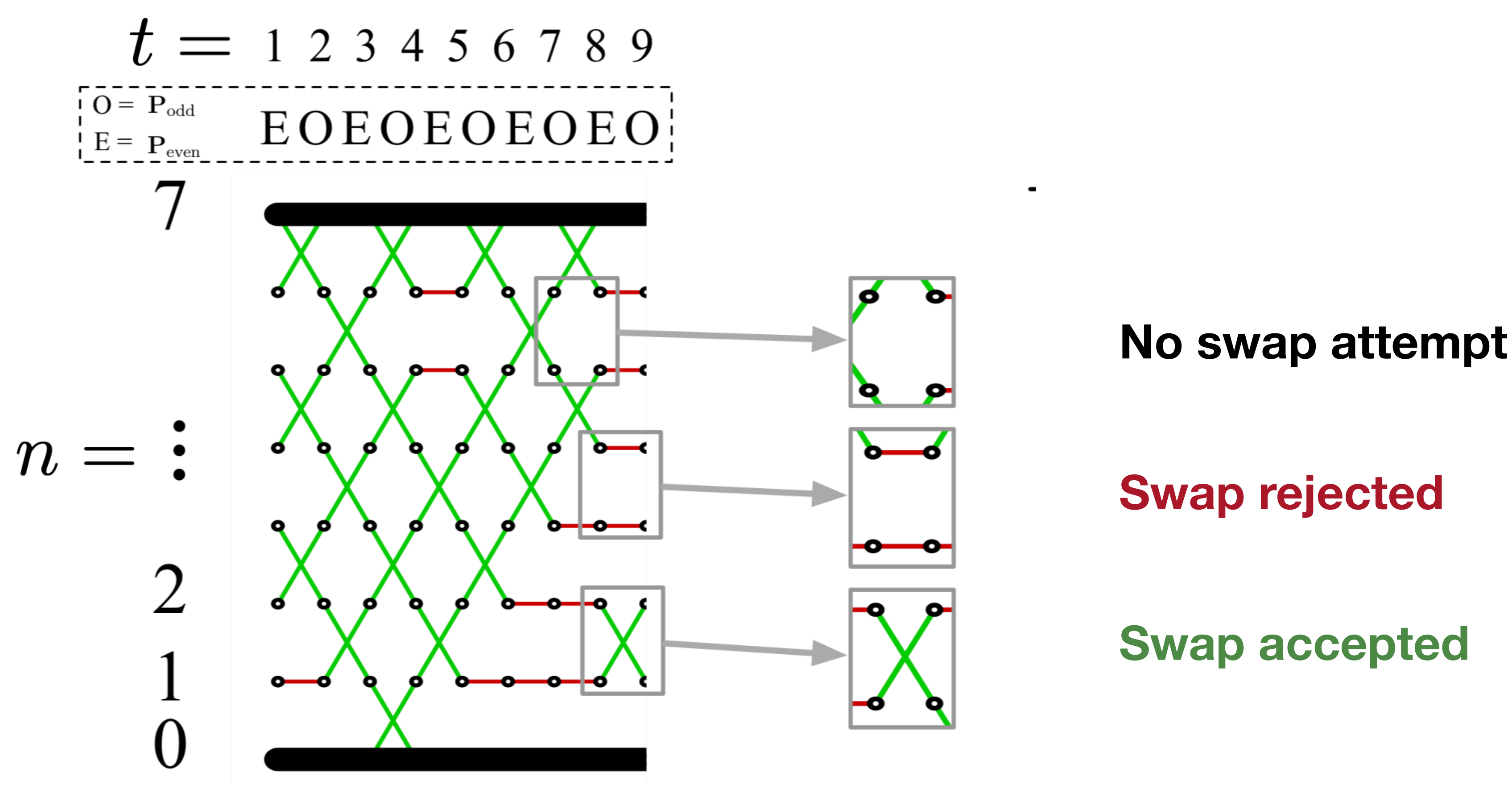
- **Dual perspective on PT:** Swap annealing parameters not states!

DISTRIBUTED PT

- **Dual perspective on PT:** Swap annealing parameters not states!
  - More efficient for distributed implementation to swap numbers

DISTRIBUTED PT

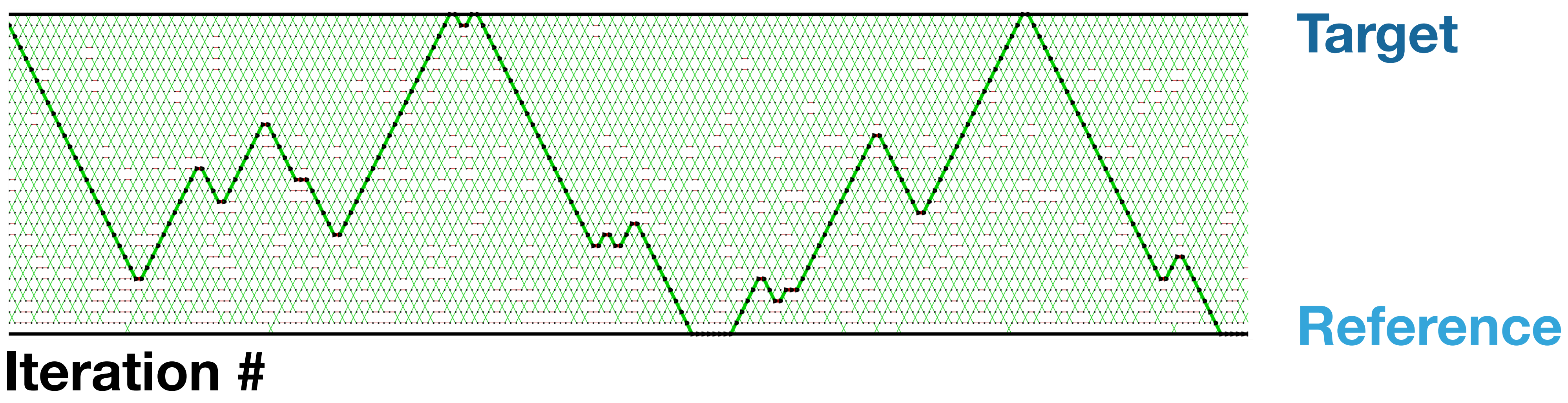
- **Dual perspective on PT:** Swap annealing parameters not states!
  - More efficient for distributed implementation to swap numbers
  - We can study communication through the dynamics of the permuted annealing parameters



OBJECTIVE OF PT

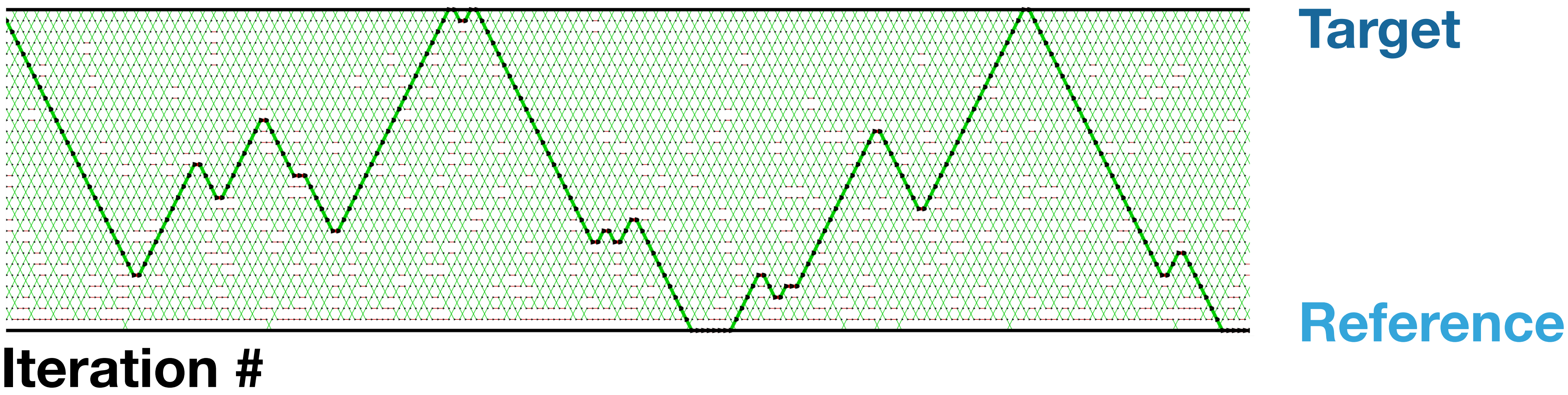
OBJECTIVE OF PT

- **Round Trip:** when a state travels from **reference** to **target** and back



OBJECTIVE OF PT

- **Round Trip:** when a state travels from **reference** to **target** and back

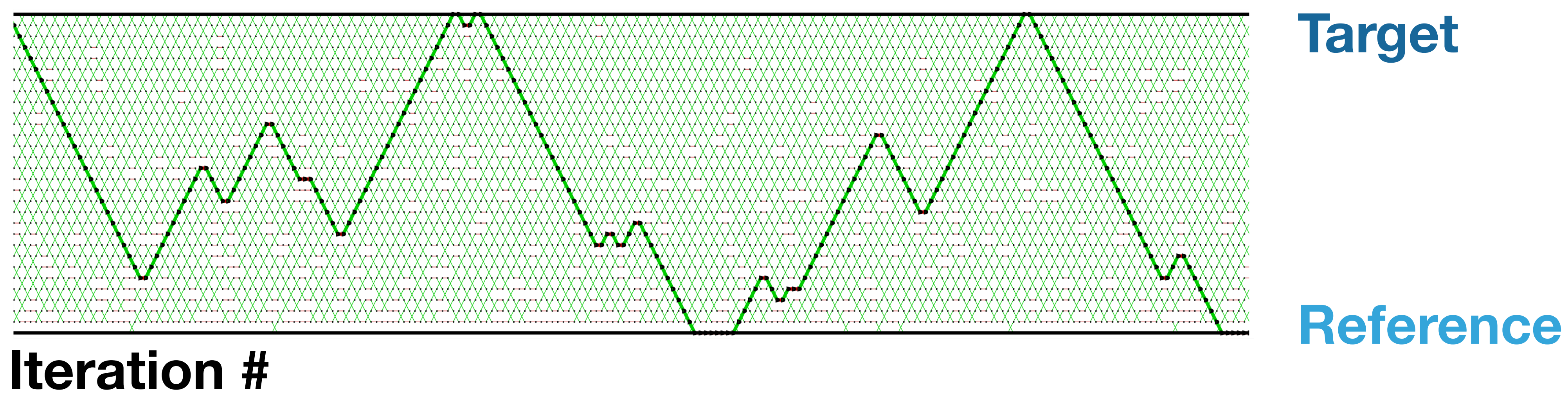


- **Round trip rate** % of iterations with round trip

$$\tau_N = \lim_{t \to \infty} \frac{\mathbb{E}[\text{total round trips by time t}]}{t}$$

OBJECTIVE OF PT

- **Round Trip:** when a state travels from **reference** to **target** and back



- **Round trip rate** % of iterations with round trip

$$\tau_N = \lim_{t \to \infty} \frac{\mathbb{E}[\text{total round trips by time t}]}{t}$$

- Objective is to maximize round trip rate

REVERSIBLE PT

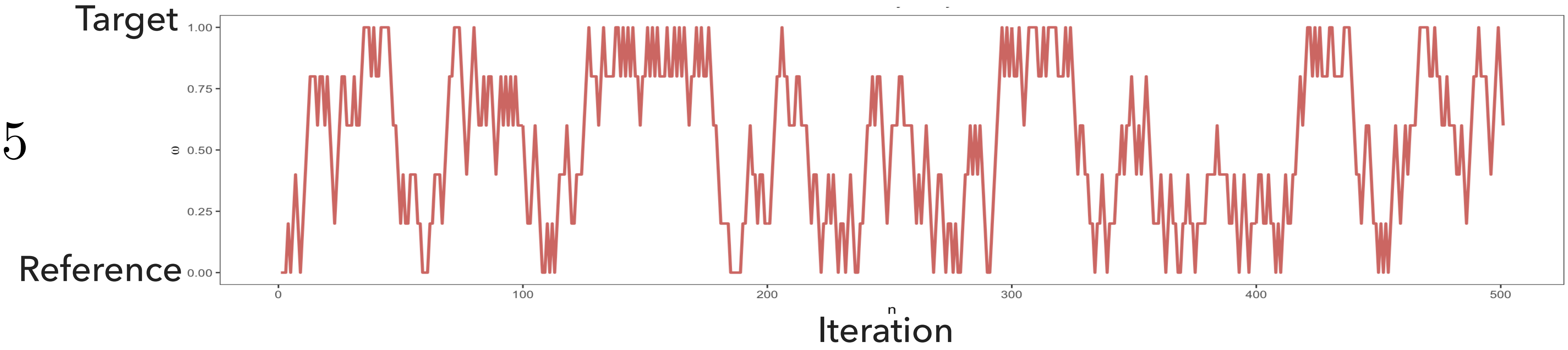
REVERSIBLE PT

- Propose swaps at random in communication phase

REVERSIBLE PT

- Propose swaps at random in communication phase

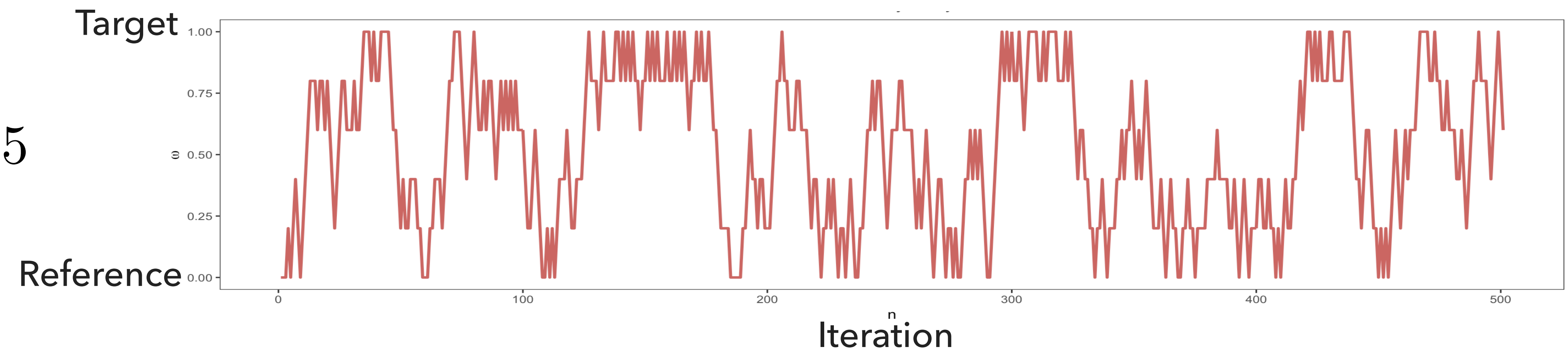
$$N = 5$$



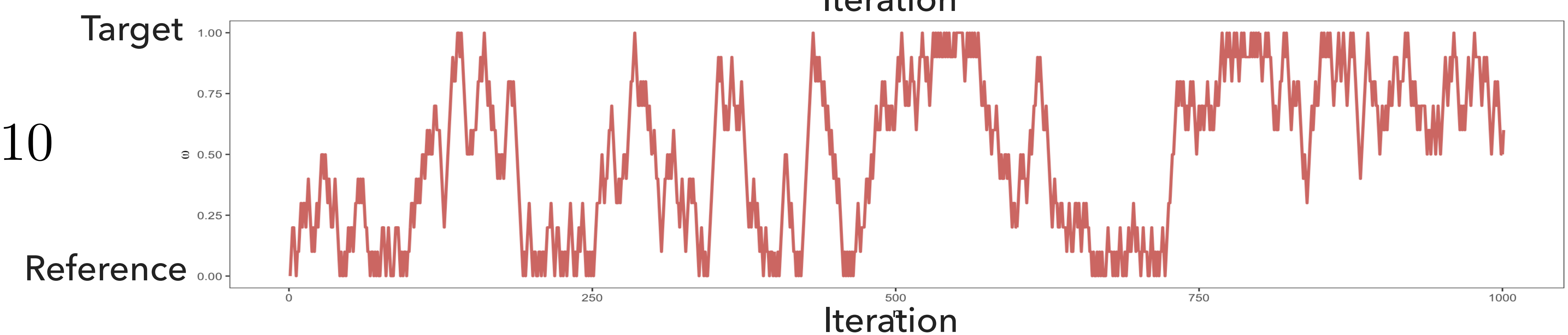
REVERSIBLE PT

- Propose swaps at random in communication phase

$N = 5$



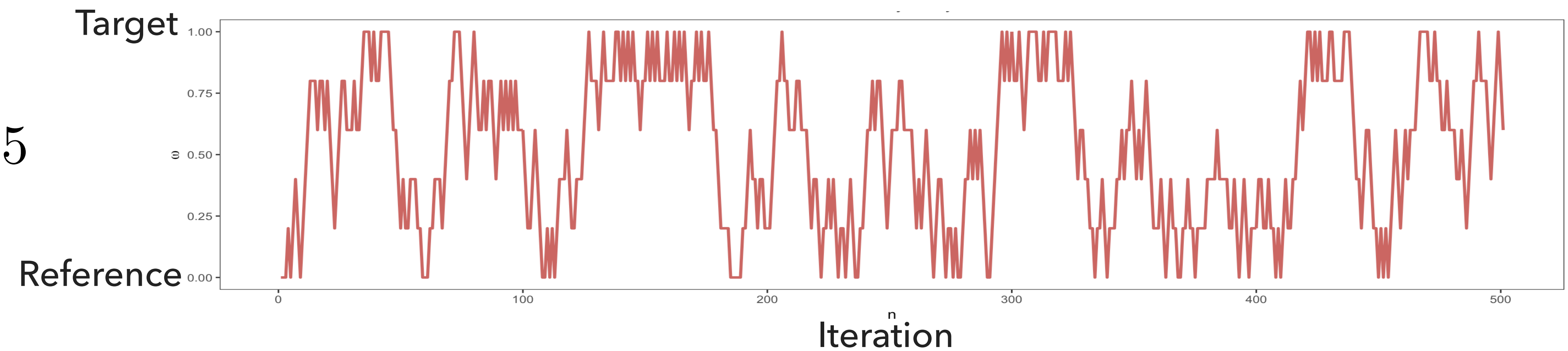
$N = 10$



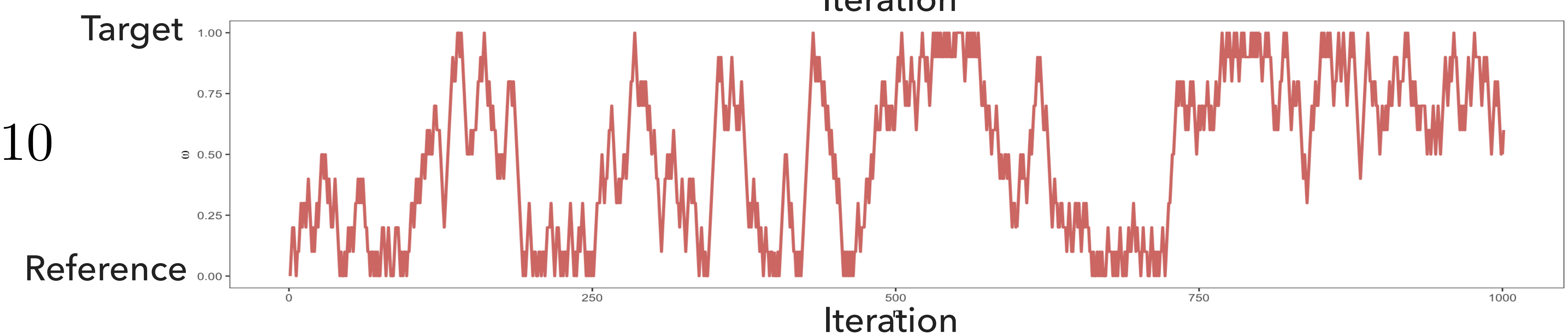
REVERSIBLE PT

- Propose swaps at random in communication phase

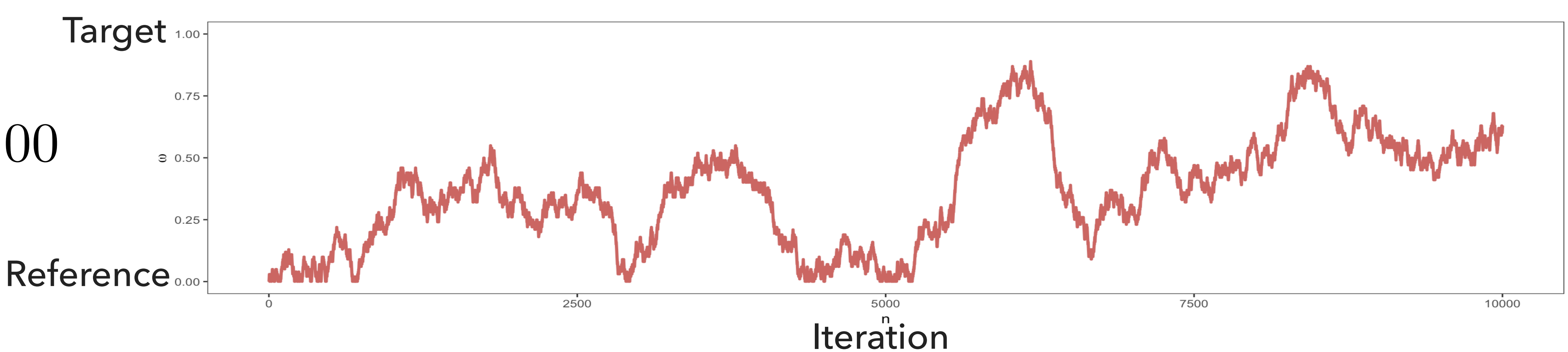
$N = 5$



$N = 10$



$N = 100$



REVERSIBLE PT

Theorem: For any annealing schedule

$$\tau_N = \left(2N + 2\sum_{n=0}^{N-1} \frac{r_n}{1-r_n}\right)^{-1}$$

where r_n is the probability swap is rejected between chain n and $n+1$

REVERSIBLE PT

Theorem: For any annealing schedule

$$\tau_N = \left(2N + 2\sum_{n=0}^{N-1} \frac{r_n}{1 - r_n}\right)^{-1}$$

where r_n is the probability swap is rejected between chain n and $n+1$

- Reversible PT deteriorates with too many chains

$$\tau_N \leq \frac{1}{2N}$$

REVERSIBLE PT

Theorem: For any annealing schedule

$$\tau_N = \left(2N + 2\sum_{n=0}^{N-1} \frac{r_n}{1-r_n}\right)^{-1}$$

where r_n is the probability swap is rejected between chain n and $n+1$

- Reversible PT deteriorates with too many chains

$$\tau_N \leq \frac{1}{2N}$$

- Very sensitive to schedule and number of chains

REVERSIBLE PT

Theorem: For any annealing schedule

$$\tau_N = \left(2N + 2\sum_{n=0}^{N-1} \frac{r_n}{1-r_n}\right)^{-1}$$

where r_n is the probability swap is rejected between chain n and $n+1$

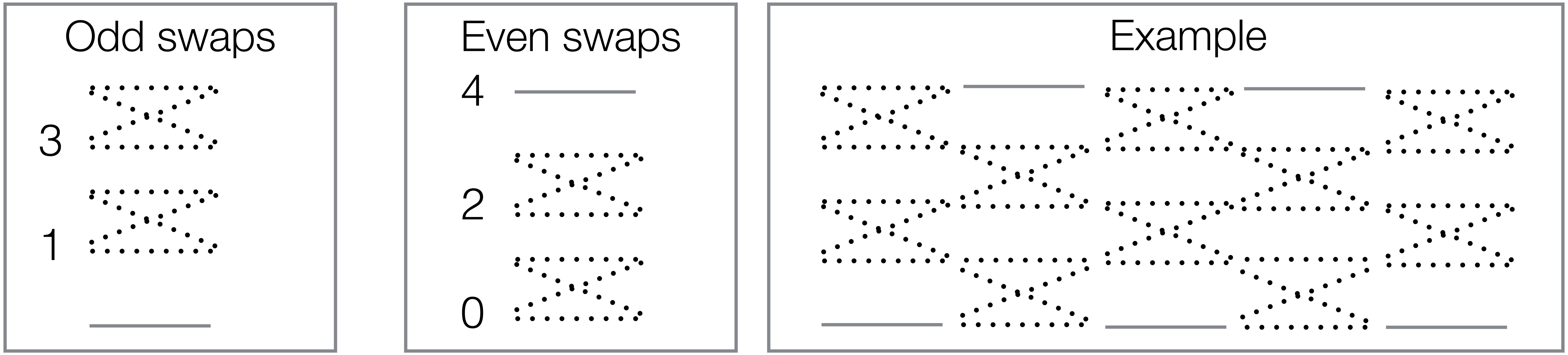
- Reversible PT deteriorates with too many chains

$$\tau_N \leq \frac{1}{2N}$$

- Very sensitive to schedule and number of chains
- Not scalable to GPUs

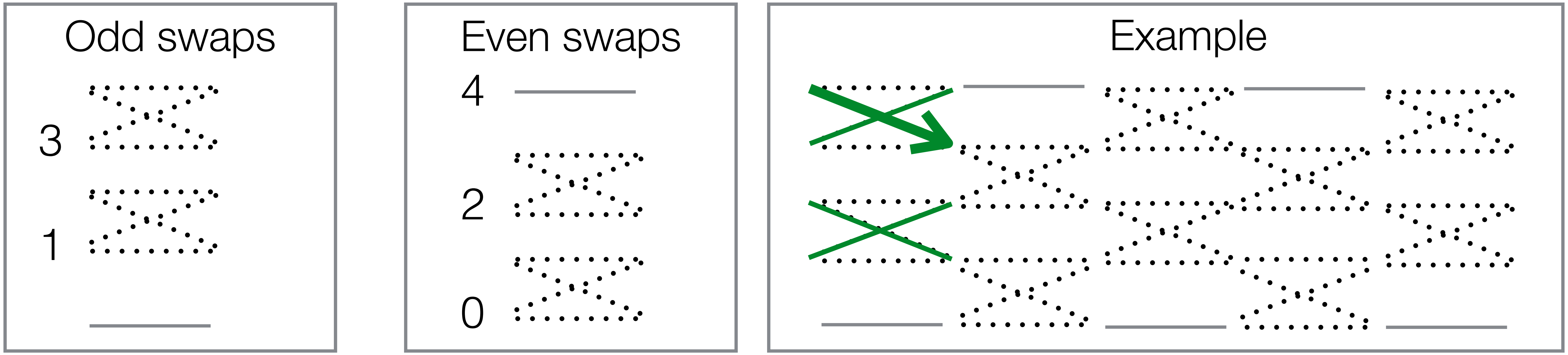
NON-REVERSIBLE PT (OKABE ET AL, 2001)

- Deterministically alternate between **Odd** and **Even** swaps



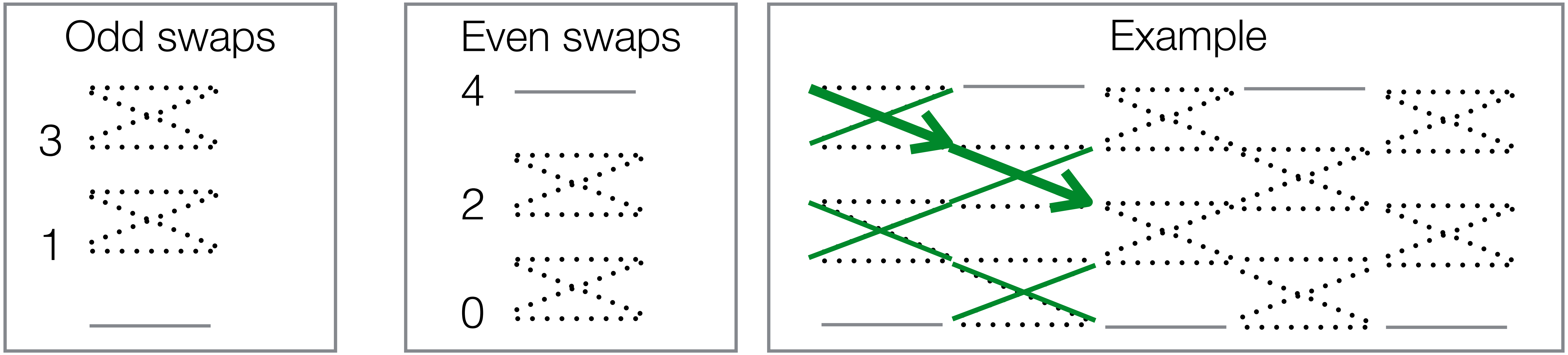
NON-REVERSIBLE PT (OKABE ET AL, 2001)

- Deterministically alternate between **Odd** and **Even** swaps



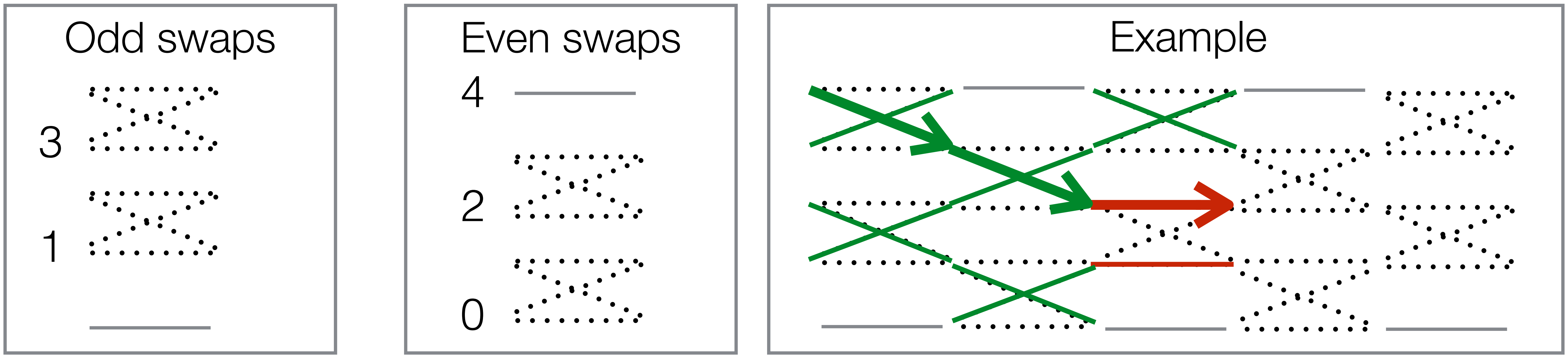
NON-REVERSIBLE PT (OKABE ET AL, 2001)

- Deterministically alternate between **Odd** and **Even** swaps



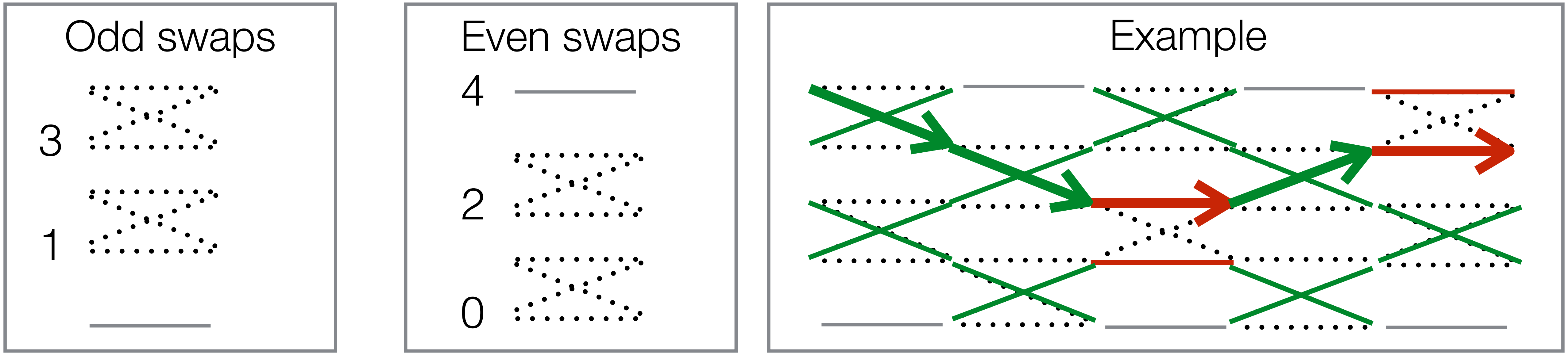
NON-REVERSIBLE PT (OKABE ET AL, 2001)

- Deterministically alternate between **Odd** and **Even** swaps



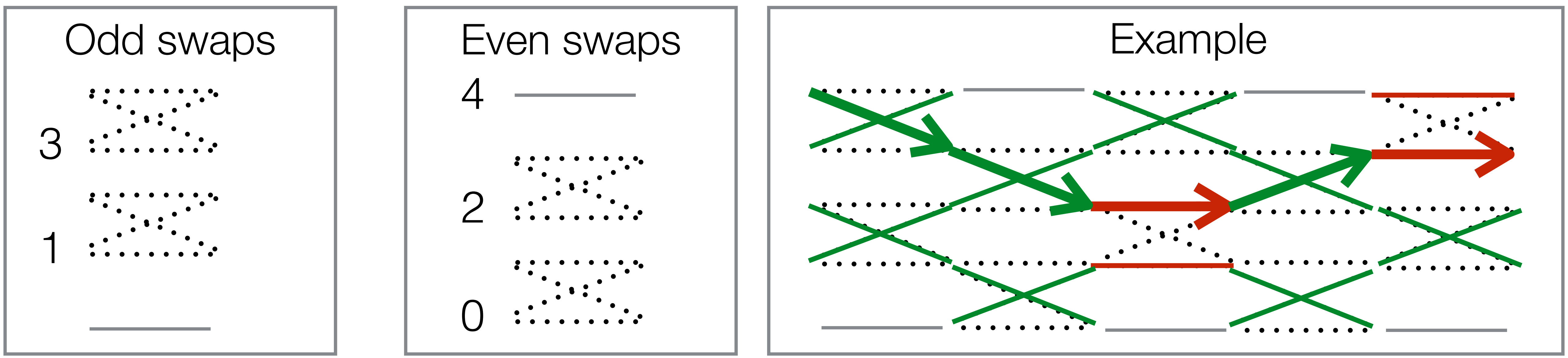
NON-REVERSIBLE PT (OKABE ET AL, 2001)

- Deterministically alternate between **Odd** and **Even** swaps



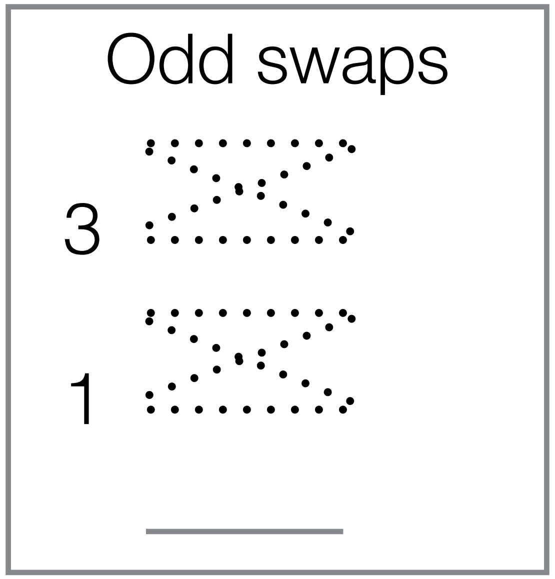
NON-REVERSIBLE PT (OKABE ET AL, 2001)

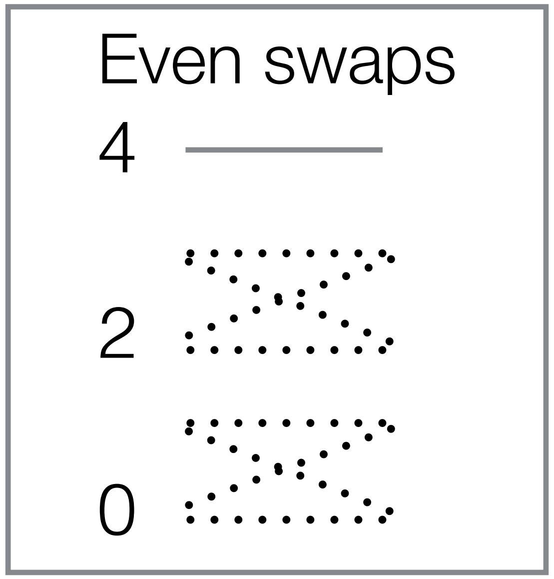
- Deterministically alternate between **Odd** and **Even** swaps
- Non-reversible PT trajectories have momentum!

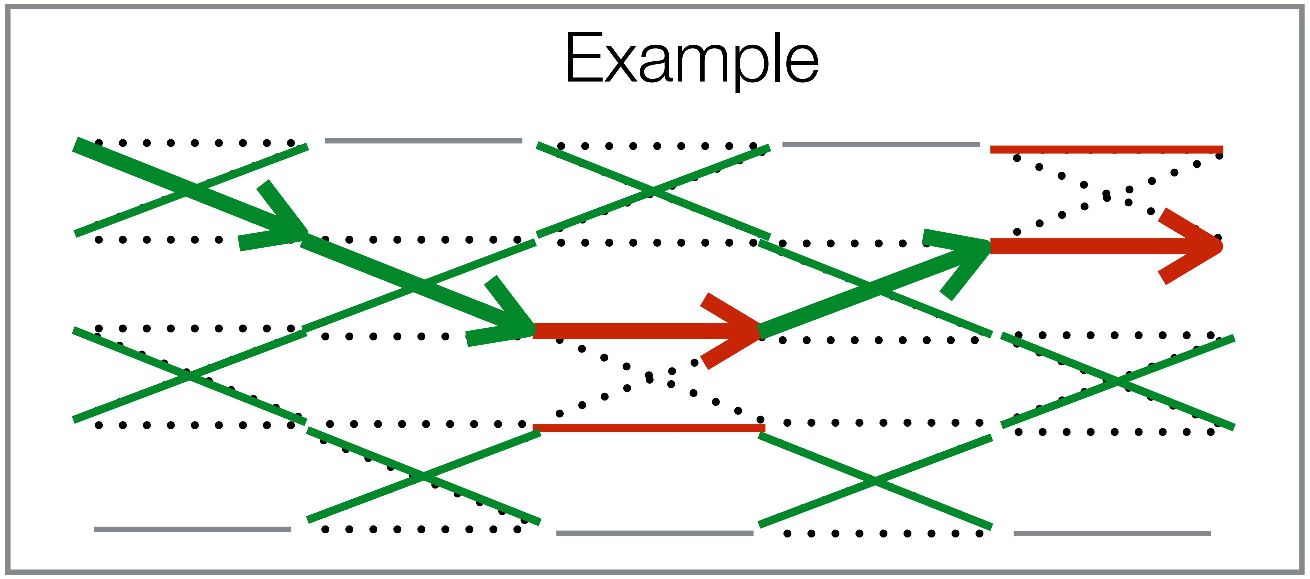


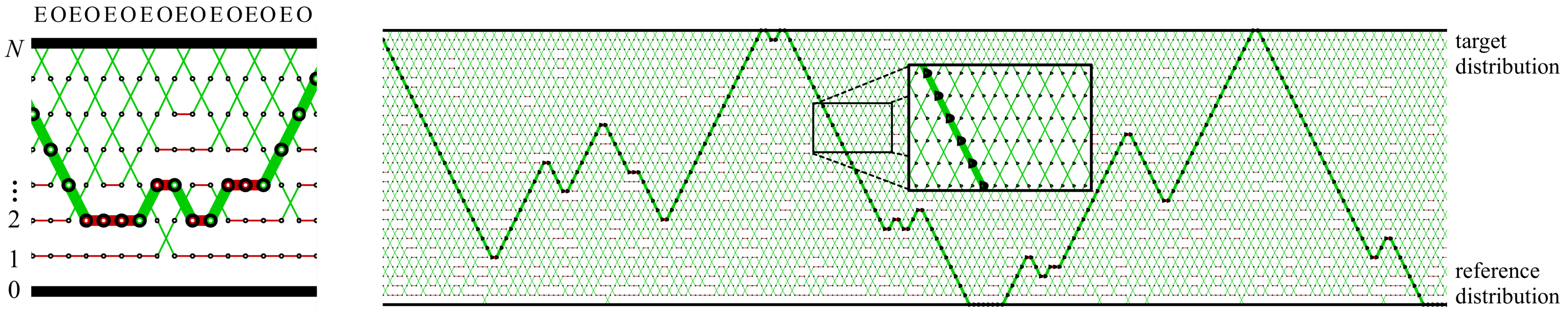
NON-REVERSIBLE PT (OKABE ET AL, 2001)

- Deterministically alternate between **Odd** and **Even** swaps
- Non-reversible PT trajectories have momentum!









NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule

$$\tau_N = \left(2 + 2\sum_{n=0}^{N-1} \frac{r_n}{1 - r_n}\right)^{-1}$$

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule

$$\tau_N = \left(2 + 2\sum_{n=0}^{N-1} \frac{r_n}{1-r_n}\right)^{-1}$$

(b) As $N \to \infty$ and $\max_{n \leq N} |\beta_n - \beta_{n-1}| \to 0$,

$$\tau_N \longrightarrow \frac{1}{2+2\Lambda}$$

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule

$$\tau_N = \left(2 + 2\sum_{n=0}^{N-1} \frac{r_n}{1 - r_n}\right)^{-1}$$

$2N$

(b) As $N \to \infty$ and $\max_{n \le N} |\beta_n - \beta_{n-1}| \to 0$,

$$\tau_N \longrightarrow \frac{1}{2 + 2\Lambda}$$

- Non-asymptotically dominates **reversible PT**

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule

$$\tau_N = \left(2 + 2\sum_{n=0}^{N-1} \frac{r_n}{1 - r_n}\right)^{-1}$$

$2N$

(b) As $N \to \infty$ and $\max_{n \leq N} |\beta_n - \beta_{n-1}| \to 0,$

$$\tau_N \longrightarrow \frac{1}{2 + 2\Lambda}$$

"Communication barrier"

- Non-asymptotically dominates **reversible PT**
- Improves with more chains (with marginal gains) controlled by **communication barrier**

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule

$$\tau_N = \left(\underset{2N}{2} + 2 \sum_{n=0}^{N-1} \frac{r_n}{1 - r_n} \right)^{-1}$$

(b) As $N \to \infty$ and $\max_{n \leq N} |\beta_n - \beta_{n-1}| \to 0$,

$$\tau_N \longrightarrow \frac{1}{2 + 2\Lambda}$$

"Communication barrier"

- Non-asymptotically dominates **reversible PT**
- Improves with more chains (with marginal gains) controlled by **communication barrier**
- Robust to schedule

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule

$$\tau_N = \left(\underset{2N}{2} + 2\sum_{n=0}^{N-1} \frac{r_n}{1 - r_n}\right)^{-1}$$

(b) As $N \to \infty$ and $\max_{n \le N} |\beta_n - \beta_{n-1}| \to 0$,

$$\tau_N \longrightarrow \frac{1}{2 + 2\Lambda}$$

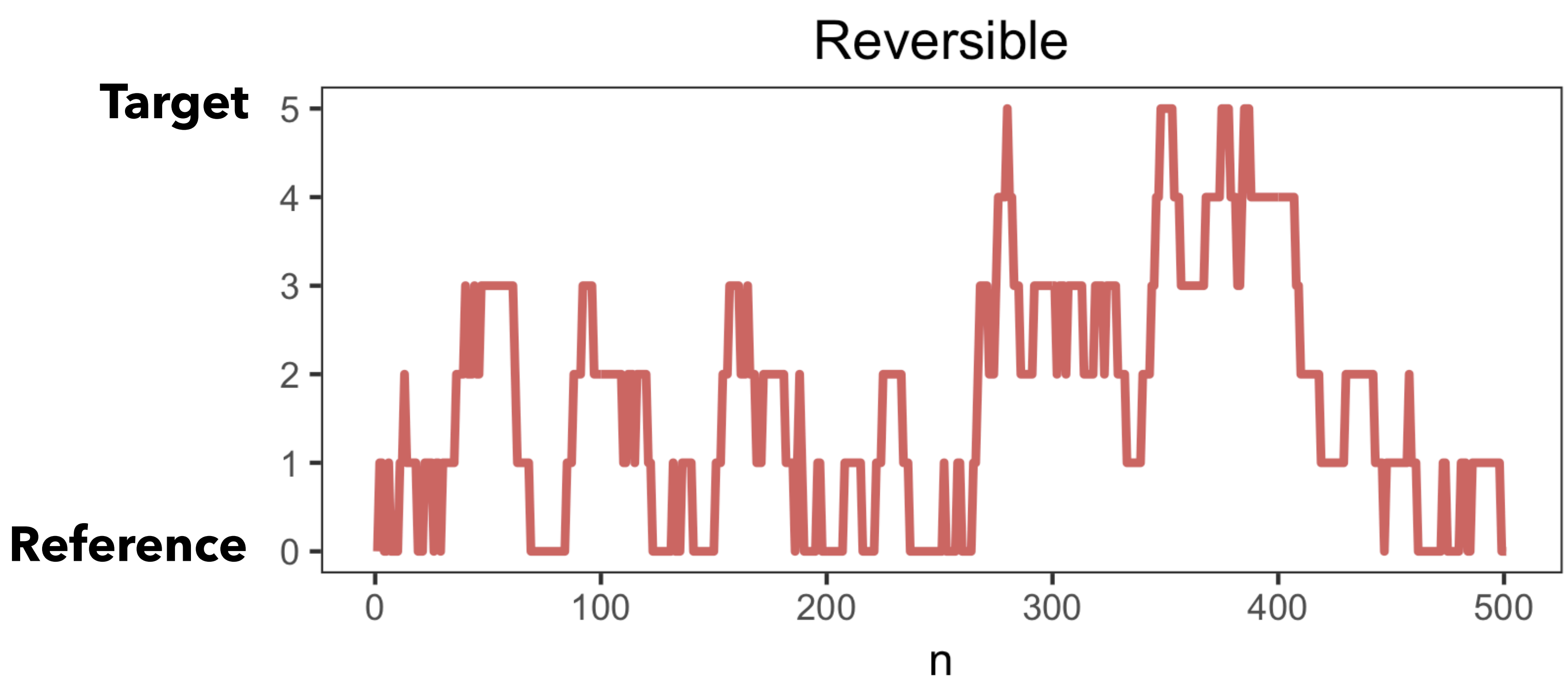
"Communication barrier"

- Non-asymptotically dominates **reversible PT**
- Improves with more chains (with marginal gains) controlled by **communication barrier**
- Robust to schedule
- Scalable to GPUs

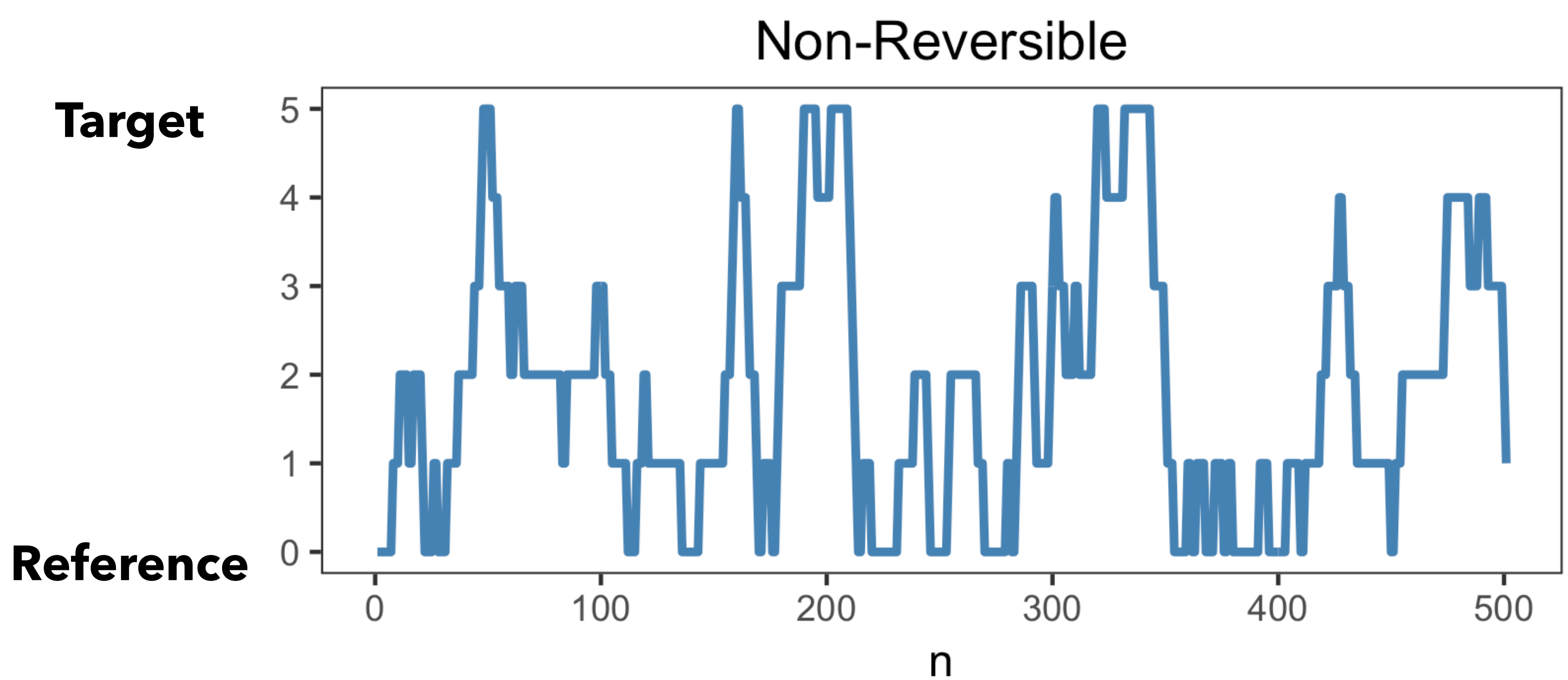
SAMPLE TRAJECTORIES

$$N = 5$$

 # Round Trips



$$1$$

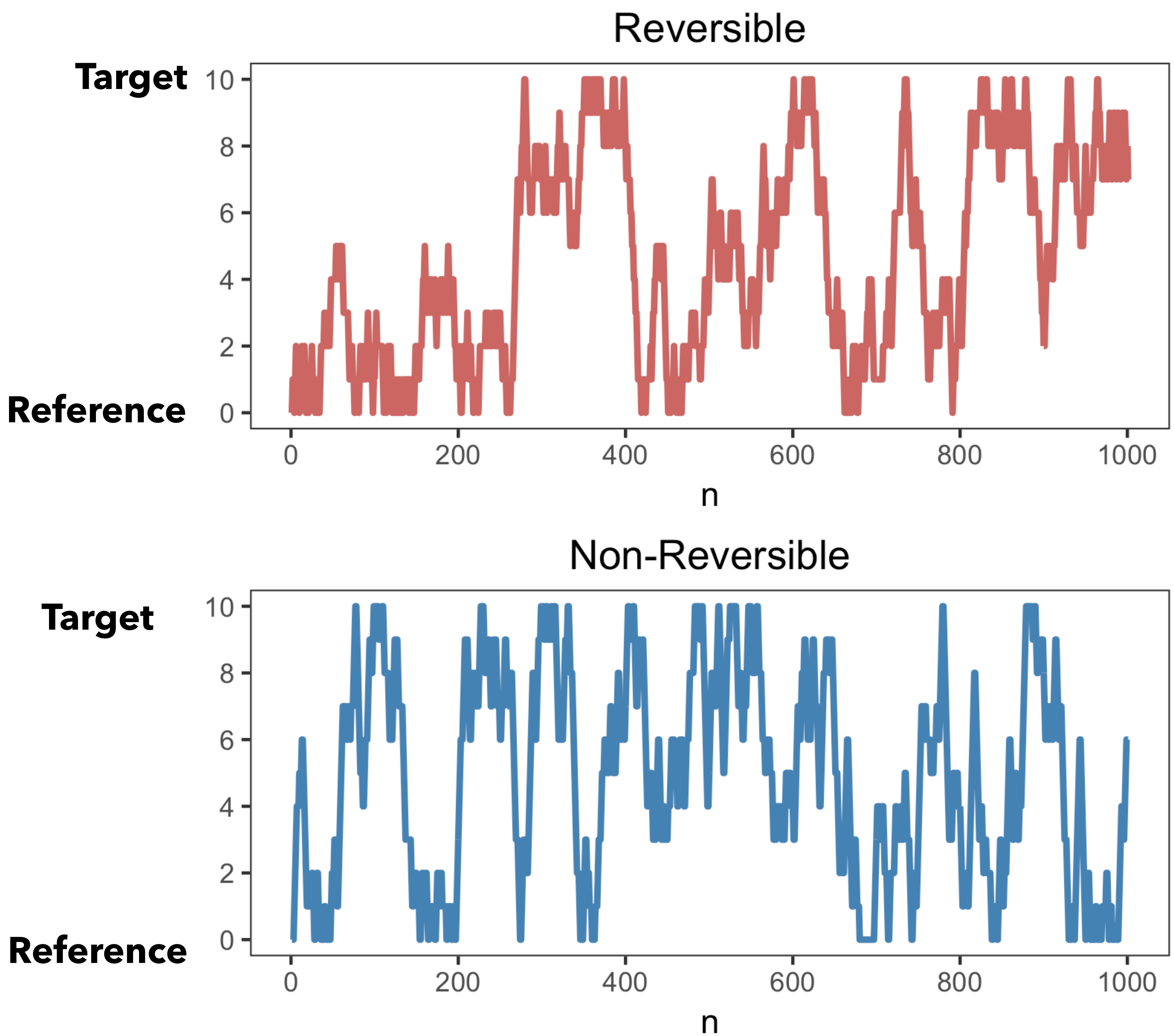


$$4$$

SAMPLE TRAJECTORIES

$$N = 10$$

 # Round Trips



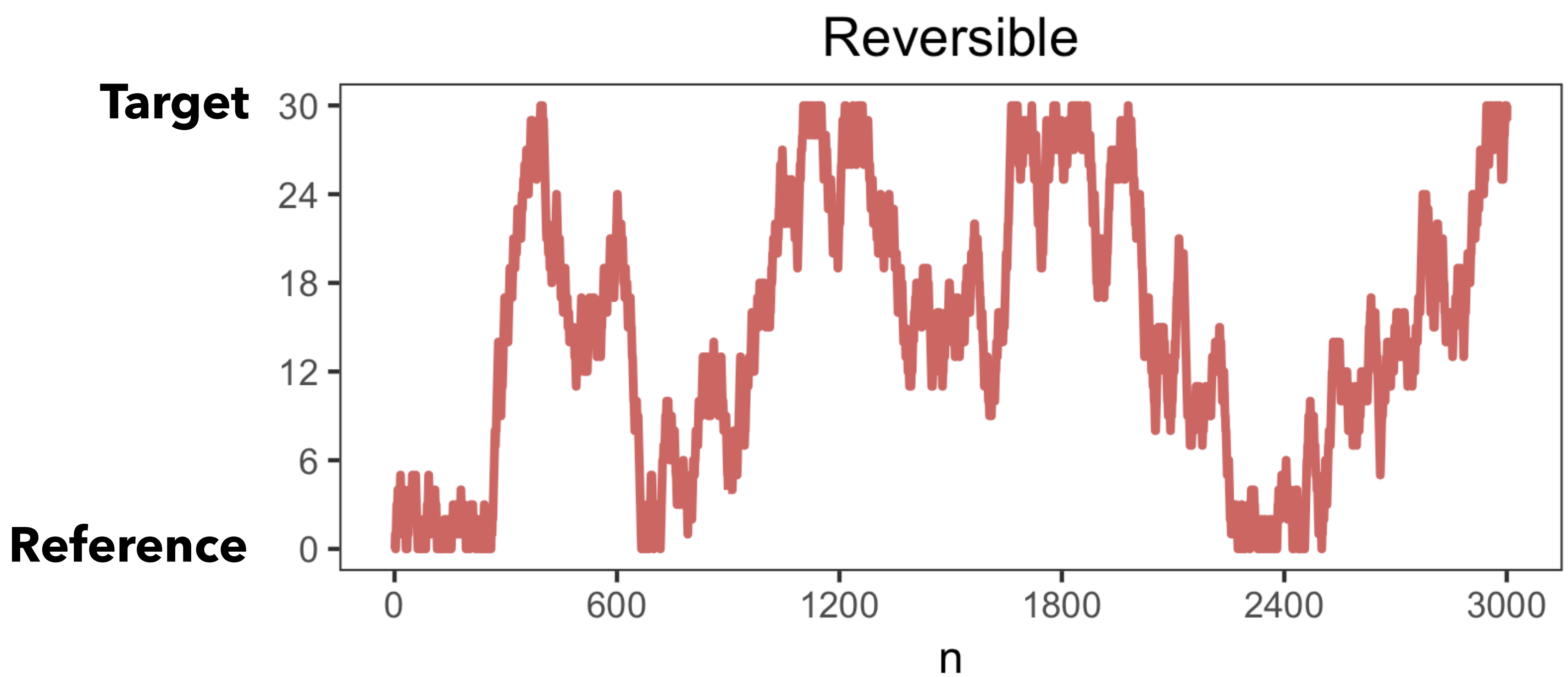
3

6

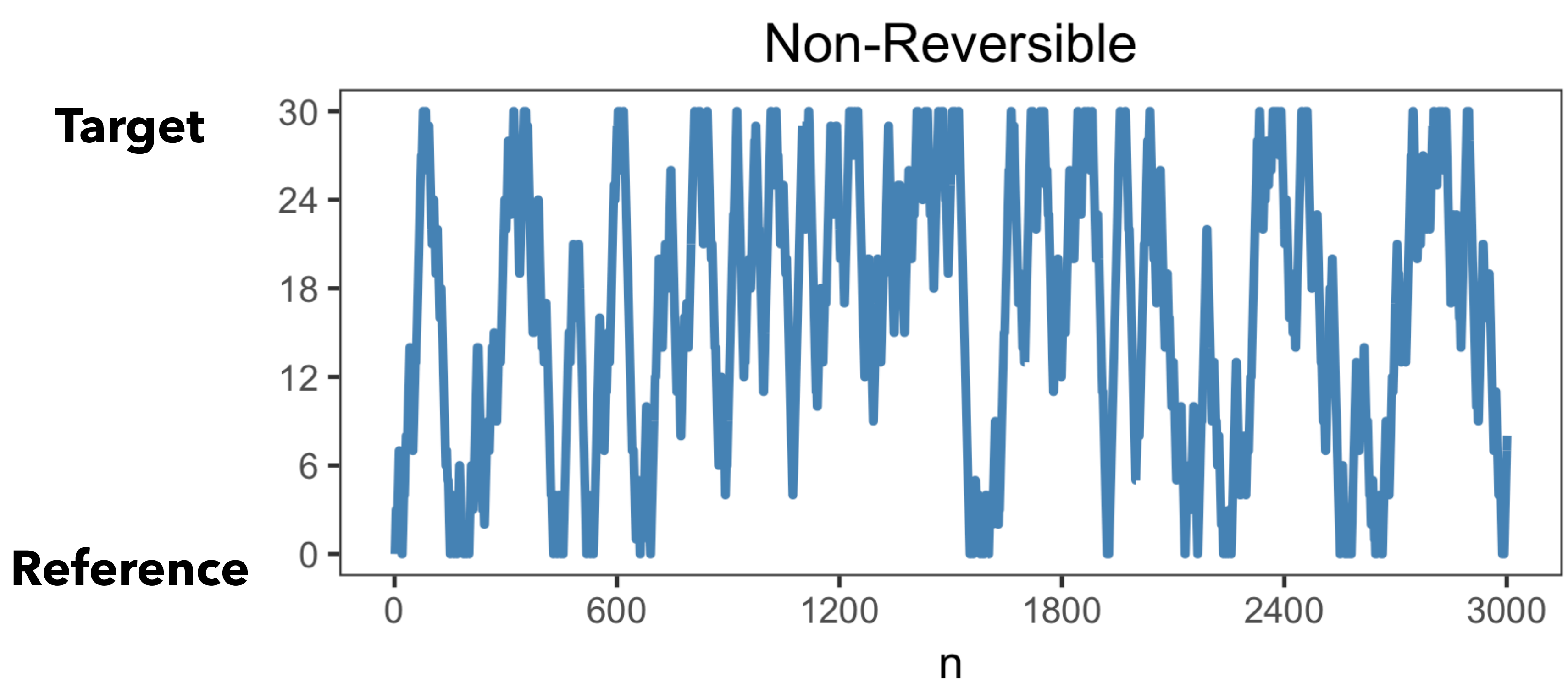
SAMPLE TRAJECTORIES

$$N = 30$$

Round Trips



2

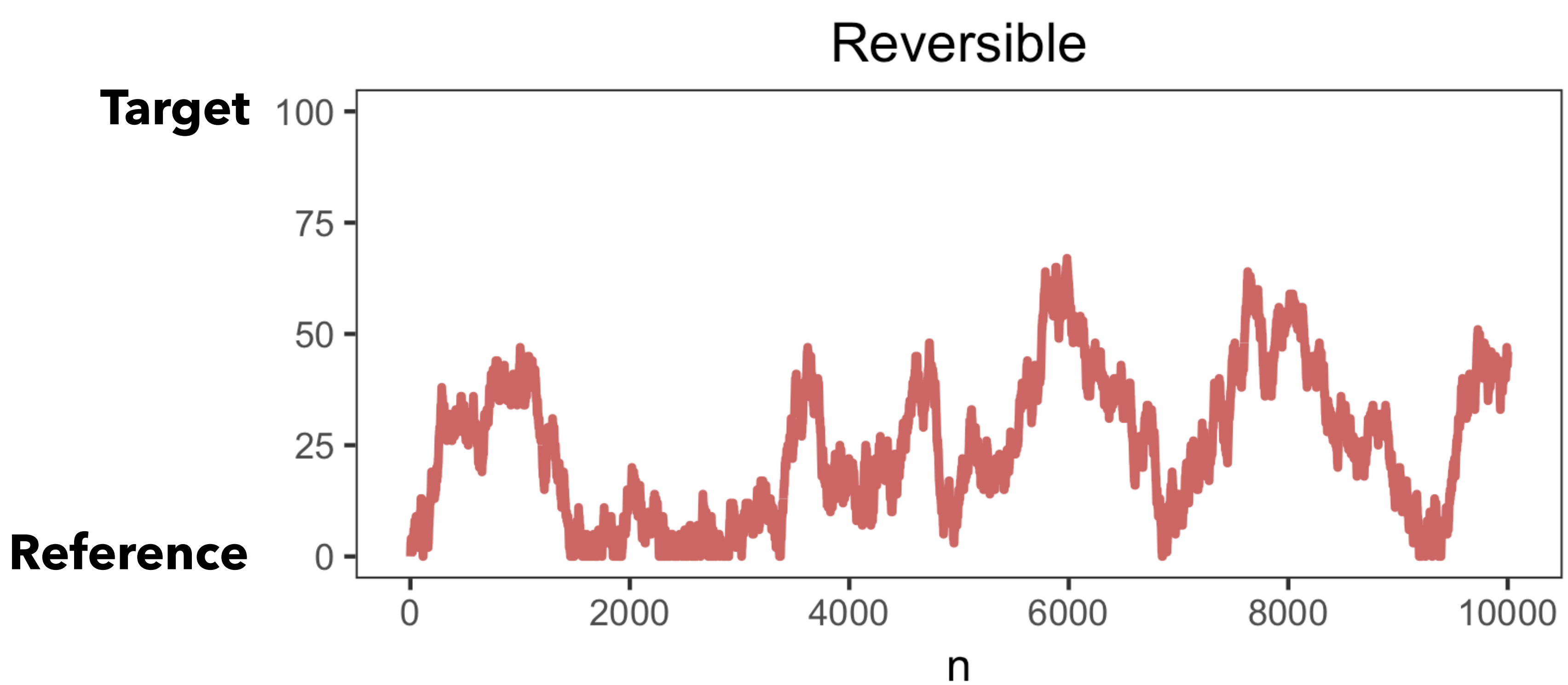


8

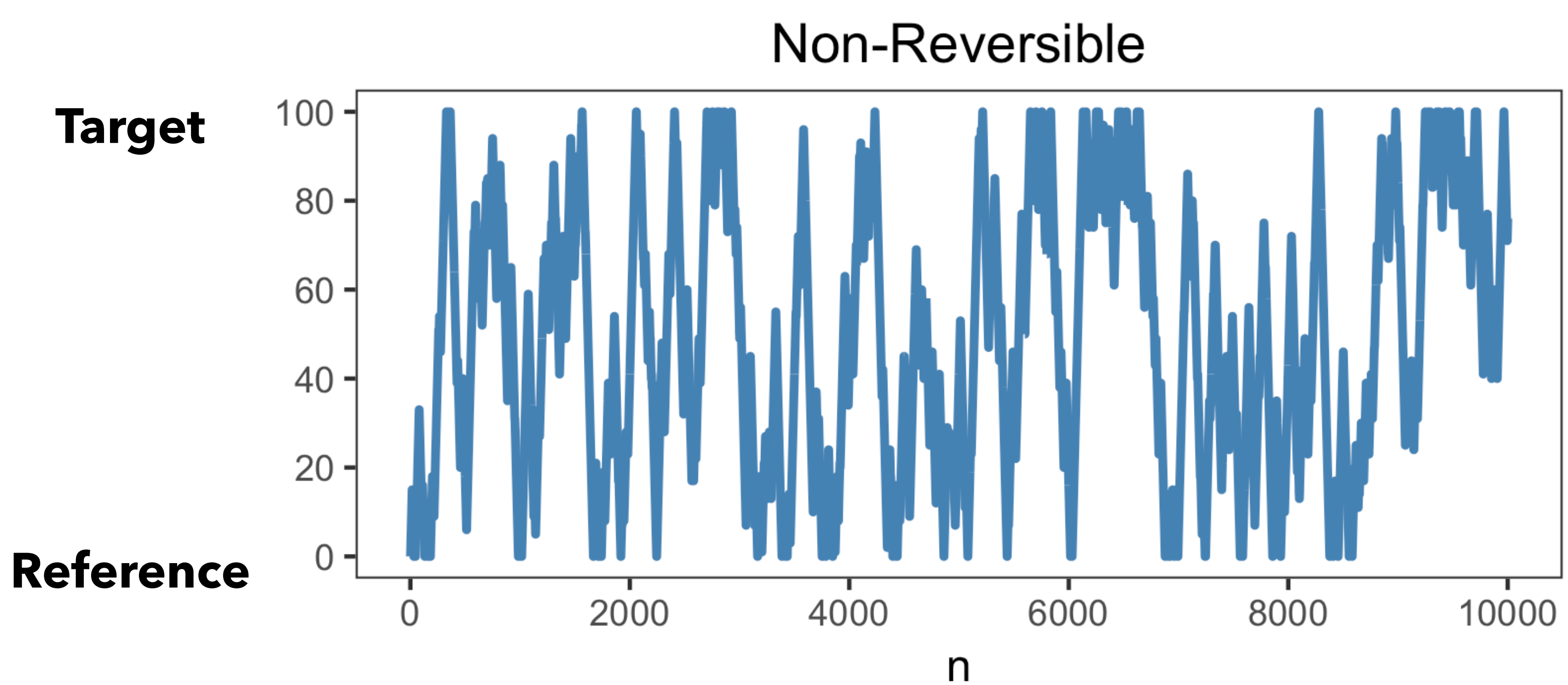
SAMPLE TRAJECTORIES

$$N = 100$$

 # Round Trips



0



9

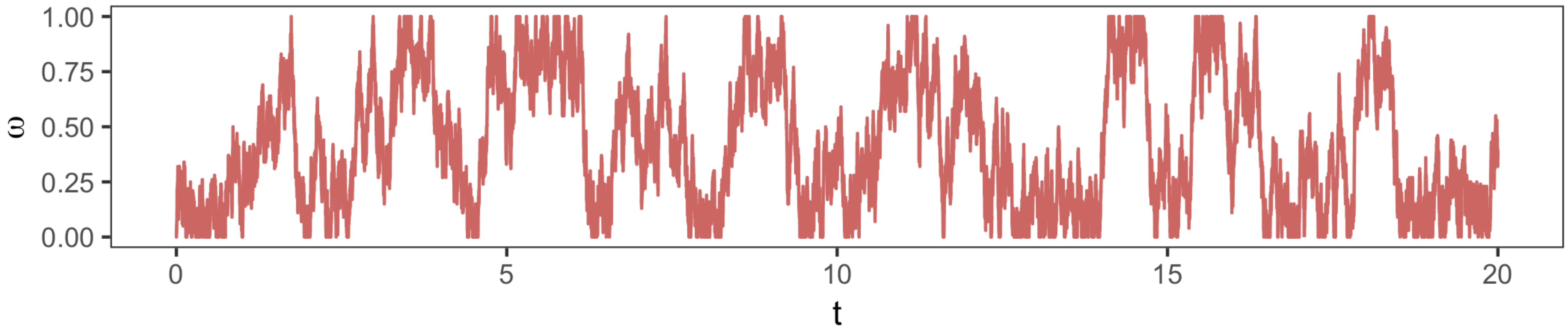
SCALING LIMIT (REVERSIBLE)

SCALING LIMIT (REVERSIBLE)

Theorem: Scaling time by N^2, the trajectories for reversible PT converge to a diffusion independent of target, reference, or annealing path.

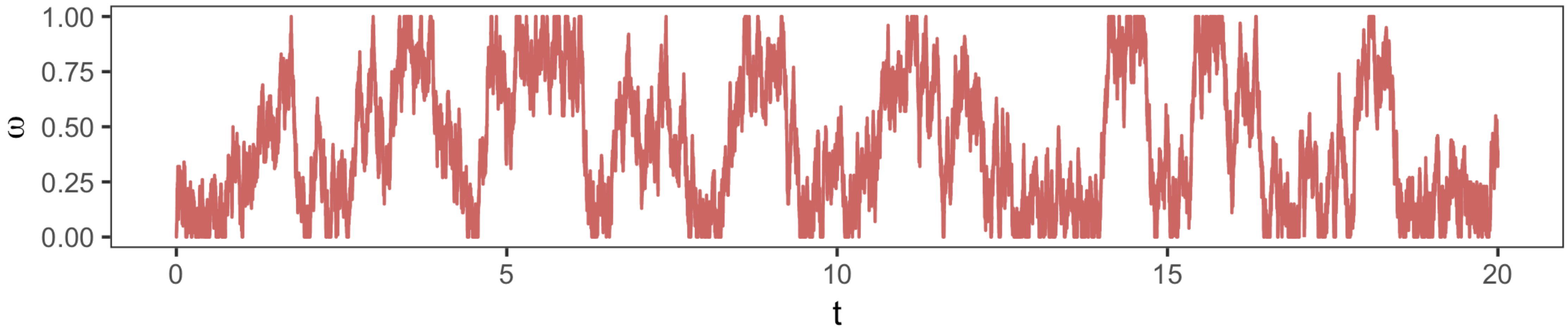
SCALING LIMIT (REVERSIBLE)

Theorem: Scaling time by N^2, the trajectories for reversible PT converge to a diffusion independent of target, reference, or annealing path.



SCALING LIMIT (REVERSIBLE)

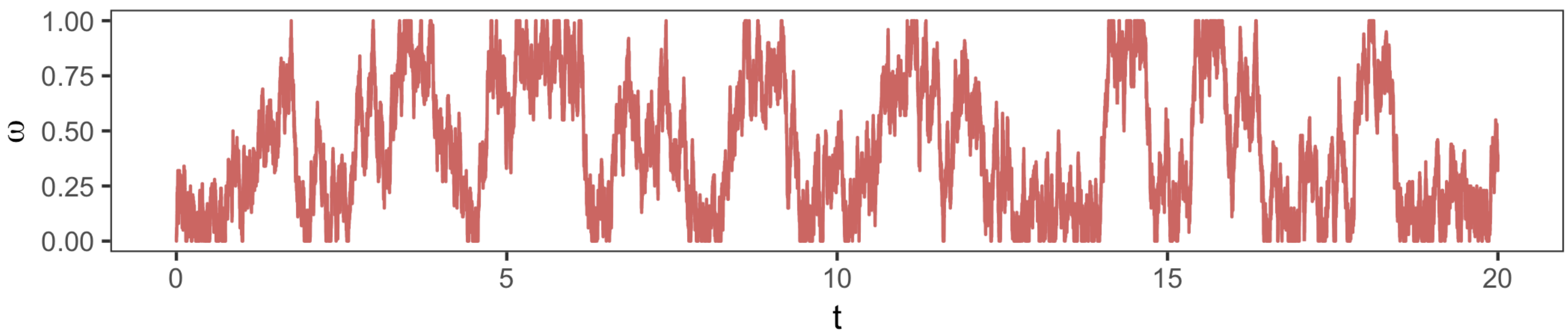
Theorem: Scaling time by N^2, the trajectories for reversible PT converge to a diffusion independent of target, reference, or annealing path.



- As N grows large, noise from diffusivity dominates

SCALING LIMIT (REVERSIBLE)

Theorem: Scaling time by N^2, the trajectories for reversible PT converge to a diffusion independent of target, reference, or annealing path.



- As N grows large, noise from diffusivity dominates
- Modifying reference, target, or path leads to marginal gains

SCALING LIMIT (NON-REVERSIBLE)

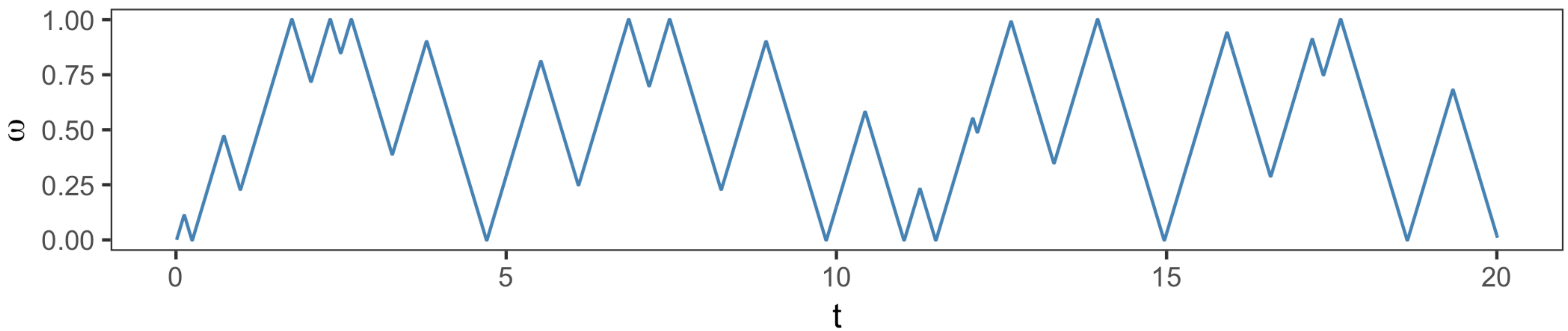
SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling time by N, the trajectories for non-reversible PT converge to piecewise deterministic Markov process depending on Λ

SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling time by N, the trajectories for non-reversible PT converge to piecewise deterministic Markov process depending on Λ

$\Lambda = 1$

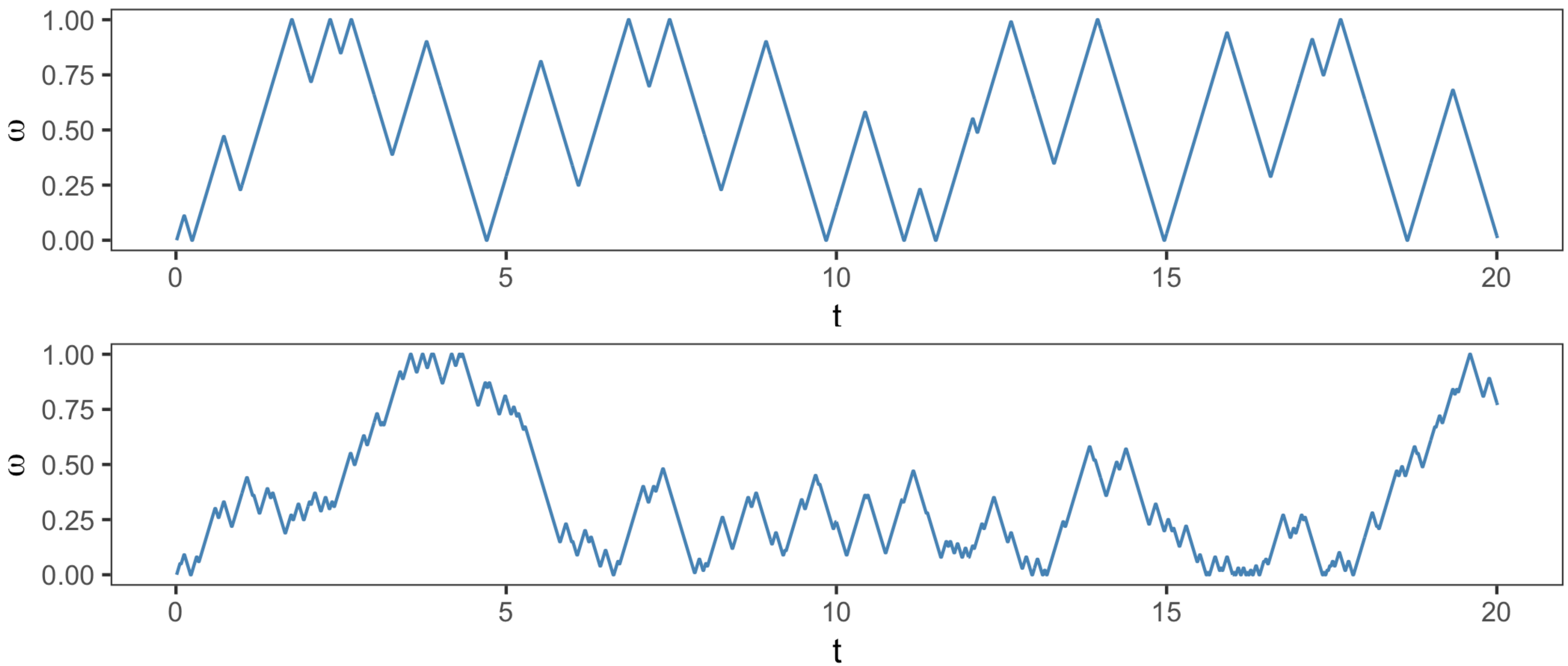


SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling time by N, the trajectories for non-reversible PT converge to piecewise deterministic Markov process depending on Λ

$\Lambda = 1$

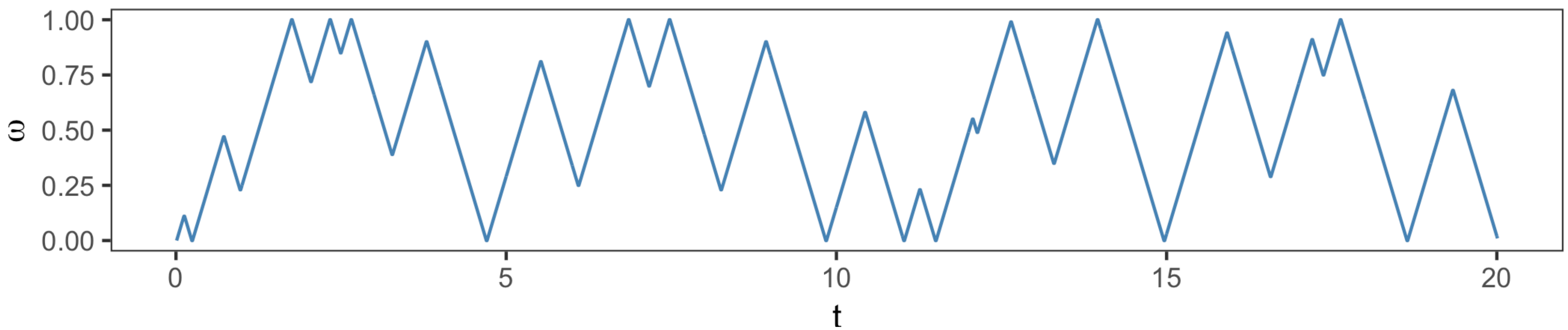
$\Lambda = 10$



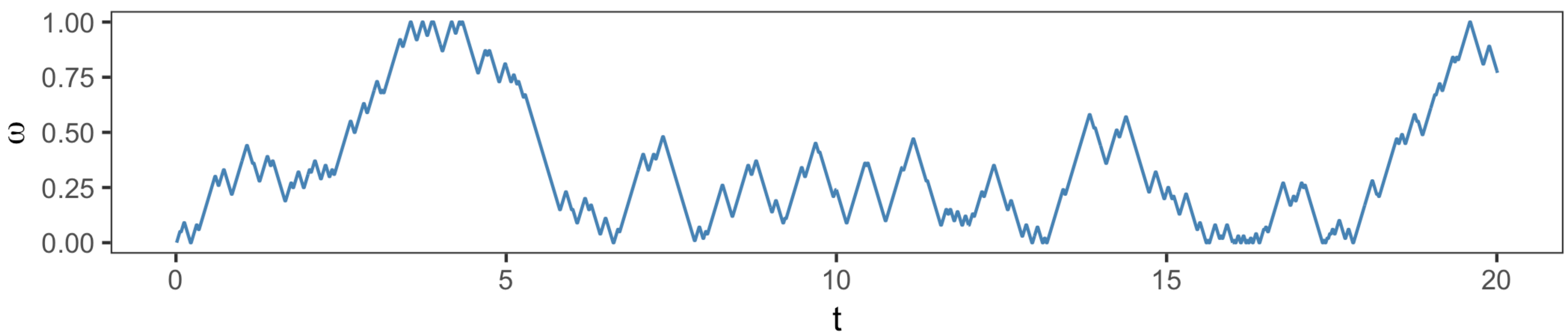
SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling time by N, the trajectories for non-reversible PT converge to piecewise deterministic Markov process depending on Λ

$\Lambda = 1$



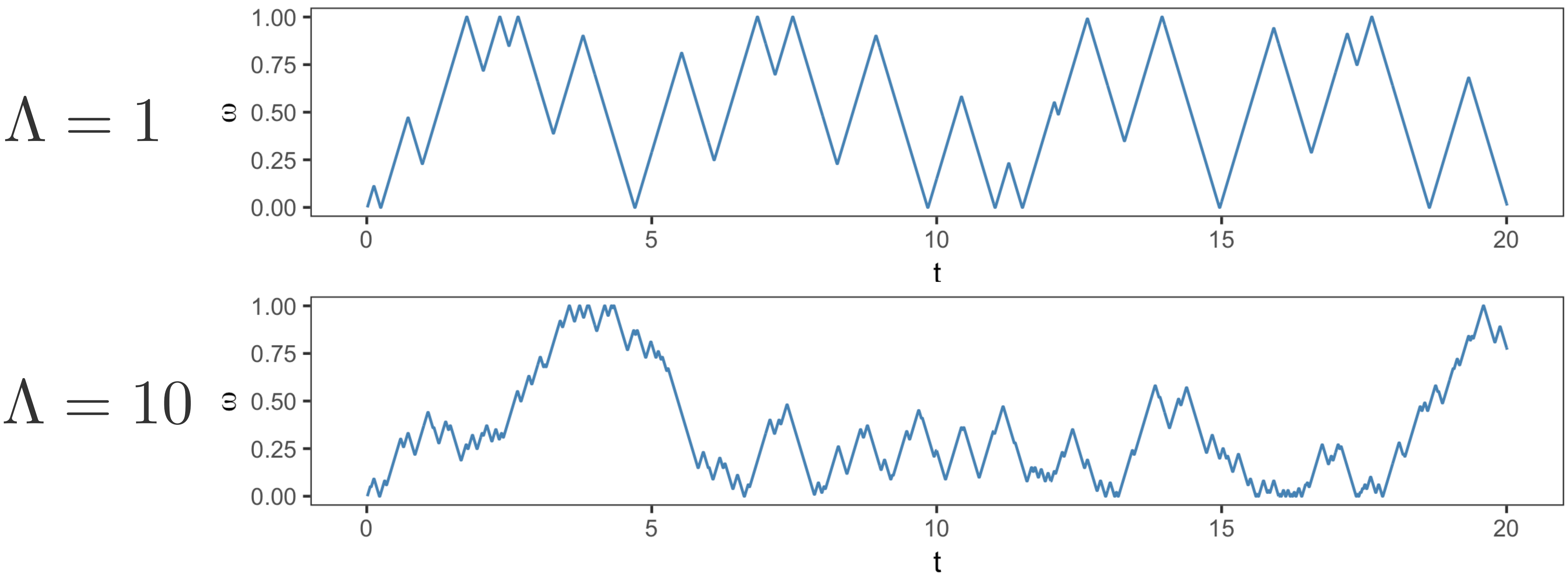
$\Lambda = 10$



- Fundamentally different behaviour compared to **reversible PT**

SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling time by N, the trajectories for non-reversible PT converge to piecewise deterministic Markov process depending on Λ

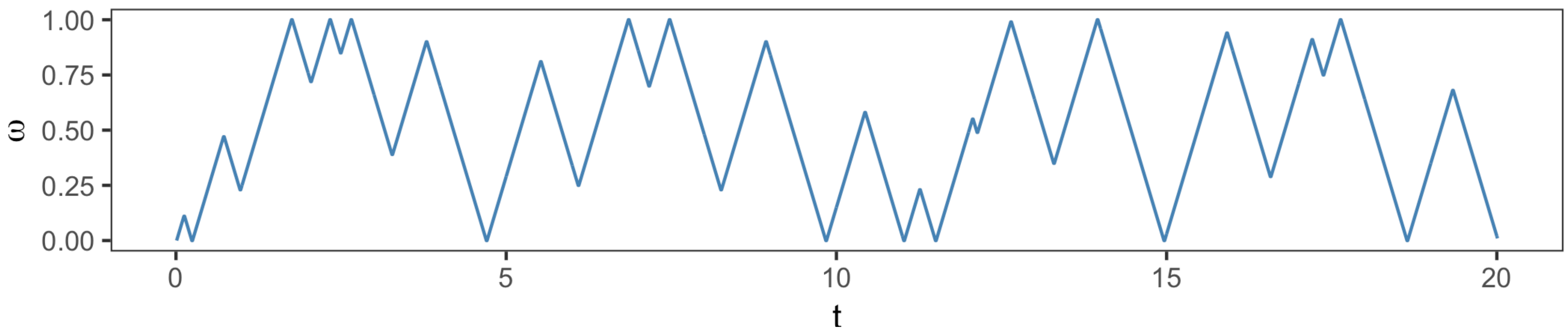


- Fundamentally different behaviour compared to **reversible PT**
- As N increases, NRPT stabilizes.

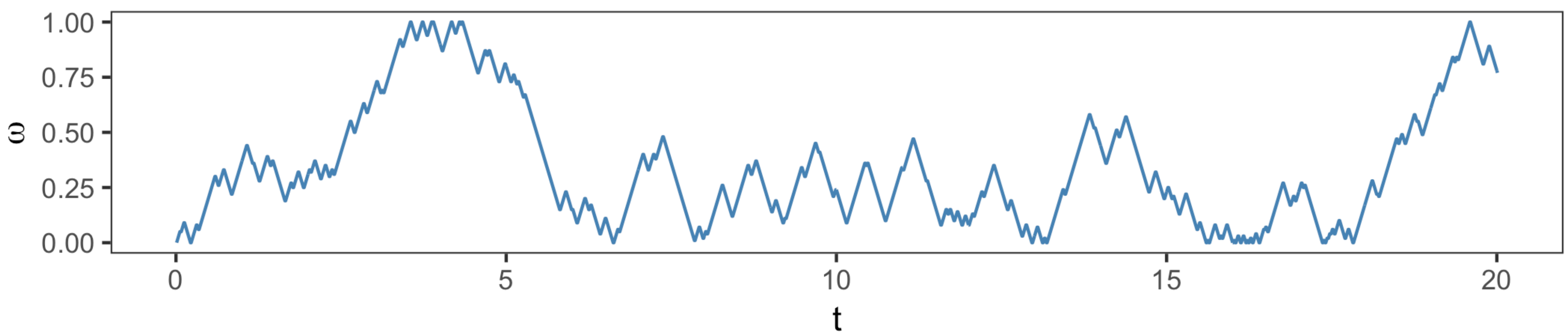
SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling time by N, the trajectories for non-reversible PT converge to piecewise deterministic Markov process depending on Λ

$\Lambda = 1$



$\Lambda = 10$



- Fundamentally different behaviour compared to **reversible PT**
- As N increases, NRPT stabilizes.
- Asymptotic behaviour of NRPT depends on the annealing path through communication barrier.

COMMUNICATION BARRIER

- **Rejection rate:** $r(\beta, \beta')$ the probability a swap rejected between π_β and $\pi_{\beta'}$

$$r_n = r(\beta_{n-1}, \beta_n)$$

COMMUNICATION BARRIER

- **Rejection rate:** $r(\beta, \beta')$ the probability a swap rejected between π_β and $\pi_{\beta'}$

$$r_n = r(\beta_{n-1}, \beta_n)$$

- **Local communication barrier:**

$$\lambda(\beta) = \lim_{\Delta\beta \to 0} \frac{r(\beta, \beta + \Delta\beta)}{|\Delta\beta|}$$

Instantaneous rate of rejection, measures how rapidly path changes at β

COMMUNICATION BARRIER

- **Rejection rate:** $r(\beta, \beta')$ the probability a swap rejected between π_β and $\pi_{\beta'}$

$$r_n = r(\beta_{n-1}, \beta_n)$$

- **Local communication barrier:**

$$\lambda(\beta) = \lim_{\Delta\beta \to 0} \frac{r(\beta, \beta + \Delta\beta)}{|\Delta\beta|}$$

Instantaneous rate of rejection, measures how rapidly path changes at β

- **Global communication barrier:**

$$\Lambda = \int_0^1 \lambda(\beta) \mathrm{d}\beta$$

Cumulative rejection rate along path

COMPUTE SCHEDULE

COMPUTE SCHEDULE

Theorem:

$$r(\beta, \beta') = \int_{\beta}^{\beta'} \lambda(u)\mathrm{d}u + O(|\beta - \beta'|^3)$$

COMPUTE SCHEDULE

Theorem:

$$r(\beta, \beta') = \int_{\beta}^{\beta'} \lambda(u)\mathrm{d}u + O(|\beta - \beta'|^3)$$

- Want to maximizing round trip rate:

$$\tau(\mathcal{B}_N) = \frac{1}{2 + 2\sum_{n=1}^{N} \frac{r_n}{1-r_n}}$$

COMPUTE SCHEDULE

Theorem:

$$r(\beta, \beta') = \int_{\beta}^{\beta'} \lambda(u)\mathrm{d}u + O(|\beta - \beta'|^3)$$

- Want to maximizing round trip rate:

$$\tau(\mathcal{B}_N) = \frac{1}{2 + 2\sum_{n=1}^{N} \frac{r_n}{1-r_n}}$$

- For any schedule:

$$\sum_{n=1}^{N} r_n = \Lambda + O(N^{-2})$$

COMPUTE SCHEDULE

Theorem:

$$r(\beta, \beta') = \int_{\beta}^{\beta'} \lambda(u)\mathrm{d}u + O(|\beta - \beta'|^3)$$

- Want to maximizing round trip rate:

$$\tau(\mathcal{B}_N) = \frac{1}{2 + 2\sum_{n=1}^{N} \frac{r_n}{1-r_n}}$$

- For any schedule:

$$\sum_{n=1}^{N} r_n = \Lambda + O(N^{-2})$$

- By ignoring error term, we have a constraint optimization problem with solution:

$$r_n^* = \frac{\Lambda}{N}$$

COMPUTE SCHEDULE

Theorem: $$r(\beta, \beta') = \int_{\beta}^{\beta'} \lambda(u)\mathrm{d}u + O(|\beta - \beta'|^3)$$

- Want to maximizing round trip rate:

$$\tau(\mathcal{B}_N) = \frac{1}{2 + 2\sum_{n=1}^{N} \frac{r_n}{1-r_n}}$$

- For any schedule:

$$\sum_{n=1}^{N} r_n = \Lambda + O(N^{-2})$$

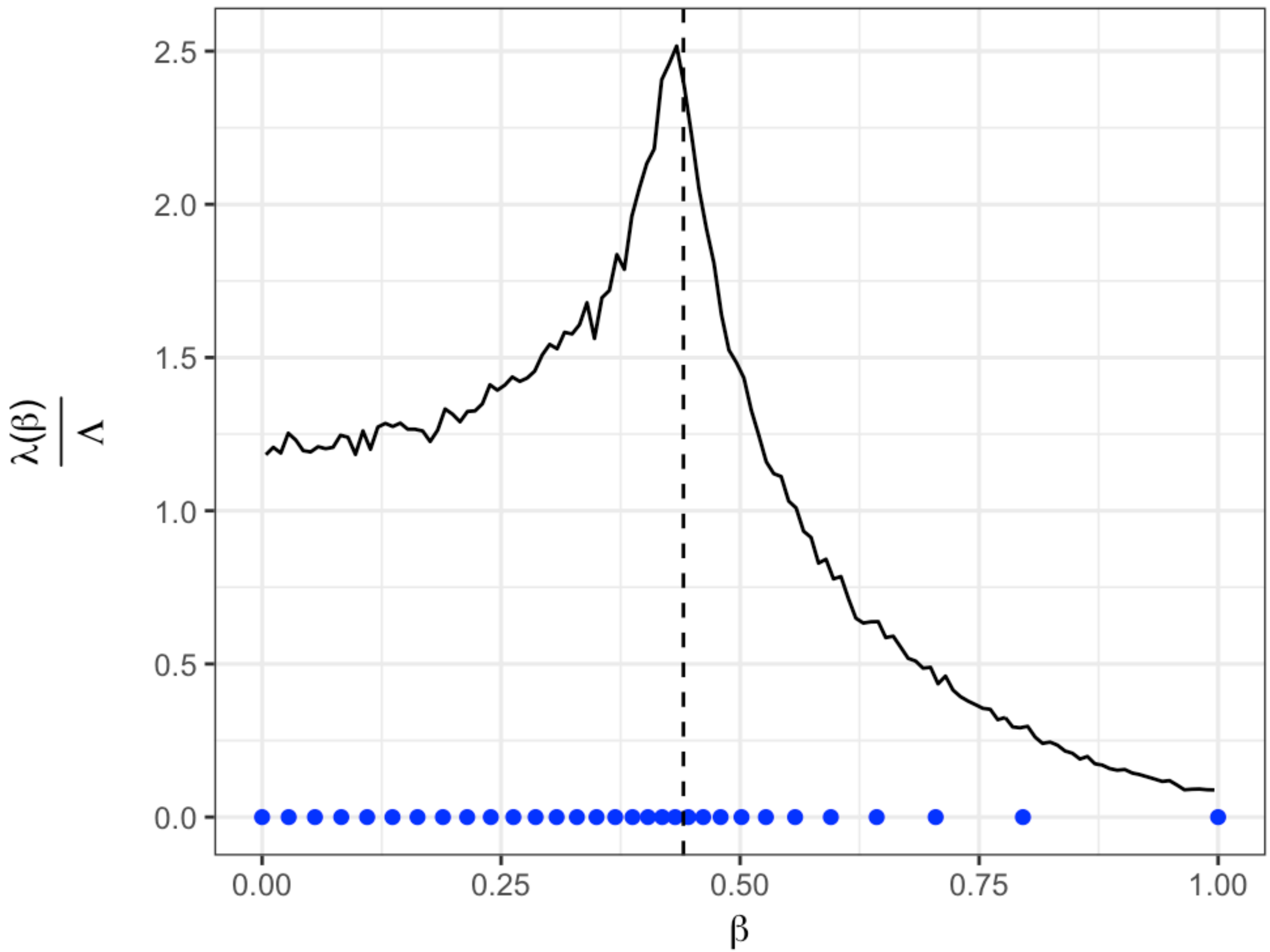
- By ignoring error term, we have a constraint optimization problem with solution:

$$r_n^* = \frac{\Lambda}{N}$$

- Theorem implies:

$$\int_{\beta_{n-1}}^{\beta_n} \lambda(\beta)\mathrm{d}\beta = \frac{\Lambda}{N}$$

EXAMPLE ISING MODEL



COMPUTE SCHEDULE

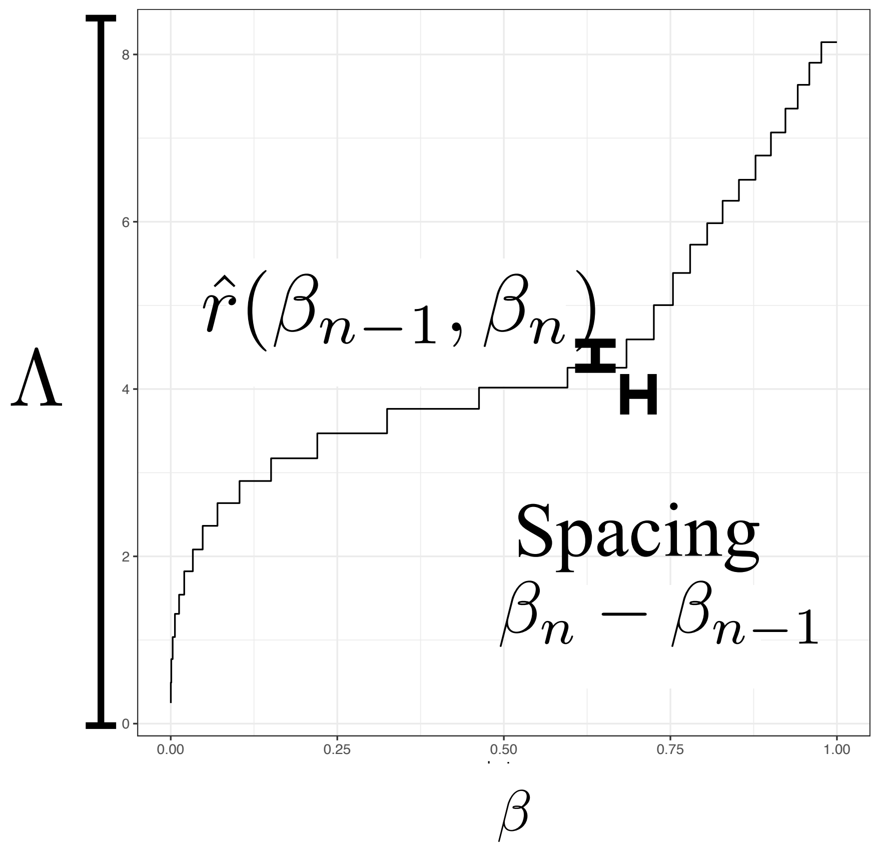
COMPUTE SCHEDULE

$$\hat{r}(\beta_{n-1}, \beta_n) \leftarrow \texttt{PT}(\mathcal{B}_N)$$

COMPUTE SCHEDULE

$$\hat{r}(\beta_{n-1}, \beta_n) \leftarrow \texttt{PT}(\mathcal{B}_N)$$

$$\mathcal{B}_N = (\beta_0, \ldots, \beta_N)$$

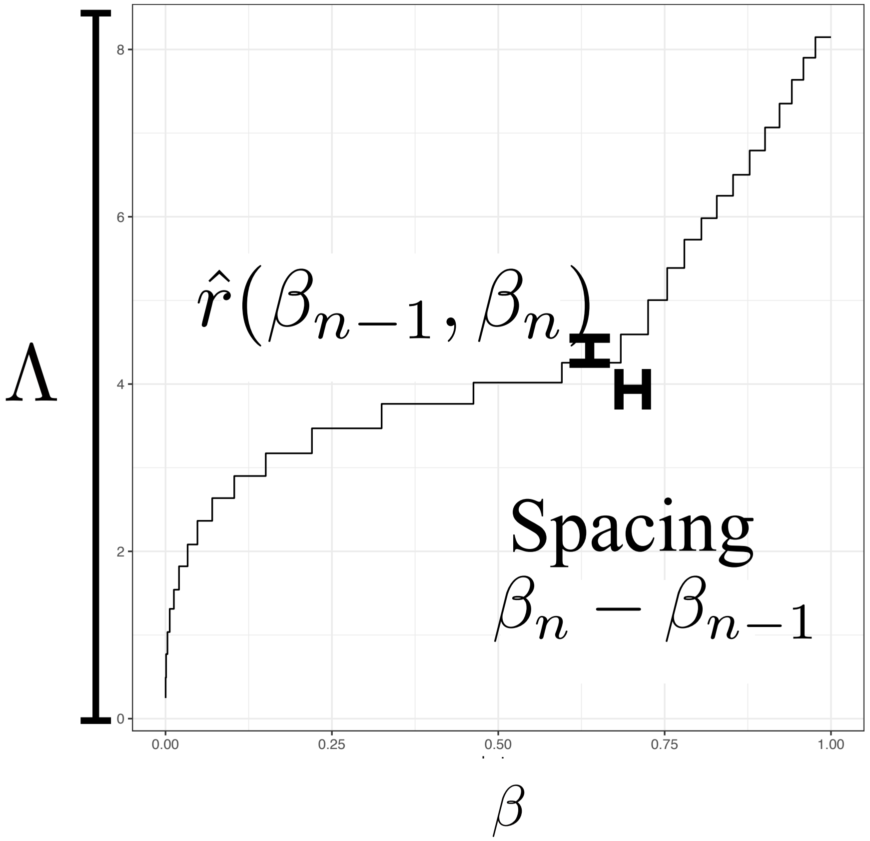


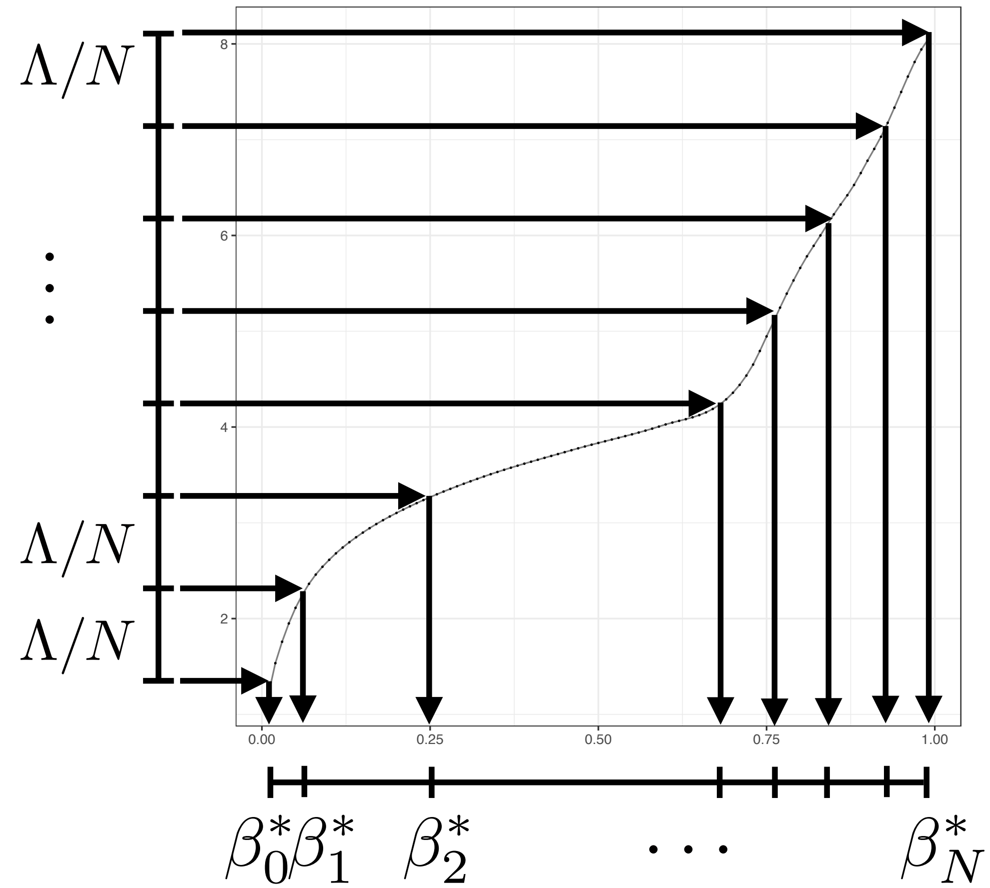
COMPUTE SCHEDULE

$$\hat{r}(\beta_{n-1}, \beta_n) \leftarrow \mathtt{PT}(\mathcal{B}_N)$$

$$\mathcal{B}_N = (\beta_0, \ldots, \beta_N)$$

$$\mathcal{B}_N^* = (\beta_0^*, \ldots, \beta_N^*)$$



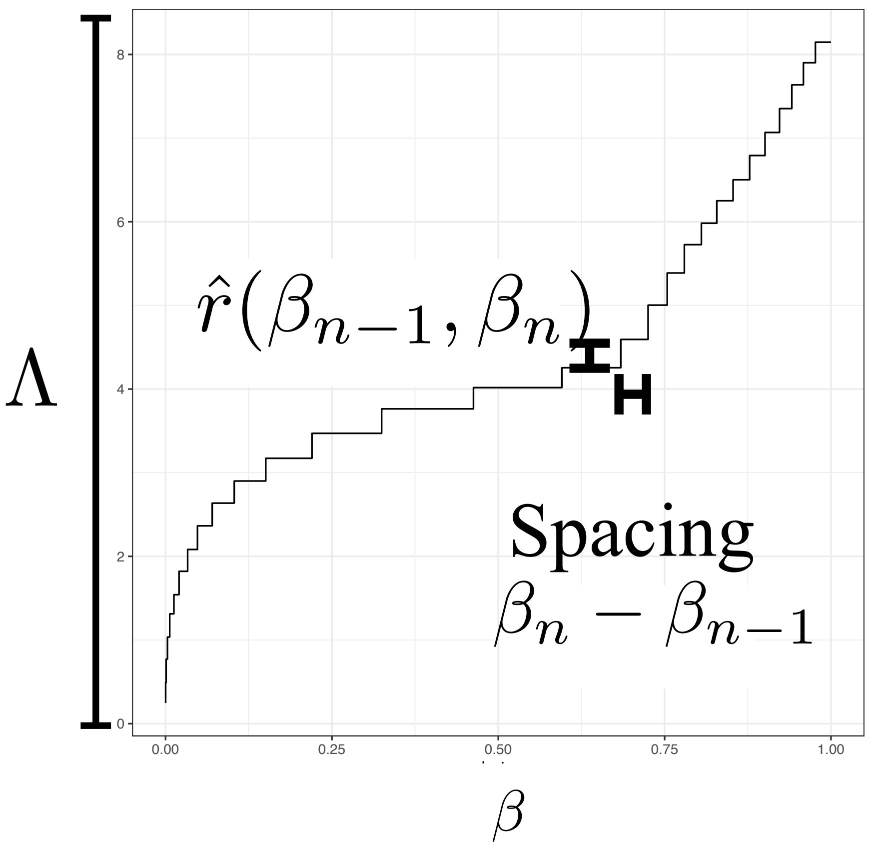


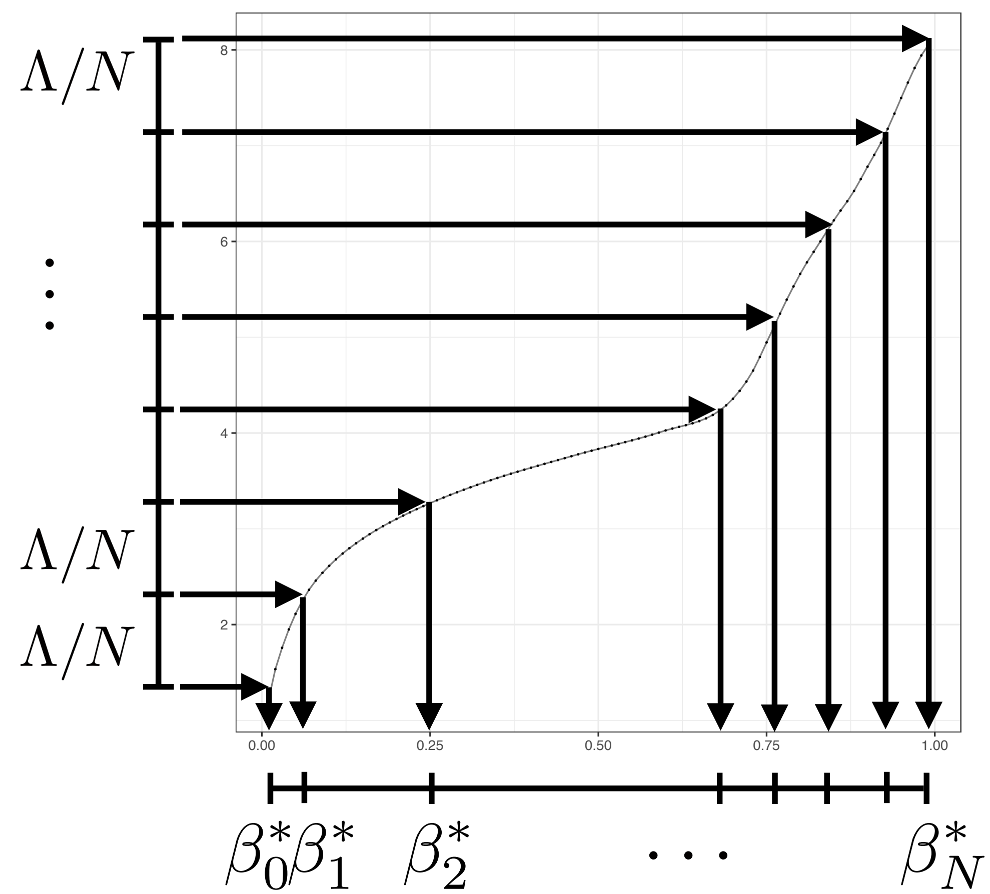
COMPUTE SCHEDULE

$$\hat{r}(\beta_{n-1}, \beta_n) \leftarrow \mathtt{PT}(\mathcal{B}_N)$$

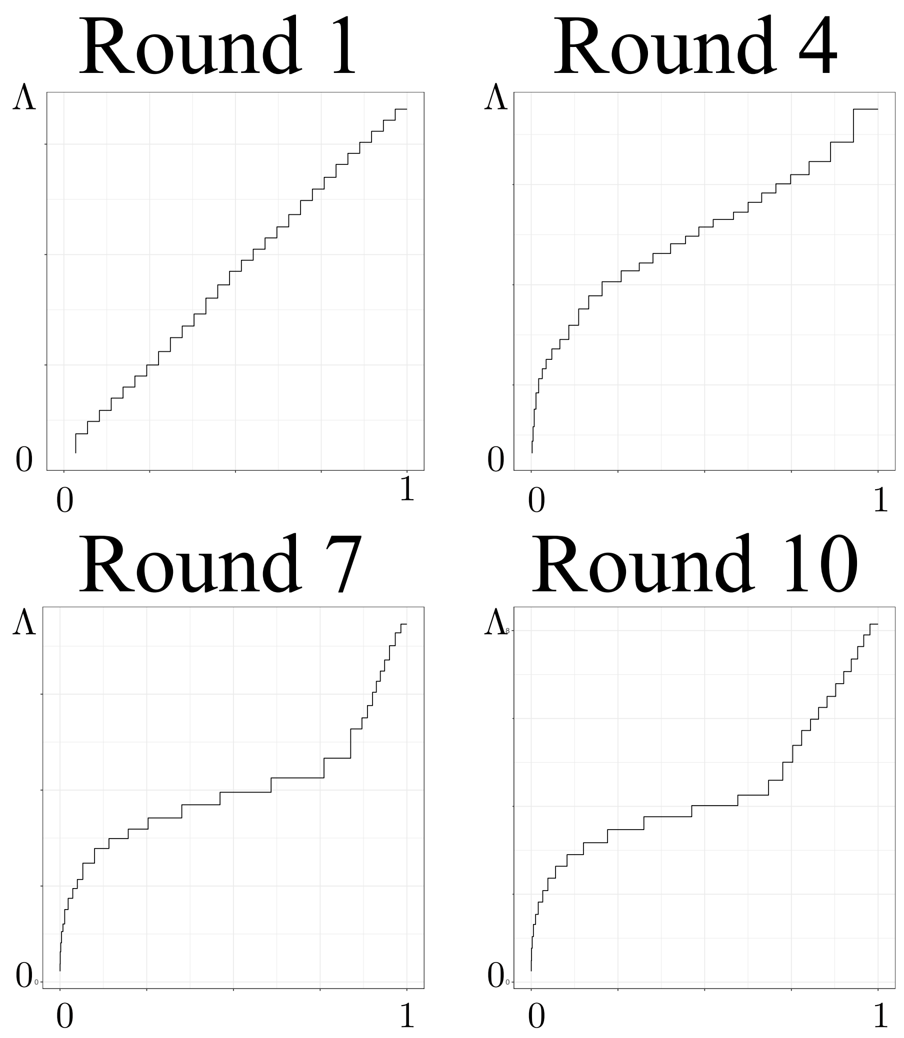
$$\mathcal{B}_N = (\beta_0, \ldots, \beta_N)$$

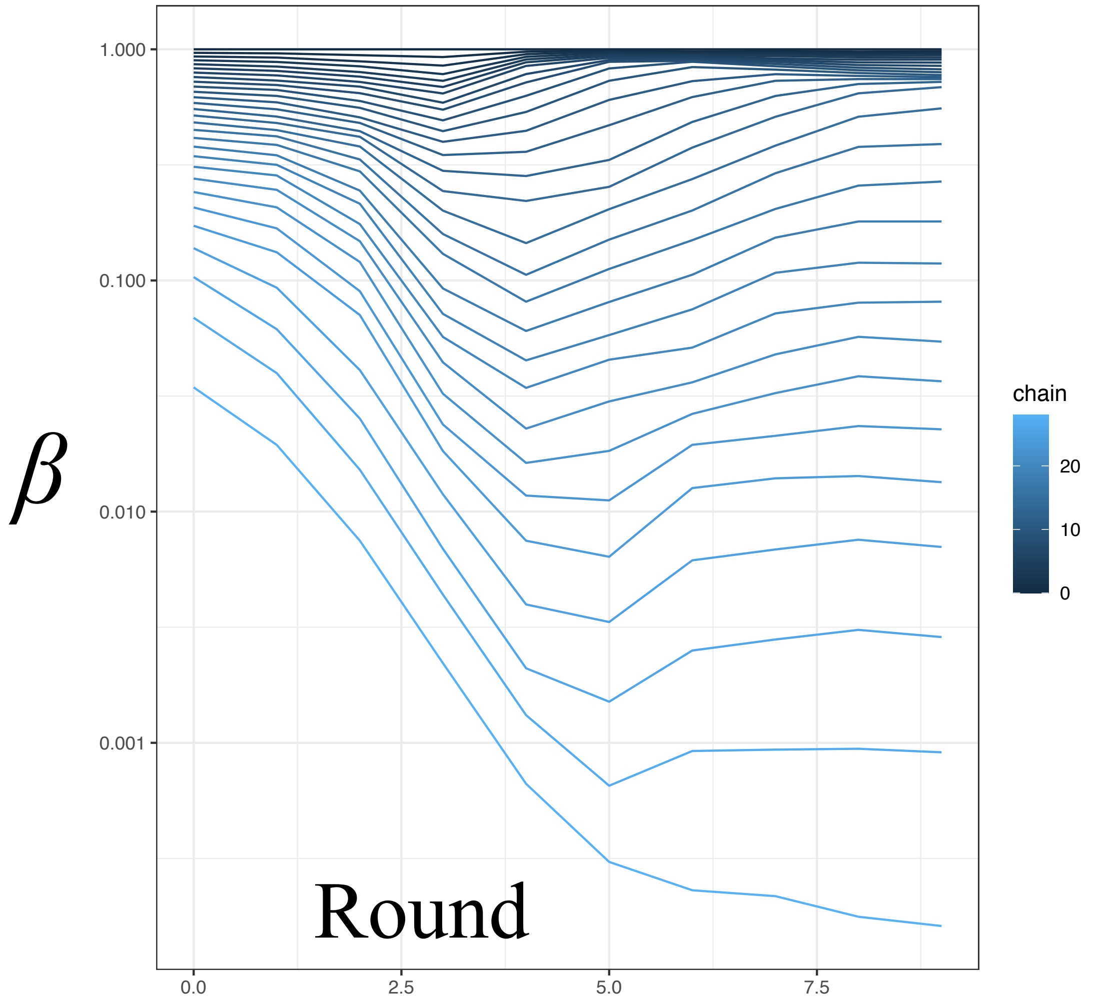
$$\mathcal{B}_N^* = (\beta_0^*, \ldots, \beta_N^*)$$



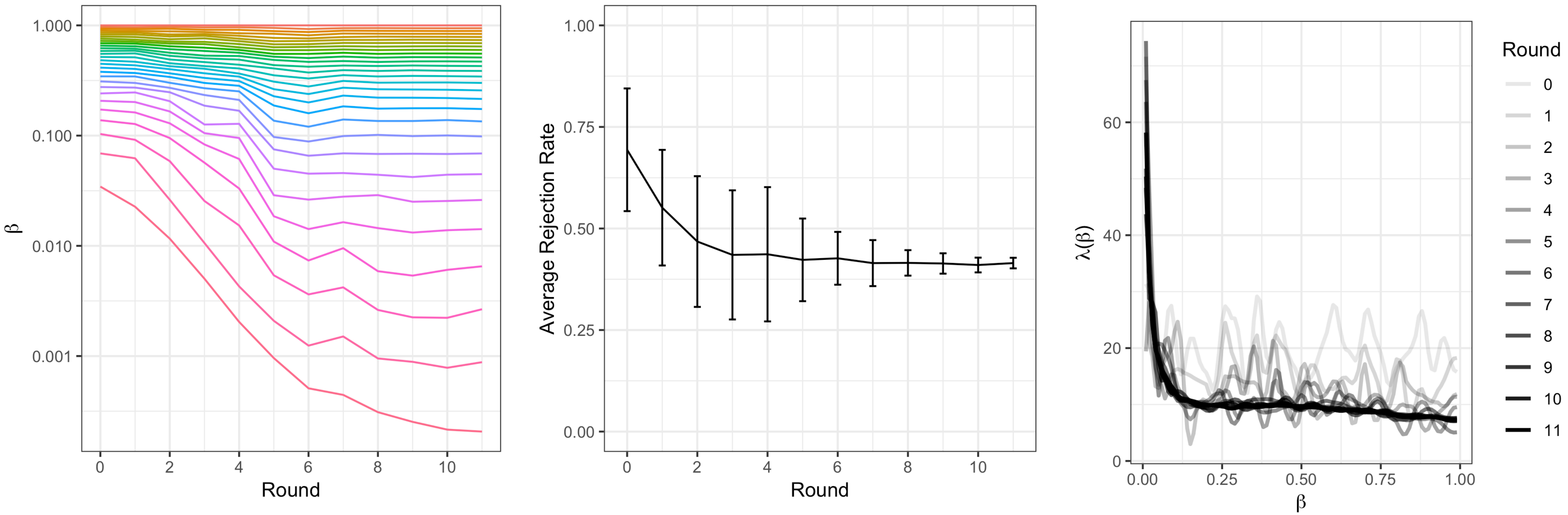


COMPUTE SCHEDULE



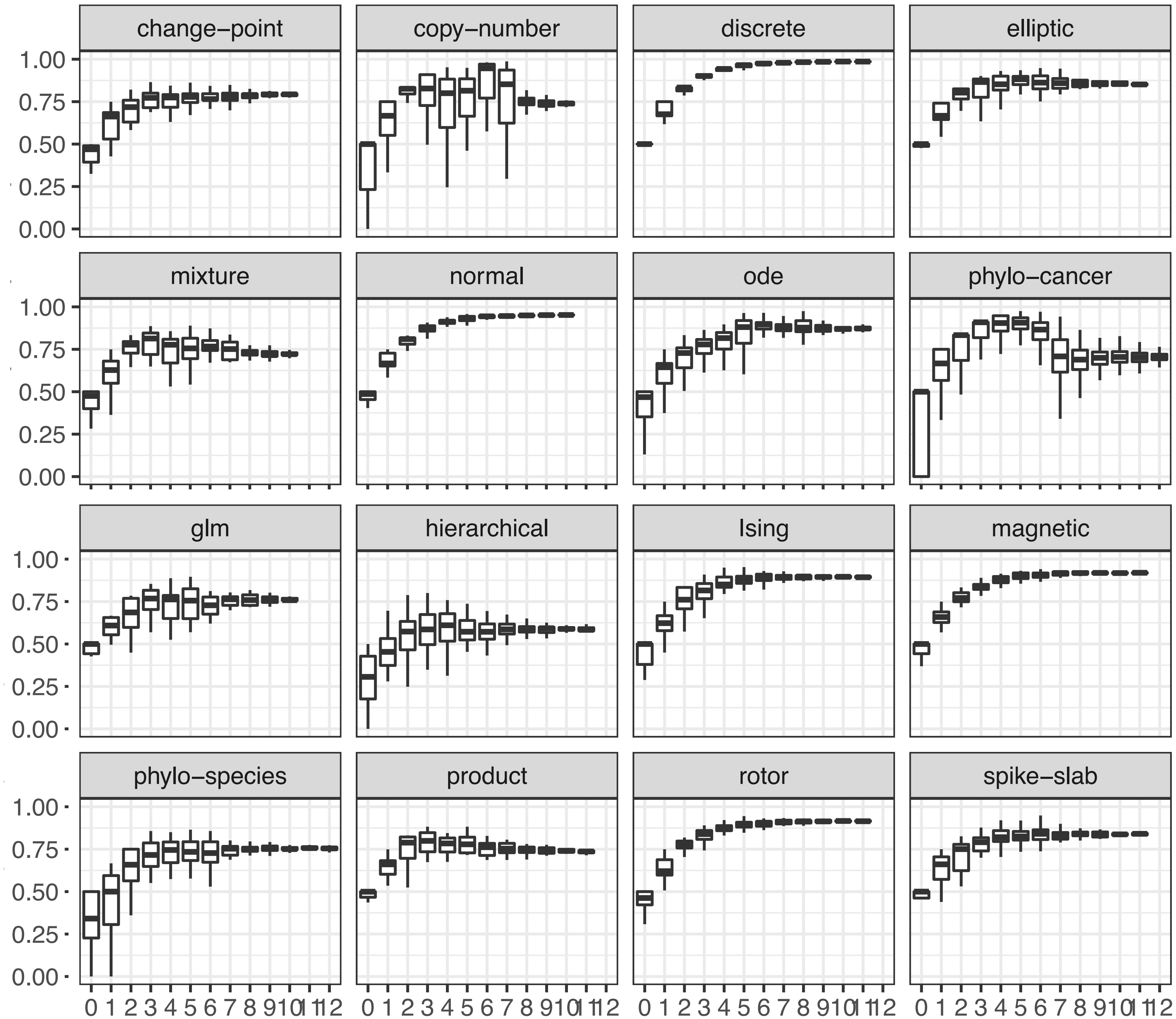


HIERARCHICAL BAYESIAN MODEL (D = 369)



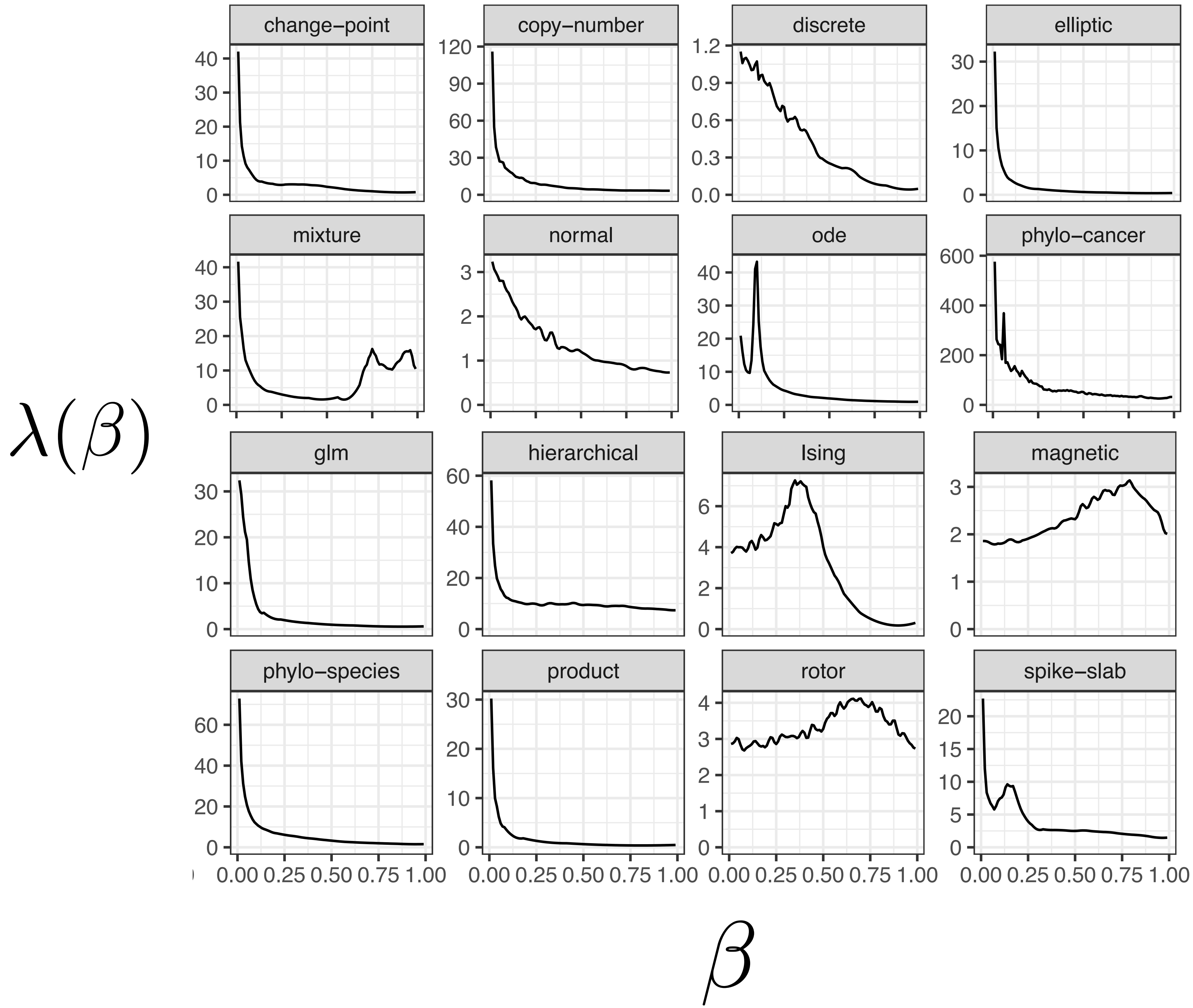
NRPT IN PRACTICE

Acceptance rates



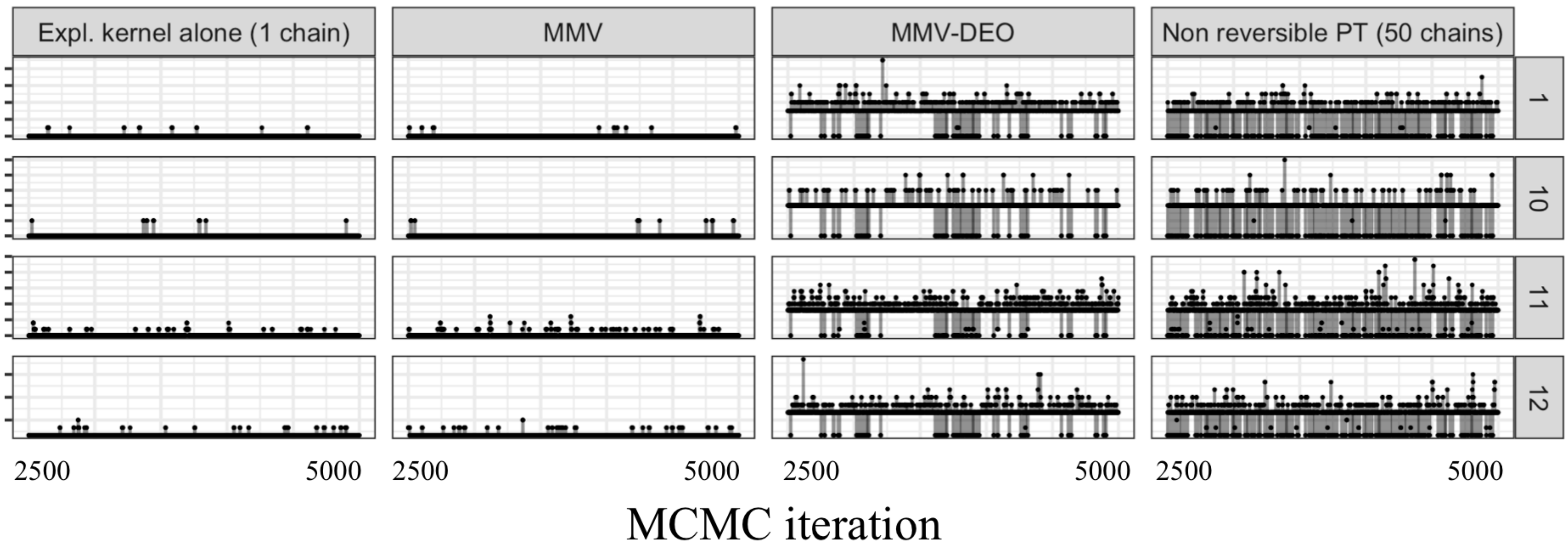
Round

LOCAL COMMUNICATION BARRIER

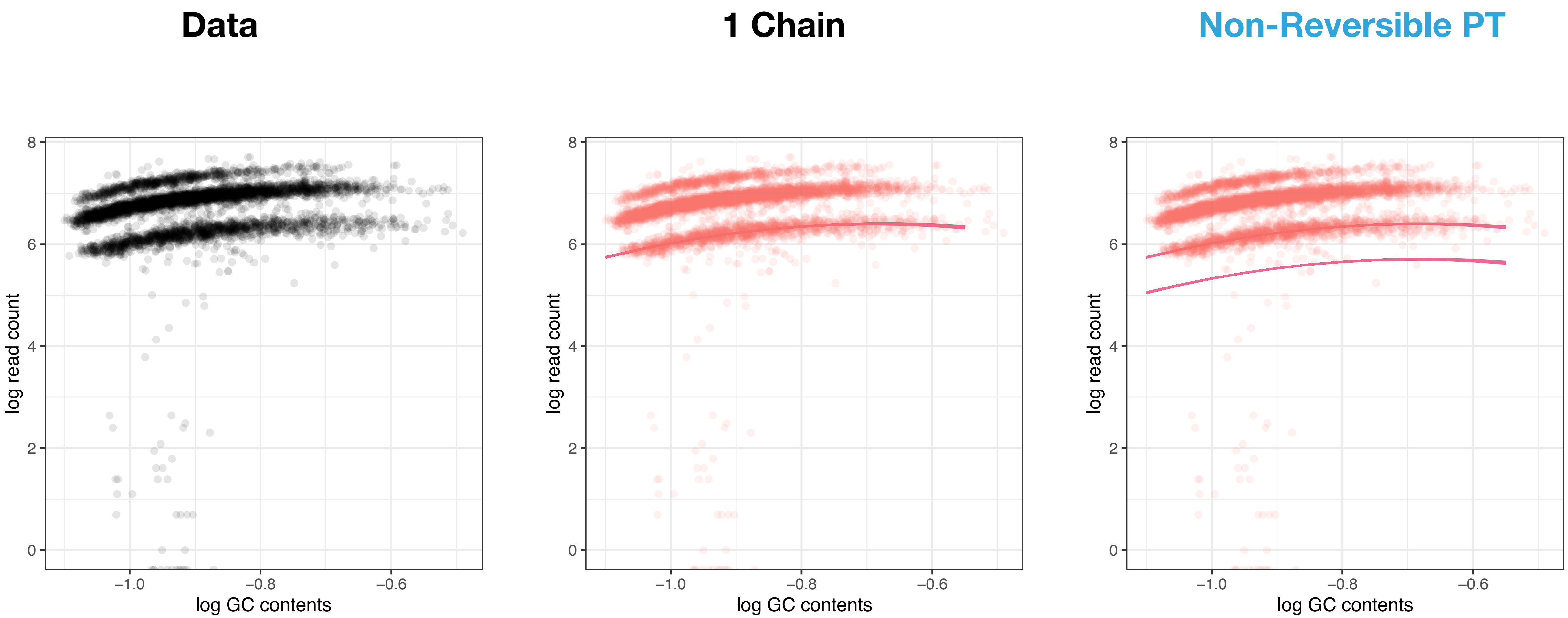


EXAMPLE: COPY NUMBER INFERENCE (D = 30)

1 Chain | Reversible PT | Reversible PT + Schedule Tuning | Non-Reversible PT + Schedule Tuning

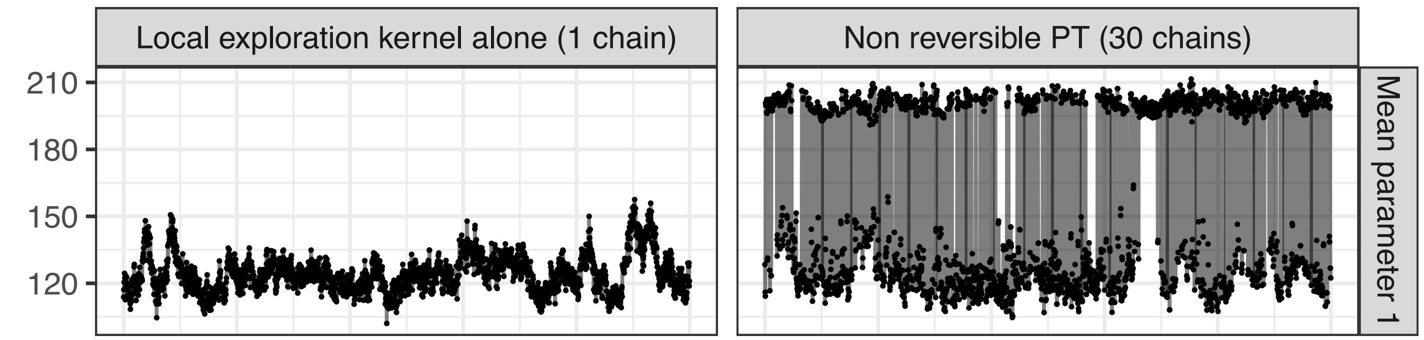


EXAMPLE: COPY NUMBER INFERENCE (D = 30)

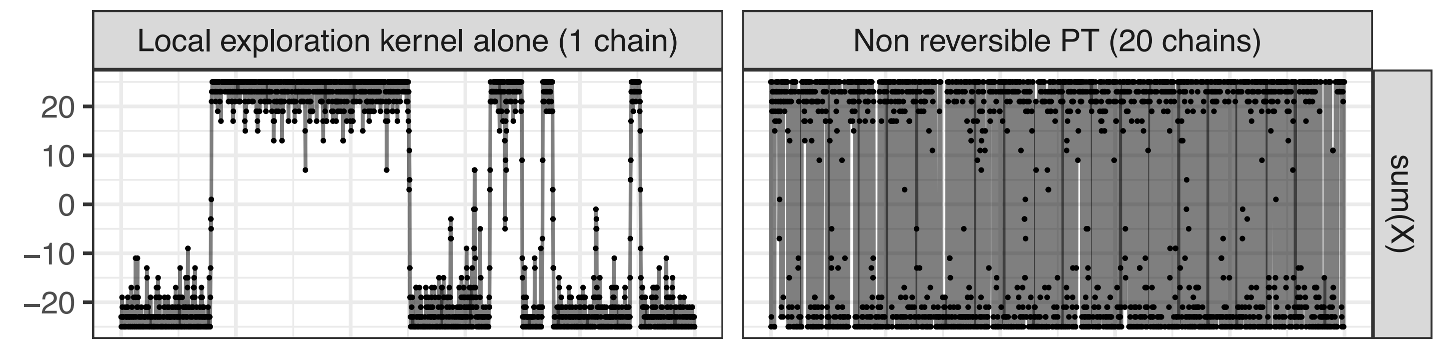


EXAMPLE: MULTIMODAL TARGETS

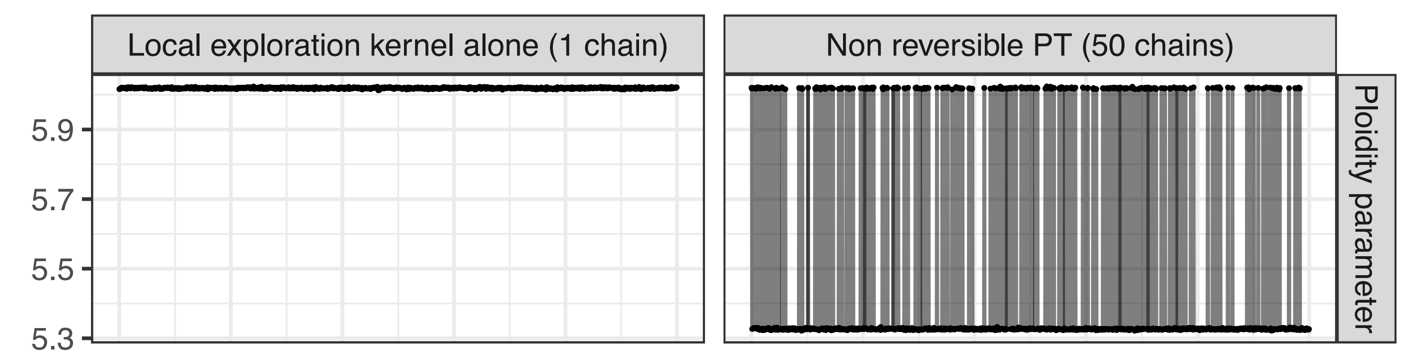
Bayesian Mixture Model
(d = 155)



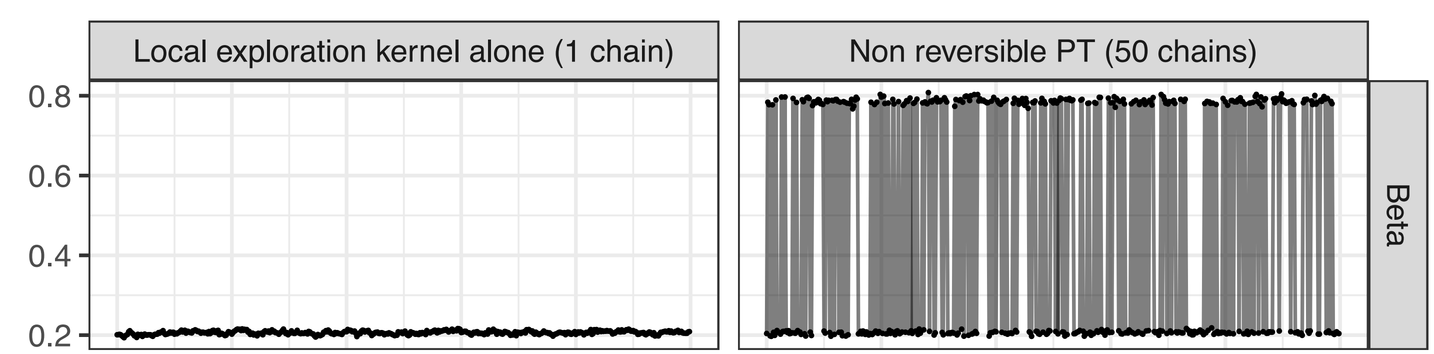
Ising Model
(d = 25)



Copy number inference
(d = 30)

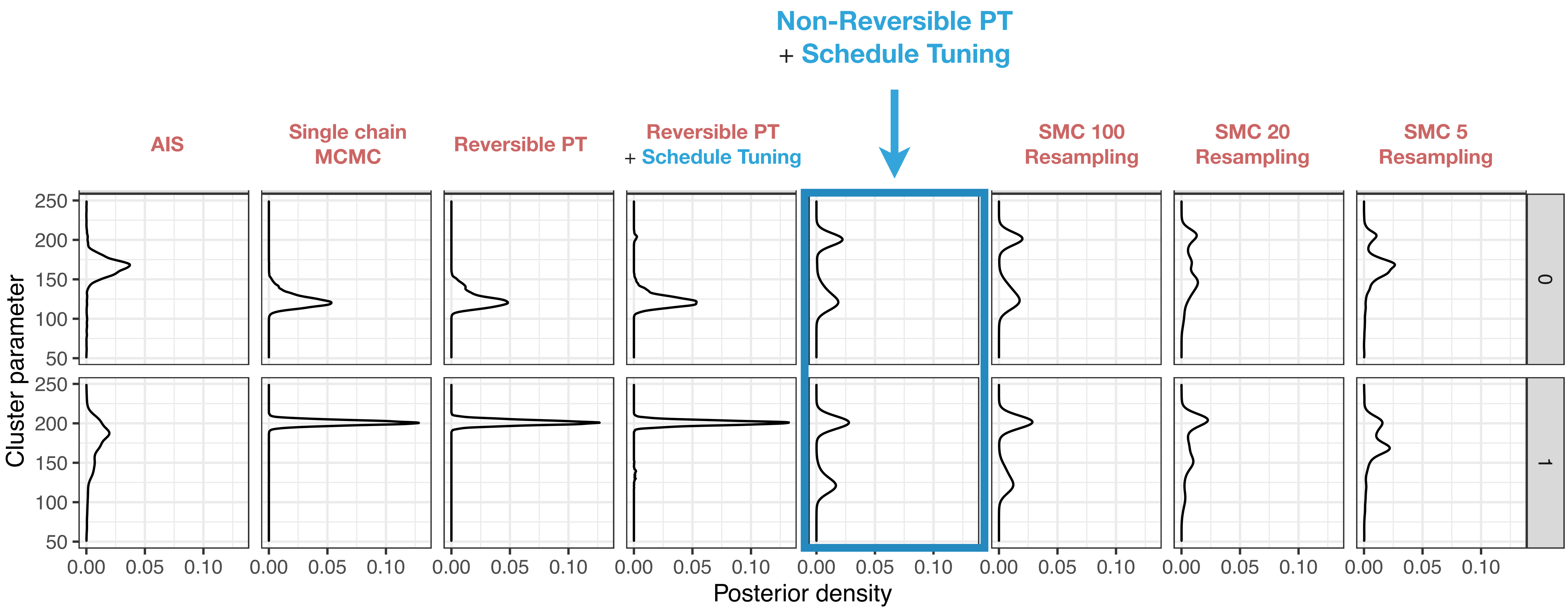


ODE parameter inference
(d = 9)



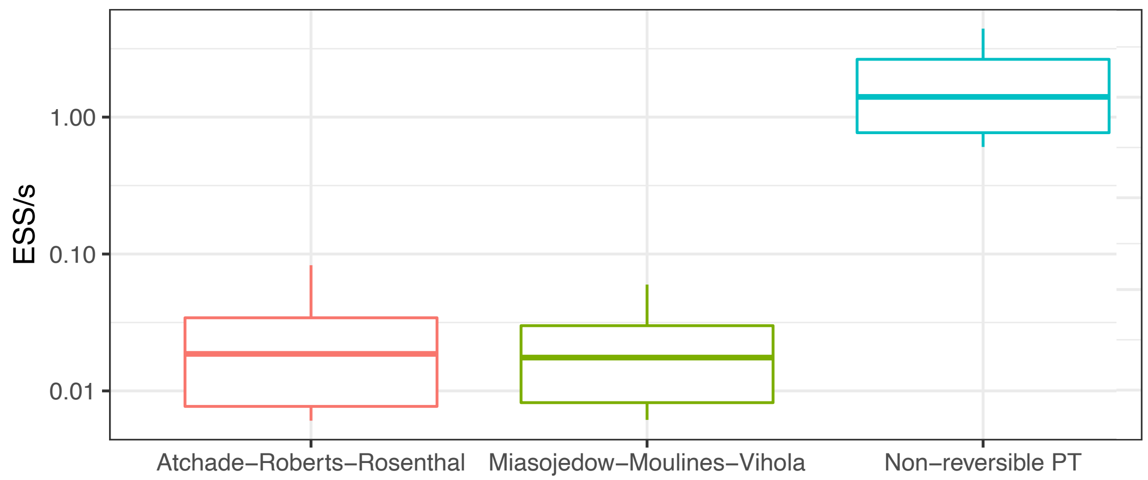
COMPARISON TO STATE OF THE ART

Bayesian Mixture Model (d = 155)

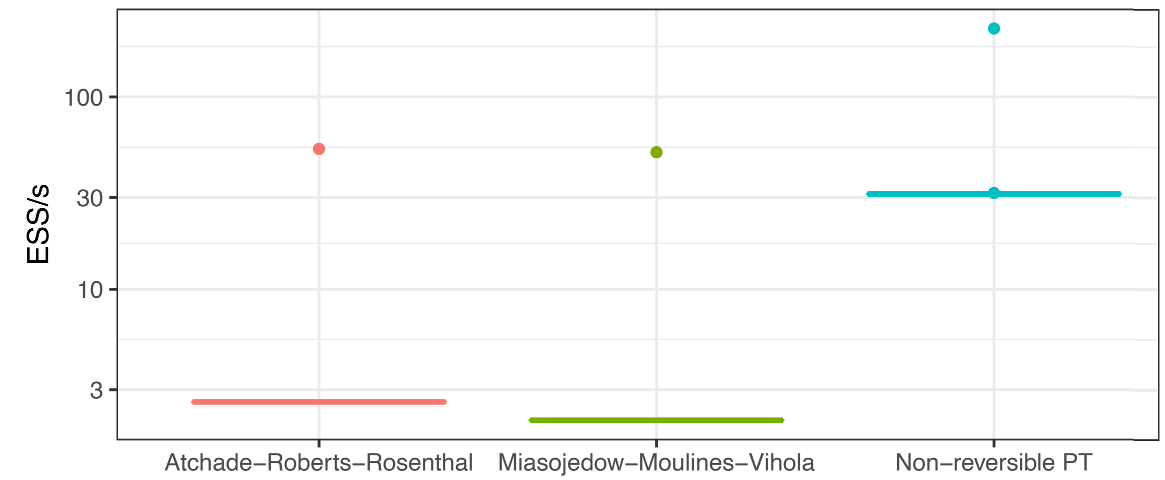


ESS COMPARED TO REVERSIBLE PT

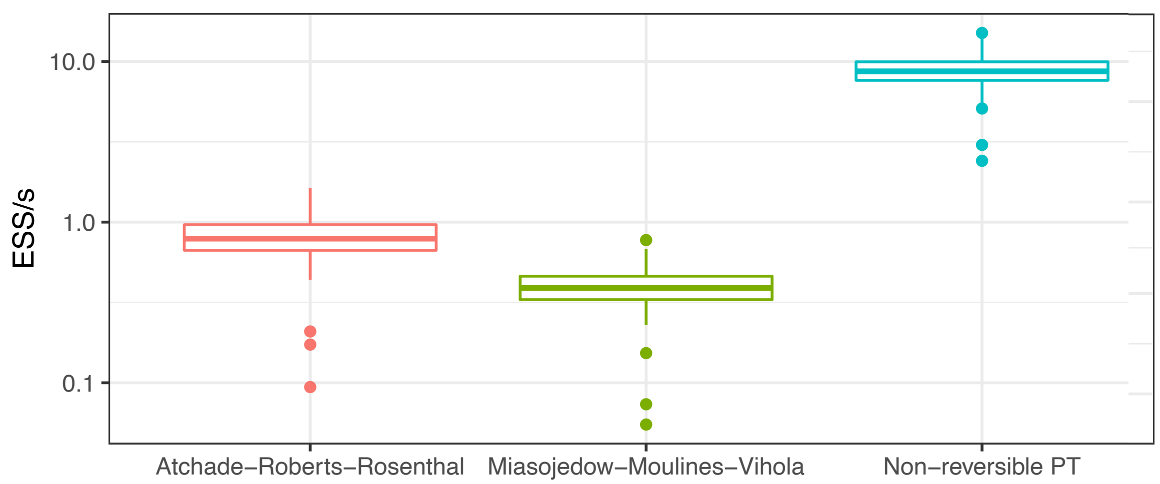
Spike–and–Slab (RMS Titanic passengers dataset)



Ising



Hierarchical model (rocket launches dataset 1957–2017)



Wright–Fisher SDE

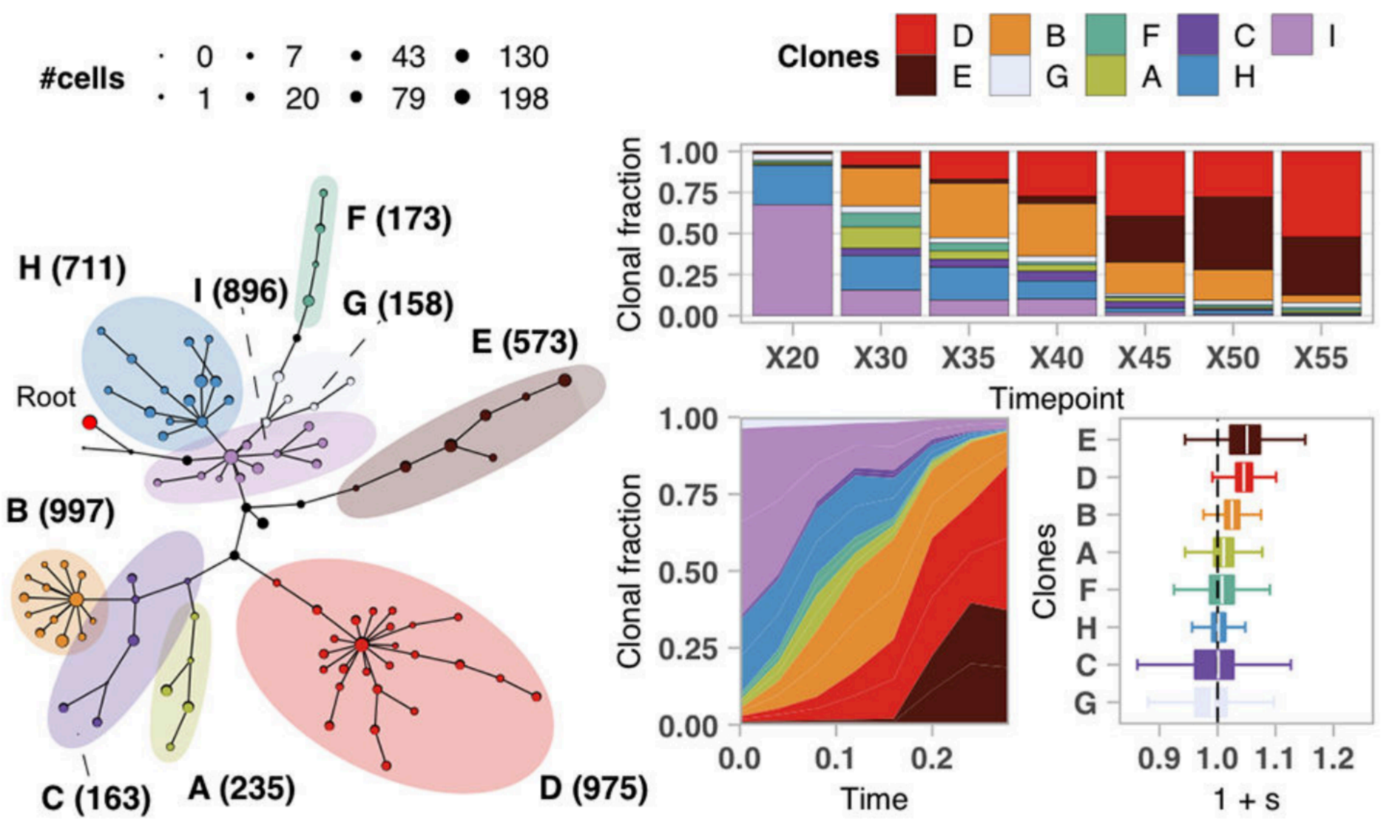
State of the art for Reversible PT

Non-Reversible PT

APPLICATIONS

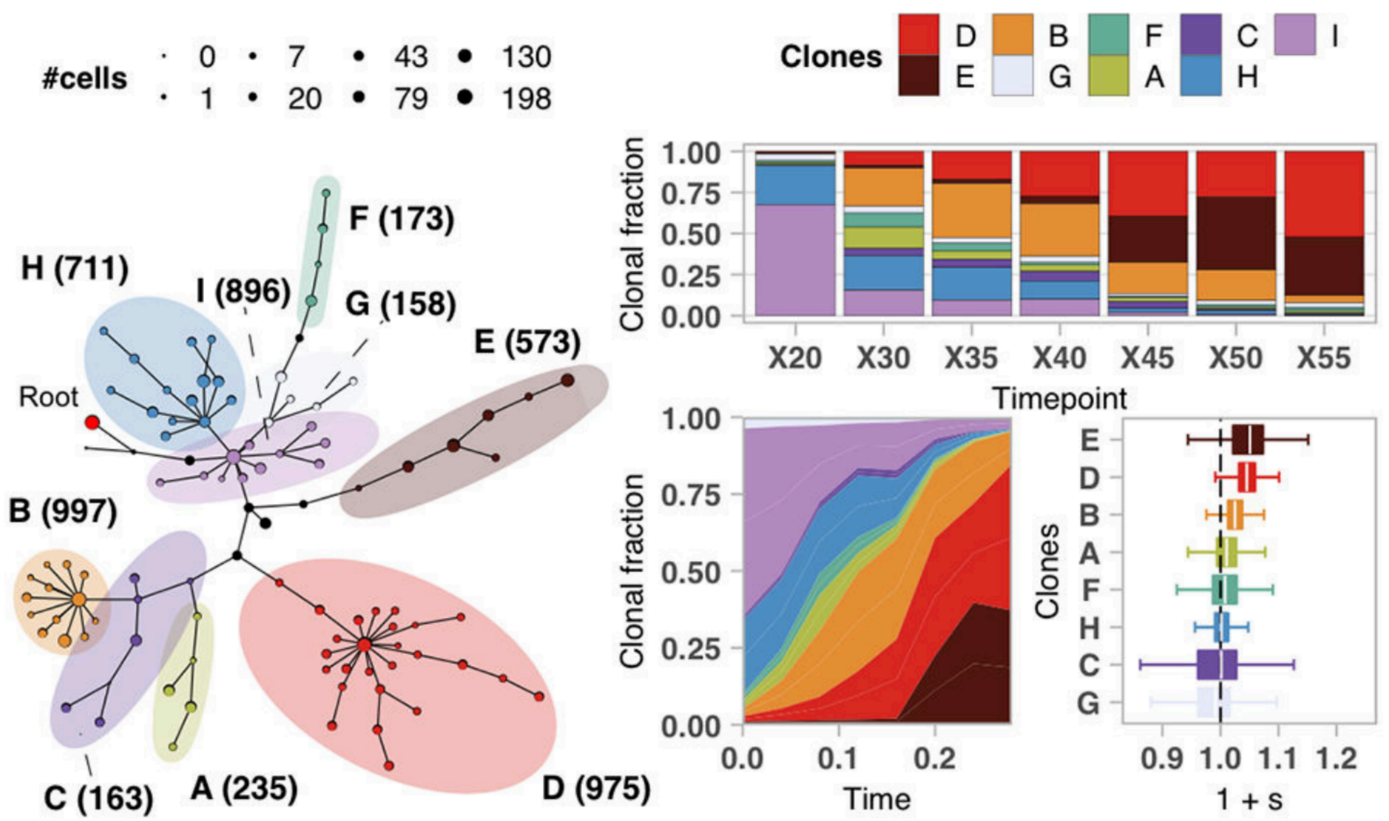
APPLICATIONS

- BC Cancer Research Center and MSK used NRPT to model time-series of single-cell cancer genomes
  (Dorri et al 2020, Salehi et al, 2021) in Nature



APPLICATIONS

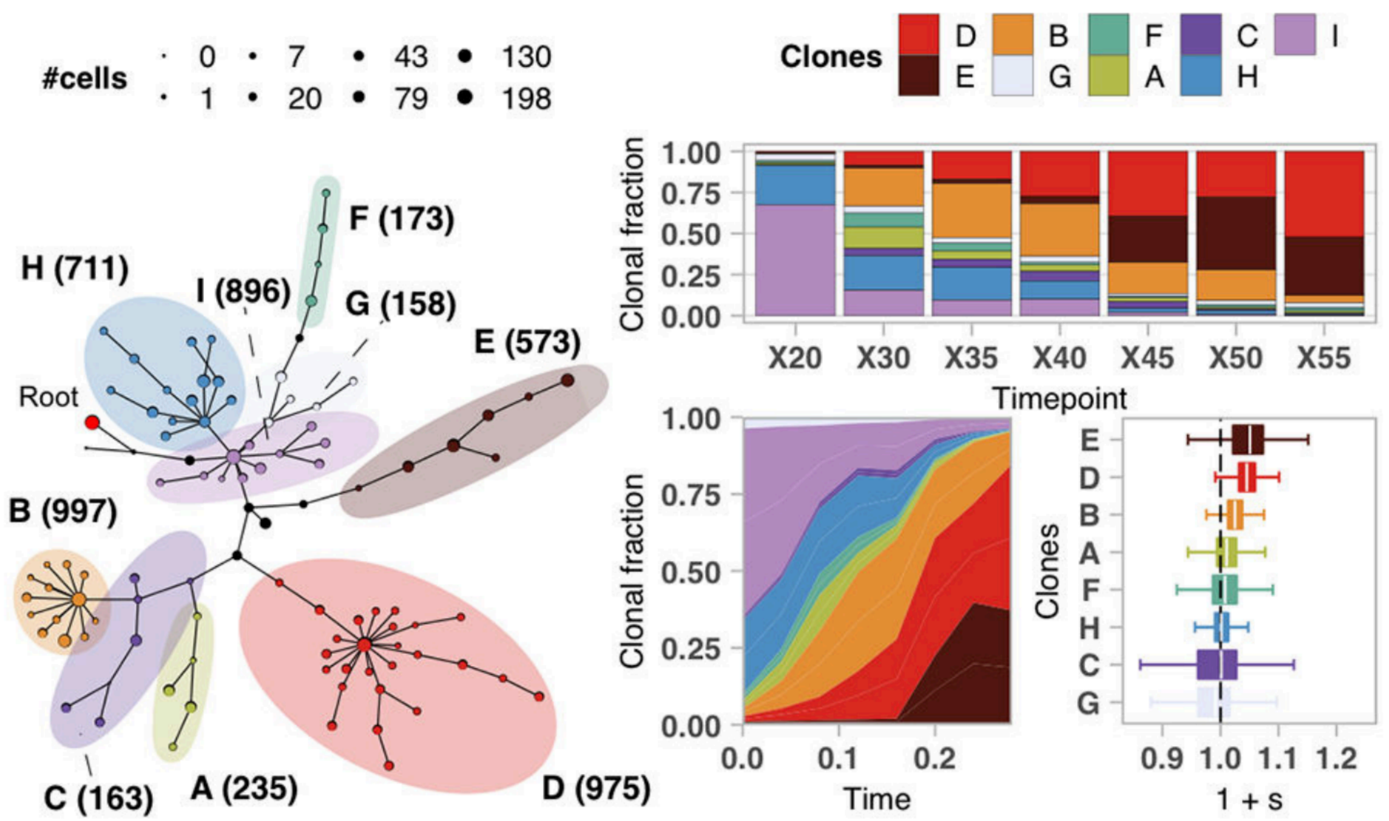
- BC Cancer Research Center and MSK used NRPT to model time-series of single-cell cancer genomes
  (Dorri et al 2020, Salehi et al, 2021) in Nature



- NRPT useful inference engine for probabilistic programming languages (eg. Blang, Themis, turing.jl, tensorflow probability)

APPLICATIONS

- BC Cancer Research Center and MSK used NRPT to model time-series of single-cell cancer genomes
(Dorri et al 2020, Salehi et al, 2021) in Nature



- NRPT useful inference engine for probabilistic programming languages (eg. Blang, Themis, turing.jl, tensorflow probability)
- Google research used NRPT for tackle ill-conditioned Bayesian inverse problems applied to nuclear fusion *(Langmore, et al 2021)*

EVENT HORIZON TELESCOPE

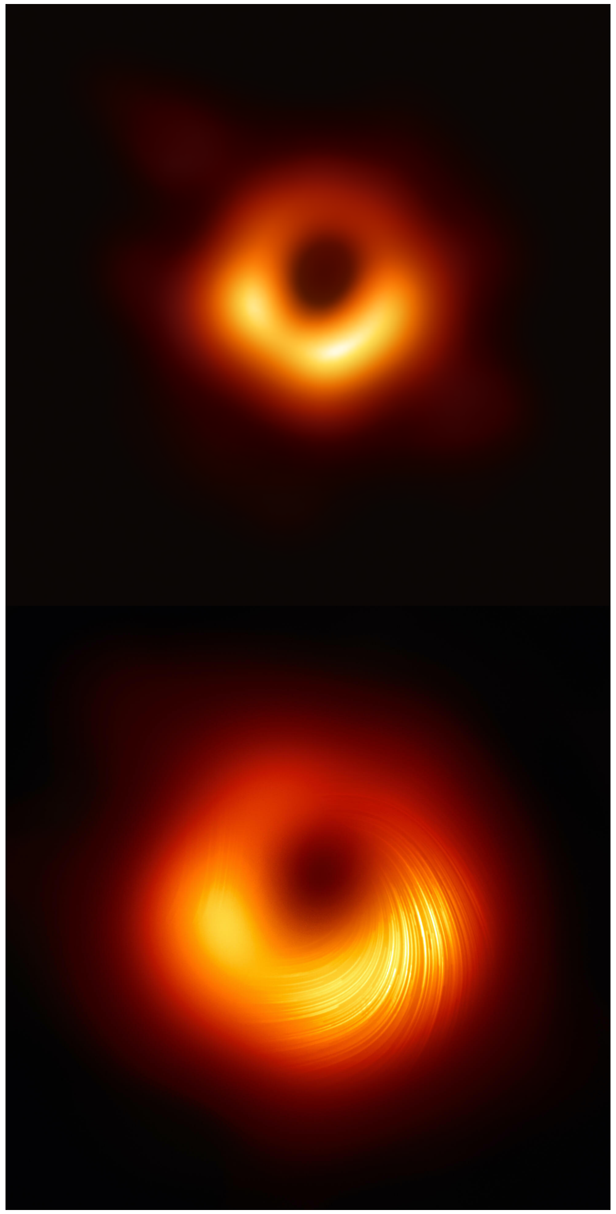
EVENT HORIZON TELESCOPE

- Event horizon telescope (EHT) used NRPT+HMC to more reliably recreate original photo of supermassive blackhole at M87 within 2% of computation budget (Tiede, 2021)

EVENT HORIZON TELESCOPE

- Event horizon telescope (EHT) used NRPT+HMC to more reliably recreate original photo of supermassive blackhole at M87 within 2% of computation budget (Tiede, 2021)

- NRPT one of the computational advancements required to discover magnetic polarization in M87 (EHT, 2021)

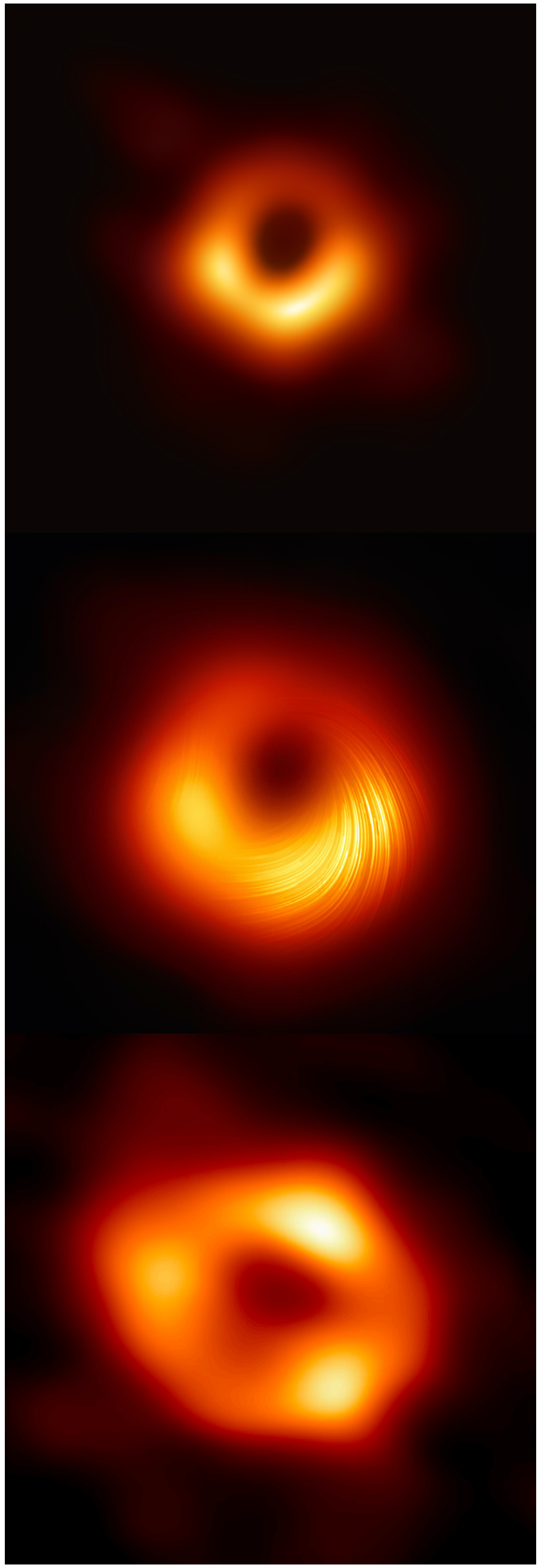


EVENT HORIZON TELESCOPE

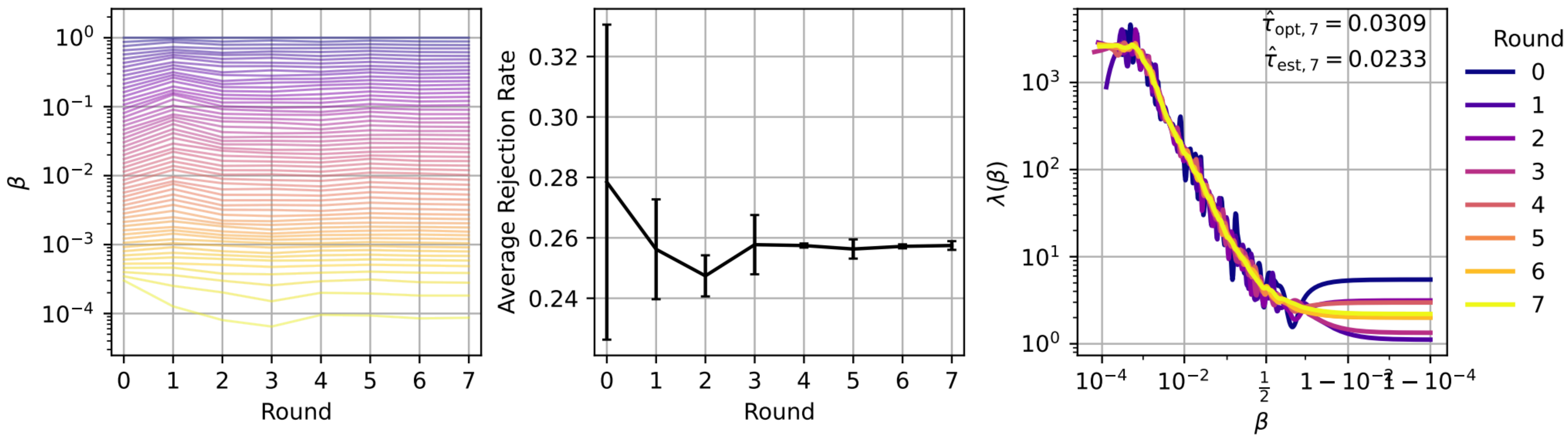
- Event horizon telescope (EHT) used NRPT+HMC to more reliably recreate original photo of supermassive blackhole at M87 within 2% of computation budget
  (Tiede, 2021)

- NRPT one of the computational advancements required to discover magnetic polarization in M87
  (EHT, 2021)

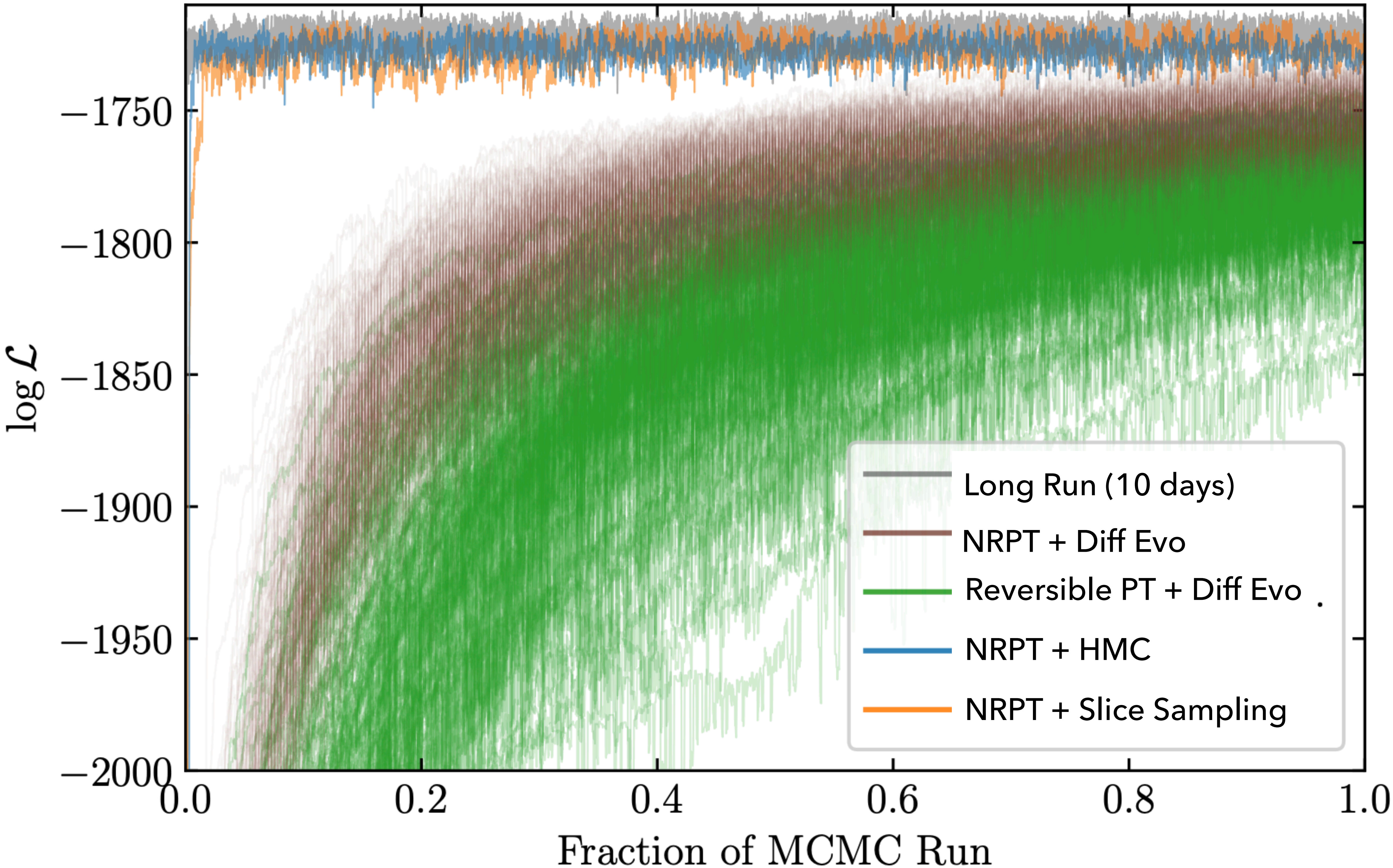
- NRPT used to capture photo of Sagittarius A*, the suppermassive blackhole in center of Milky Way
  (EHT, 2022)



NRPT FOR BLACK HOLE PHOTO



NRPT FOR BLACK HOLE PHOTO



REJECTION RATE AS A DIVERGENCE

REJECTION RATE AS A DIVERGENCE

- **Recall:** Performance of PT with linear path (i.e. $\pi_\beta \propto \pi_0^{1-\beta}\pi_1^{\beta}$) depends on communication barrier

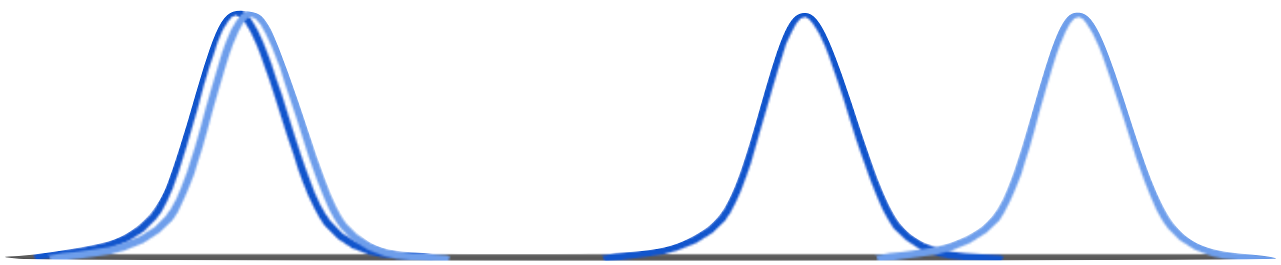
$$\tau = \frac{1}{2 + 2\Lambda} \qquad \Lambda \approx \sum_{n=0}^{n-1} r_n$$

REJECTION RATE AS A DIVERGENCE

- **Recall:** Performance of PT with linear path (i.e. $\pi_\beta \propto \pi_0^{1-\beta}\pi_1^{\beta}$) depends on communication barrier

$$\tau = \frac{1}{2+2\Lambda} \qquad \Lambda \approx \sum_{n=0}^{n-1} r_n$$

- Can show rejection rate measures the "overlap" between two distributions



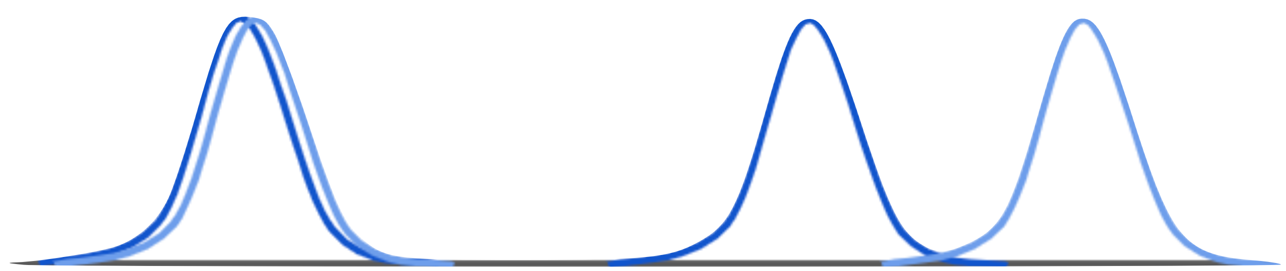
$r_n \approx 0$ $\qquad$ $r_n \approx 1$

REJECTION RATE AS A DIVERGENCE

- **Recall:** Performance of PT with linear path (i.e. $\pi_\beta \propto \pi_0^{1-\beta}\pi_1^{\beta}$) depends on communication barrier

$$\tau = \frac{1}{2+2\Lambda} \qquad \Lambda \approx \sum_{n=0}^{n-1} r_n$$

- Can show rejection rate measures the "overlap" between two distributions



$$r_n \approx 0 \qquad\qquad r_n \approx 1$$

- Rejection rate is a divergence measuring overlap between distributions

$$r_n = r\left(\pi_{\beta_{n-1}}, \pi_{\beta_n}\right)$$

$$r(p,q) = \|p \times q - q \times p\|_{\mathrm{TV}}$$

PT ON GENERAL ANNEALING PATHS

PT ON GENERAL ANNEALING PATHS

- The theory extends naturally to "general" annealing paths π_β

$$\lambda(\beta) = \lim_{\Delta\beta \to 0} \frac{r(\pi_\beta, \pi_{\beta+\Delta\beta})}{|\Delta\beta|}$$

$$\Lambda = \int_0^1 \lambda(\beta) \mathrm{d}\beta$$

PT ON GENERAL ANNEALING PATHS

- The theory extends naturally to "general" annealing paths π_β

$$\lambda(\beta) = \lim_{\Delta\beta \to 0} \frac{r(\pi_\beta, \pi_{\beta+\Delta\beta})}{|\Delta\beta|}$$

$$\Lambda = \int_0^1 \lambda(\beta)\mathrm{d}\beta$$

- Optimal performance controlled by communication barrier functional

$$\tau = \frac{1}{2+2\Lambda} \qquad \Lambda \approx \sum_{n=0}^{n-1} r_n$$

PT ON GENERAL ANNEALING PATHS

- The theory extends naturally to "general" annealing paths π_β

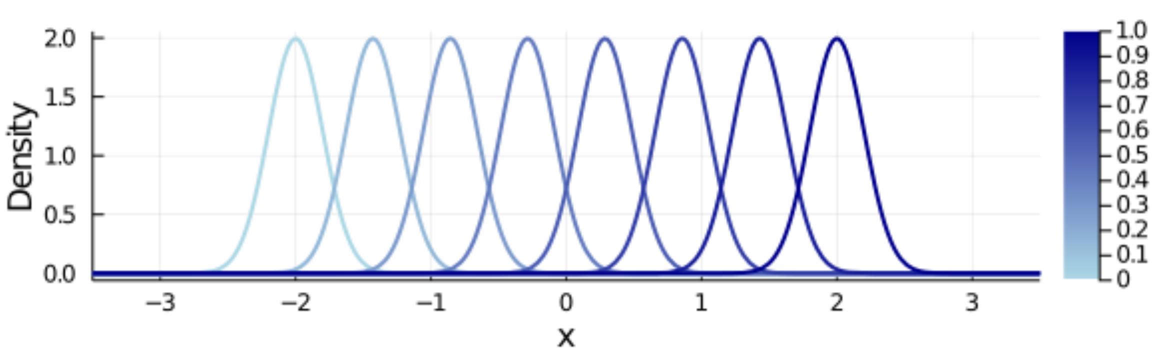
$$\lambda(\beta) = \lim_{\Delta\beta \to 0} \frac{r(\pi_\beta, \pi_{\beta+\Delta\beta})}{|\Delta\beta|}$$

$$\Lambda = \int_0^1 \lambda(\beta) \mathrm{d}\beta$$

- Optimal performance controlled by communication barrier functional

$$\tau = \frac{1}{2 + 2\Lambda} \qquad \Lambda \approx \sum_{n=0}^{n-1} r_n$$

- Can we do better than the linear path $\pi_\beta \propto \pi_0^{1-\beta}\pi_1^\beta$?



PT ON GENERAL ANNEALING PATHS

- The theory extends naturally to "general" annealing paths π_β

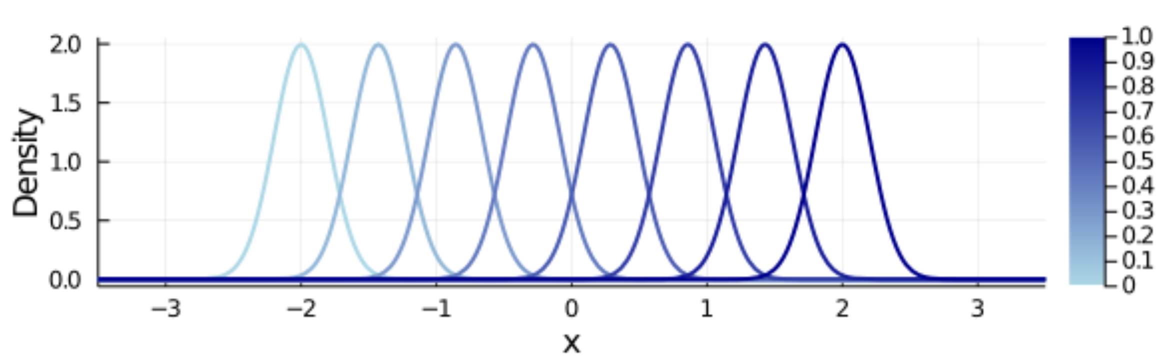
$$\lambda(\beta) = \lim_{\Delta\beta \to 0} \frac{r(\pi_\beta, \pi_{\beta+\Delta\beta})}{|\Delta\beta|}$$

$$\Lambda = \int_0^1 \lambda(\beta) \mathrm{d}\beta$$

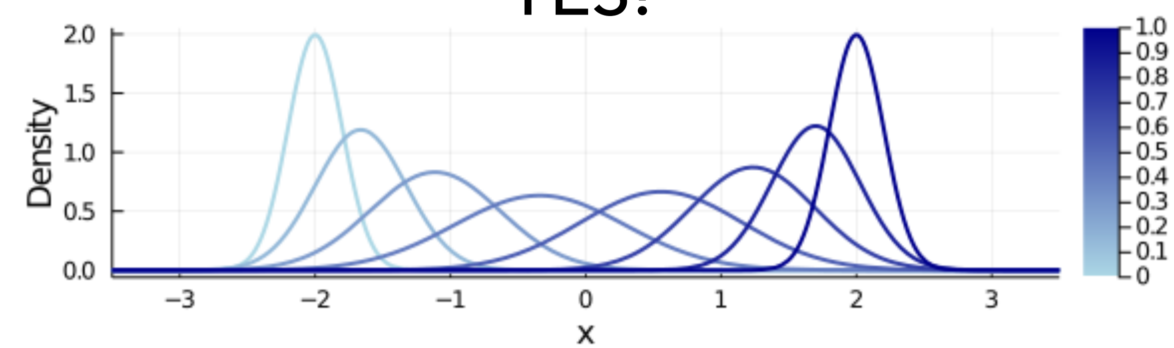
- Optimal performance controlled by communication barrier functional

$$\tau = \frac{1}{2 + 2\Lambda} \qquad \Lambda \approx \sum_{n=0}^{n-1} r_n$$

- Can we do better than the linear path $\pi_\beta \propto \pi_0^{1-\beta} \pi_1^\beta$?



YES!

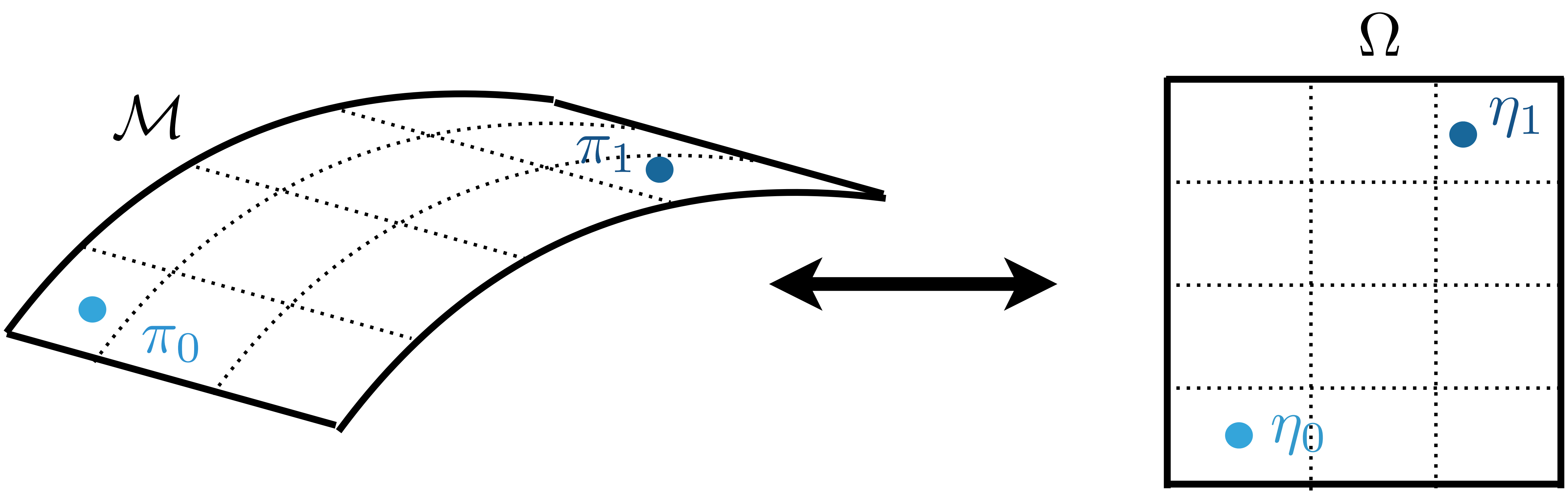


ANNEALING PATHS ON PARAMETRIC FAMILIES

ANNEALING PATHS ON PARAMETRIC FAMILIES

- Suppose $\pi_0, \pi_1 \in \mathcal{M}$ is a parametric family of densities supported on $\mathcal{X}$

$$\mathcal{M} = \{p_\eta : \eta \in \Omega \subset \mathbb{R}^d\}$$



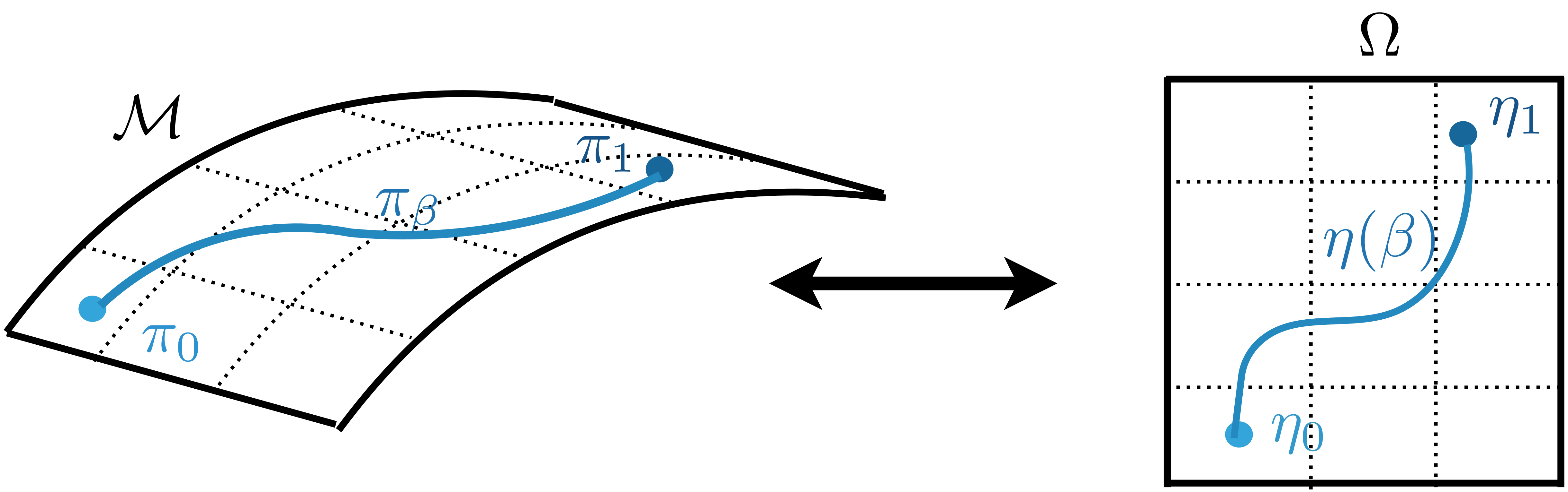
ANNEALING PATHS ON PARAMETRIC FAMILIES

- Suppose $\pi_0, \pi_1 \in \mathcal{M}$ is a parametric family of densities supported on $\mathcal{X}$

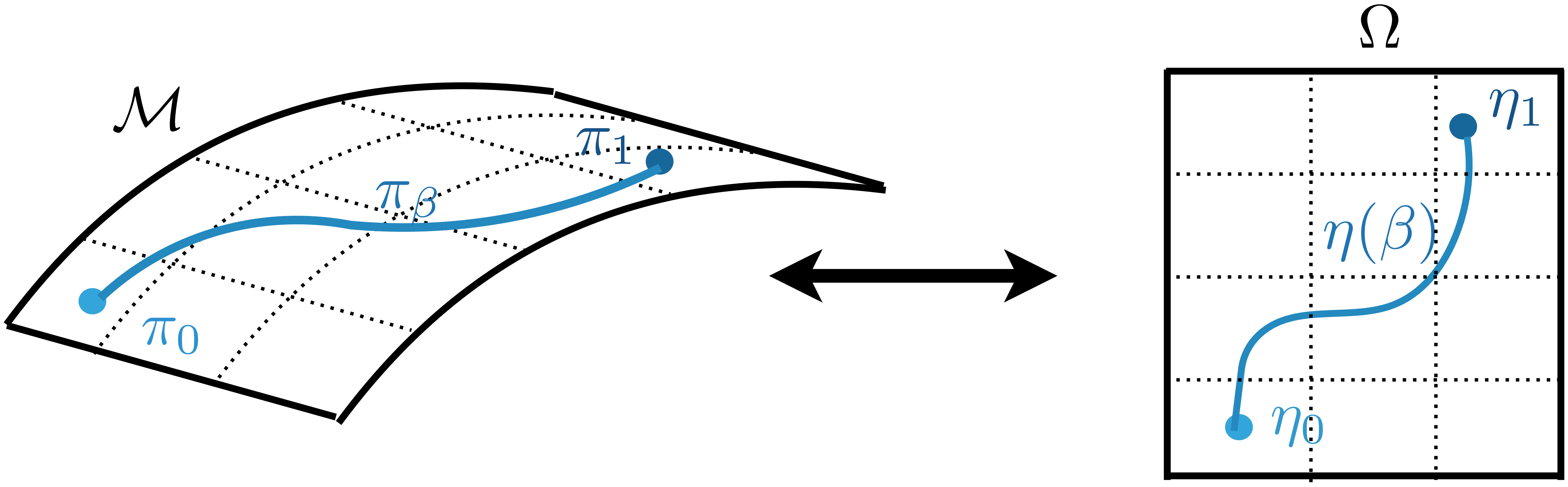
$$\mathcal{M} = \{p_\eta : \eta \in \Omega \subset \mathbb{R}^d\}$$

- An annealing path in $\mathcal{M}$ is function of densities connecting π_0 and π_1

$$\pi_\beta = p_{\eta(\beta)}$$

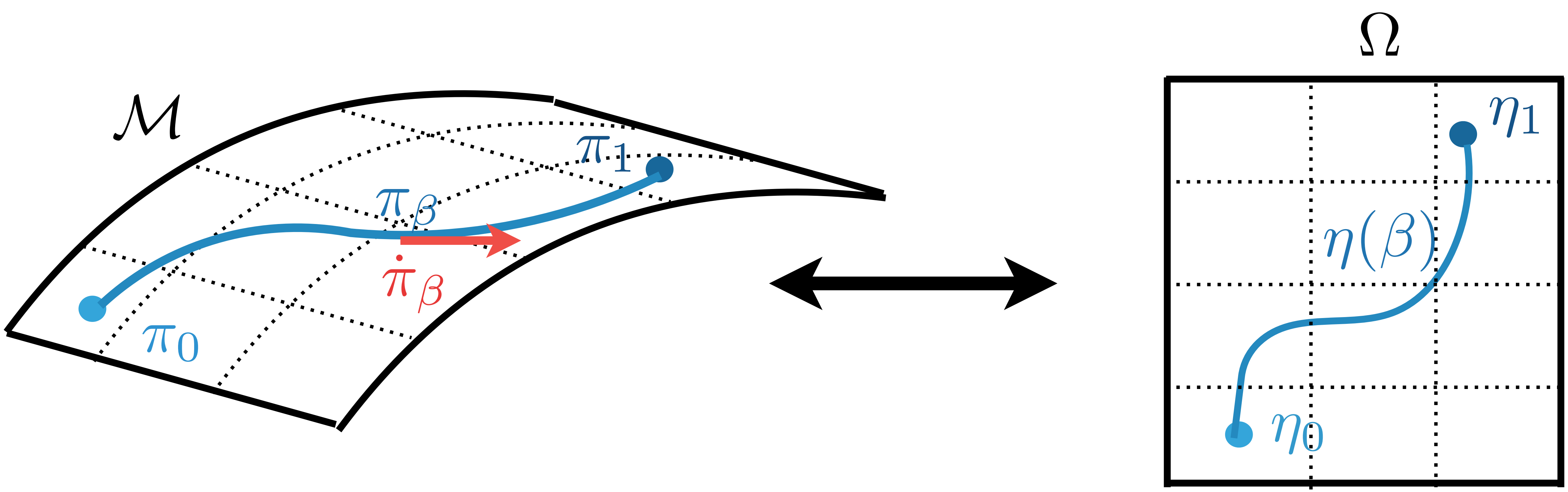


DIFFERENTIABLE ANNEALING PATHS



- We will say the path is differentiable with **velocity** equal to:

$$\dot{\pi}_\beta(x) = \frac{\mathrm{d} \log \pi_\beta(x)}{\mathrm{d}\beta}$$



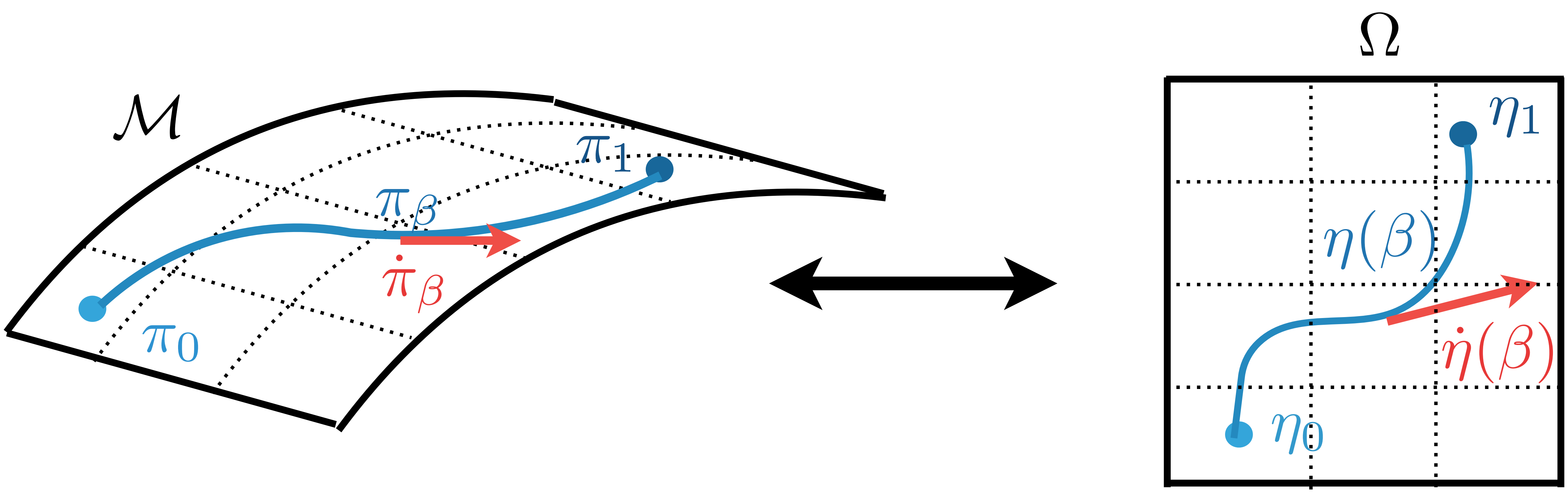
DIFFERENTIABLE ANNEALING PATHS

- We will say the path is differentiable with **velocity** equal to:

$$\dot{\pi}_\beta(x) = \frac{\mathrm{d}\log \pi_\beta(x)}{\mathrm{d}\beta}$$

- Differentiable annealing paths in $\mathcal{M}$ are in bijective correspondence to differentiable parametric paths in Ω via chain rule.

$$\dot{\pi}_\beta = \dot{\eta}(\beta)^T S_{\eta(\beta)} \qquad S_\eta(x) = \nabla_\eta \log p_\eta(x)$$



GEOMETRY OF COMMUNICATION BARRIER

GEOMETRY OF COMMUNICATION BARRIER

Theorem: Local and global communication barriers are "speed" and "length" of the path

$$\lambda(\beta) = \|\dot{\pi}_\beta\|_{\pi_\beta}, \quad \Lambda = \int_0^1 \|\dot{\pi}_\beta\|_{\pi_\beta} \mathrm{d}\beta$$

GEOMETRY OF COMMUNICATION BARRIER

Theorem: Local and global communication barriers are "speed" and "length" of the path

$$\lambda(\beta) = \|\dot{\pi}_\beta\|_{\pi_\beta}, \quad \Lambda = \int_0^1 \|\dot{\pi}_\beta\|_{\pi_\beta} \mathrm{d}\beta$$

- Optimal algorithm corresponds to path minimising length, i.e. **Geodesics!**

GEOMETRY OF COMMUNICATION BARRIER

Theorem: Local and global communication barriers are "speed" and "length" of the path

$$\lambda(\beta) = \|\dot{\pi}_\beta\|_{\pi_\beta}, \quad \Lambda = \int_0^1 \|\dot{\pi}_\beta\|_{\pi_\beta} \mathrm{d}\beta$$

- Optimal algorithm corresponds to path minimising length, i.e. **Geodesics!**
- Note norm is **not** quadratic

$$\|\dot{p}\|_p = \frac{1}{2}\mathbb{E}[\|\dot{p}(X) - \dot{p}(X')\|], \quad X, X' \overset{iid}{\sim} p$$

GEOMETRY OF COMMUNICATION BARRIER

Theorem: Local and global communication barriers are "speed" and "length" of the path

$$\lambda(\beta) = \|\dot{\pi}_\beta\|_{\pi_\beta}, \quad \Lambda = \int_0^1 \|\dot{\pi}_\beta\|_{\pi_\beta} \mathrm{d}\beta$$

- Optimal algorithm corresponds to path minimising length, i.e. **Geodesics!**
- Note norm is **not** quadratic

$$\|\dot{p}\|_p = \frac{1}{2}\mathbb{E}[\|\dot{p}(X) - \dot{p}(X')\|], \quad X, X' \overset{iid}{\sim} p$$

- We can find geodesics in Riemannian manifold $(\mathcal{M}, g)$, where g is the Fisher information metric

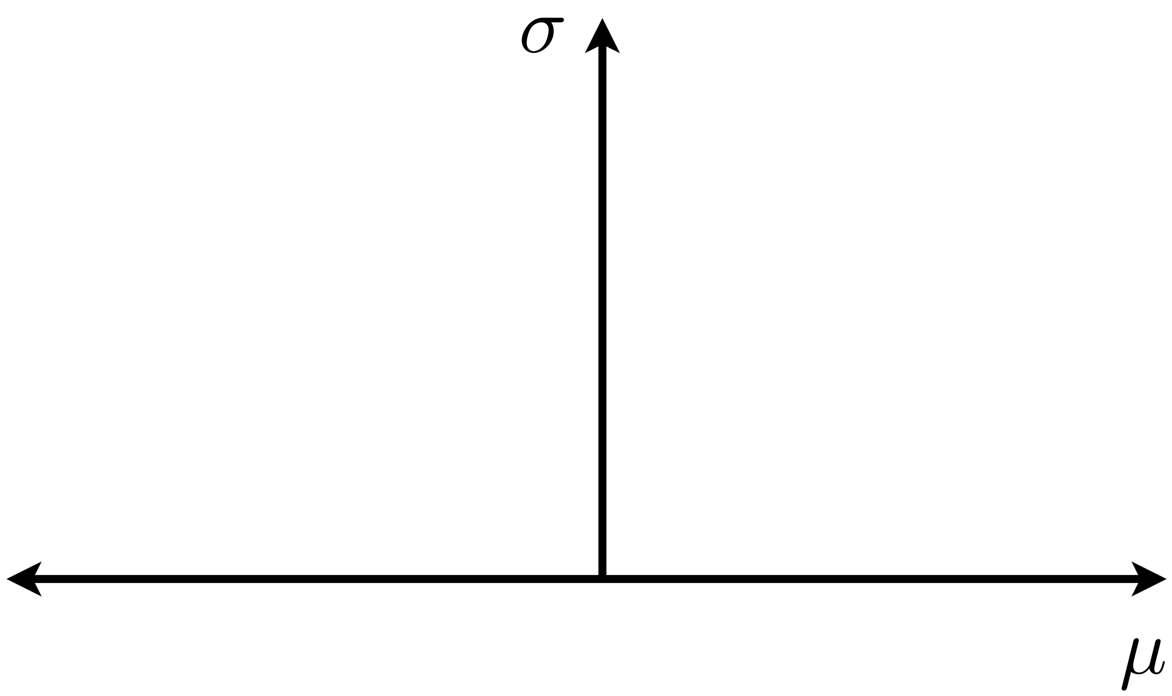
$$\lambda(\beta)^2 \leq \dot{\pi}_\beta^T g_{\pi_\beta} \pi_\beta \qquad\qquad g_p(\dot{p}) = \mathrm{Var}_p(\dot{p})$$

EXAMPLE GAUSSIAN

EXAMPLE GAUSSIAN

- For example univariate Gaussians:

$$\mathcal{M} = \{N(\mu, \sigma^2) : (\mu, \sigma) \in \mathbb{R} \times \mathbb{R}_+$$



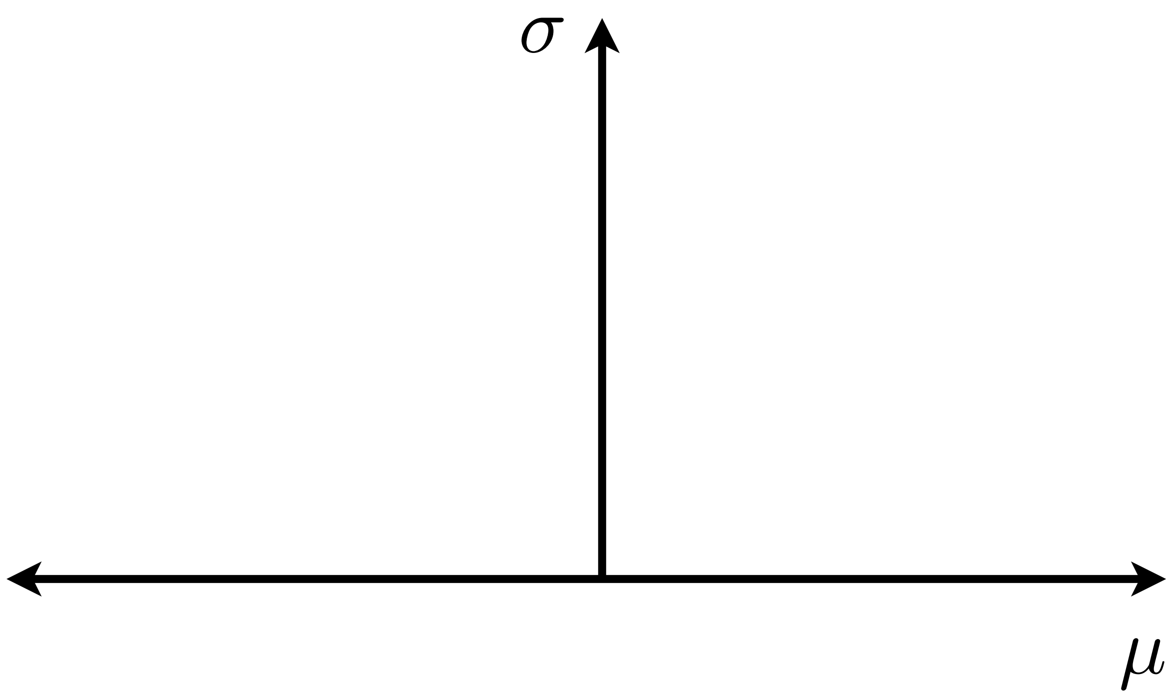
EXAMPLE GAUSSIAN

- For example univariate Gaussians:

$$\mathcal{M} = \{N(\mu, \sigma^2) : (\mu, \sigma) \in \mathbb{R} \times \mathbb{R}_+$$

- Fisher information corresponds to a scaled hyperbolic space

$$g_{(\mu,\sigma)} = \frac{1}{\sigma^2}\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$



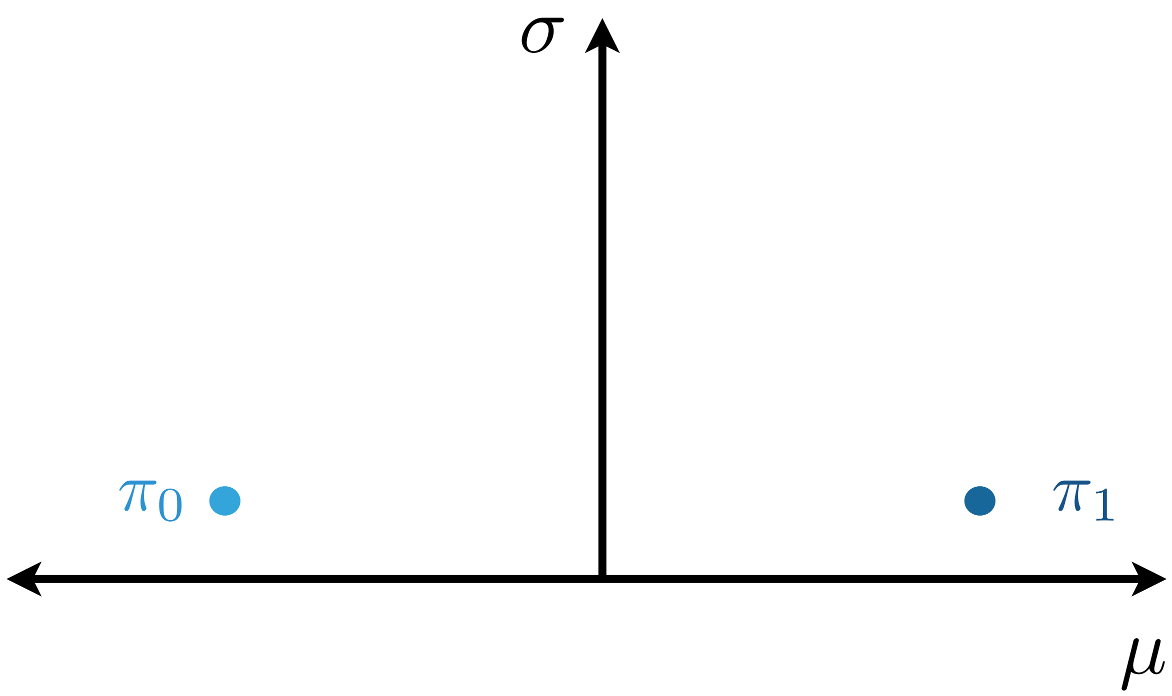
EXAMPLE GAUSSIAN

- For example univariate Gaussians:

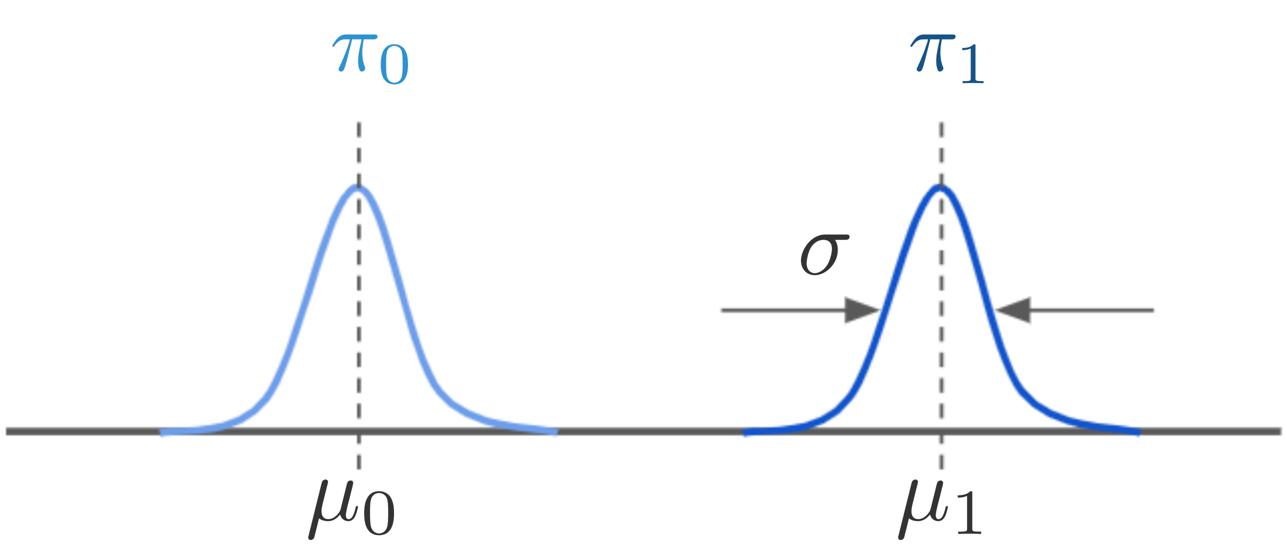
$$\mathcal{M} = \{N(\mu, \sigma^2) : (\mu, \sigma) \in \mathbb{R} \times \mathbb{R}_+$$

- Fisher information corresponds to a scaled hyperbolic space

$$g_{(\mu,\sigma)} = \frac{1}{\sigma^2} \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$



- Eg. Gaussian reference and target



EXAMPLE GAUSSIAN

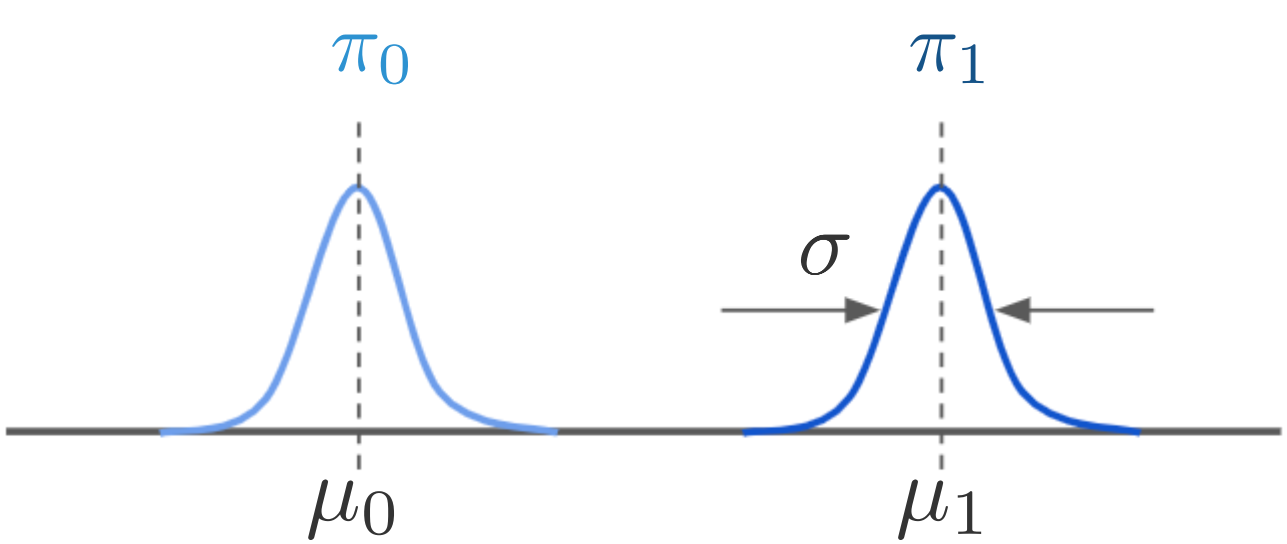
- For example univariate Gaussians:

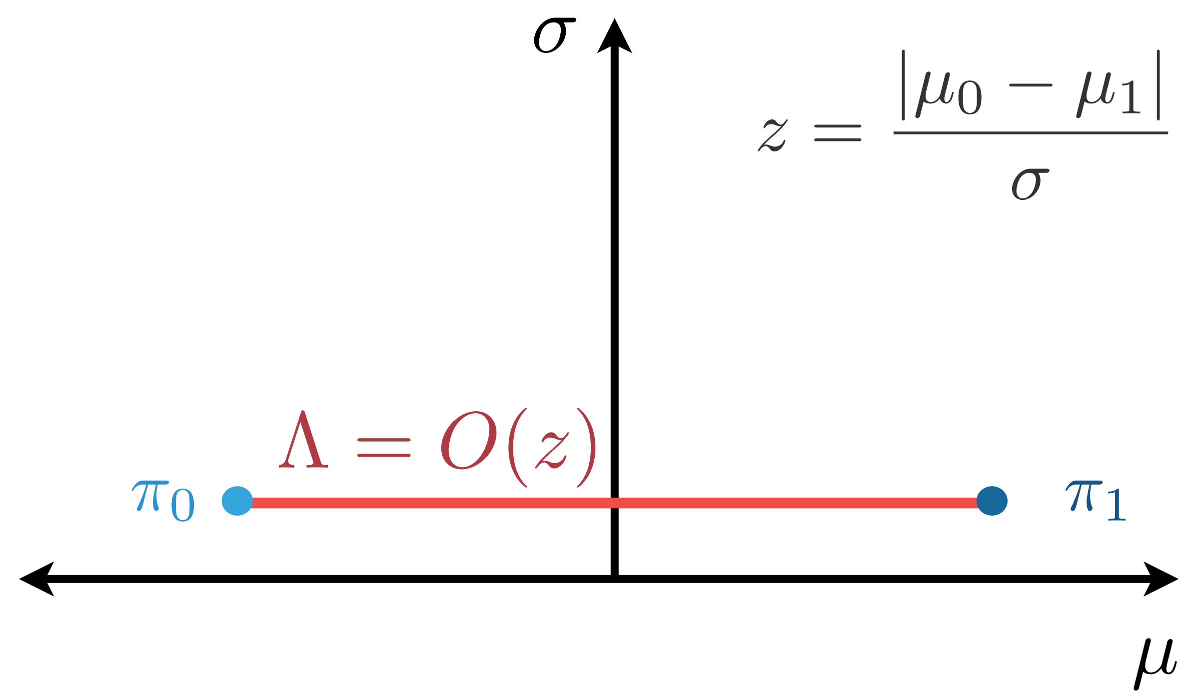
$$\mathcal{M} = \{N(\mu, \sigma^2) : (\mu, \sigma) \in \mathbb{R} \times \mathbb{R}_+$$

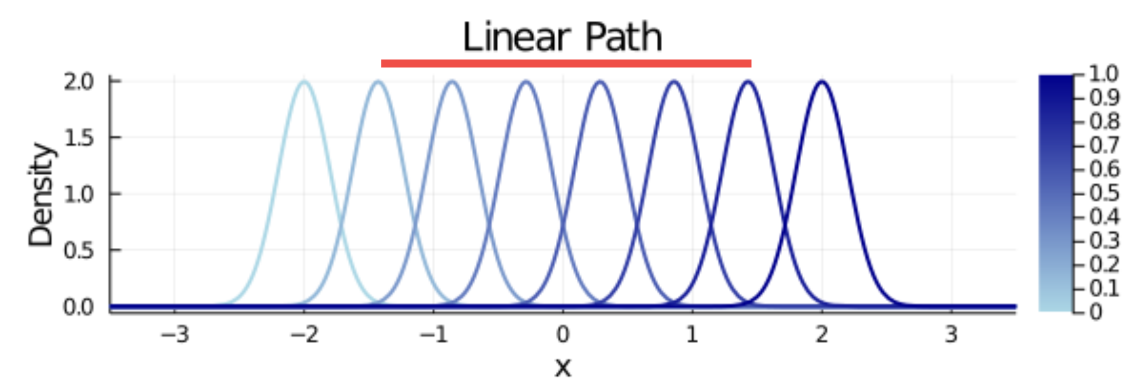
- Fisher information corresponds to a scaled hyperbolic space

$$g_{(\mu,\sigma)} = \frac{1}{\sigma^2}\begin{bmatrix}1 & 0\\ 0 & 2\end{bmatrix}$$

- Eg. Gaussian reference and target







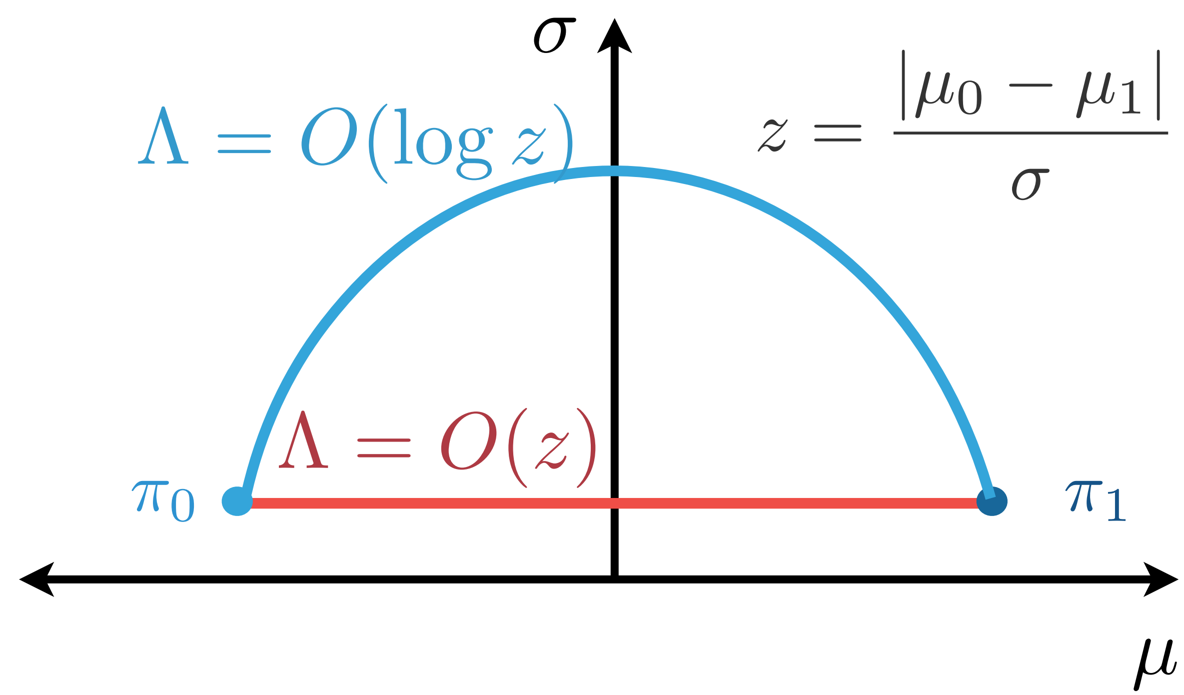
EXAMPLE GAUSSIAN

- For example univariate Gaussians:

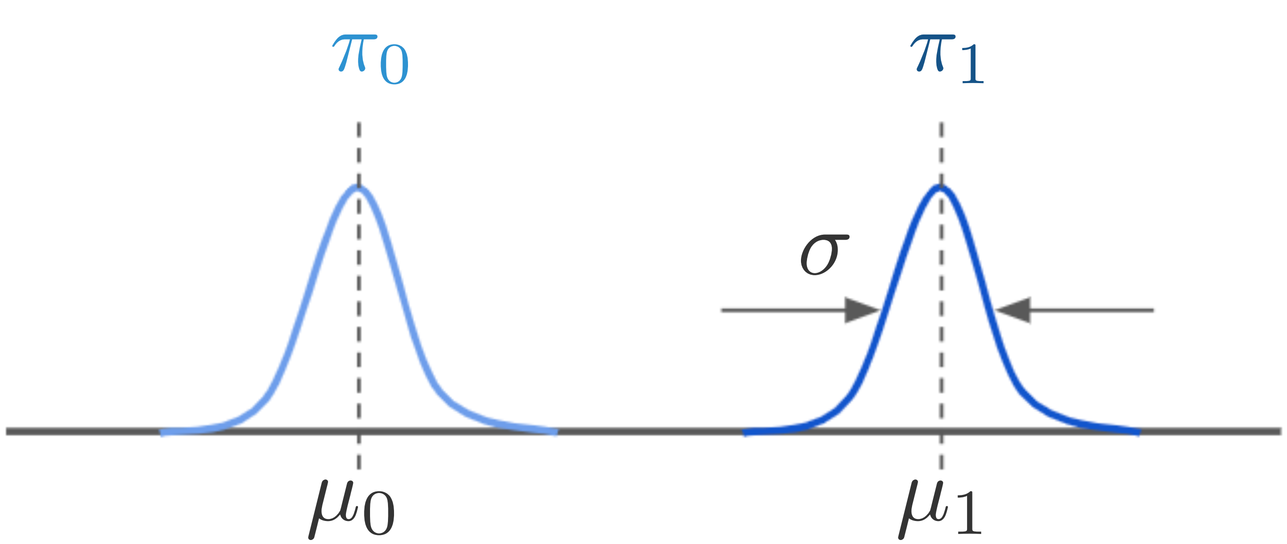
$$\mathcal{M} = \{N(\mu, \sigma^2) : (\mu, \sigma) \in \mathbb{R} \times \mathbb{R}_+$$

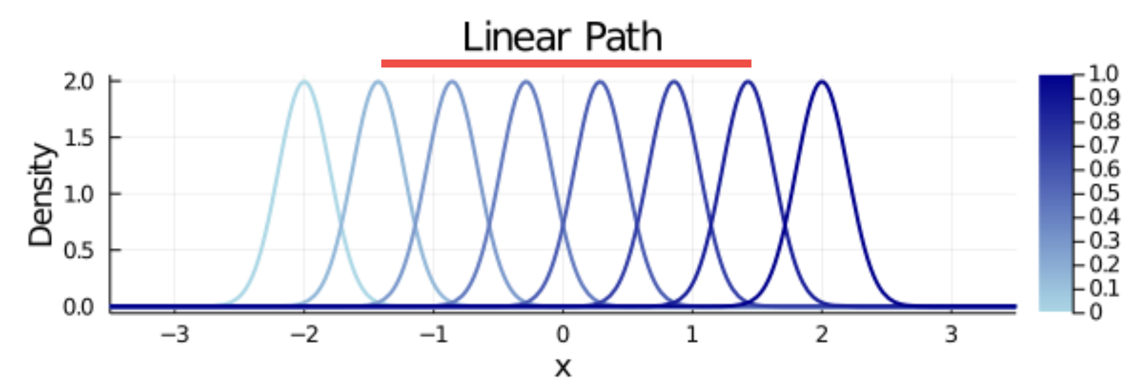
- Fisher information corresponds to a scaled hyperbolic space

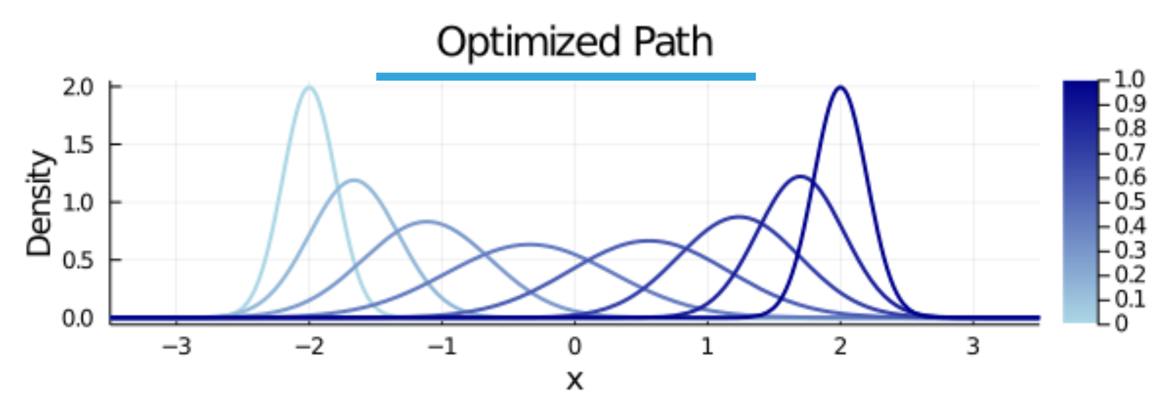
$$g_{(\mu,\sigma)} = \frac{1}{\sigma^2}\begin{bmatrix}1 & 0\\ 0 & 2\end{bmatrix}$$



- Eg. Gaussian reference and target







EXPONENTIAL ANNEALING FAMILY

EXPONENTIAL ANNEALING FAMILY

- **Deserata:**
  1. Linear path should be in the family
  2. Should not depend on details of reference or target
  3. Should allow for larger "overlap" than linear path

EXPONENTIAL ANNEALING FAMILY

- **Deserata:**
  1. Linear path should be in the family
  2. Should not depend on details of reference or target
  3. Should allow for larger "overlap" than linear path

- Exponential annealing family

$$\mathcal{M} = \{p_\eta : \eta \in \Omega\}$$

$$p_\eta(x) = \frac{1}{Z(\eta)} \pi_0(x)^{\eta_0} \pi_1(x)^{\eta_1}$$

EXPONENTIAL ANNEALING FAMILY

- **Deserata:**
  1. Linear path should be in the family
  2. Should not depend on details of reference or target
  3. Should allow for larger "overlap" than linear path
- Exponential annealing family

$$\mathcal{M} = \{p_\eta : \eta \in \Omega\}$$

$$p_\eta(x) = \frac{1}{Z(\eta)} \pi_0(x)^{\eta_0} \pi_1(x)^{\eta_1}$$

- Natural parameter space is convex and contains reference and target

$$\Omega = \{\eta : Z(\eta) < \infty\} \subset \mathbb{R}^2$$

EXPONENTIAL ANNEALING FAMILY

- **Deserata:**
  1. Linear path should be in the family
  2. Should not depend on details of reference or target
  3. Should allow for larger "overlap" than linear path

- Exponential annealing family

$$\mathcal{M} = \{p_\eta : \eta \in \Omega\}$$

$$p_\eta(x) = \frac{1}{Z(\eta)} \pi_0(x)^{\eta_0} \pi_1(x)^{\eta_1}$$

- Natural parameter space is convex and contains reference and target

$$\Omega = \{\eta : Z(\eta) < \infty\} \subset \mathbb{R}^2$$

- Linear annealing paths in $\mathcal{M}$ correspond to linear parametric paths in Ω

SPLINE PATH FAMILY

SPLINE PATH FAMILY

- All annealing paths correspond to some $\eta : [0, 1] \to \Omega$

$$\pi_\beta \propto \pi_0^{\eta_0(\beta)} \pi_1^{\eta_1(\beta)}$$

$$\eta(1) = (0, 1)$$

$$\eta(0) = (1, 0)$$

SPLINE PATH FAMILY

- All annealing paths correspond to some $\eta : [0, 1] \to \Omega$

$$\pi_\beta \propto \pi_0^{\eta_0(\beta)} \pi_1^{\eta_1(\beta)}$$

$$\eta(1) = (0, 1)$$

$$\eta(0) = (1, 0)$$

- Suppose $\eta(\beta)$ linear spline with K knots

SPLINE PATH FAMILY

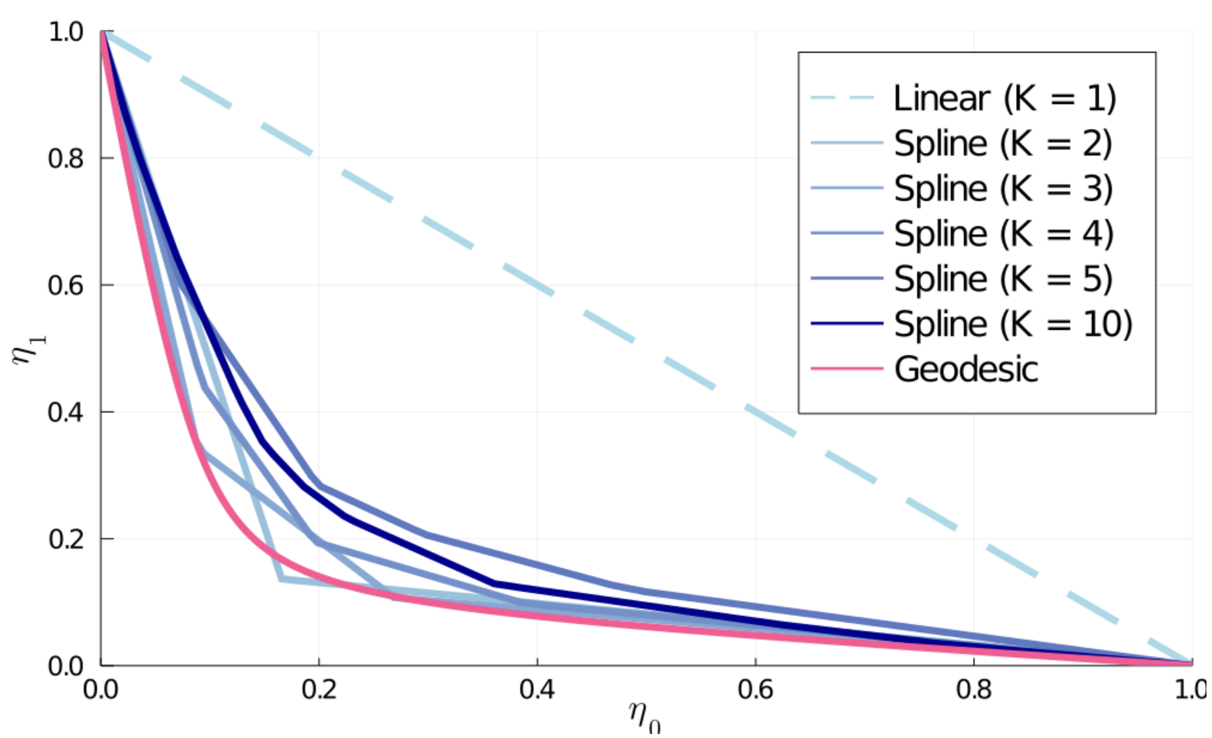
- All annealing paths correspond to some $\eta : [0,1] \to \Omega$

$$\pi_\beta \propto \pi_0^{\eta_0(\beta)} \pi_1^{\eta_1(\beta)}$$

$$\eta(1) = (0,1)$$

$$\eta(0) = (1,0)$$

- Suppose $\eta(\beta)$ linear spline with K knots



SPLINE PATH FAMILY

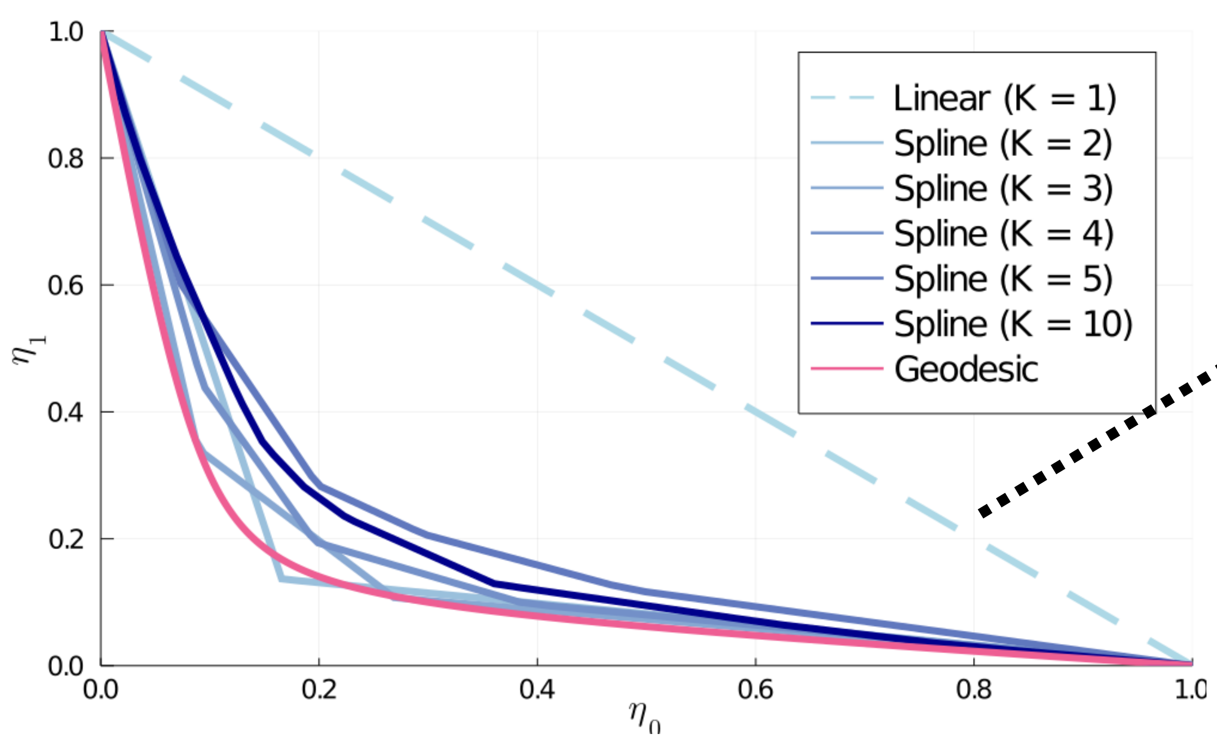
- All annealing paths correspond to some $\eta : [0,1] \to \Omega$

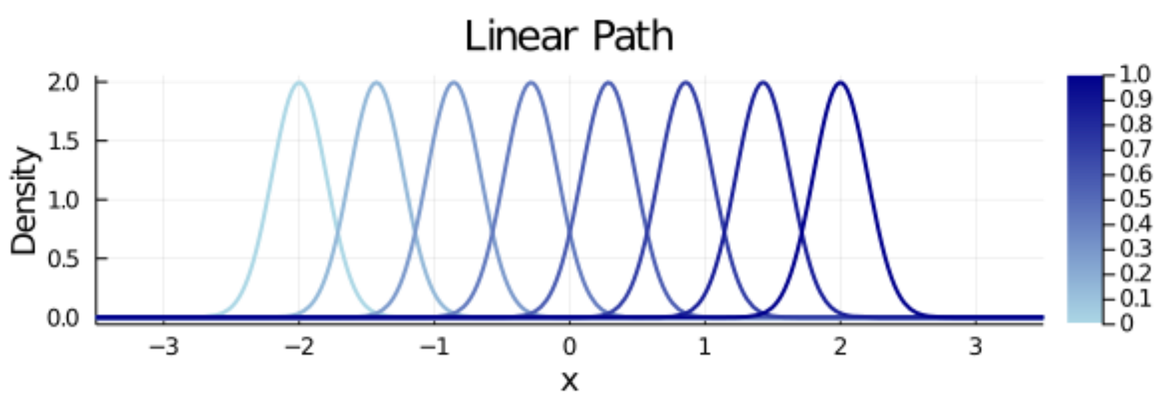
$$\pi_\beta \propto \pi_0^{\eta_0(\beta)} \pi_1^{\eta_1(\beta)}$$

$$\eta(1) = (0,1)$$

$$\eta(0) = (1,0)$$

- Suppose $\eta(\beta)$ linear spline with K knots





SPLINE PATH FAMILY

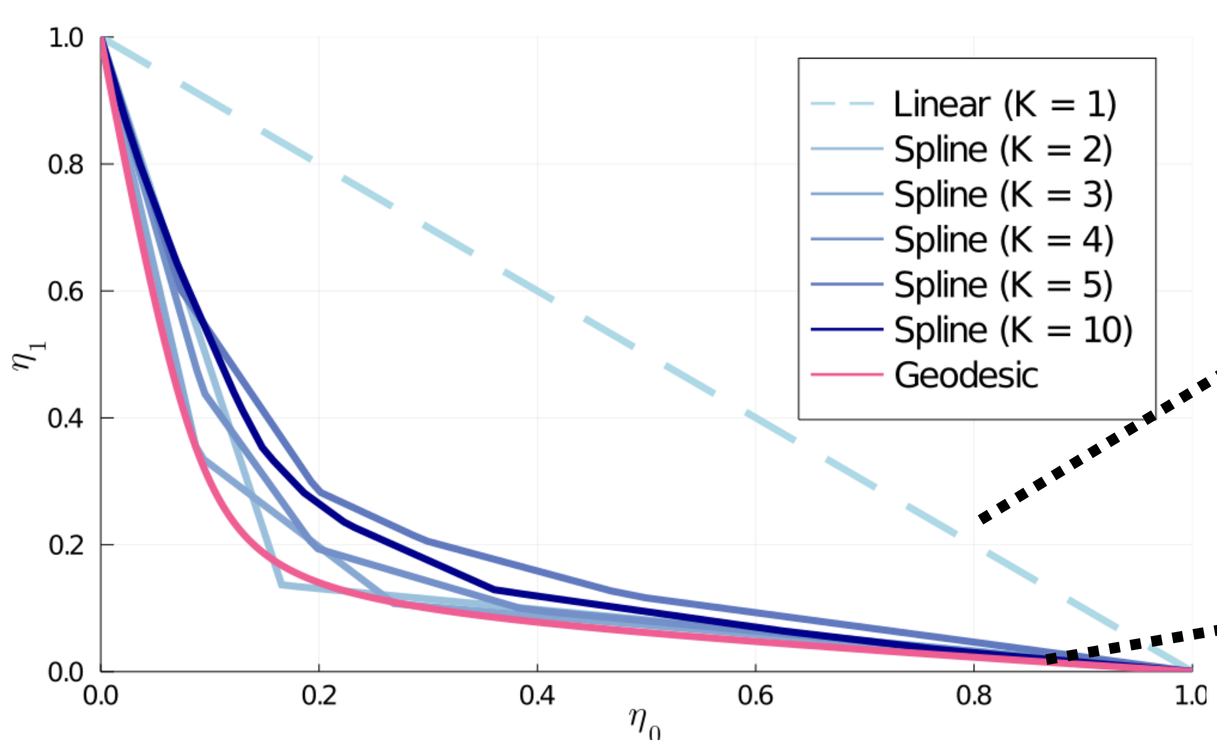
- All annealing paths correspond to some $\eta : [0,1] \to \Omega$

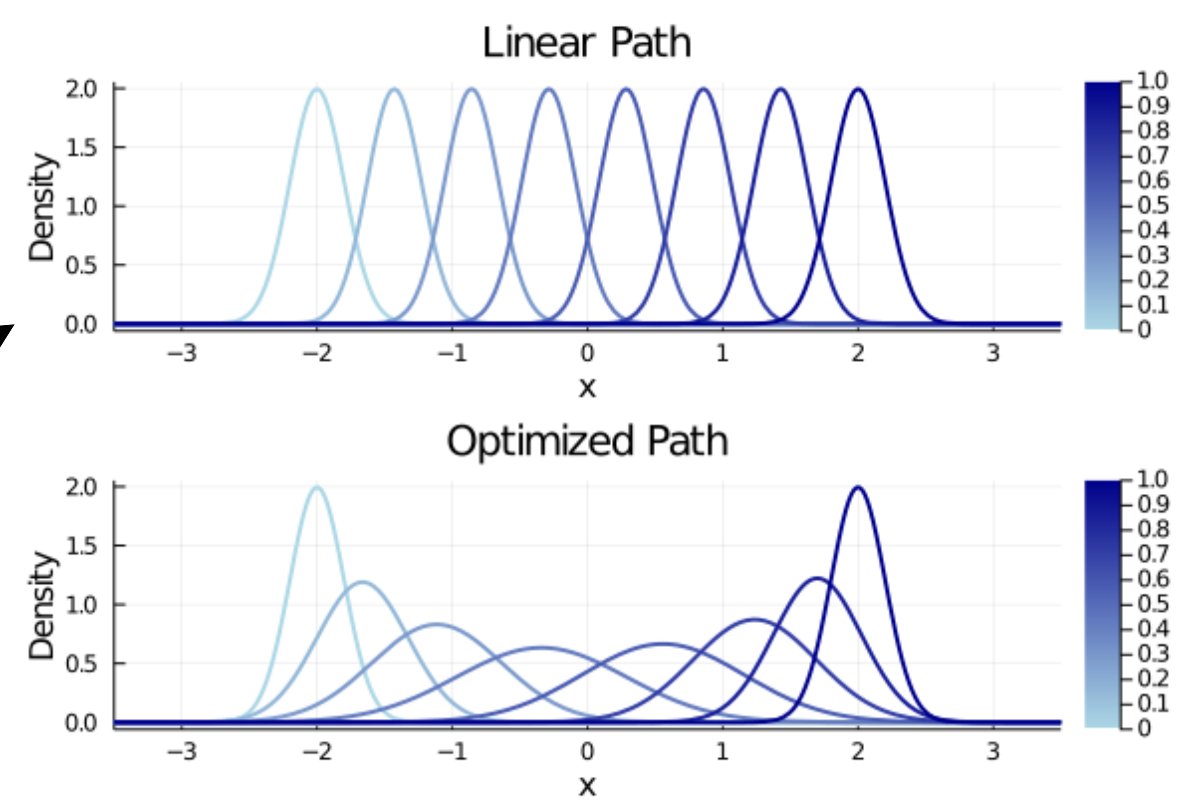
$$\pi_\beta \propto \pi_0^{\eta_0(\beta)} \pi_1^{\eta_1(\beta)}$$

$$\eta(1) = (0,1)$$

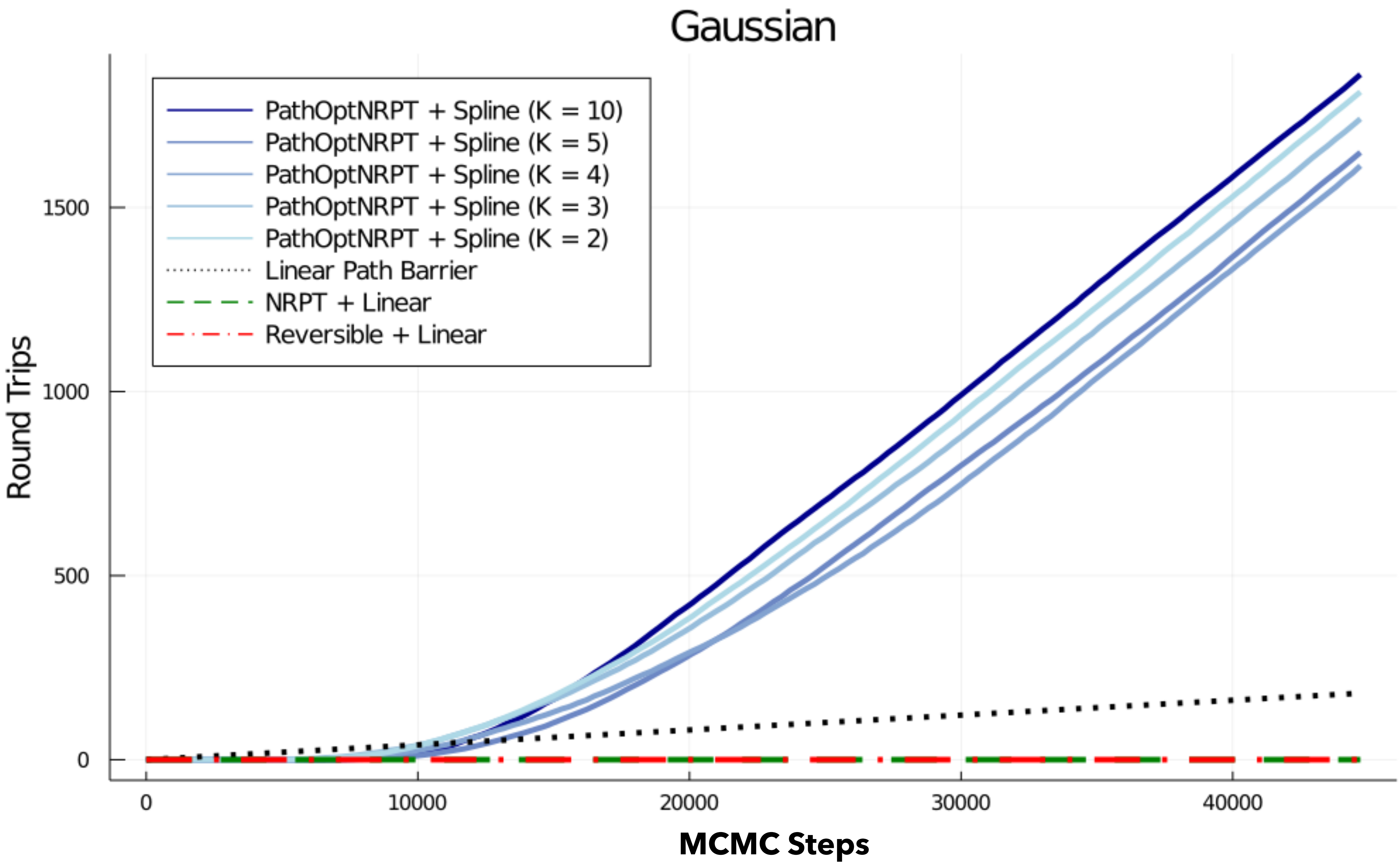
$$\eta(0) = (1,0)$$

- Suppose $\eta(\beta)$ linear spline with K knots



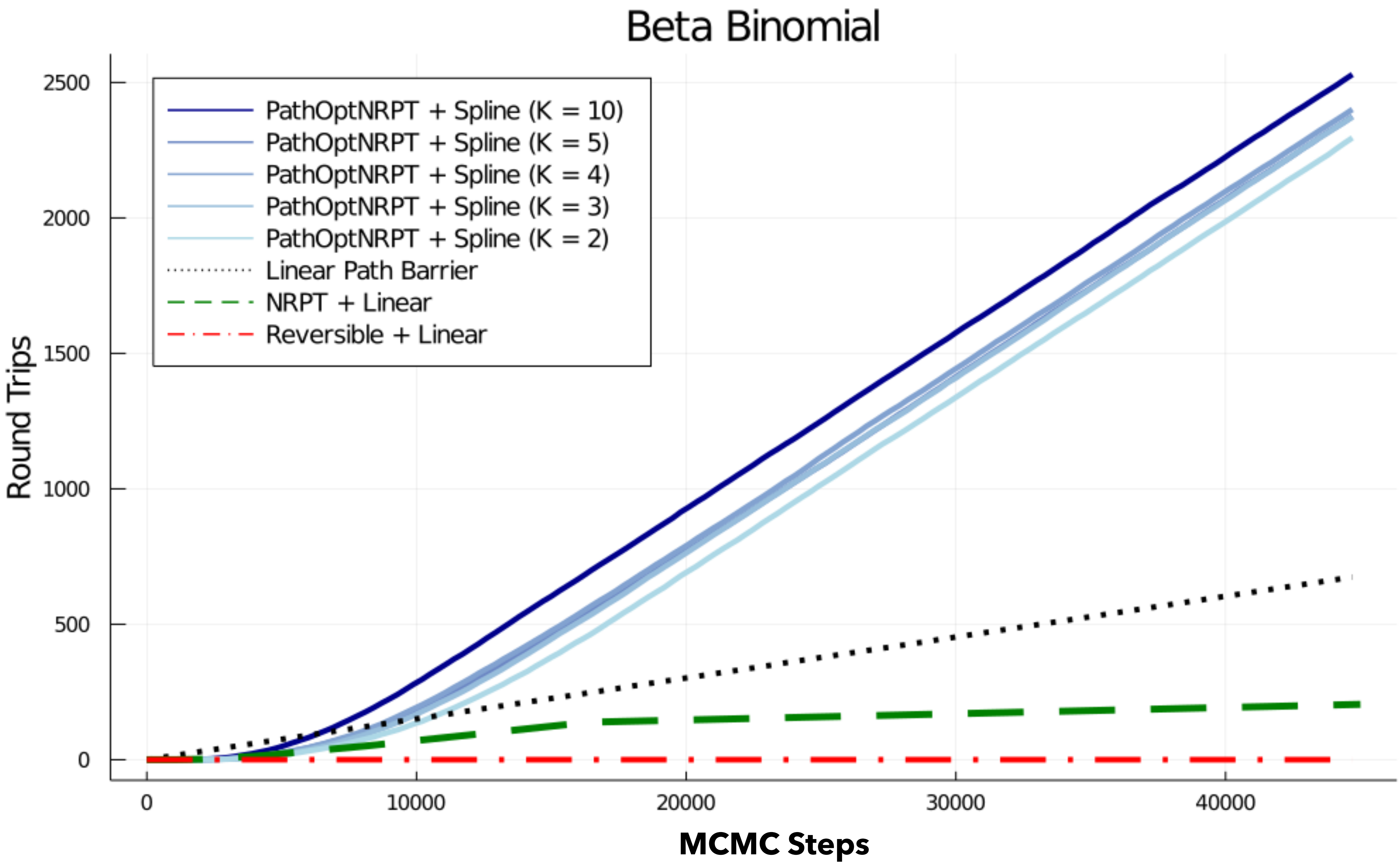


EXAMPLES

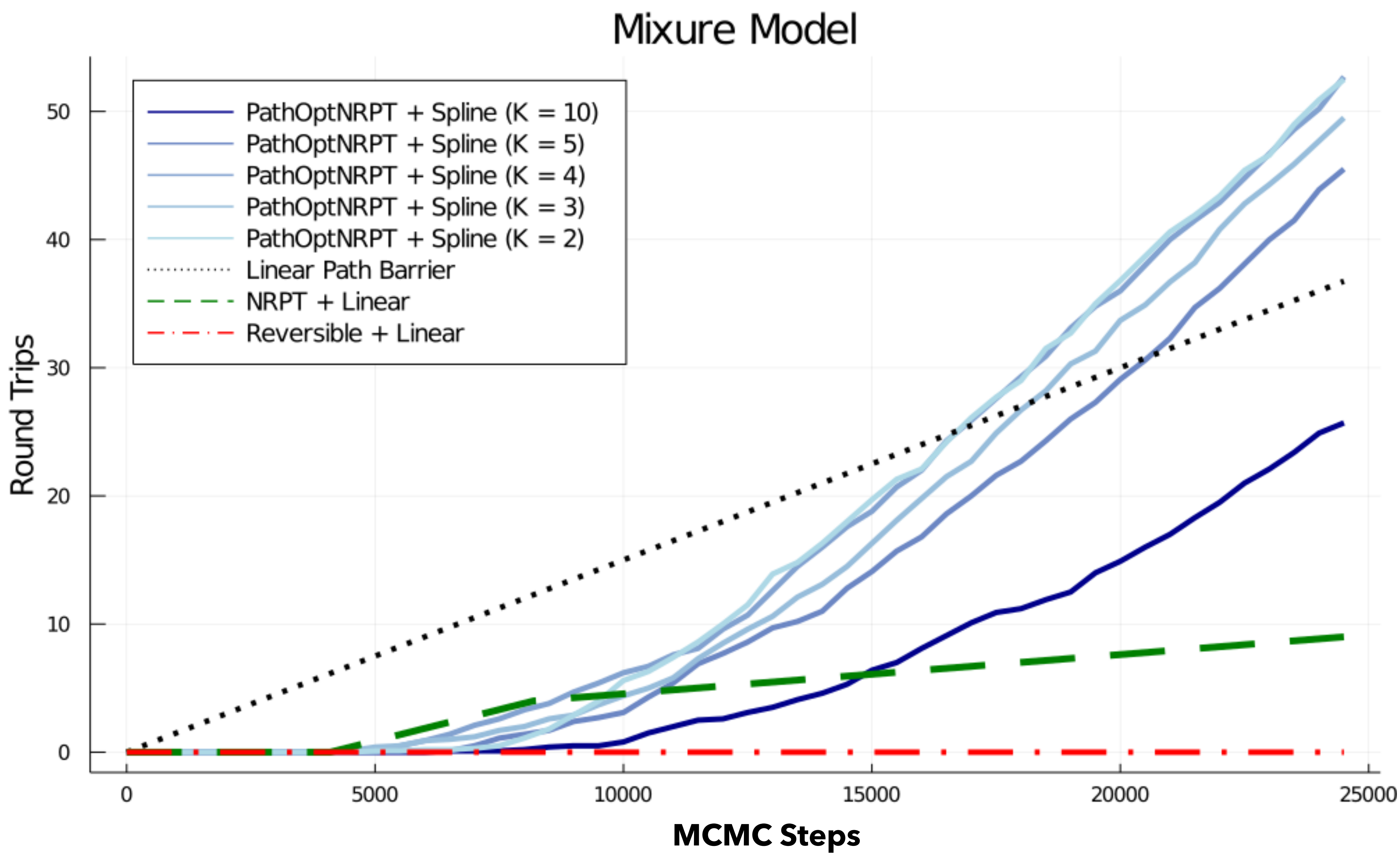


Gaussian
PathOptNRPT + Spline (K = 10)
PathOptNRPT + Spline (K = 5)
PathOptNRPT + Spline (K = 4)
PathOptNRPT + Spline (K = 3)
PathOptNRPT + Spline (K = 2)
Linear Path Barrier
NRPT + Linear
Reversible + Linear
Round Trips
1500
1000
500
0
0
10000
20000
30000
40000
MCMC Steps

EXAMPLES


Beta Binomial
PathOptNRPT + Spline (K = 10)
PathOptNRPT + Spline (K = 5)
PathOptNRPT + Spline (K = 4)
PathOptNRPT + Spline (K = 3)
PathOptNRPT + Spline (K = 2)
Linear Path Barrier
NRPT + Linear
Reversible + Linear
Round Trips
2500
2000
1500
1000
500
0
0
10000
20000
30000
40000
MCMC Steps

EXAMPLES



Mixure Model
PathOptNRPT + Spline (K = 10)
PathOptNRPT + Spline (K = 5)
PathOptNRPT + Spline (K = 4)
PathOptNRPT + Spline (K = 3)
PathOptNRPT + Spline (K = 2)
Linear Path Barrier
NRPT + Linear
Reversible + Linear
Round Trips
0
10
20
30
40
50
0
5000
10000
15000
20000
25000
MCMC Steps

SUMMARY

SUMMARY

- Non-reversible PT > reversible PT

SUMMARY

- Non-reversible PT > reversible PT
- Characterized the optimal schedule

SUMMARY

- Non-reversible PT > reversible PT
- Characterized the optimal schedule
- Can scale to GPUs

SUMMARY

- Non-reversible PT > reversible PT
- Characterized the optimal schedule
- Can scale to GPUs
- Developed natural efficiency metrics useful for practitioners

SUMMARY

- Non-reversible PT > reversible PT
- Characterized the optimal schedule
- Can scale to GPUs
- Developed natural efficiency metrics useful for practitioners
- No structural assumptions on model or state space

SUMMARY

- Non-reversible PT > reversible PT
- Characterized the optimal schedule
- Can scale to GPUs
- Developed natural efficiency metrics useful for practitioners
- No structural assumptions on model or state space
- Efficient, black box algorithm for both schedule and path tuning

SUMMARY

- Non-reversible PT > reversible PT
- Characterized the optimal schedule
- Can scale to GPUs
- Developed natural efficiency metrics useful for practitioners
- No structural assumptions on model or state space
- Efficient, black box algorithm for both schedule and path tuning

THANK YOU!

MODEL ASSUMPTIONS

MODEL ASSUMPTIONS

- **(A1) Stationarity:**
  - $\mathbf{X}_0 \sim \boldsymbol{\pi}$

MODEL ASSUMPTIONS

- **(A1) Stationarity:**
  - $\mathbf{X}_0 \sim \boldsymbol{\pi}$
- **(A2) Efficient Local Exploration (ELE):**
  - If $X \sim \pi_\beta$, $X' \sim K_\beta(X, \mathrm{d}x')$ then $V(X)$ and $V(X')$ are independent

$$V(x) = \log \frac{\pi_1(x)}{\pi_0(x)}$$

  - Weaker than assuming X and X' are independent.
  - Tool to isolate the communication from the (problem specific) local exploration
  - Methodology robust to severe violations in practice!!

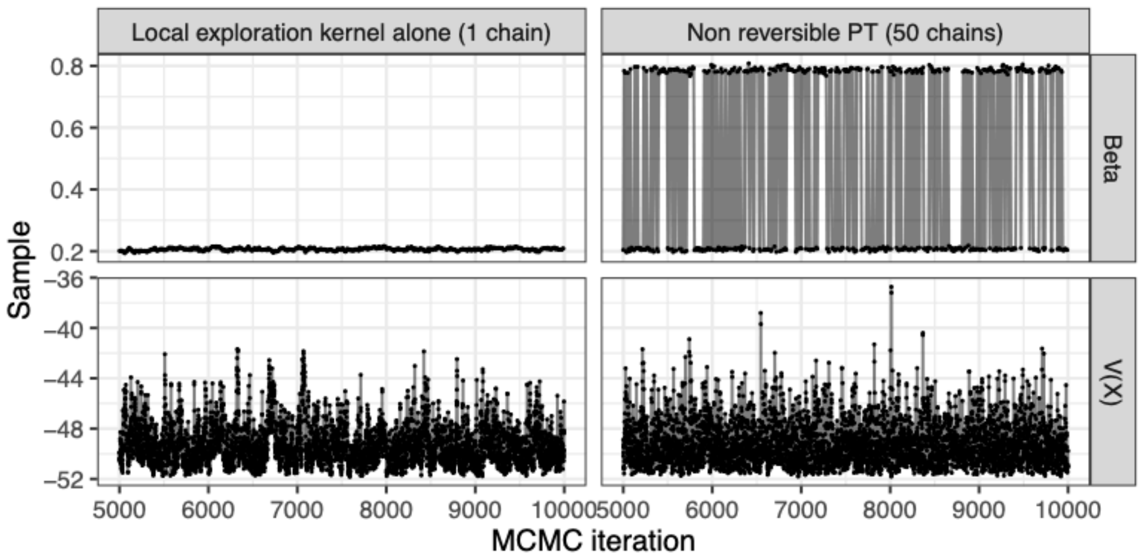
MODEL ASSUMPTIONS

- **(A1) Stationarity:**
  - $\mathbf{X}_0 \sim \boldsymbol{\pi}$
- **(A2) Efficient Local Exploration (ELE):**
  - If $X \sim \pi_\beta$, $X' \sim K_\beta(X, \mathrm{d}x')$ then $V(X)$ and $V(X')$ are independent
$$V(x) = \log \frac{\pi_1(x)}{\pi_0(x)}$$
  - Weaker than assuming X and X' are independent.
  - Tool to isolate the communication from the (problem specific) local exploration
  - Methodology robust to severe violations in practice!!
- **(A3) Integrability**
  - $\pi_0(|V|^3), \pi_1(|V|^3) < \infty$

VIOLATION OF ELE

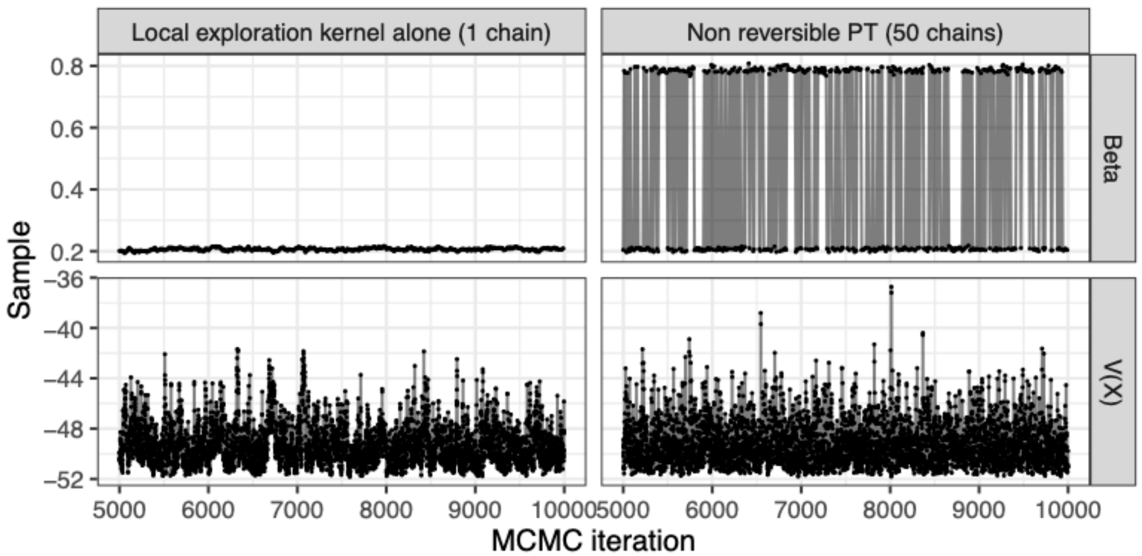
VIOLATION OF ELE

ODE parameter estimation

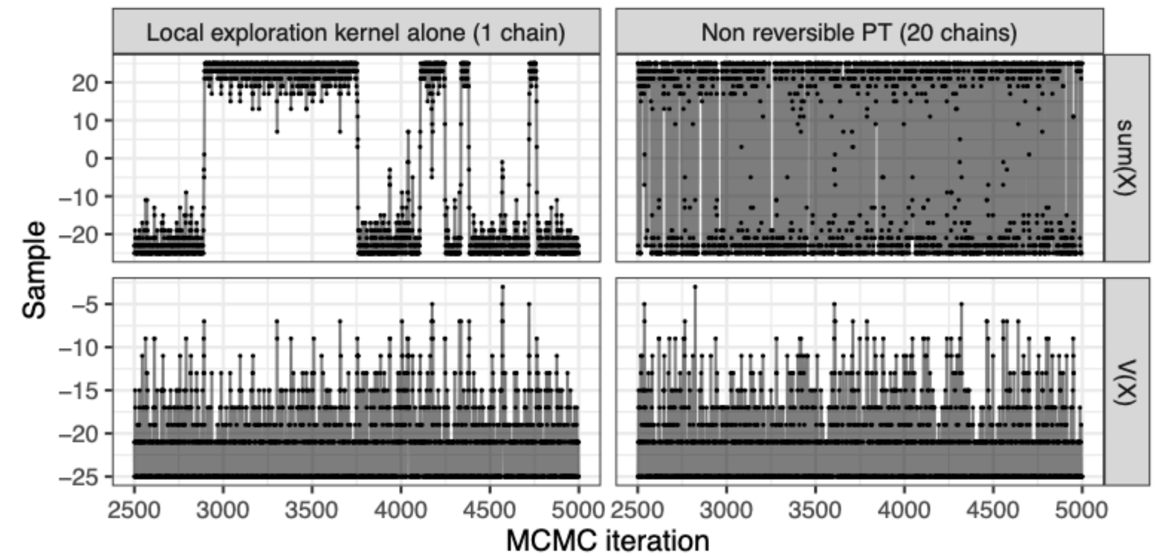


VIOLATION OF ELE

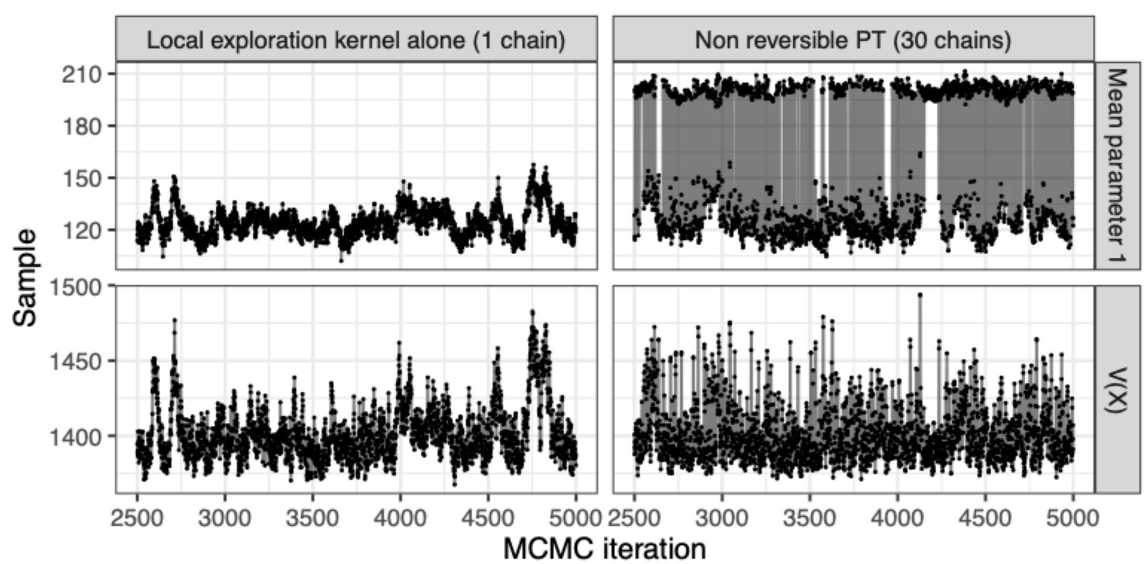
ODE parameter estimation



Ising model



Bayesian Mixture Model



Copy-number inference