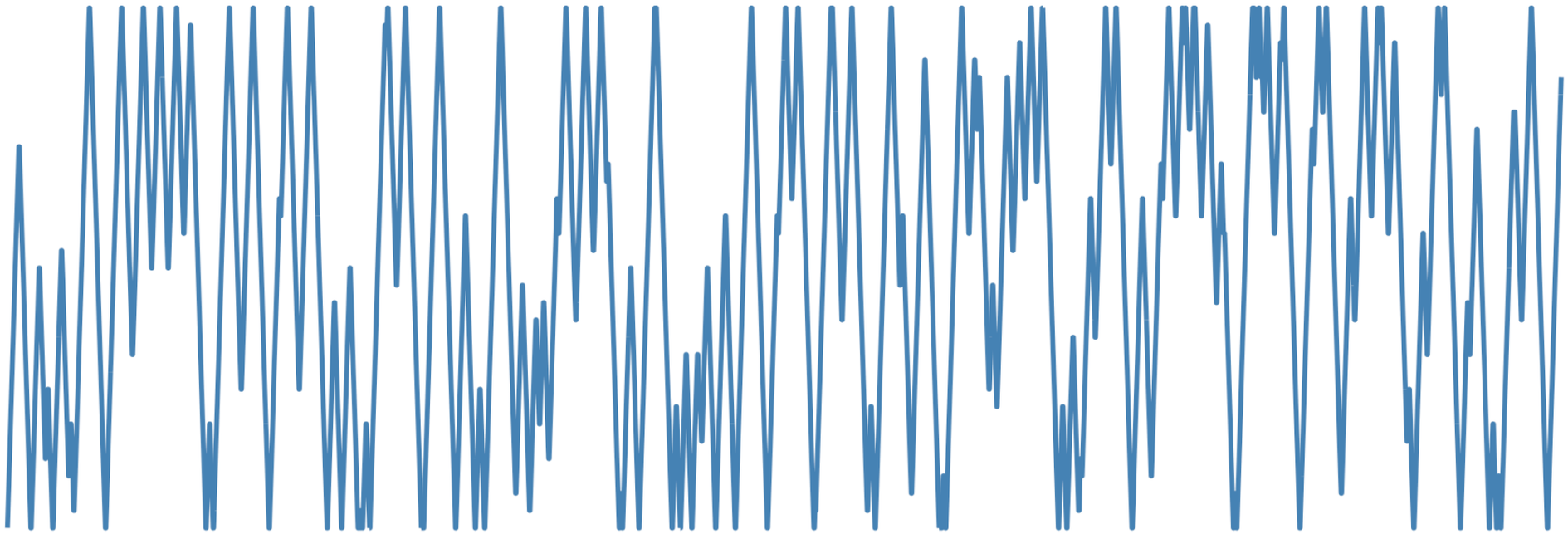
NON-REVERSIBLE PARALLEL TEMPERING



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

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

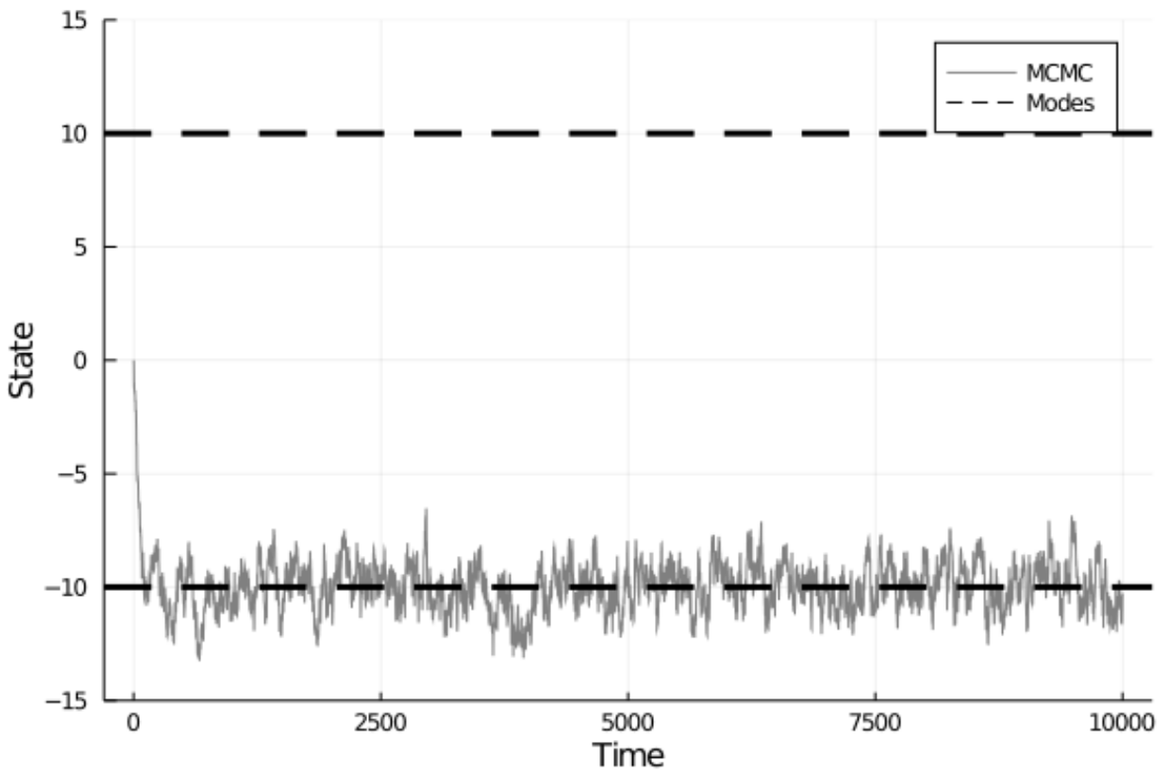
- π_1 is a general probability distribution
- $\mathcal{X}$ is a general state space
- $f(x)$ is a general function

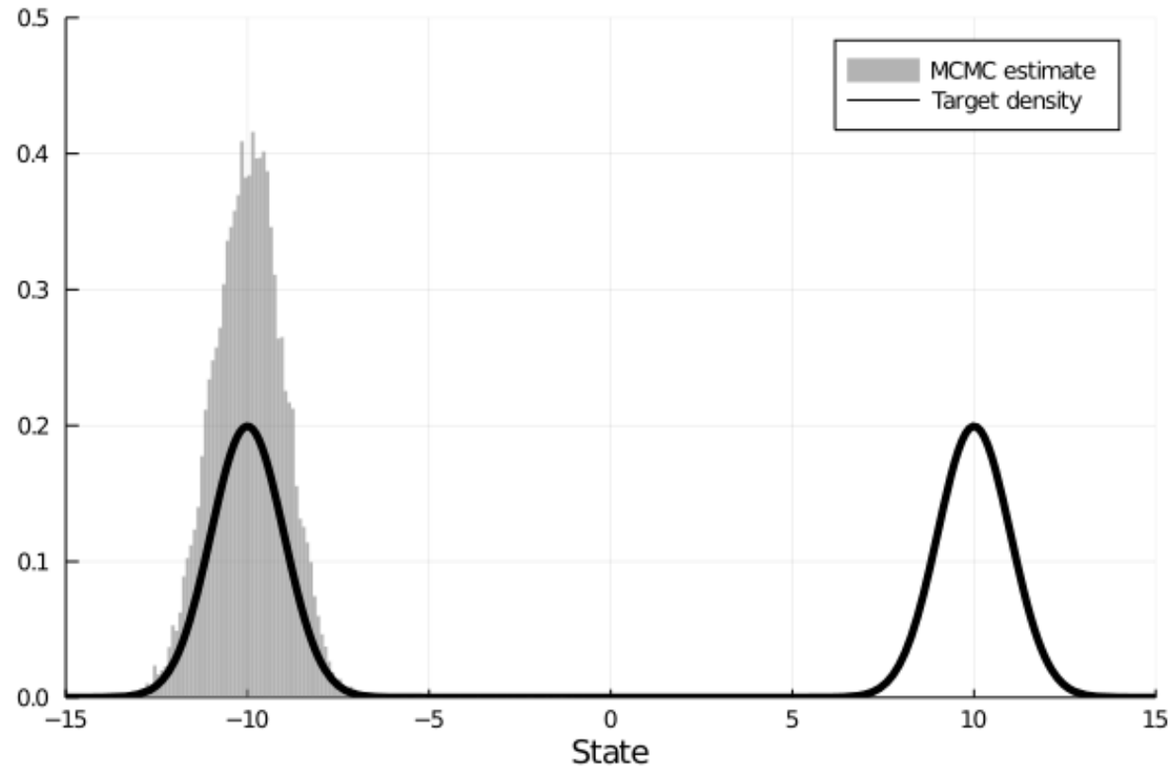
- **Solution:** Numerically approximate expectation using MCMC
- Run Markov chain stationary with respect to target and use ergodic theorem.

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

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





PARALLEL COMPUTING

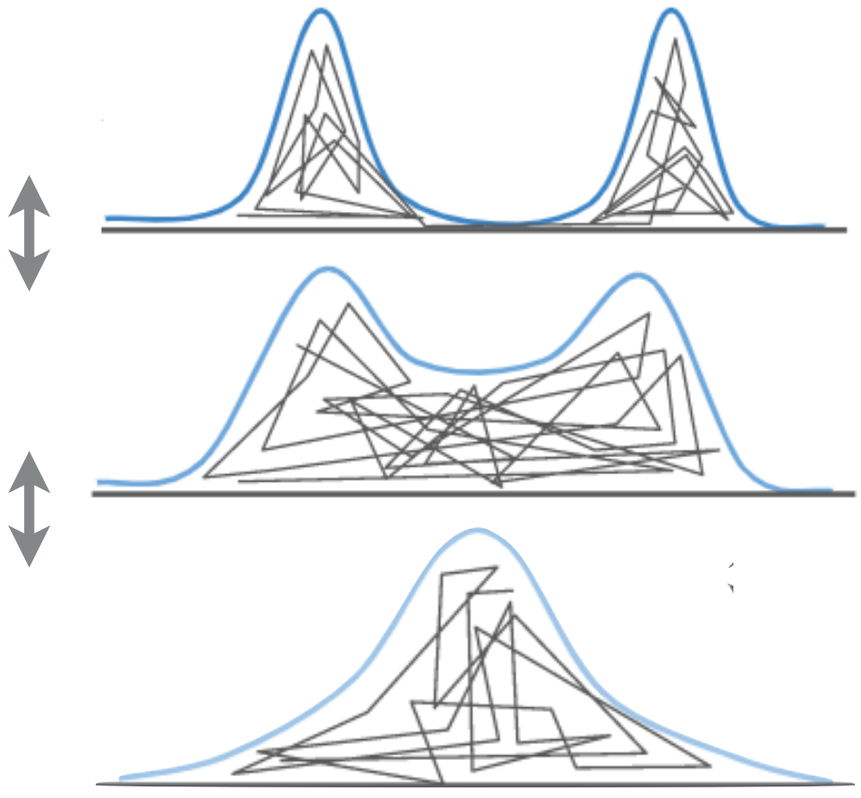
PARALLEL COMPUTING

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

PARALLEL COMPUTING

- **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

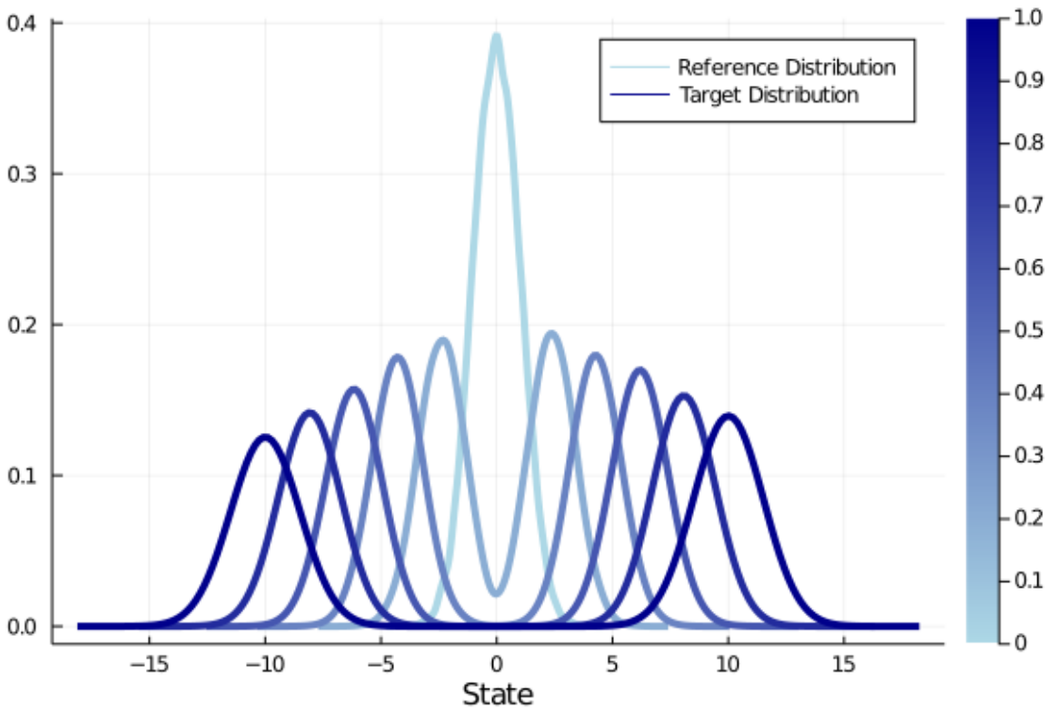
$$\pi_n = \pi_{\beta_n}$$

ANNEALING

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

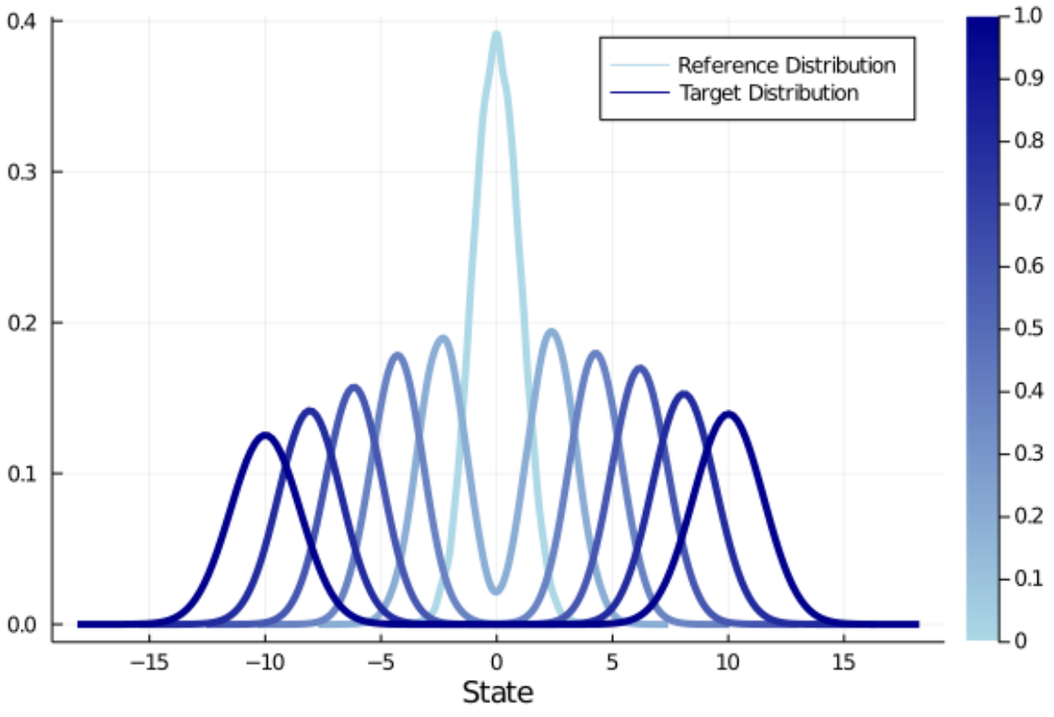
ANNEALING

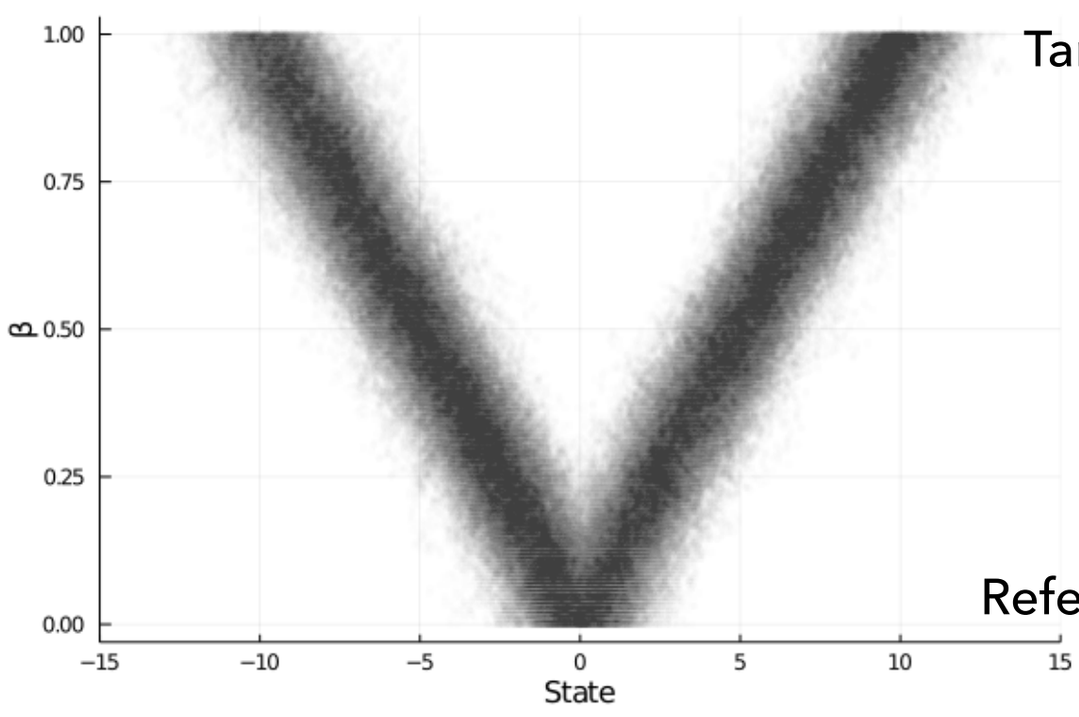
$$\pi_\beta(x) \propto \pi_0^{1-\beta}(x)\pi_1(x)^\beta$$



ANNEALING

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



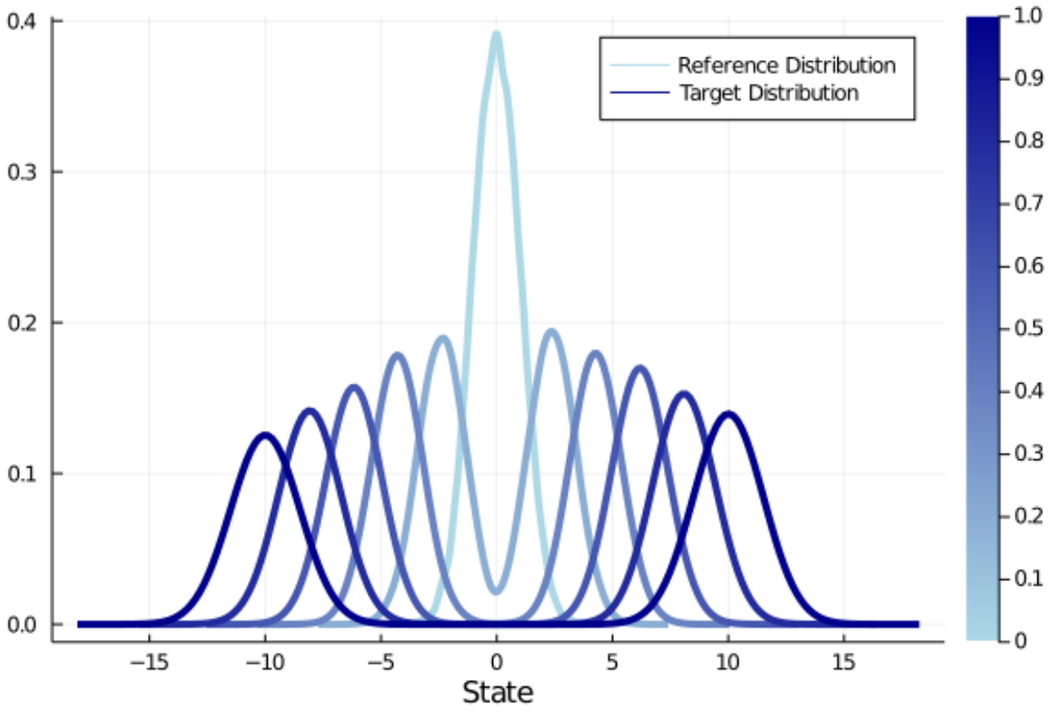


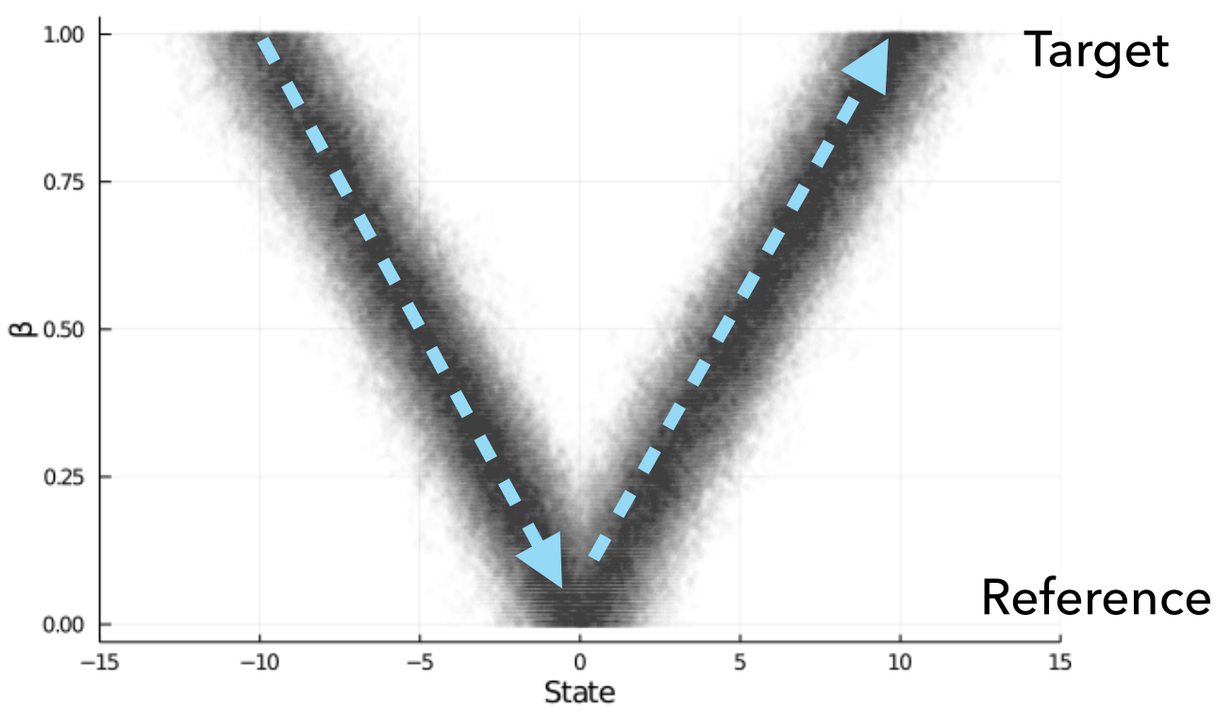
Target

Reference

ANNEALING

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





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 $\boldsymbol{\pi}$:

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

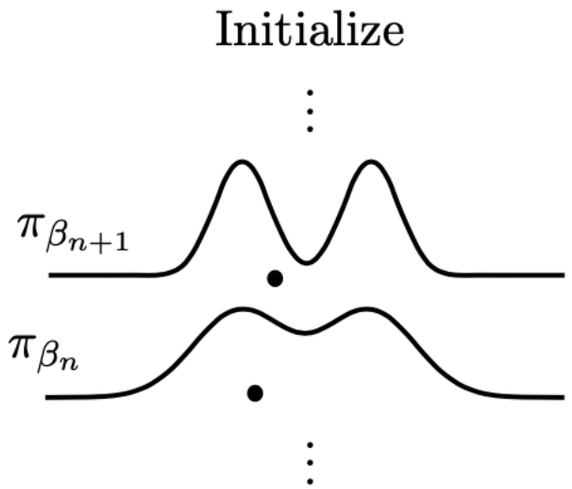
$$\mathbb{E}[f] = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} f(X_t^N)$$

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
   - Propose swap components n and $n+1$ for $n \in S_t \subset \{0, \ldots, N-1\}$
   - Accept with probability:

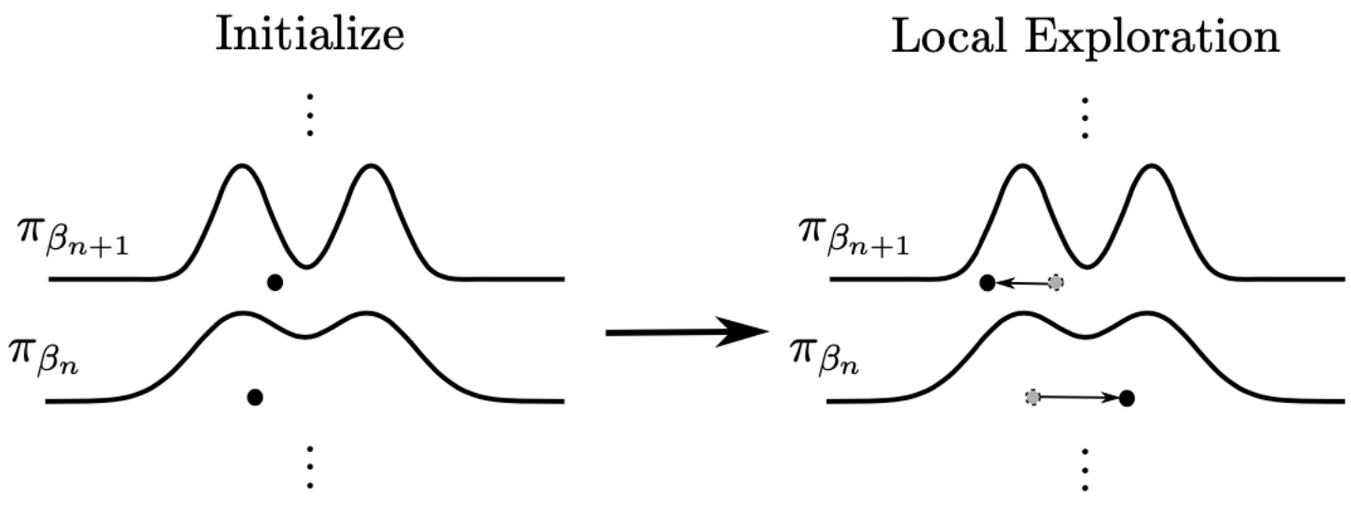
$$\alpha_{n,n+1} = 1 \wedge \frac{\pi_n(x^{n+1})\pi_{n+1}(x^n)}{\pi_n(x^n)\pi_{n+1}(x^{n+1})}$$

PARALLEL TEMPERING

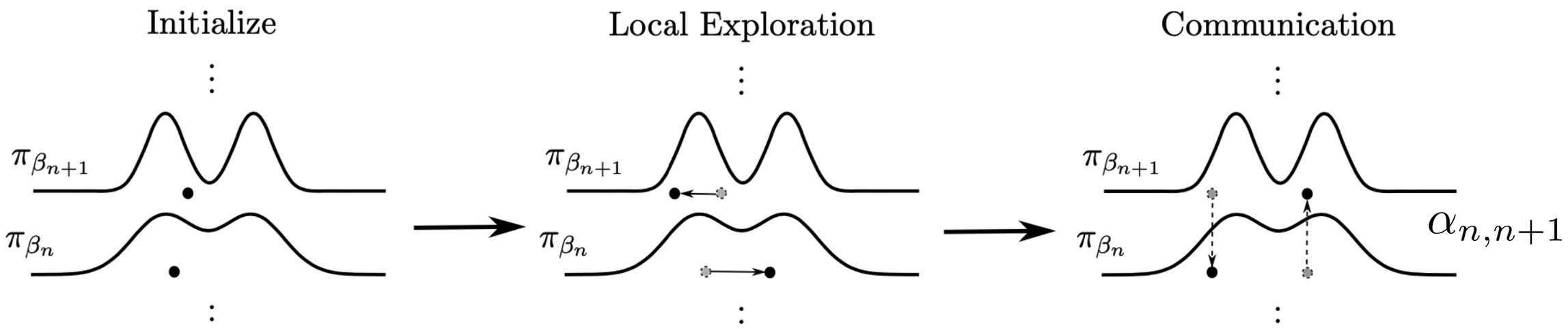
PARALLEL TEMPERING



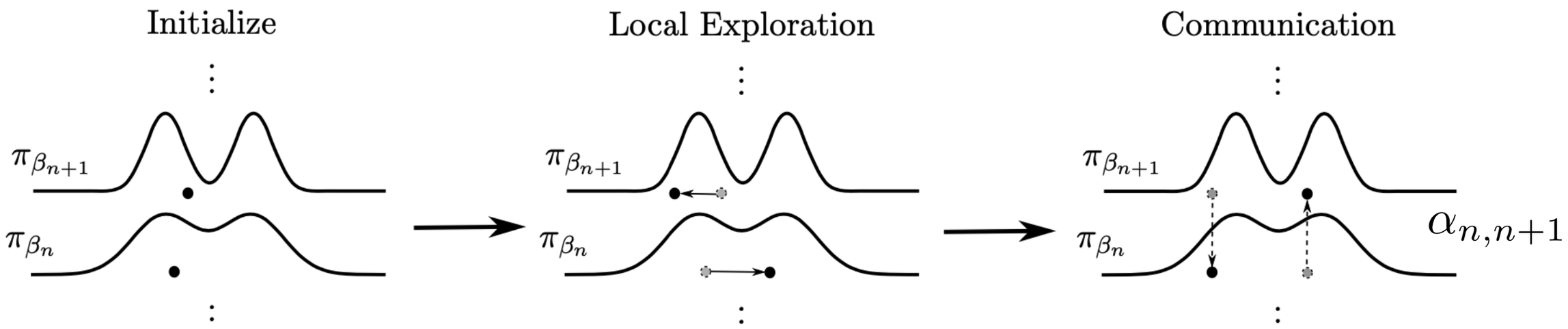
PARALLEL TEMPERING



PARALLEL TEMPERING



PARALLEL TEMPERING



- Acceptance probability:

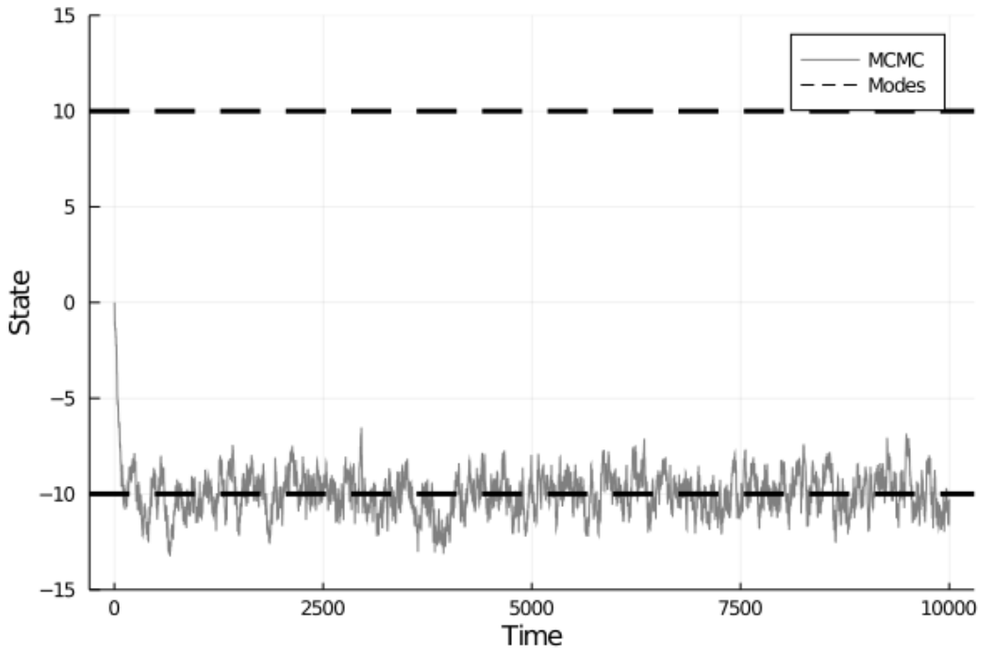
$$\alpha_{n,n+1} = 1 \wedge \frac{\pi_n(x^{n+1})\pi_{n+1}(x^n)}{\pi_n(x^n)\pi_{n+1}(x^{n+1})}$$

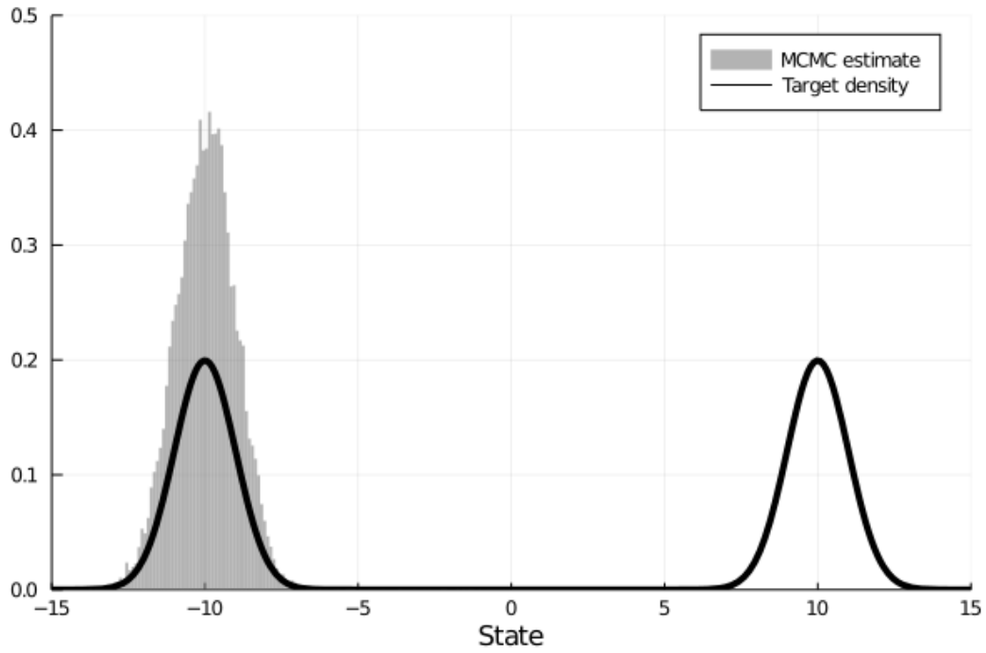
  - Notice swap probability only depends on likelihood ratio of states
  - Agnostic to location of modes

MCMC VS PARALLEL TEMPERING

MCMC VS PARALLEL TEMPERING

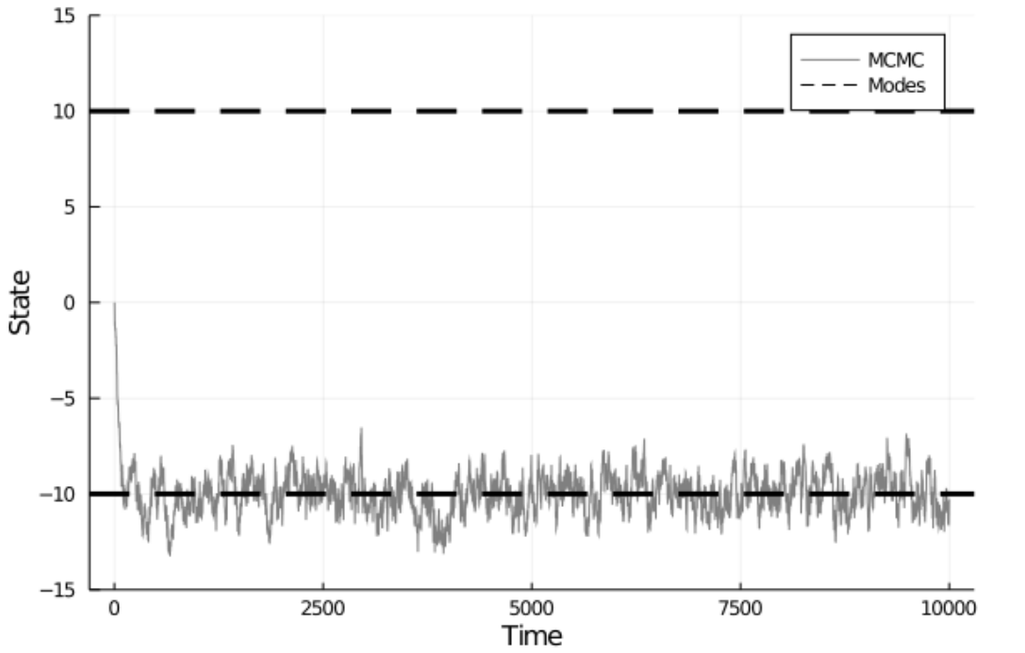
Single chain MCMC

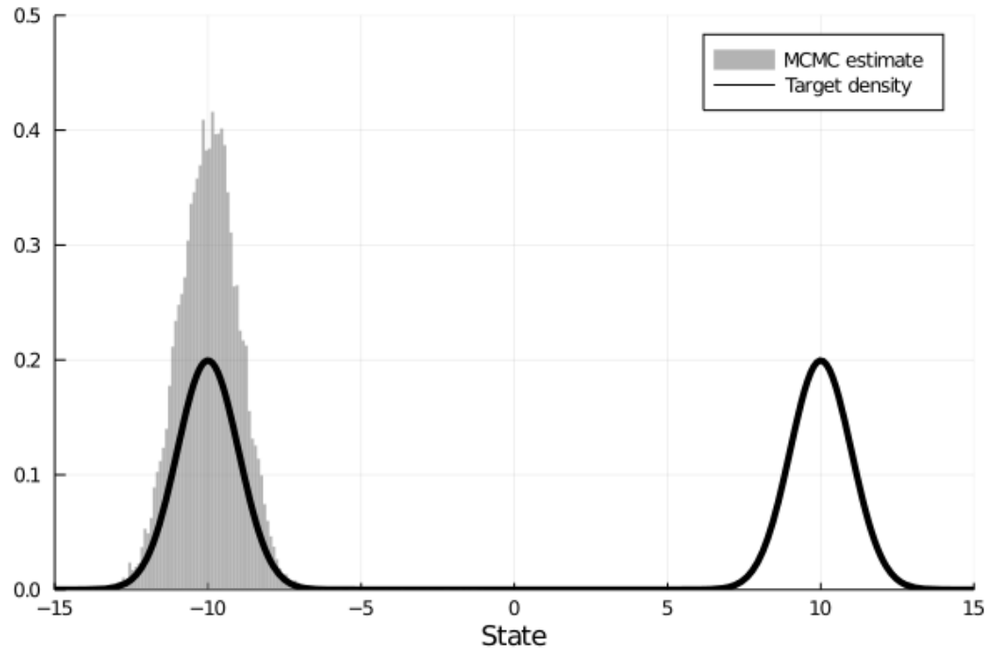




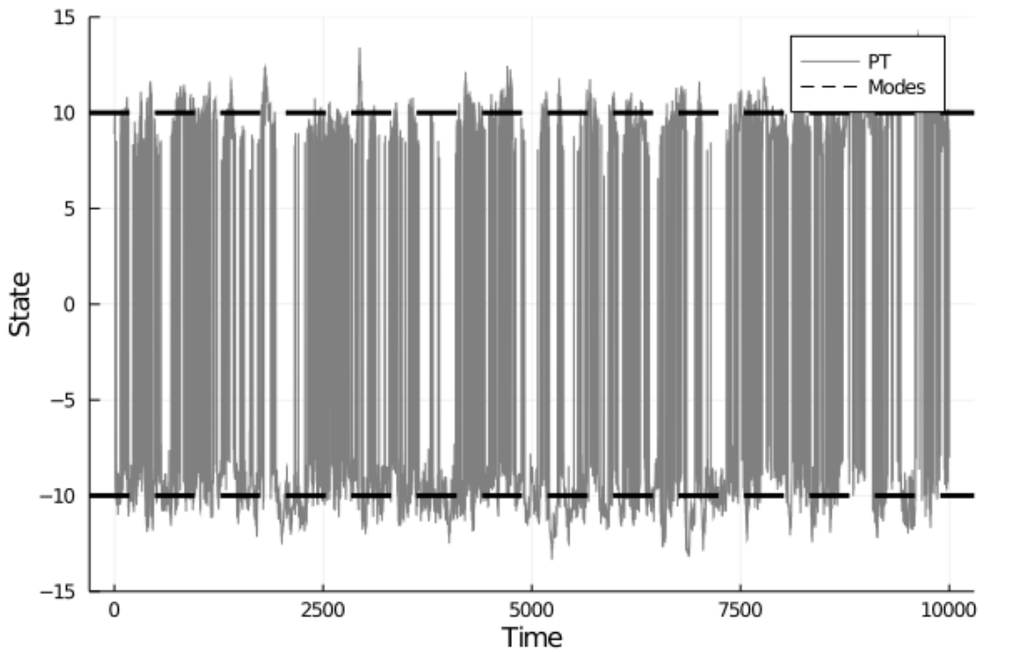
MCMC VS PARALLEL TEMPERING

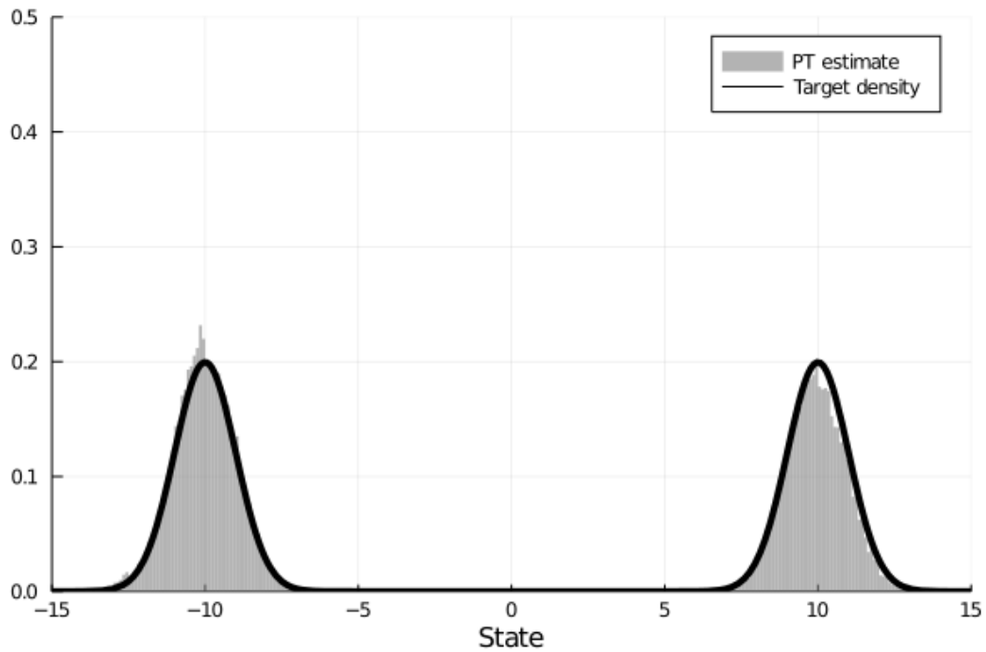
Single chain MCMC





Parallel Tempering (N = 10)





TUNING PT

TUNING PT

- How to propose swaps?
- How to pick annealing 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)
- How to pick annealing 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)
- How to pick annealing schedule? — Reject 77% of swaps
- How to pick number of chains? — $N \approx 10$ and adaptively reduce
- How to pick path?
- Hard problem or bad implementation?

TUNING PT

- How to propose swaps? — At random (reversible)
- How to pick annealing schedule? — Reject 77% of swaps
- How to pick number of chains? — $N \approx 10$ and adaptively reduce
- How to pick path? — Always 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)
- How to pick annealing schedule? — Reject 77% of swaps
- How to pick number of chains? — $N \approx 10$ and adaptively reduce
- How to pick path? — Always linear path : $\pi_\beta \propto \pi_0^{1-\beta}\pi_1^{\beta}$
- Hard problem or bad implementation?

OUTLINE

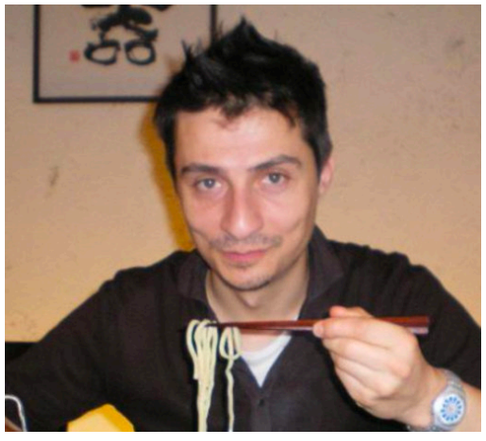
- **Non-reversible parallel tempering (JRSS-B, 2021)**
  - Distributed PT and the index process
  - Non-Reversible vs Reversible PT
  - Scaling behavior
  - Tuning non-reversible PT
  - Experiments

OUTLINE

- **Non-reversible parallel tempering (JRSS-B, 2021)**
  - Distributed PT and the index process
  - Non-Reversible vs Reversible PT
  - Scaling behavior
  - Tuning non-reversible PT
  - Experiments

Alexandre Bouchard-Côté

George Deligiannidis



Arnaud Doucet

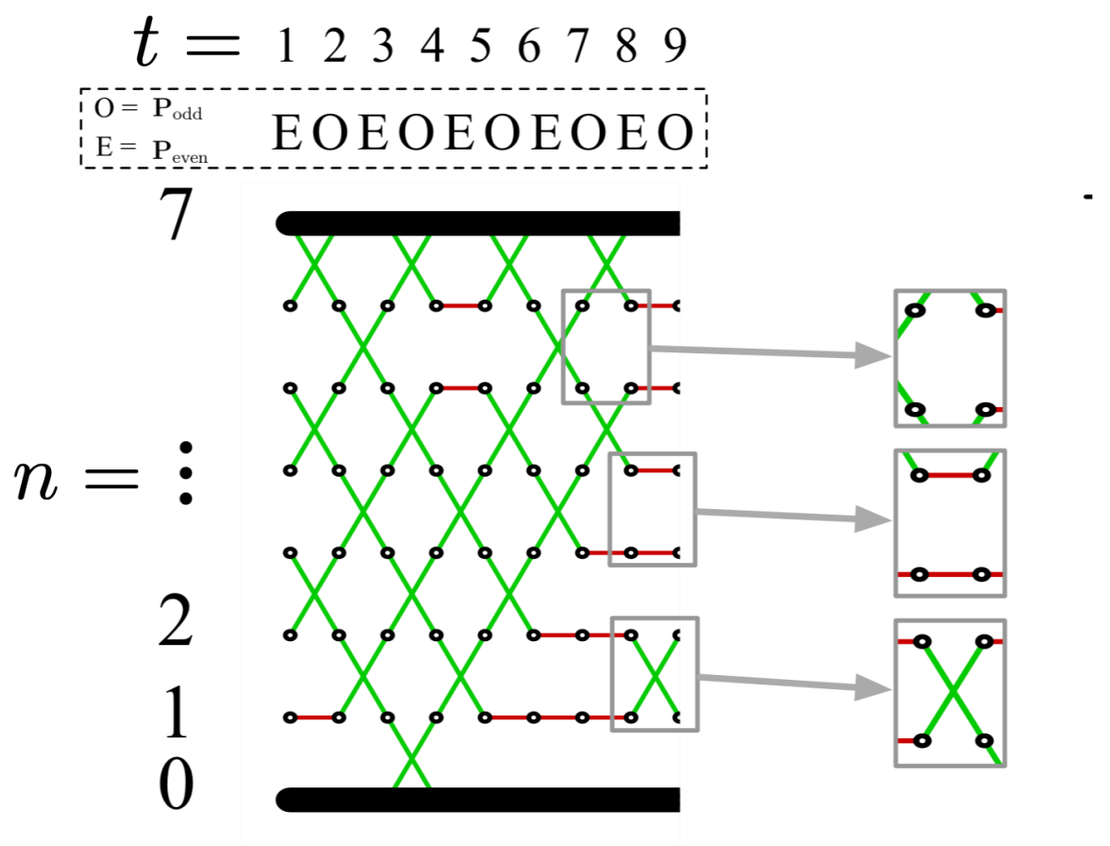
DISTRIBUTED PT

DISTRIBUTED PT

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

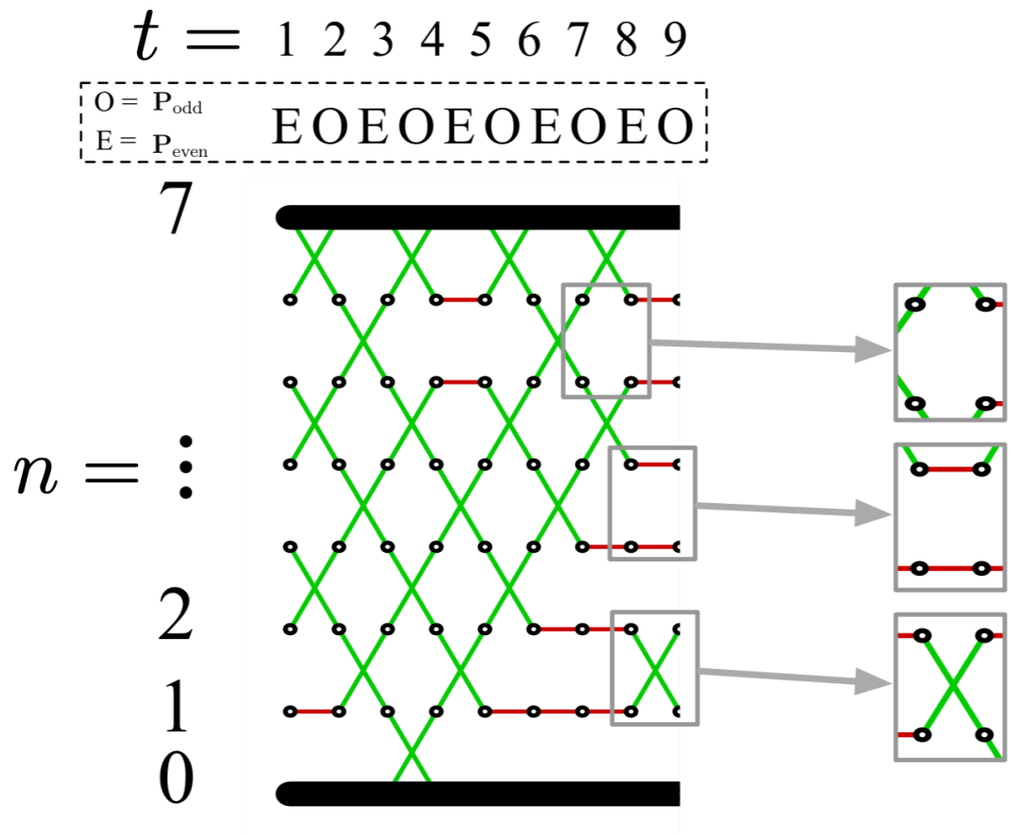
DISTRIBUTED PT

- **Dual perspective on PT:** Swap annealing parameters not states!
  - We can study communication through the dynamics of the permuted annealing parameters



DISTRIBUTED PT

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



DISTRIBUTED PT

DISTRIBUTED PT

- Machine m stores:
  - State: $Y_t^m \in \mathcal{X}$
  - Annealing parameter: $\beta_{I_t^m} \in \mathcal{B}_N$
  - Proposal direction: $\varepsilon_t^m \in \{-1, 1\}$

DISTRIBUTED PT

- Machine m stores:
  - State: $Y_t^m \in \mathcal{X}$
  - Annealing parameter: $\beta_{I_t^m} \in \mathcal{B}_N$
  - Proposal direction: $\varepsilon_t^m \in \{-1, 1\}$

- Distributed PT state:

$$\mathbf{Y}_t = (Y_t^0, \ldots, Y_t^N)$$

$$\mathbf{I}_t = (I_t^0, ..., I_t^N)$$

$$\boldsymbol{\varepsilon}_t = (\varepsilon_t^0, \ldots, \varepsilon_t^N)$$

DISTRIBUTED PT

- Machine m stores:
  - State: $Y_t^m \in \mathcal{X}$
  - Annealing parameter: $\beta_{I_t^m} \in \mathcal{B}_N$
  - Proposal direction: $\varepsilon_t^m \in \{-1, 1\}$

- Distributed PT state:

$$\mathbf{Y}_t = (Y_t^0, \ldots, Y_t^N)$$

$$\mathbf{I}_t = (I_t^0, ..., I_t^N)$$

$$\boldsymbol{\varepsilon}_t = (\varepsilon_t^0, \ldots, \varepsilon_t^N)$$

- We note that $\mathbf{Y}_t$ is $\mathbf{X}_t$ shuffled according to permutation $\mathbf{I}_t$

DISTRIBUTED PT

- Machine m stores:
  - State: $Y_t^m \in \mathcal{X}$
  - Annealing parameter: $\beta_{I_t^m} \in \mathcal{B}_N$
  - Proposal direction: $\varepsilon_t^m \in \{-1, 1\}$
- Distributed PT state:

$$\mathbf{Y}_t = (Y_t^0, \ldots, Y_t^N)$$

$$\mathbf{I}_t = (I_t^0, ..., I_t^N)$$

$$\boldsymbol{\varepsilon}_t = (\varepsilon_t^0, \ldots, \varepsilon_t^N)$$

- We note that $\mathbf{Y}_t$ is $\mathbf{X}_t$ shuffled according to permutation $\mathbf{I}_t$

- **Local exploration:**
  - For each m update Y_t^m with MCMC move targeting $\pi_{I_t^m}$

DISTRIBUTED PT

- Machine m stores:
  - State: $Y_t^m \in \mathcal{X}$
  - Annealing parameter: $\beta_{I_t^m} \in \mathcal{B}_N$
  - Proposal direction: $\varepsilon_t^m \in \{-1, 1\}$

- Distributed PT state:

$$\mathbf{Y}_t = (Y_t^0, \ldots, Y_t^N)$$

$$\mathbf{I}_t = (I_t^0, ..., I_t^N)$$

$$\boldsymbol{\varepsilon}_t = (\varepsilon_t^0, \ldots, \varepsilon_t^N)$$

- We note that $\mathbf{Y}_t$ is $\mathbf{X}_t$ shuffled according to permutation $\mathbf{I}_t$

- **Local exploration:**
  - For each m update Y_t^m with MCMC move targeting $\pi_{I_t^m}$

- **Communication:**
  - For each m propose swap of annealing parameter across machines storing $\beta_{I_t^m}$ and $\beta_{I_t^m + \varepsilon_t^m}$

OBJECTIVE OF PT

OBJECTIVE OF PT

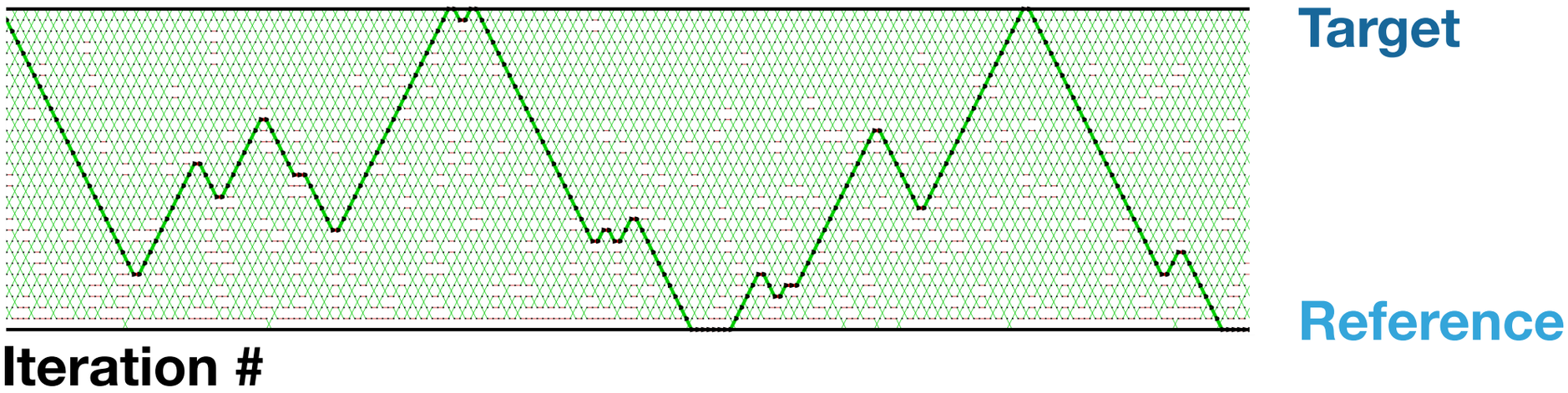
- We can study the communication through index processes I_t^m

OBJECTIVE OF PT

- We can study the communication through index processes I_t^m
- When I_t^m travels from 0 to N, reference and target have communicated on machine m

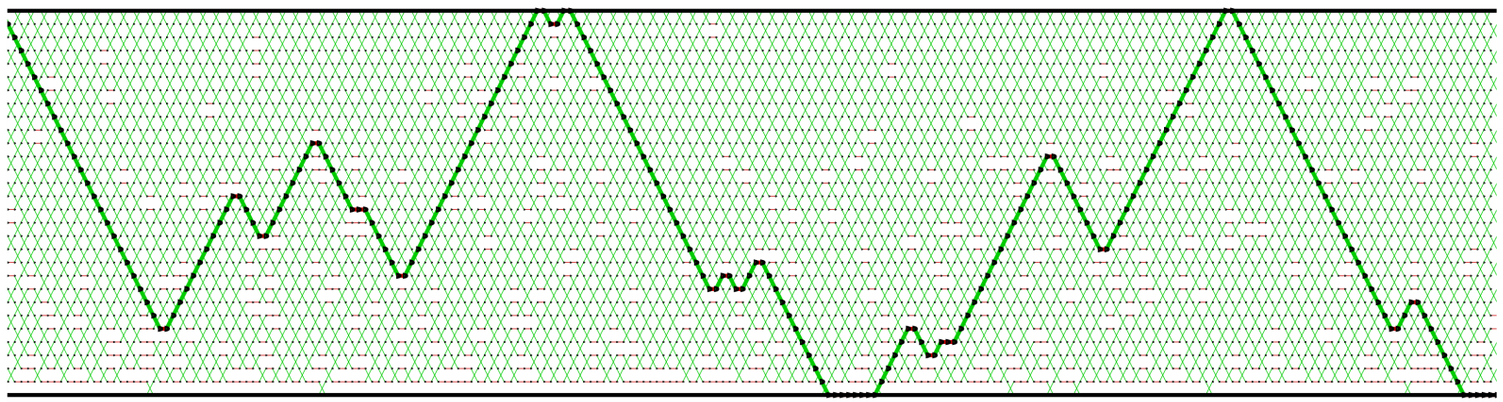
OBJECTIVE OF PT

- We can study the communication through index processes I_t^m
- When I_t^m travels from 0 to N, reference and target have communicated on machine m
- **Round Trip:** when a state travels from **reference** to **target** and back



OBJECTIVE OF PT

- We can study the communication through index processes I_t^m
- When I_t^m travels from 0 to N, reference and target have communicated on machine m
- **Round Trip:** when a state travels from **reference** to **target** and back

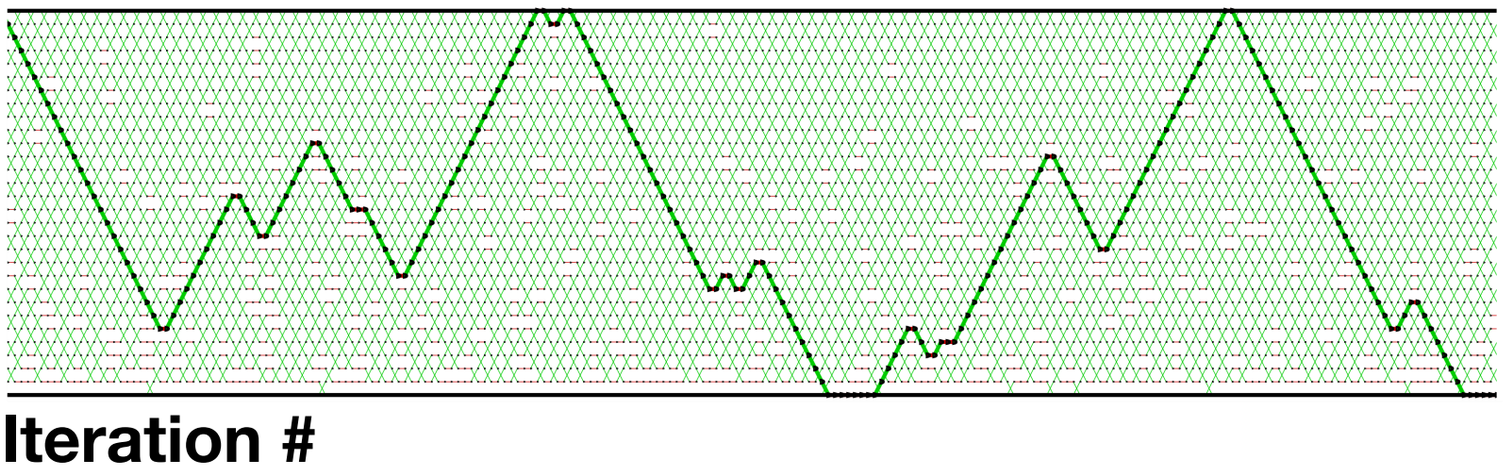


- **Round trip rate** % of iterations with round trip

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

OBJECTIVE OF PT

- We can study the communication through index processes I_t^m
- When I_t^m travels from 0 to N, reference and target have communicated on machine m
- **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

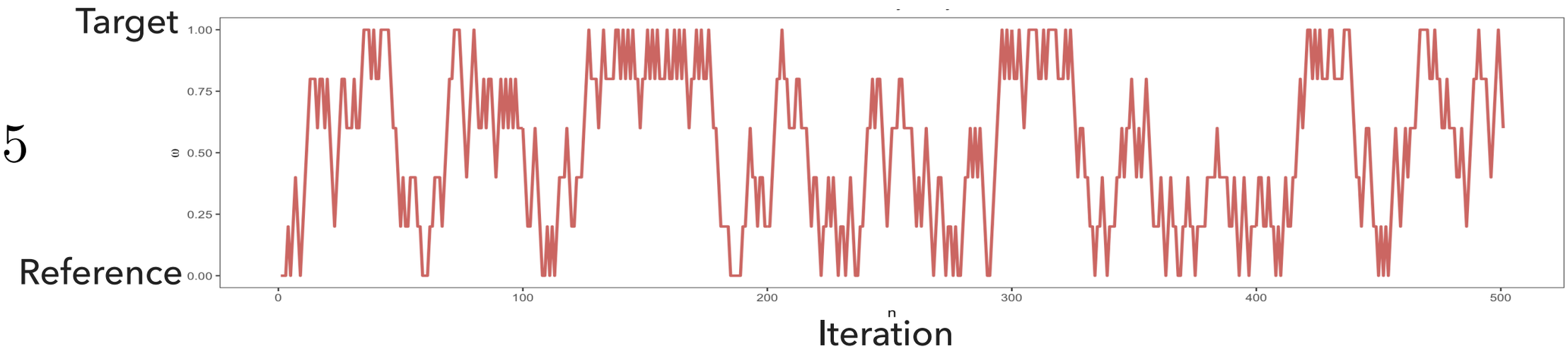
REVERSIBLE PT

- Propose swaps at random in communication phase

REVERSIBLE PT

- Propose swaps at random in communication phase

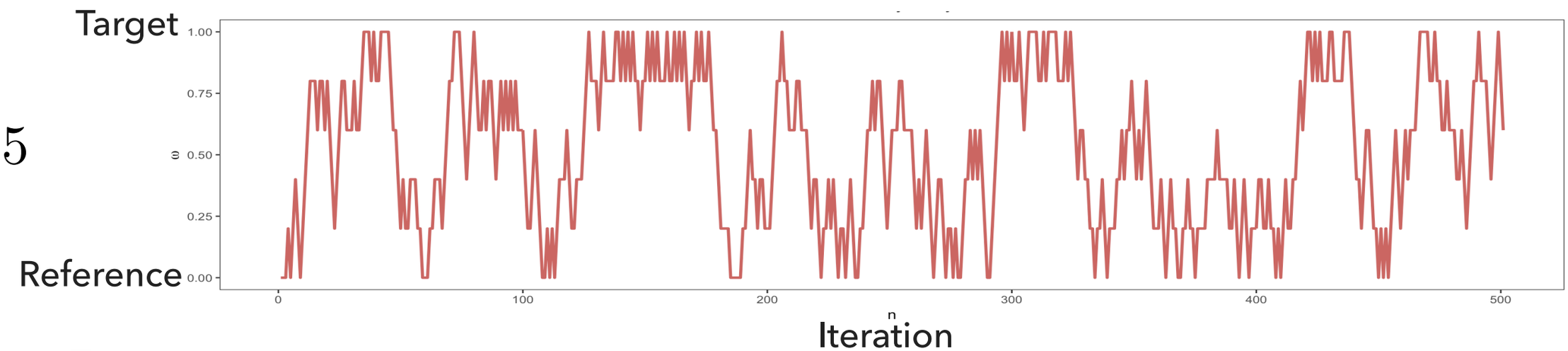
$N = 5$



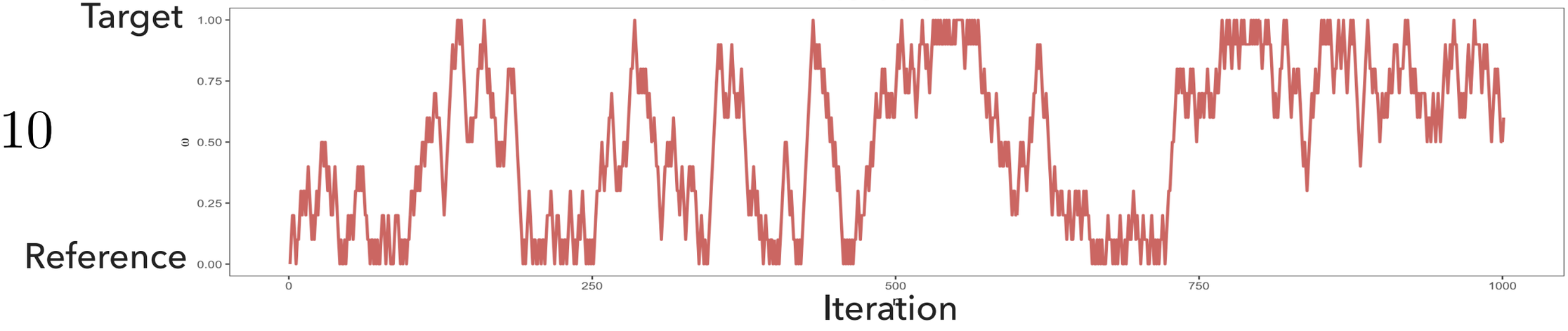
REVERSIBLE PT

- Propose swaps at random in communication phase

$N = 5$



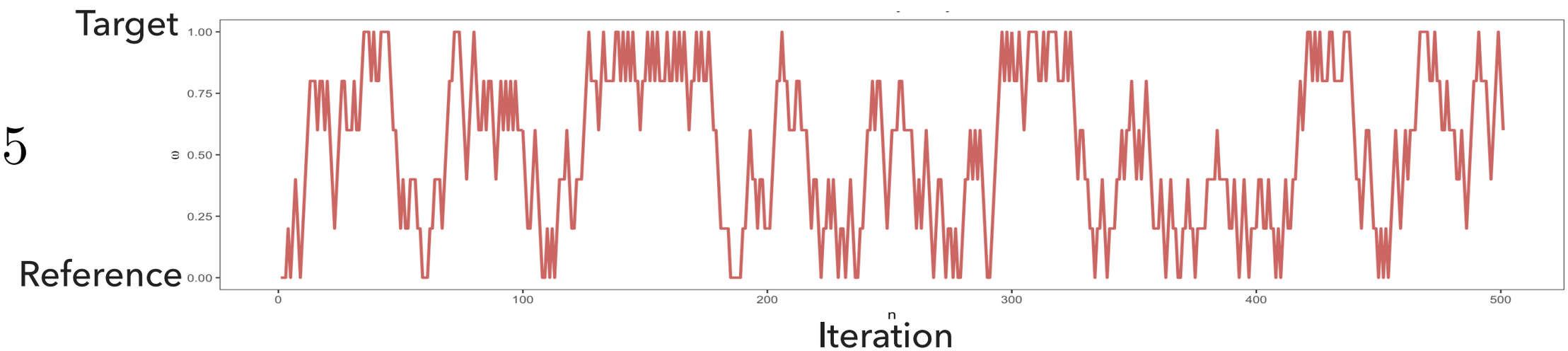
$N = 10$



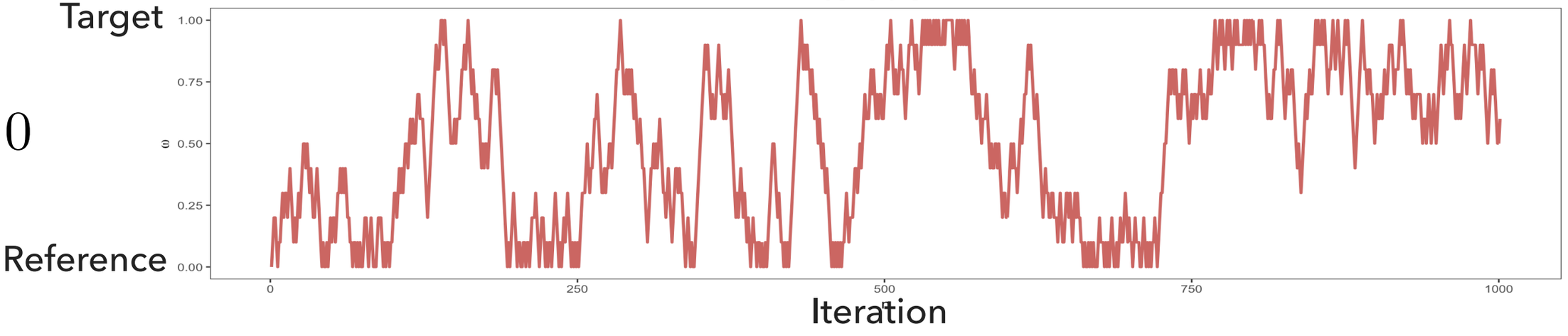
REVERSIBLE PT

- Propose swaps at random in communication phase

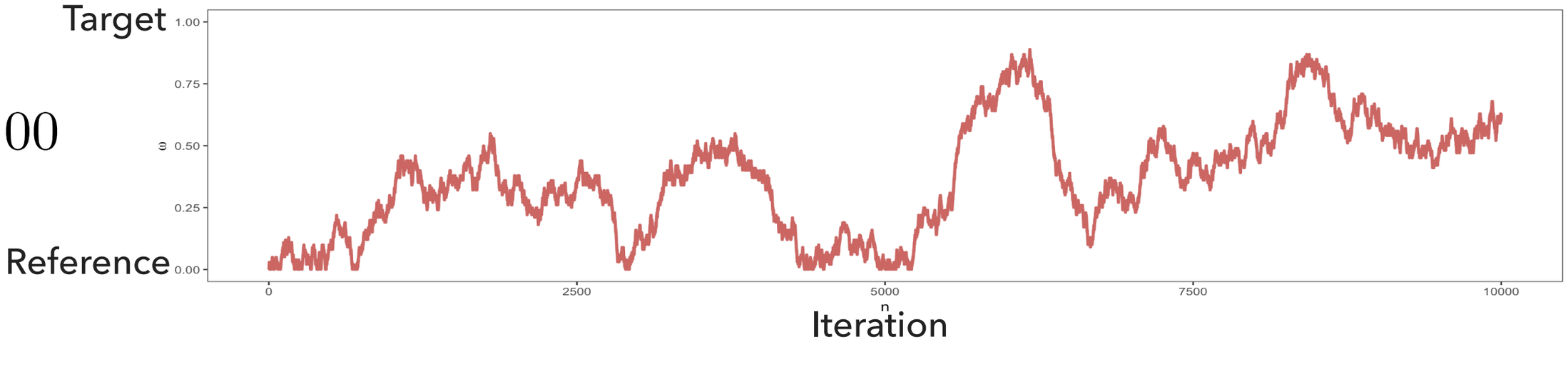
$N = 5$



$N = 10$

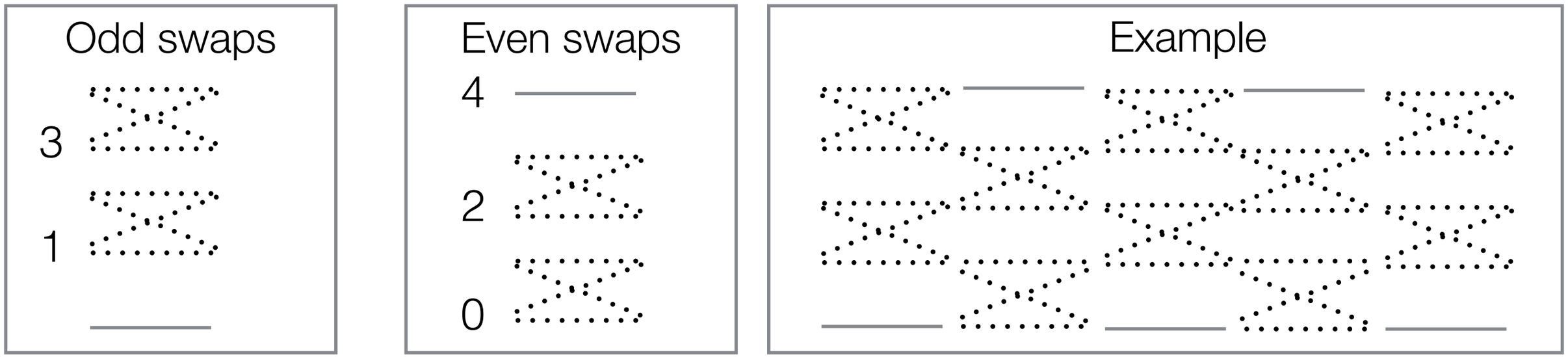


$N = 100$



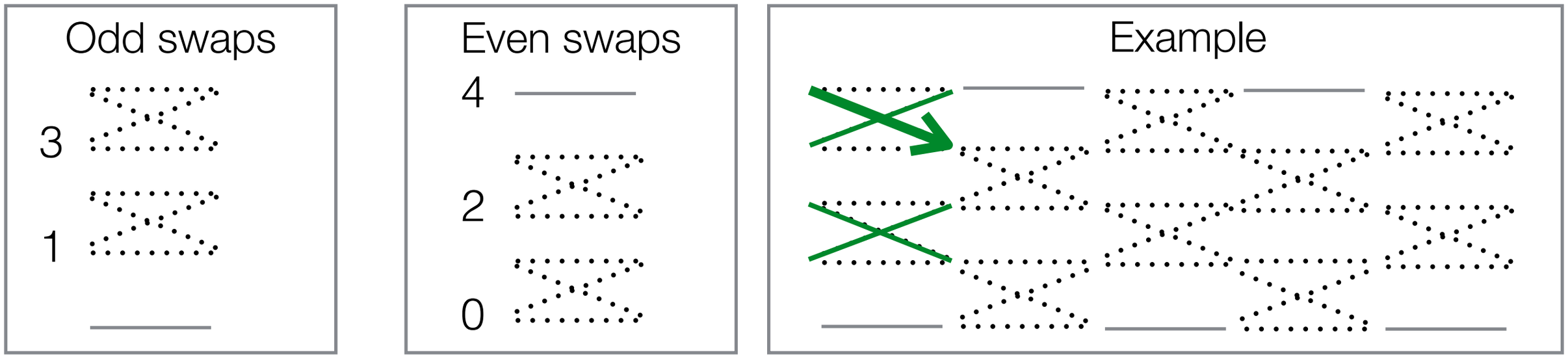
NON-REVERSIBLE PT (OKABE ET AL., 2001)

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



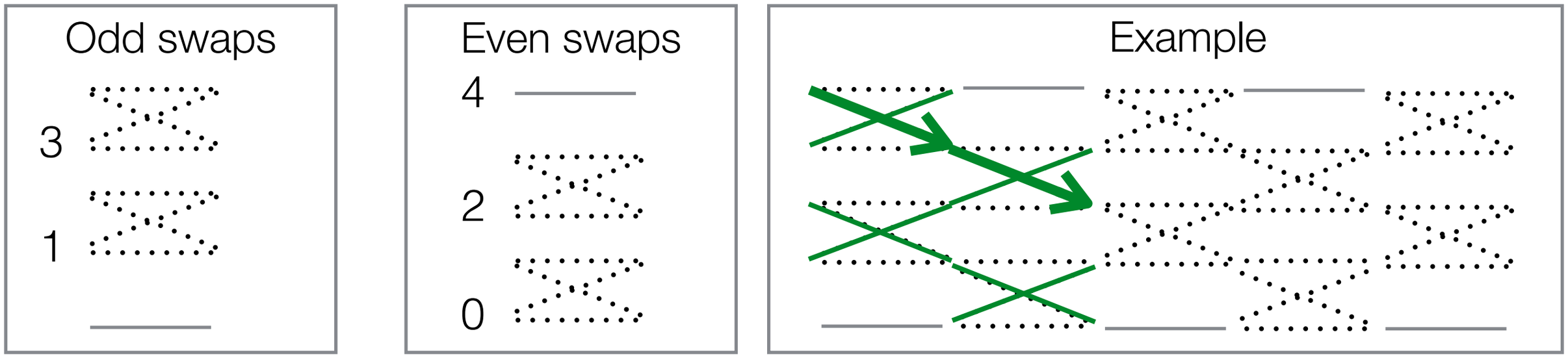
NON-REVERSIBLE PT (OKABE ET AL. 2001)

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



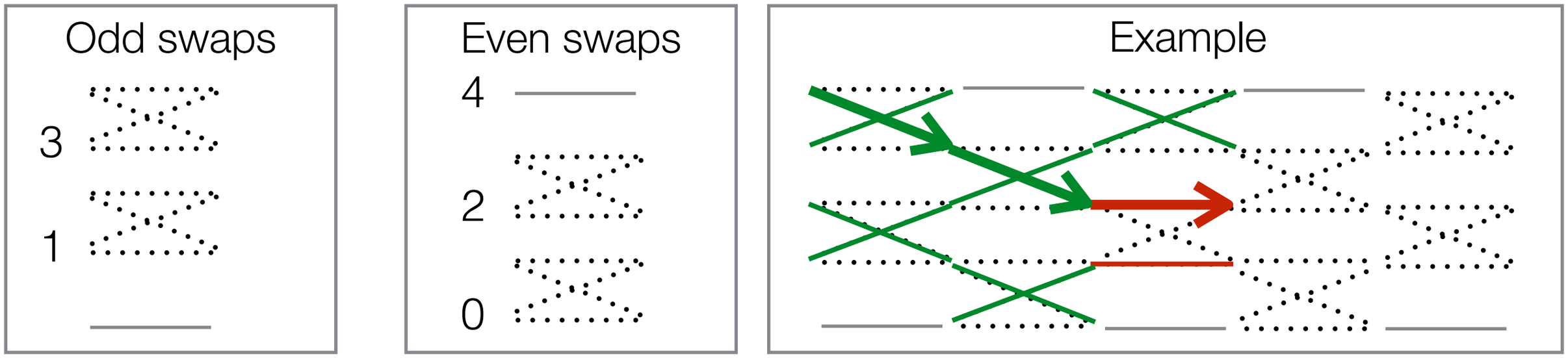
NON-REVERSIBLE PT (OKABE ET AL. 2001)

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



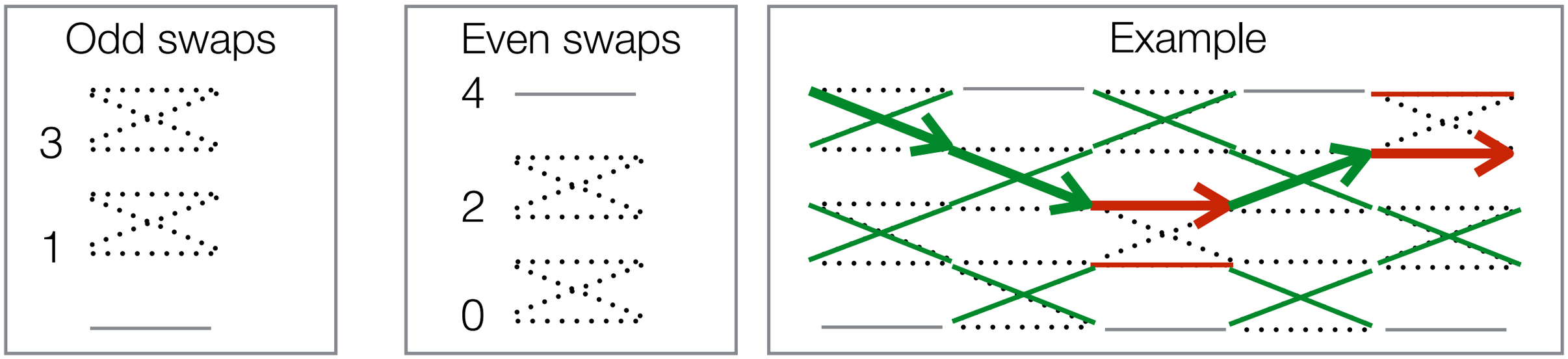
NON-REVERSIBLE PT (OKABE ET AL. 2001)

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



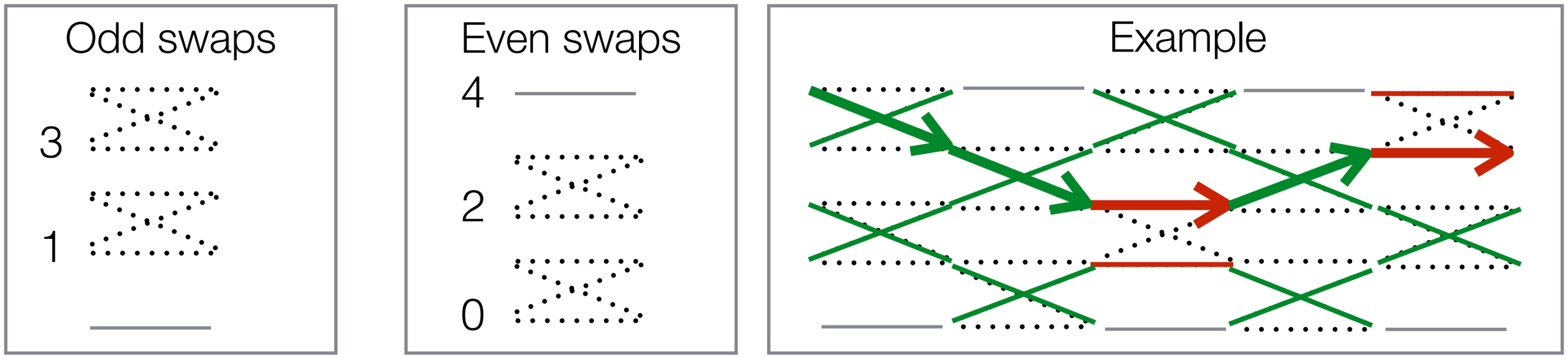
NON-REVERSIBLE PT (OKABE ET AL., 2001)

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



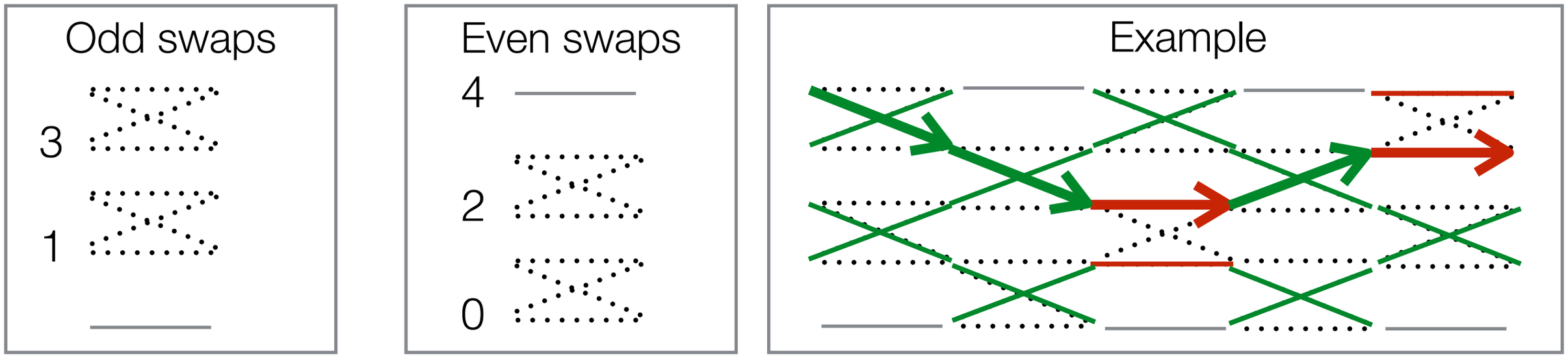
NON-REVERSIBLE PT (OKABE ET AL. 2001)

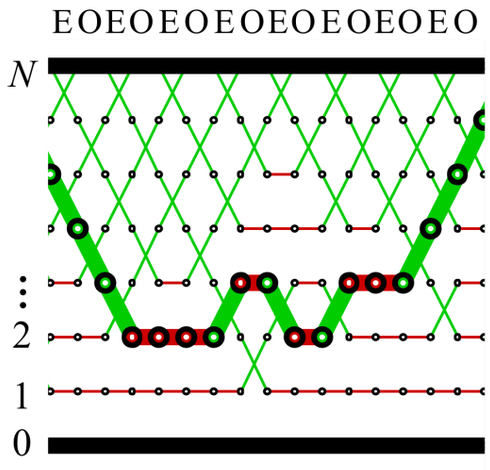
- Deterministically alternate between **Odd** and **Even** swaps
- Non-reversible PT trajectories have momentum!

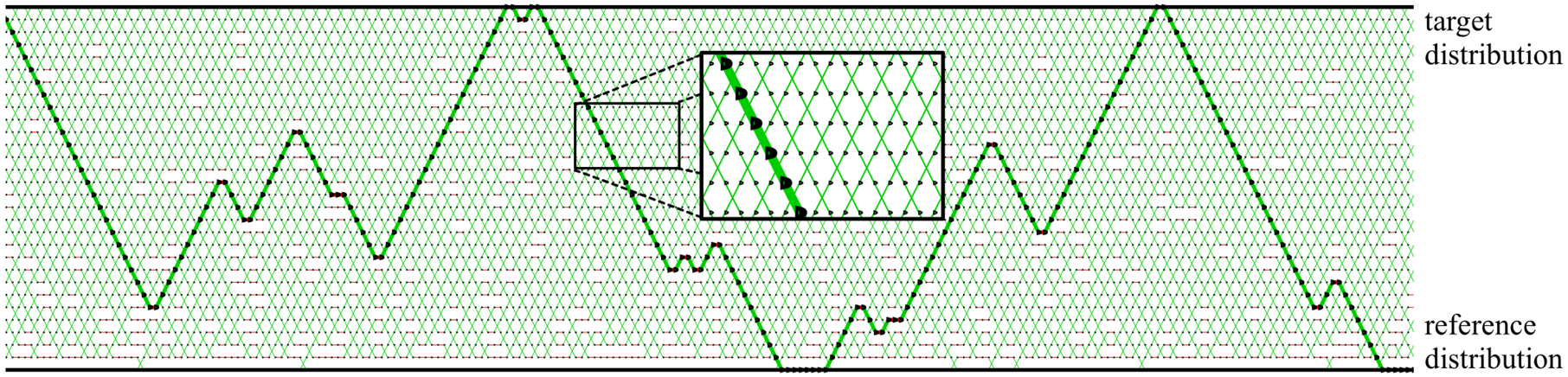


NON-REVERSIBLE PT (OKABE ET AL. 2001)

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







MODEL ASSUMPTIONS

- **(A1) Stationarity:**
  - $\mathbf{X}_t \sim \boldsymbol{\pi}$ for all t

- **(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

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

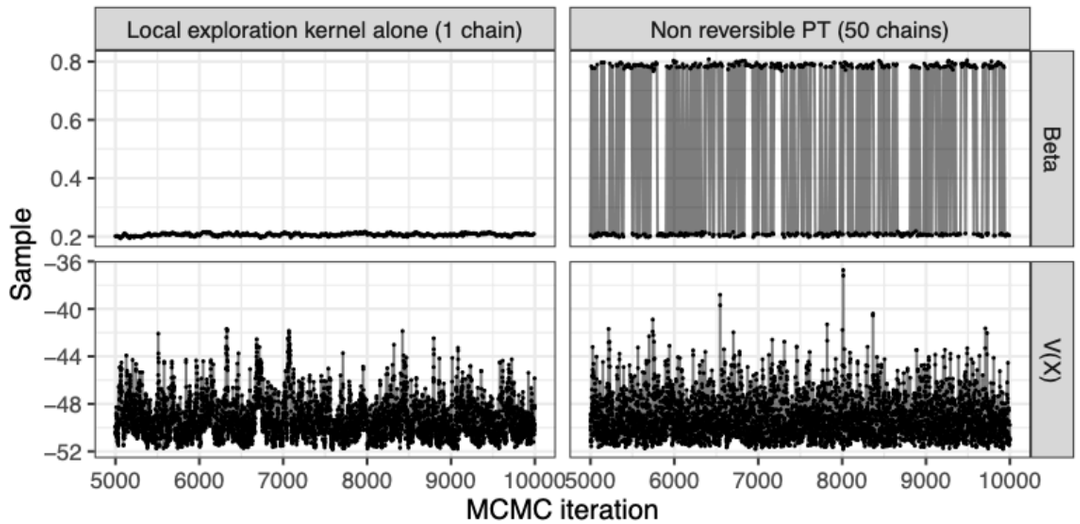
  - Weaker than assuming X and X' are independent!
  - Sufficient to study communication move independent of (problem specific) local exploration move.
  - Methodology and analysis robust to severe violations in practice

- **(A3) Integrability**
  - $\pi_0(|V|^3), \pi_1(|V|^3) < \infty$

VIODLATION OF ELE

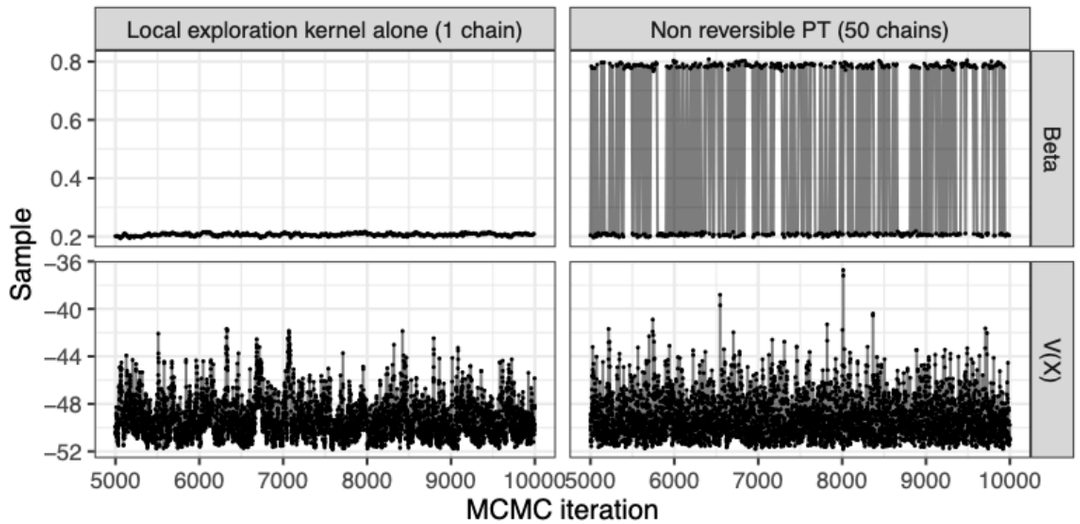
VIODLATION OF ELE

ODE parameter estimation

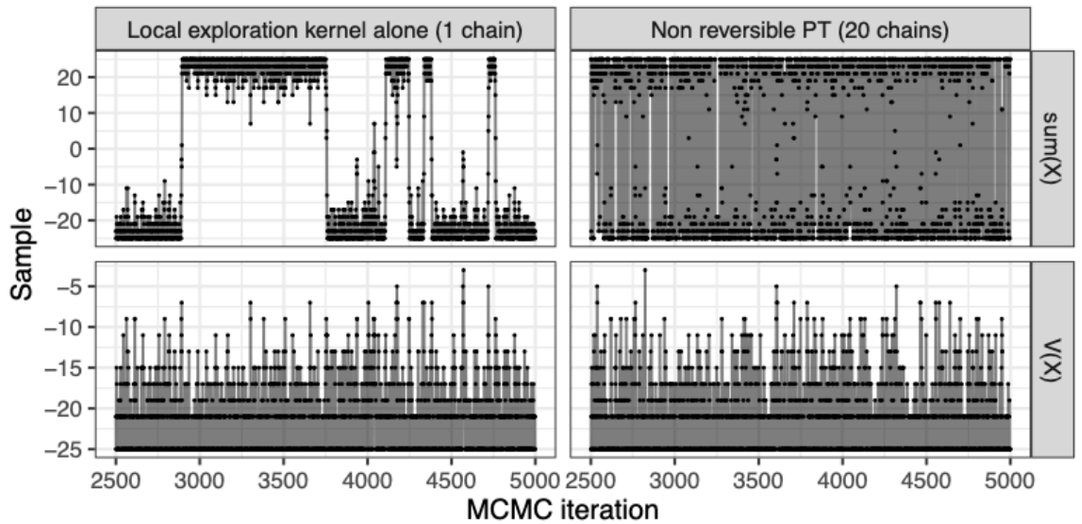


VIODLATION OF ELE

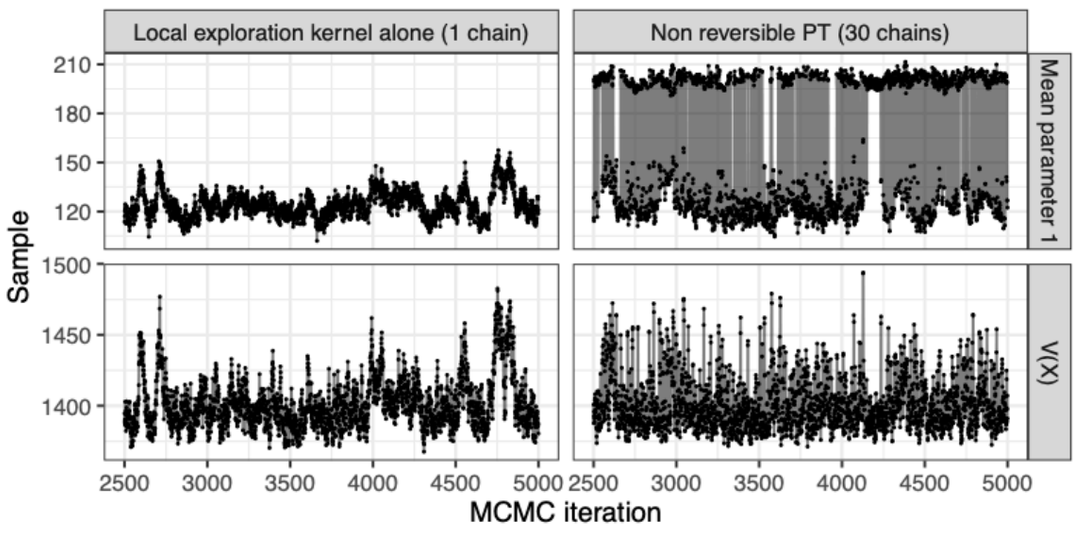
ODE parameter estimation



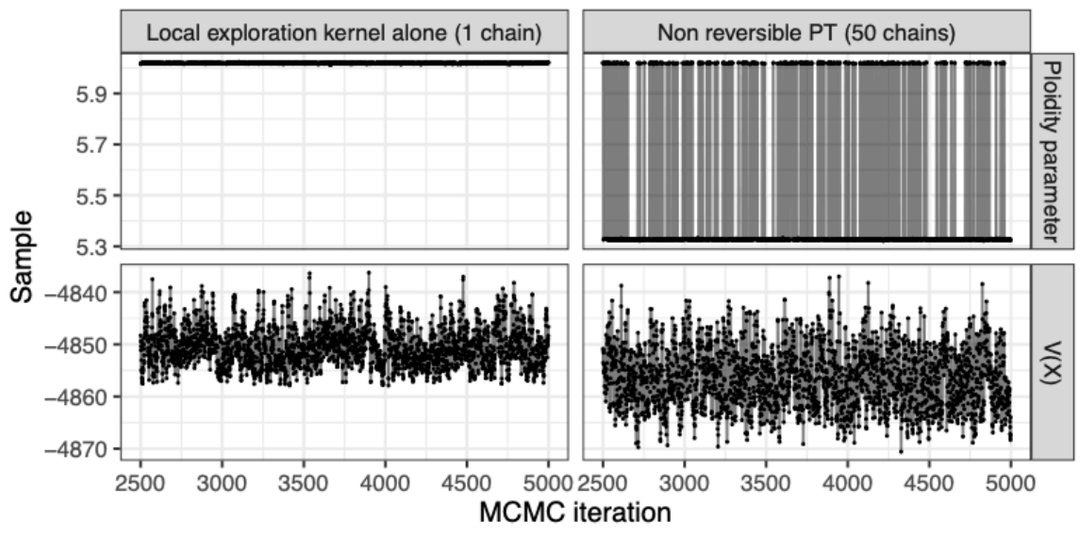
Ising model



Bayesian Mixture Model



Copy-number inference



INDEX PROCESS AS MARKOV CHAIN

INDEX PROCESS AS MARKOV CHAIN

- Makes index process (I_t^m, ε_t^m) a Markov Chain for each machine m.

$$I_{t+1}^m = \begin{cases} I_t^m + \varepsilon_t^m & w.p. \quad s_{n,n+\varepsilon_t^m} \\ I_t^m & w.p. \quad 1 - s_{n,n+\varepsilon_t^m} \end{cases}$$

$$s_{i,j} = \mathbb{E}[\alpha_{i,j}]$$

INDEX PROCESS AS MARKOV CHAIN

- Makes index process (I_t^m, ε_t^m) a Markov Chain for each machine m.

$$I_{t+1}^m = \begin{cases} I_t^m + \varepsilon_t^m & w.p. \quad s_{n,n+\varepsilon_t^m} \\ I_t^m & w.p. \quad 1 - s_{n,n+\varepsilon_t^m} \end{cases}$$

$$s_{i,j} = \mathbb{E}[\alpha_{i,j}]$$

  - For **reversible** PT:

$$\varepsilon_{t+1}^m \sim \text{Uniform}(\{-1, 1\})$$

INDEX PROCESS AS MARKOV CHAIN

- Makes index process (I_t^m, ε_t^m) a Markov Chain for each machine m.

$$I_{t+1}^m = \begin{cases} I_t^m + \varepsilon_t^m & w.p. \quad s_{n,n+\varepsilon_t^m} \\ I_t^m & w.p. \quad 1 - s_{n,n+\varepsilon_t^m} \end{cases}$$

$$s_{i,j} = \mathbb{E}[\alpha_{i,j}]$$

  - For **reversible** PT:

$$\varepsilon_{t+1}^m \sim \text{Uniform}(\{-1, 1\})$$

  - For **non-reversible** PT:

$$\varepsilon_{t+1}^m = \begin{cases} \varepsilon_t^m & \text{if} \quad I_{t+1}^m = I_t^m + \varepsilon_t^m \\ -\varepsilon_t^m & \text{otherwise} \end{cases}$$

INDEX PROCESS AS MARKOV CHAIN

- Makes index process (I_t^m, ε_t^m) a Markov Chain for each machine m.

$$I_{t+1}^m = \begin{cases} I_t^m + \varepsilon_t^m & w.p. \quad s_{n,n+\varepsilon_t^m} \\ I_t^m & w.p. \quad 1 - s_{n,n+\varepsilon_t^m} \end{cases}$$

$$s_{i,j} = \mathbb{E}[\alpha_{i,j}]$$

  - For **reversible** PT:

$$\varepsilon_{t+1}^m \sim \text{Uniform}(\{-1, 1\})$$

  - For **non-reversible** PT:

$$\varepsilon_{t+1}^m = \begin{cases} \varepsilon_t^m & \text{if} \quad I_{t+1}^m = I_t^m + \varepsilon_t^m \\ -\varepsilon_t^m & \text{otherwise} \end{cases}$$

- Makes a round trip a renewal event for each machine m.

INDEX PROCESS AS MARKOV CHAIN

- Makes index process (I_t^m, ε_t^m) a Markov Chain for each machine m.

$$I_{t+1}^m = \begin{cases} I_t^m + \varepsilon_t^m & w.p. \quad s_{n,n+\varepsilon_t^m} \\ I_t^m & w.p. \quad 1 - s_{n,n+\varepsilon_t^m} \end{cases}$$

$$s_{i,j} = \mathbb{E}[\alpha_{i,j}]$$

  - For **reversible** PT:

$$\varepsilon_{t+1}^m \sim \text{Uniform}(\{-1, 1\})$$

  - For **non-reversible** PT:

$$\varepsilon_{t+1}^m = \begin{cases} \varepsilon_t^m & \text{if} \quad I_{t+1}^m = I_t^m + \varepsilon_t^m \\ -\varepsilon_t^m & \text{otherwise} \end{cases}$$

- Makes a round trip a renewal event for each machine m.

- By key-renewal theorem:

$$\tau(\mathcal{B}_N) = \lim_{t \to \infty} \frac{1}{t} \mathbb{E}[\text{Total round trips by time } t]$$

$$= \frac{N+1}{\mathbb{E}[\text{round trip time per machine}]}$$

REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

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

REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

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

(b) If $N \to \infty$ and $\max_{n \le N} |\beta_n - \beta_{n-1}| \to 0$, then for some $\Lambda \ge 0$

$$\tau(\mathcal{B}_N) \sim \frac{1}{2N + 2\Lambda}$$

REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

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

(b) If $N \to \infty$ and $\max_{n \leq N} |\beta_n - \beta_{n-1}| \to 0$, then for some $\Lambda \geq 0$

$$\tau(\mathcal{B}_N) \sim \frac{1}{2N + 2\Lambda}$$

- Reversible PT deteriorates with too many chains

REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

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

(b) If $N \to \infty$ and $\max_{n \leq N} |\beta_n - \beta_{n-1}| \to 0$, then for some $\Lambda \geq 0$

$$\tau(\mathcal{B}_N) \sim \frac{1}{2N + 2\Lambda}$$

- Reversible PT deteriorates with too many chains
- Very sensitive to schedule and number of chains

REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

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

(b) If $N \to \infty$ and $\max_{n \leq N} |\beta_n - \beta_{n-1}| \to 0$, then for some $\Lambda \geq 0$

$$\tau(\mathcal{B}_N) \sim \frac{1}{2N + 2\Lambda}$$

- Reversible PT deteriorates with too many chains
- Very sensitive to schedule and number of chains
- Not scalable to GPUs

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

$2N$

- Non-asymptotically dominates **reversible PT**

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

$2N$

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

$$\tau(\mathcal{B}_N) \longrightarrow \frac{1}{2 + 2\Lambda}$$

- Non-asymptotically dominates **reversible PT**

NON-REVERSIBLE PT

Theorem: (a) For any annealing schedule $\mathcal{B}_N$,

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

$2N$

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

$$\tau(\mathcal{B}_N) \longrightarrow \frac{1}{2 + 2\Lambda}$$

"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 $\mathcal{B}_N$,

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

$2N$

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

$$\tau(\mathcal{B}_N) \longrightarrow \frac{1}{2 + 2\Lambda}$$

"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 $\mathcal{B}_N$,

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

$2N$

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

$$\tau(\mathcal{B}_N) \longrightarrow \frac{1}{2 + 2\Lambda}$$

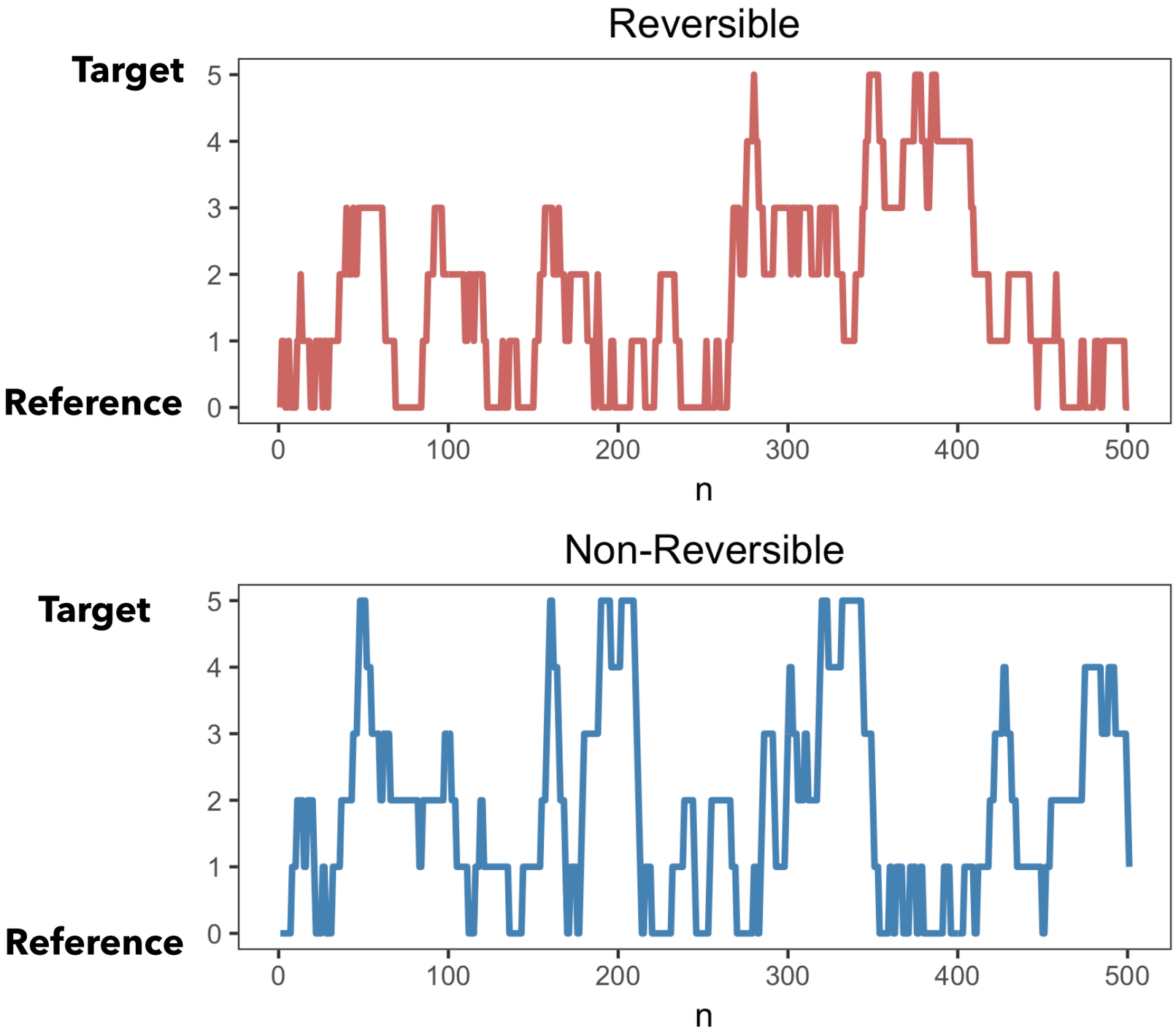
"Communication barrier"

- 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



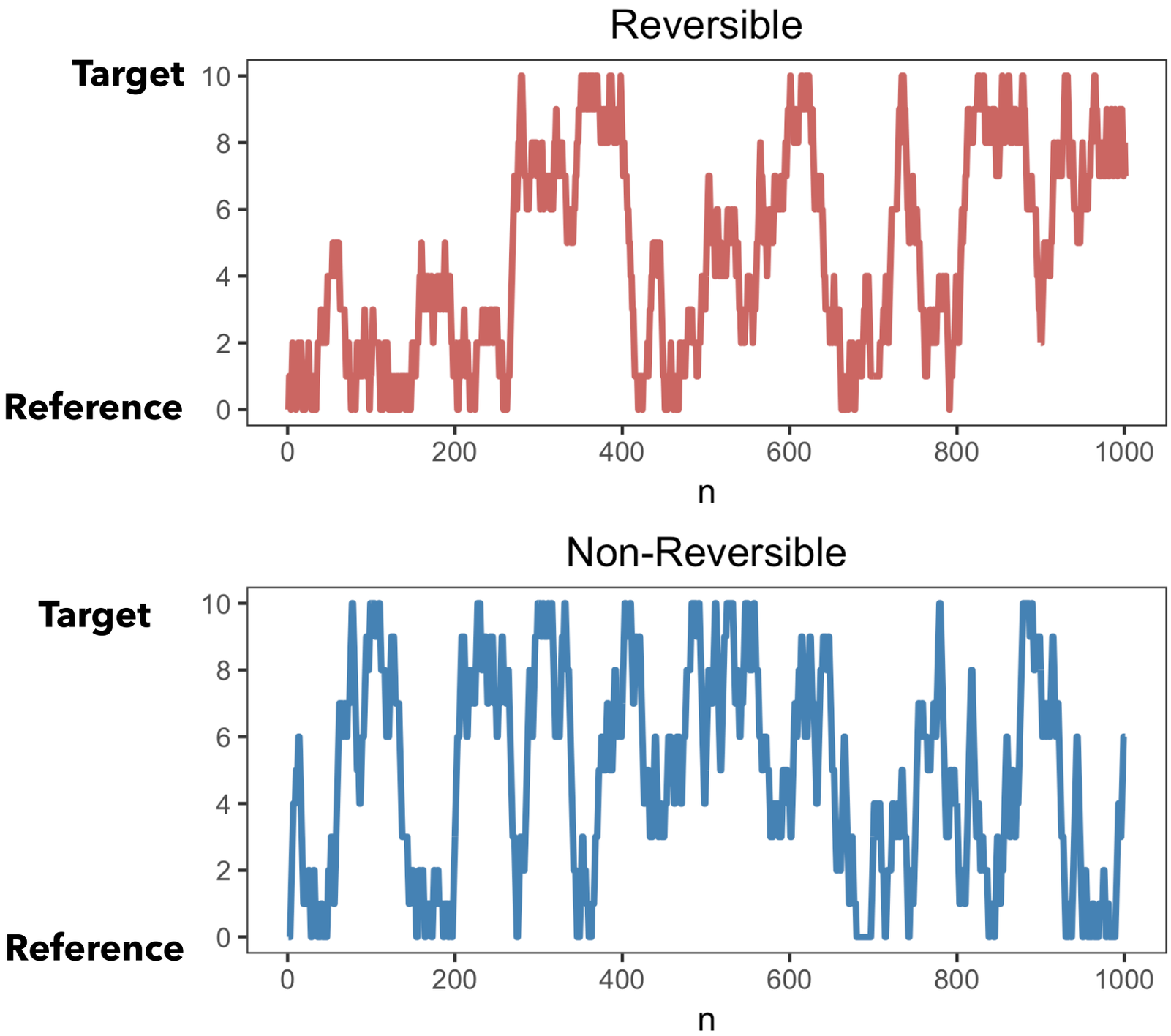
1

4

SAMPLE TRAJECTORIES

$$N = 10$$

 # Round Trips



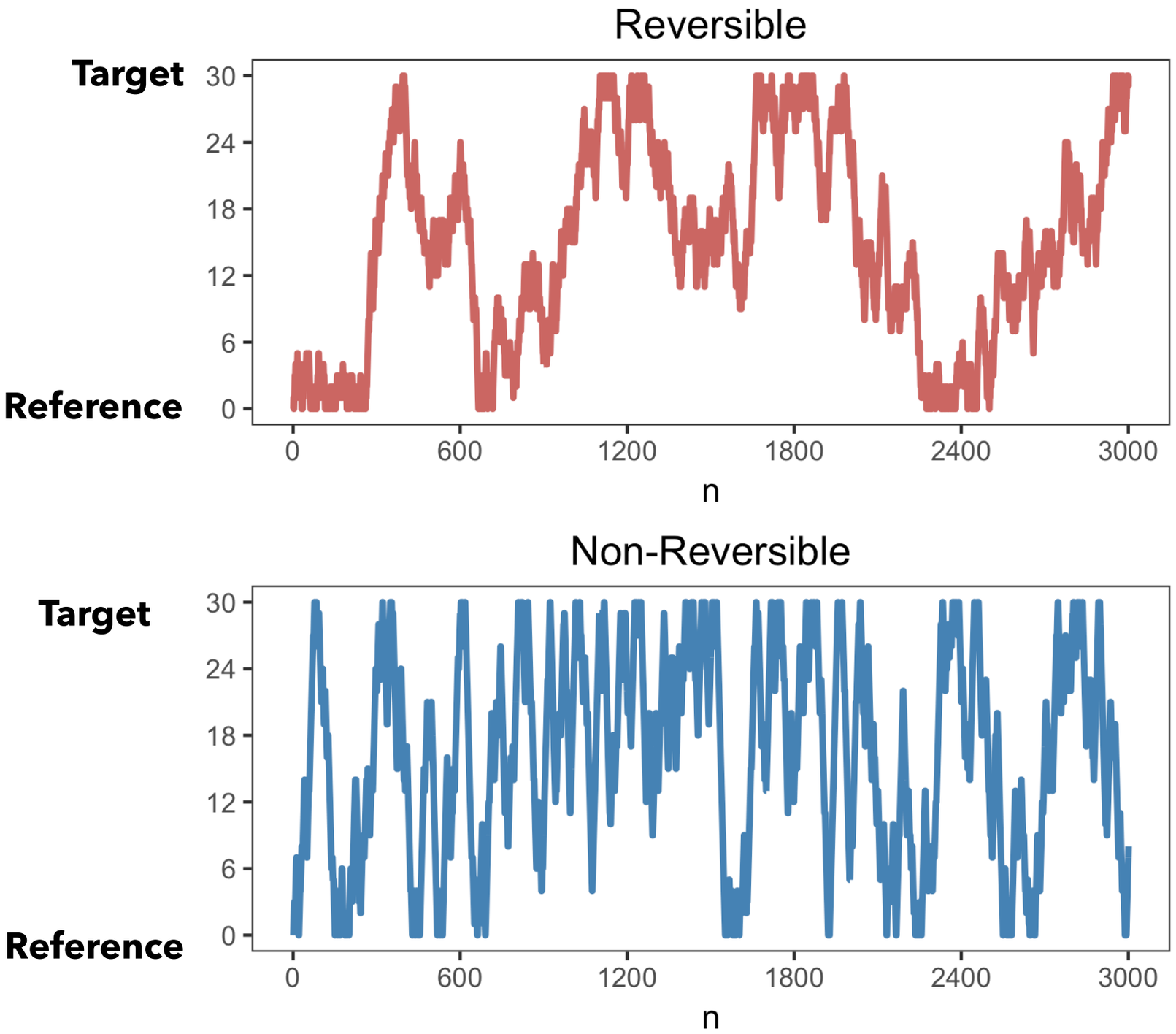
3

6

SAMPLE TRAJECTORIES

$$N = 30$$

 # Round Trips



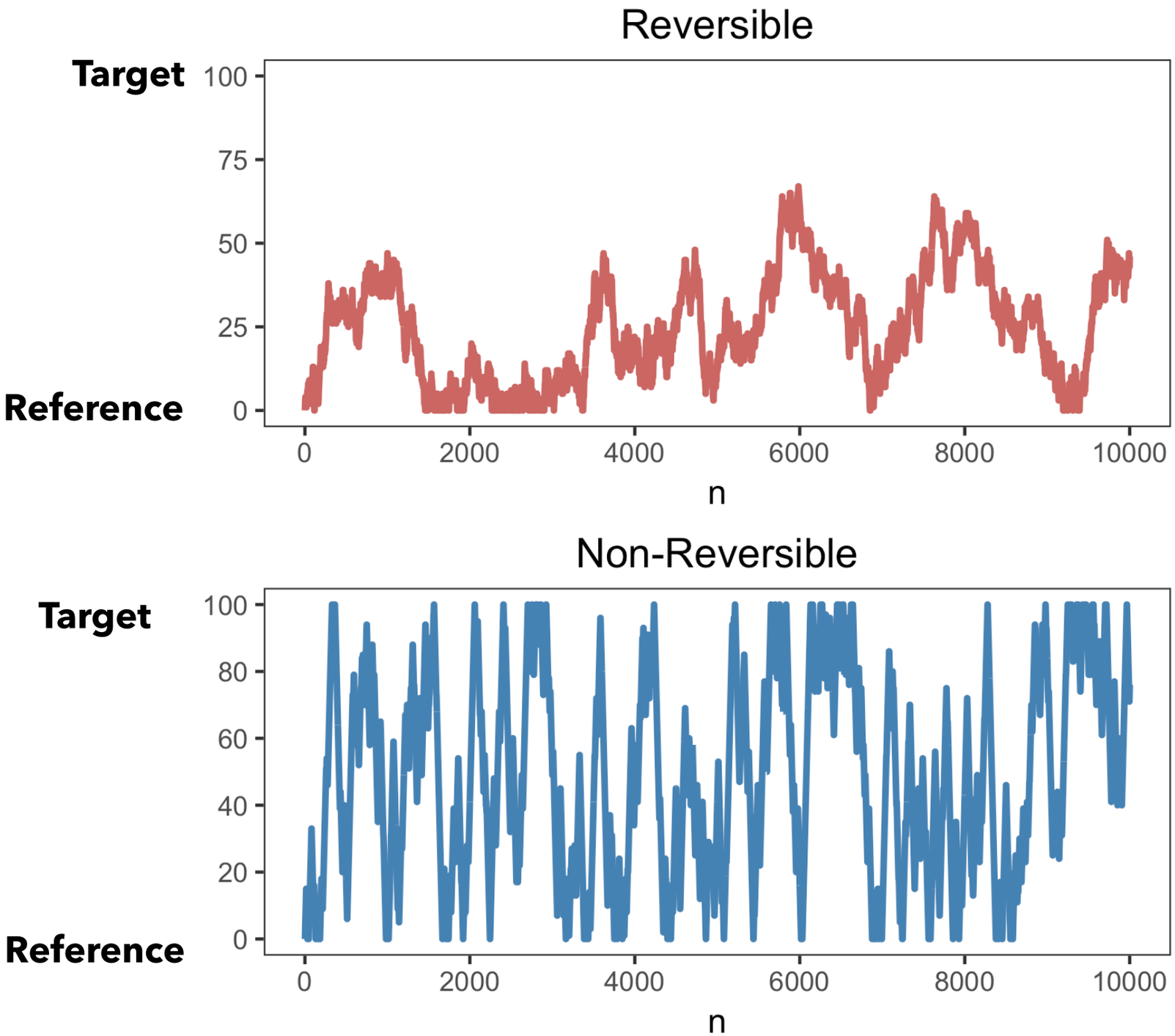
2

8

SAMPLE TRAJECTORIES

$$N = 100$$

 # Round Trips



0

9

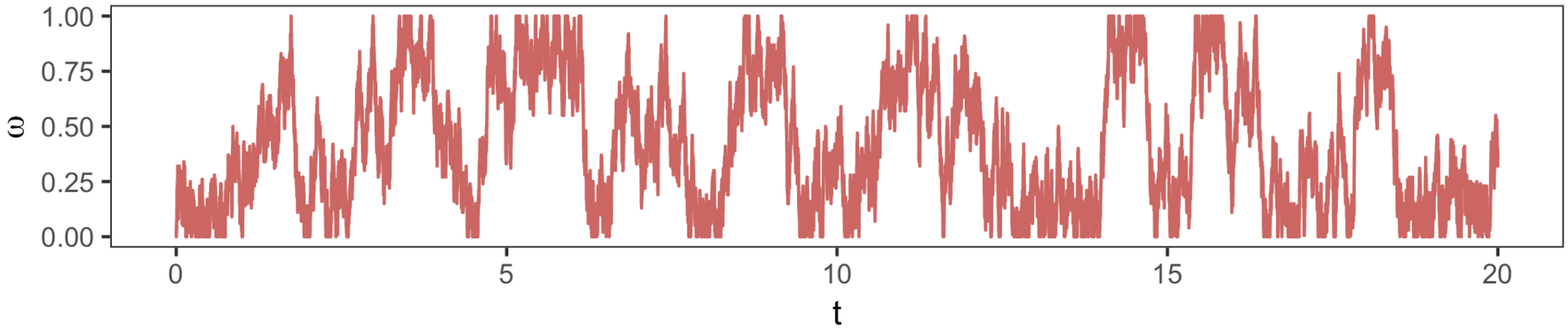
SCALING LIMIT (REVERSIBLE)

SCALING LIMIT (REVERSIBLE)

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

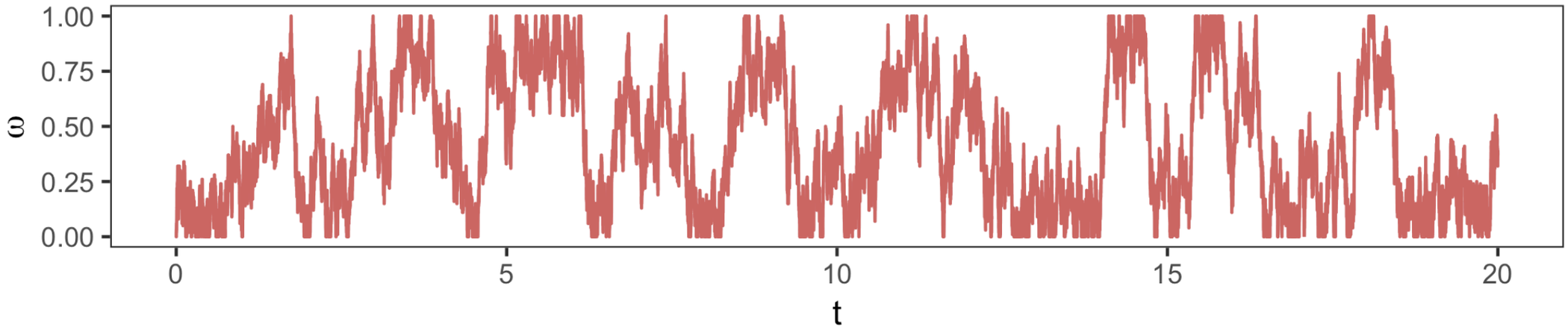
SCALING LIMIT (REVERSIBLE)

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



SCALING LIMIT (REVERSIBLE)

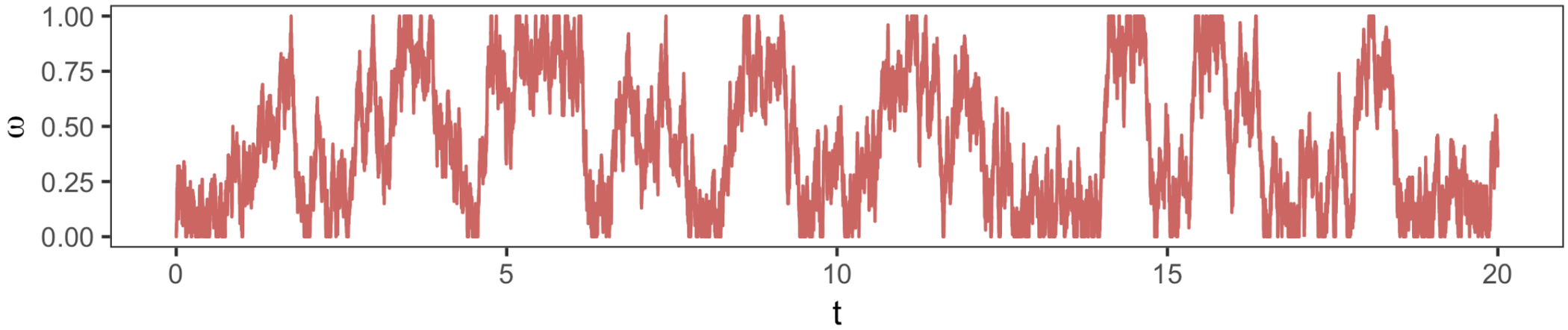
Theorem: Scaling the index by N and time by N^2, the scaled index process for reversible PT weakly converge to a diffusion independent of target, reference, or annealing path.



- As N grows large, noise from diffusivity dominates

SCALING LIMIT (REVERSIBLE)

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



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

SCALING LIMIT (NON-REVERSIBLE)

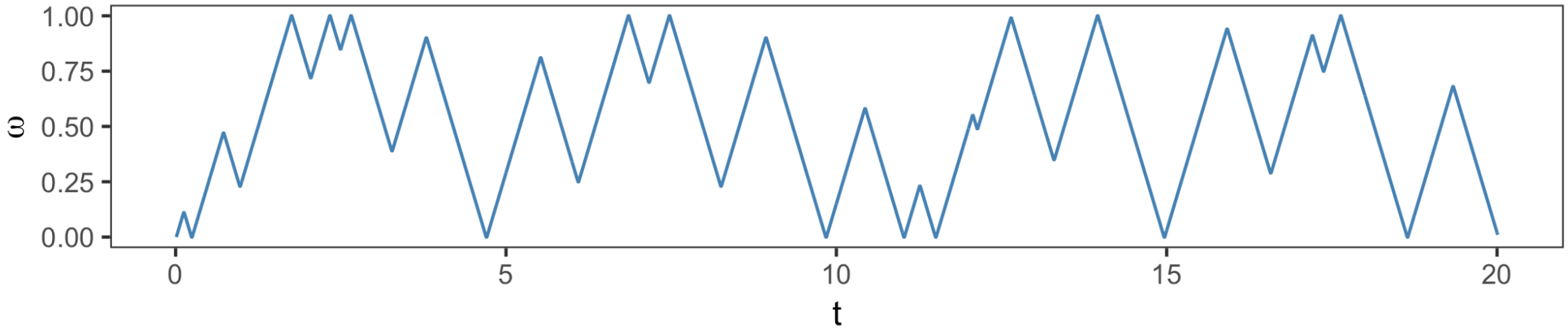
SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling the index by N and time by N, the scaled index process for non-reversible PT converge to piecewise deterministic Markov process depending on the annealing path through Λ

SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling the index by N and time by N, the scaled index process for non-reversible PT converge to piecewise deterministic Markov process depending on the annealing path through Λ

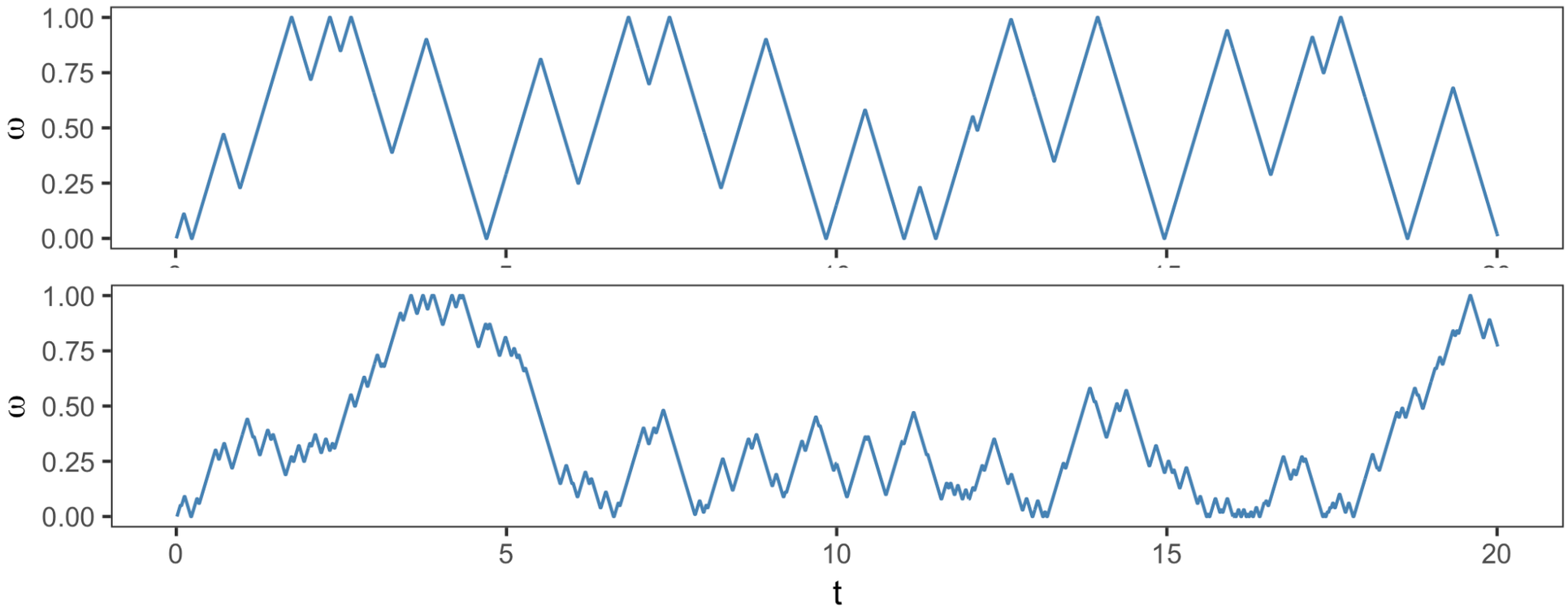
$\Lambda = 1$



SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling the index by N and time by N, the scaled index process for non-reversible PT converge to piecewise deterministic Markov process depending on the annealing path through Λ

$\Lambda = 1$

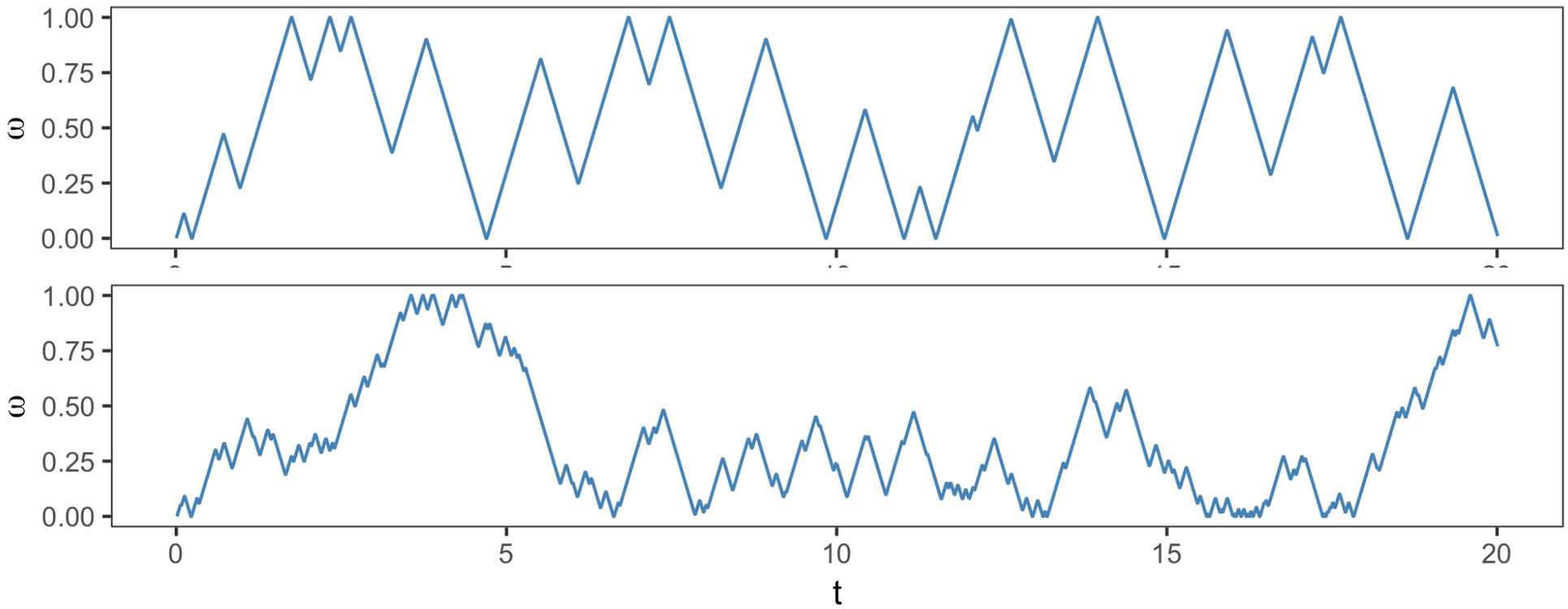


$\Lambda = 10$

SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling the index by N and time by N, the scaled index process for non-reversible PT converge to piecewise deterministic Markov process depending on the annealing path through Λ

$\Lambda = 1$



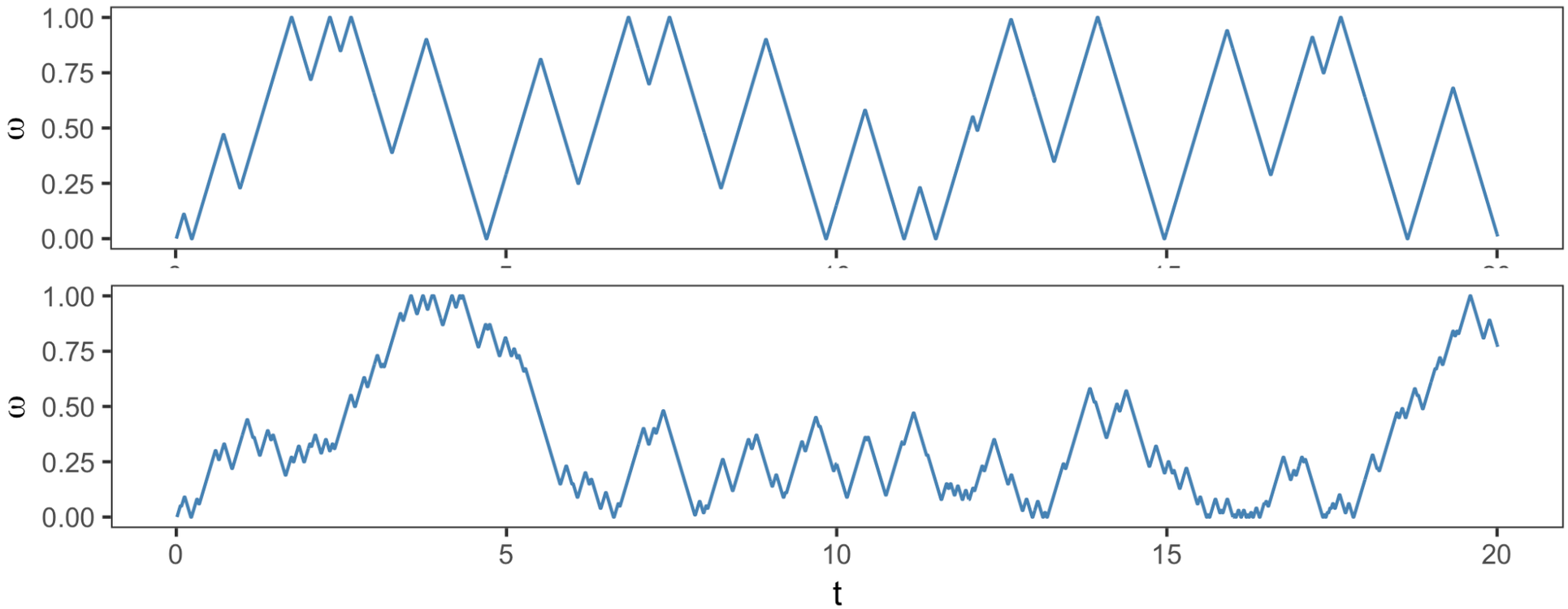
$\Lambda = 10$

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

SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling the index by N and time by N, the scaled index process for non-reversible PT converge to piecewise deterministic Markov process depending on the annealing path through Λ

$\Lambda = 1$



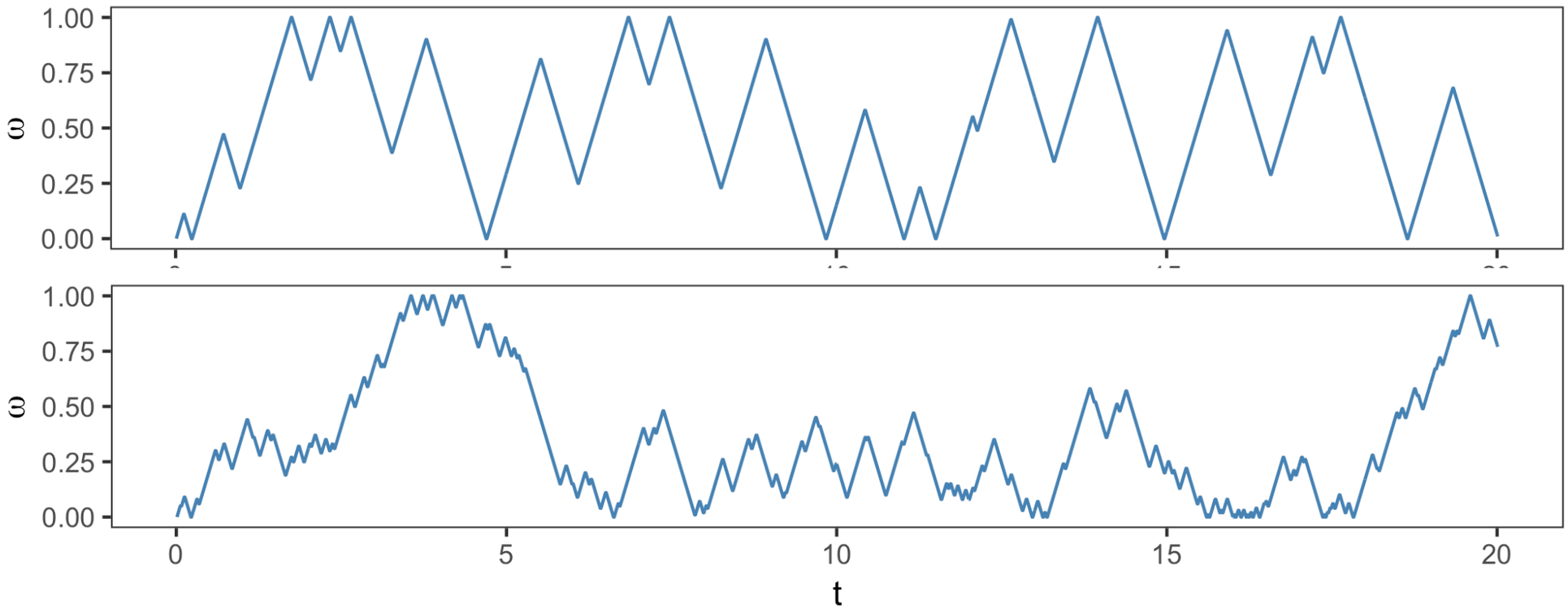
$\Lambda = 10$

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

SCALING LIMIT (NON-REVERSIBLE)

Theorem: Scaling the index by N and time by N, the scaled index process for non-reversible PT converge to piecewise deterministic Markov process depending on the annealing path through 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

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

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

COMPUTE SCHEDULE

Theorem:

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

- 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:

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

- 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:

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

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:

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

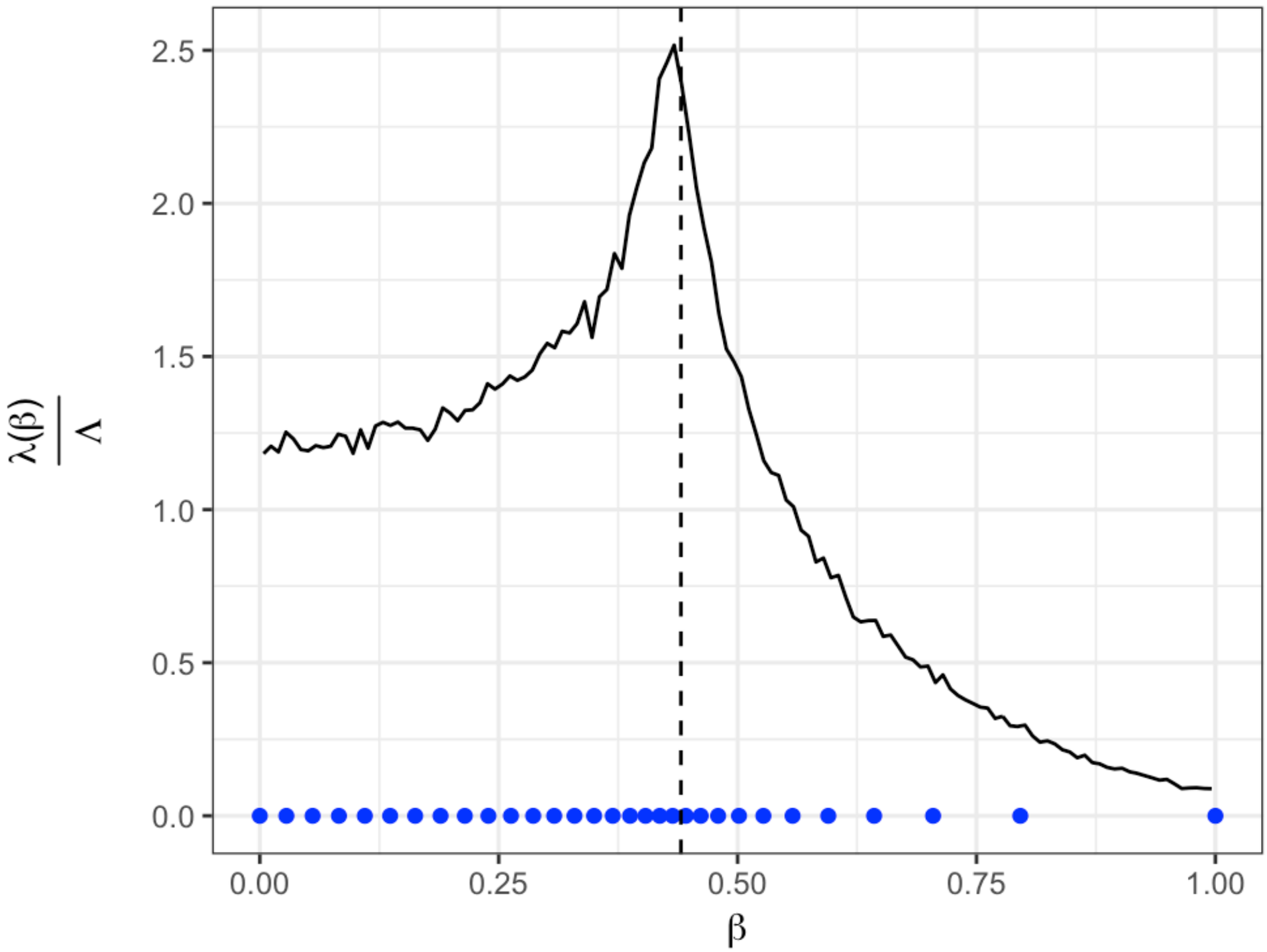
- By ignoring error term, we have a constraint optimization problem with solution:

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

- Theorem implies:

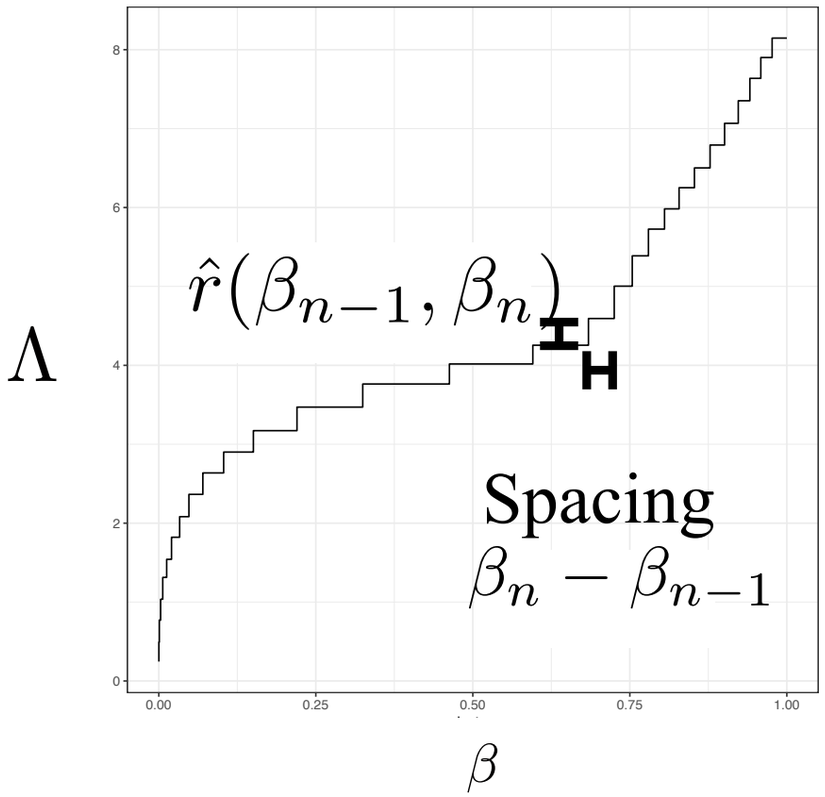
$$\int_{\beta_{n-1}}^{\beta_n} \lambda(\beta)\mathrm{d}\beta = \frac{\Lambda}{N}$$

EXAMPLE ISING MODEL

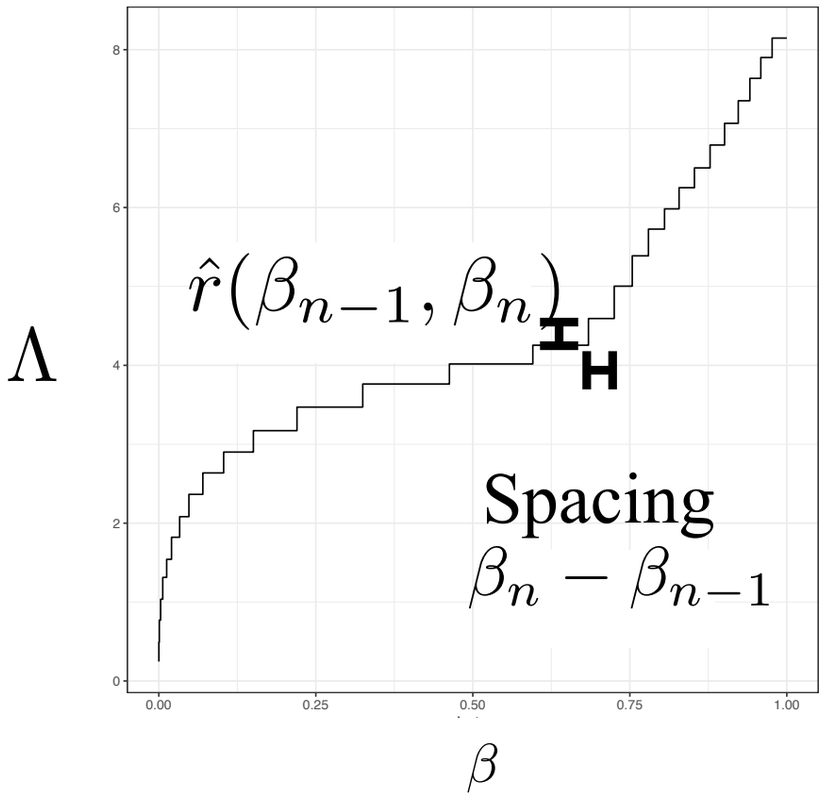


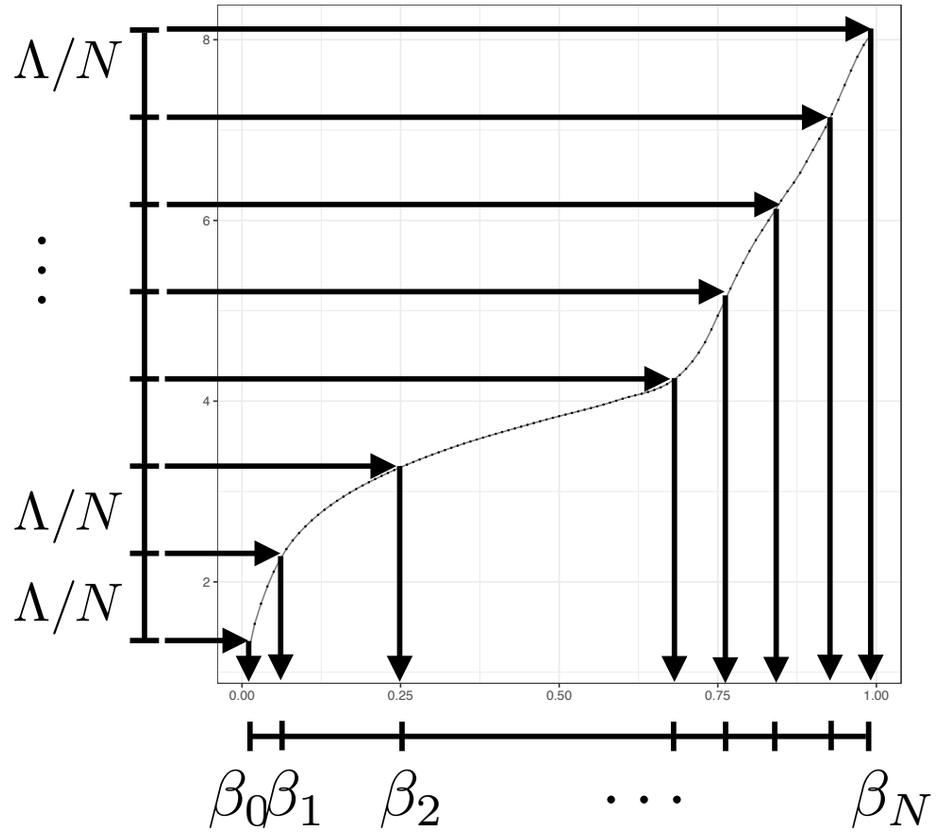
COMPUTE SCHEDULE

COMPUTE SCHEDULE

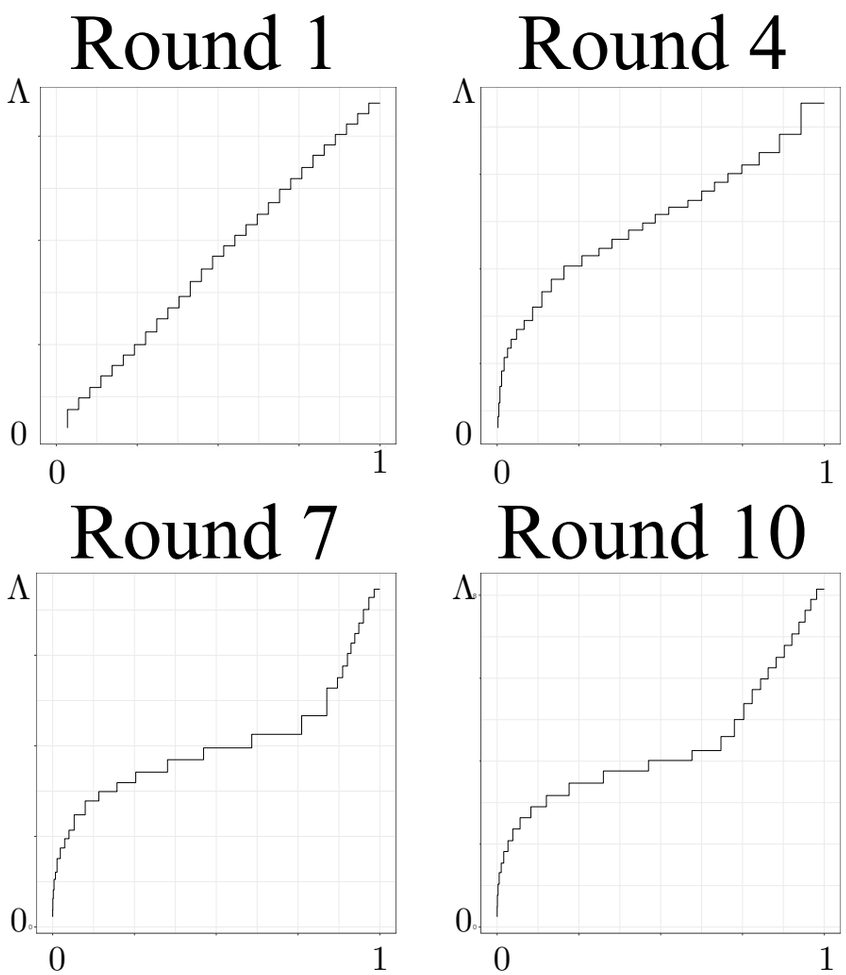


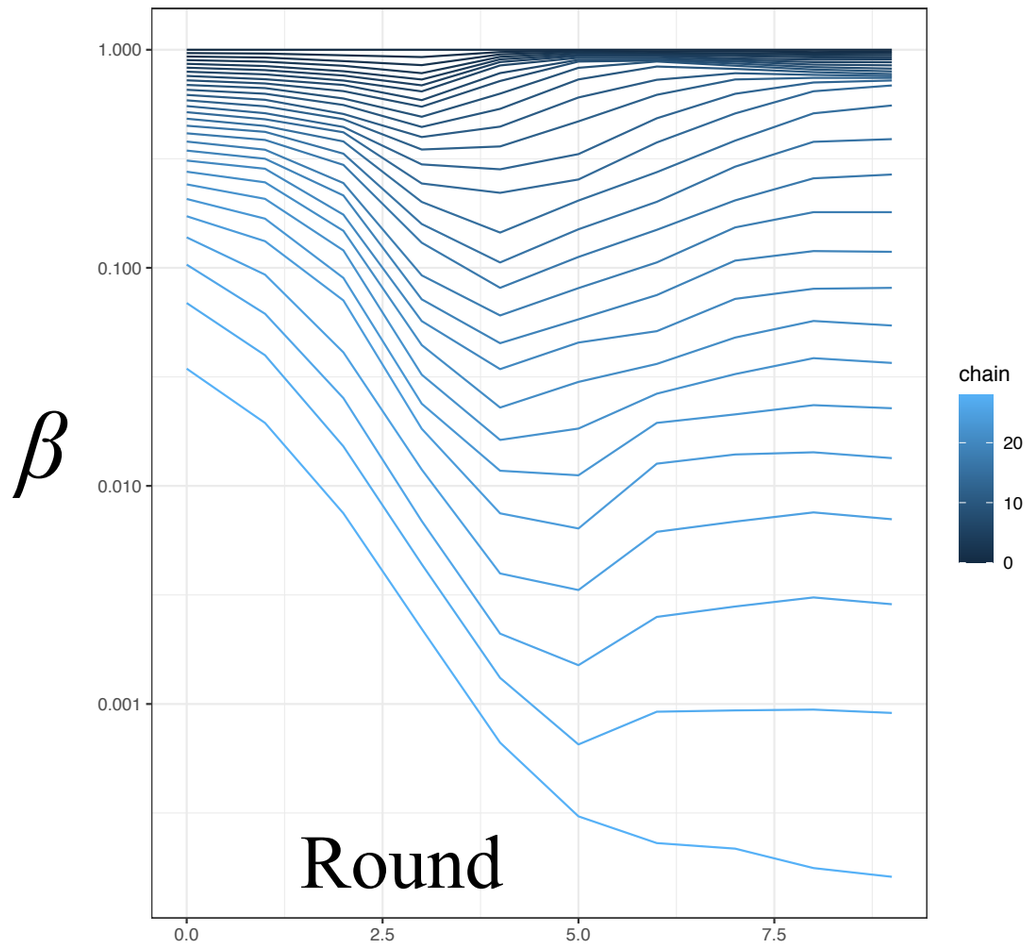
COMPUTE SCHEDULE



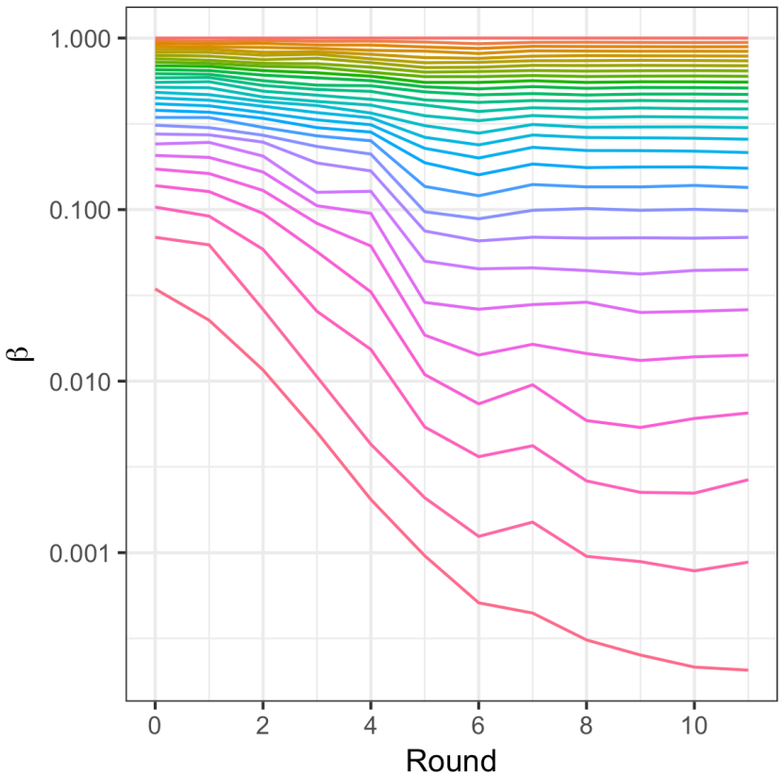


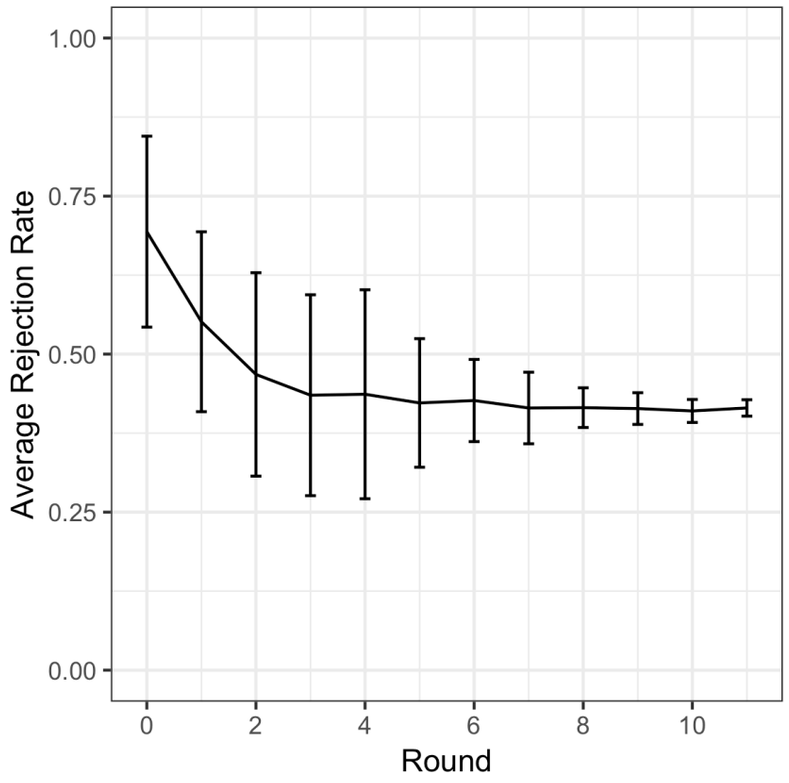
COMPUTE SCHEDULE

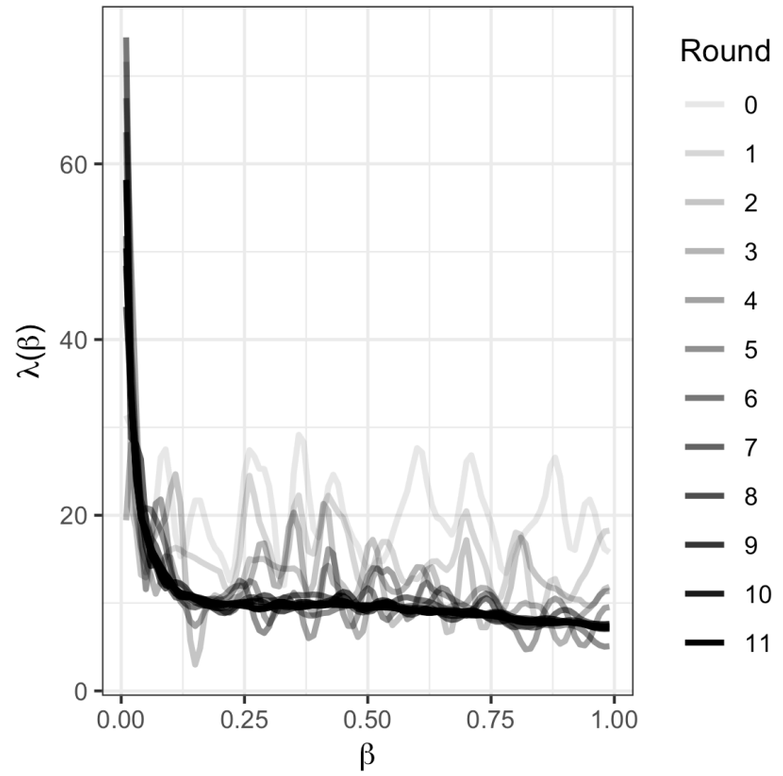




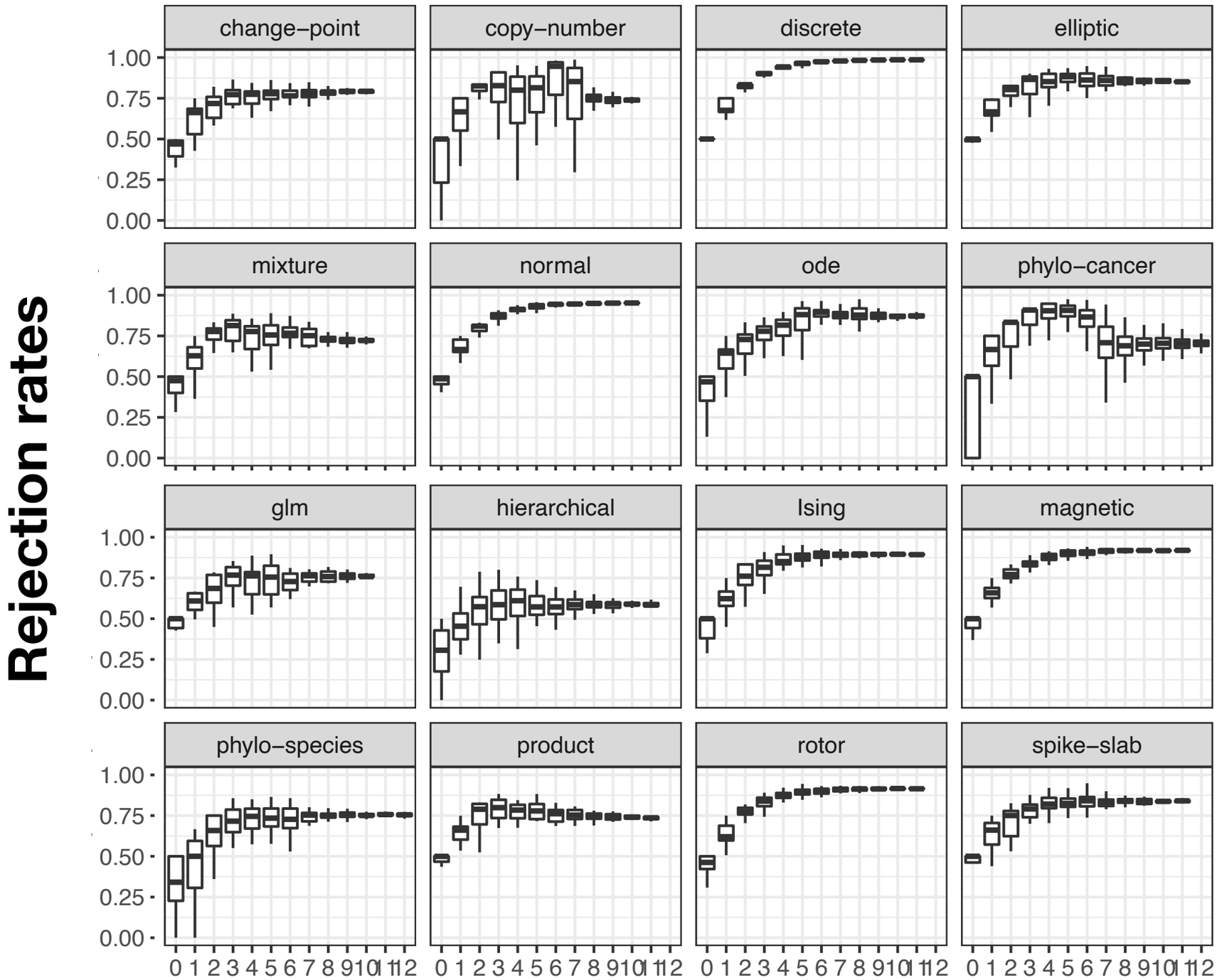
HIERARCHICAL BAYESIAN MODEL (D = 369)





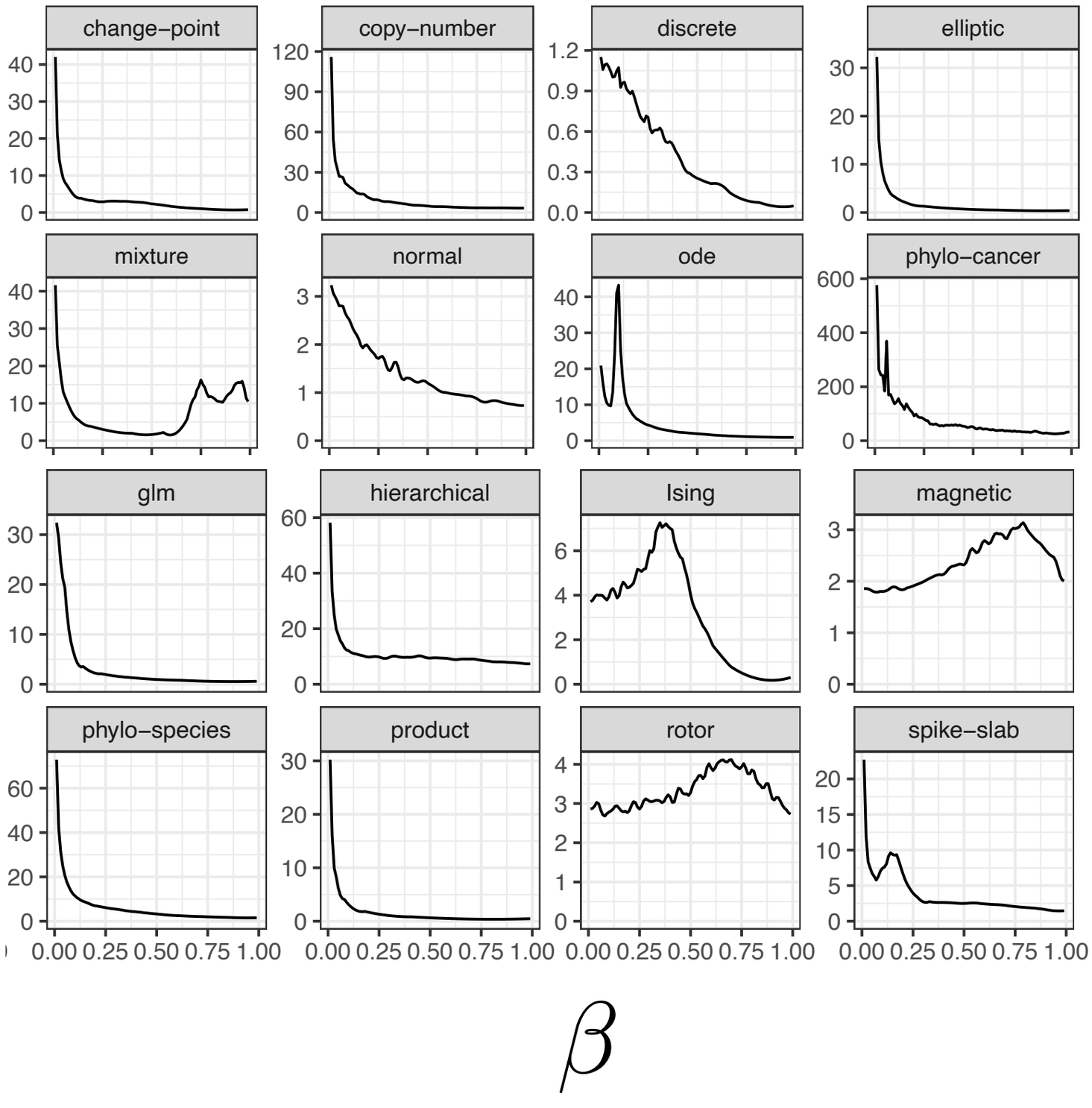


NRPT IN PRACTICE



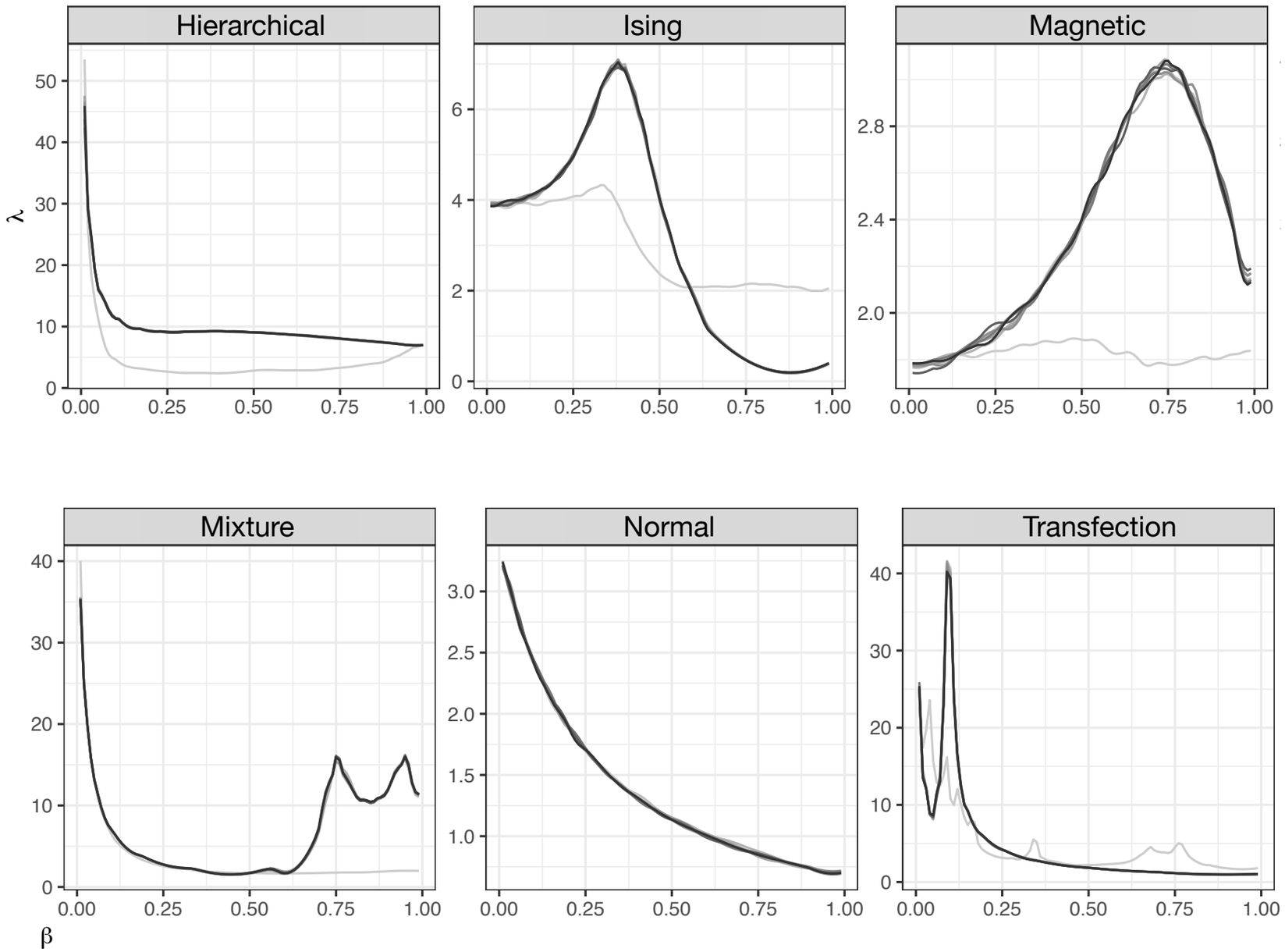
LOCAL COMMUNICATION BARRIER

$\lambda(\beta)$



β

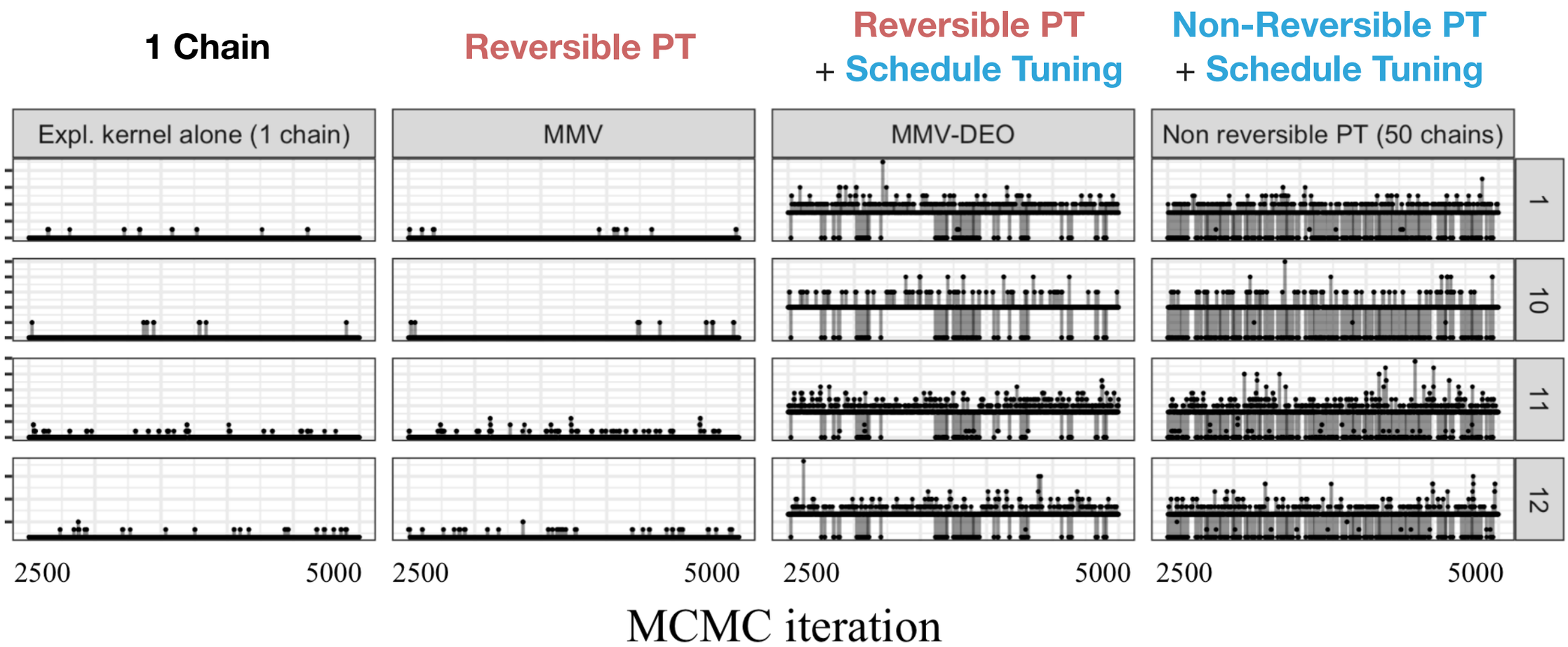
TUNING ROBUST TO ELE VIOLATION



Variables updated per local exploration move

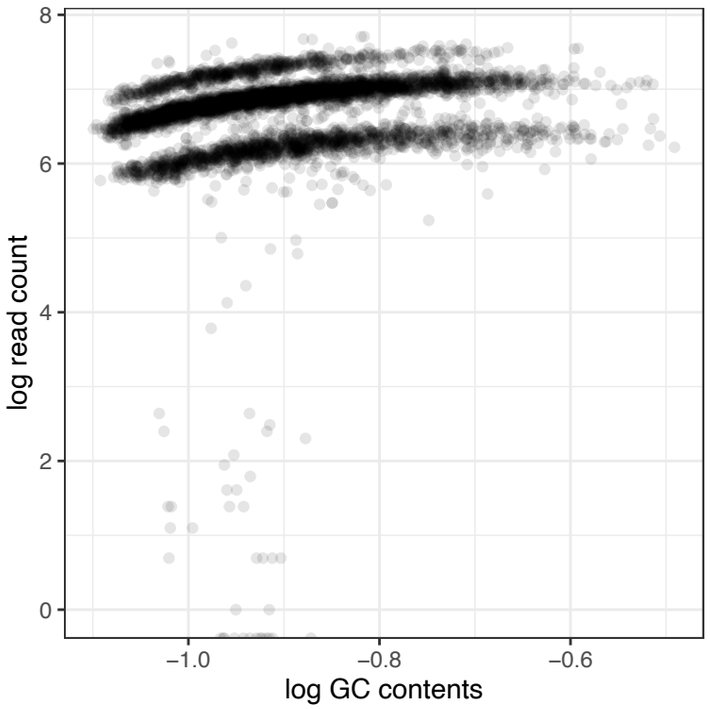
- 0
- 0.5
- 1
- 2
- 4
- 8
- 16
- 32

EXAMPLE: COPY NUMBER INFERENCE (D = 30)

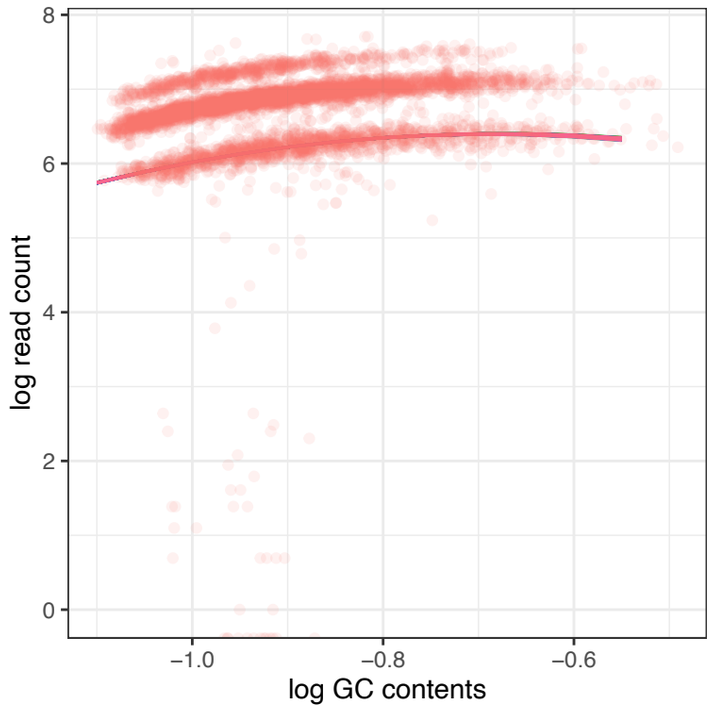


EXAMPLE: COPY NUMBER INFERENCE (D = 30)

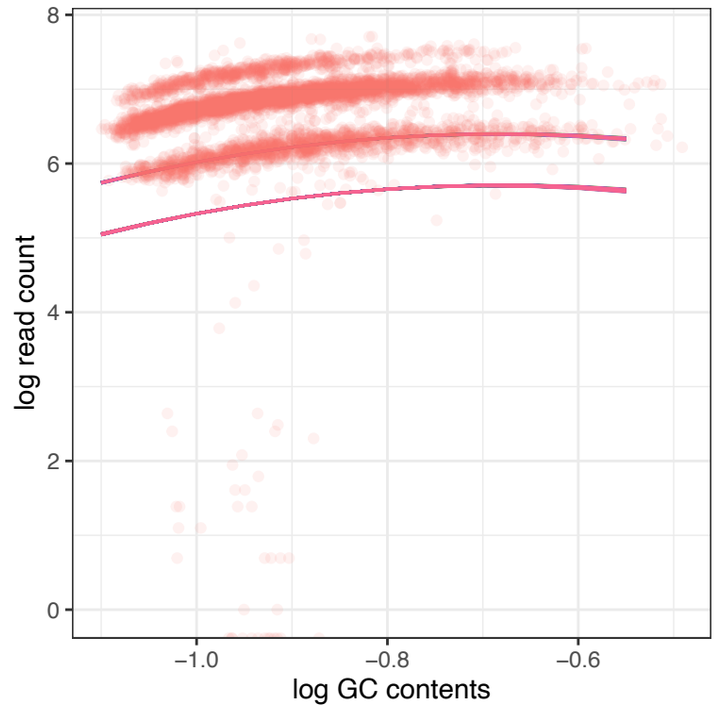
Data



1 Chain

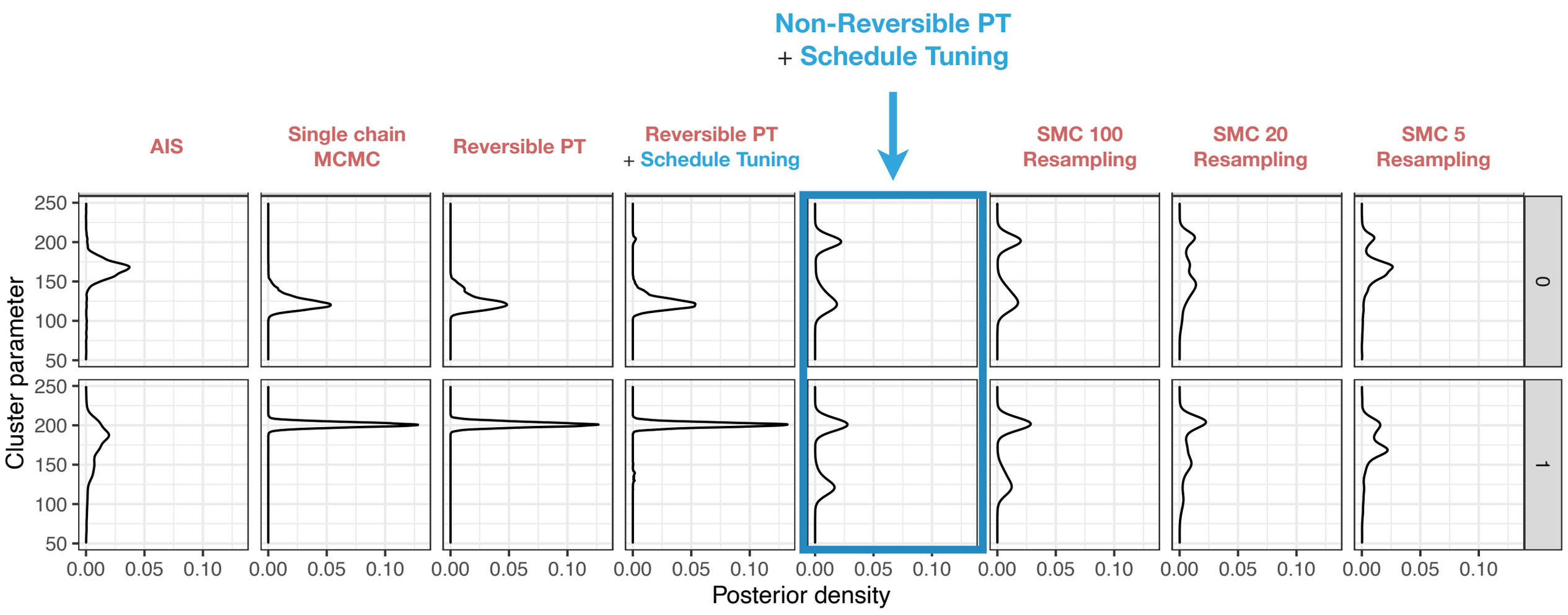


Non-Reversible PT



COMPARISON TO STATE OF THE ART

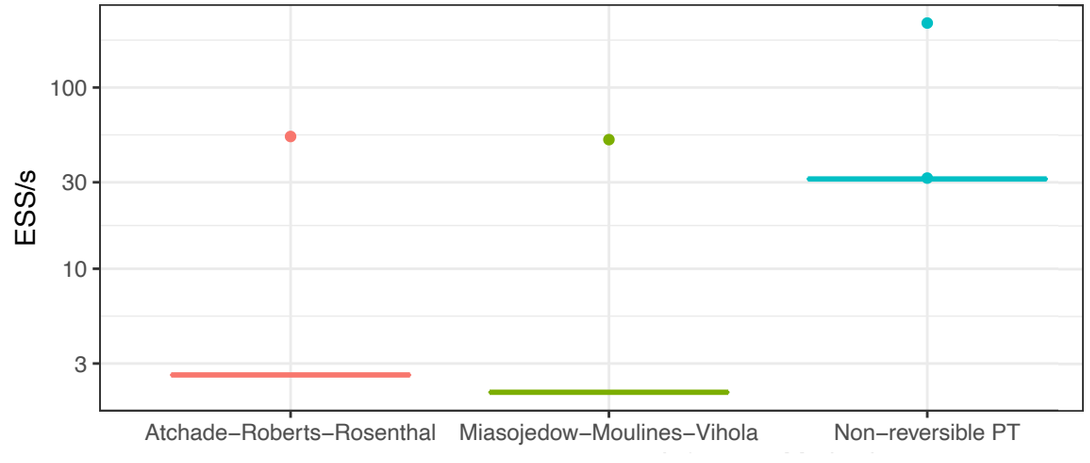
Bayesian Mixture Model (d = 155)



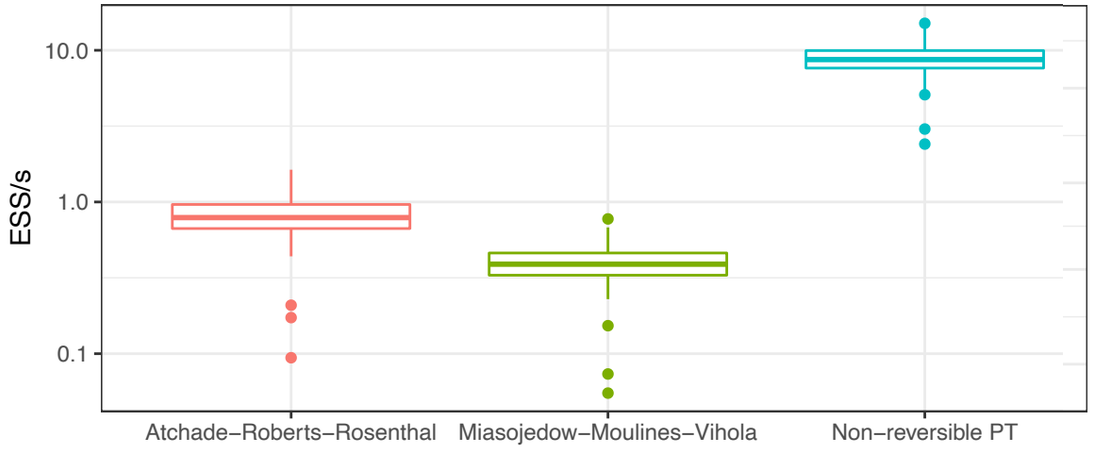
ESS COMPARED TO REVERSIBLE PT

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 (Miasojedow, et al. 2013)

Non-Reversible PT

EVENT HORIZON TELESCOPE

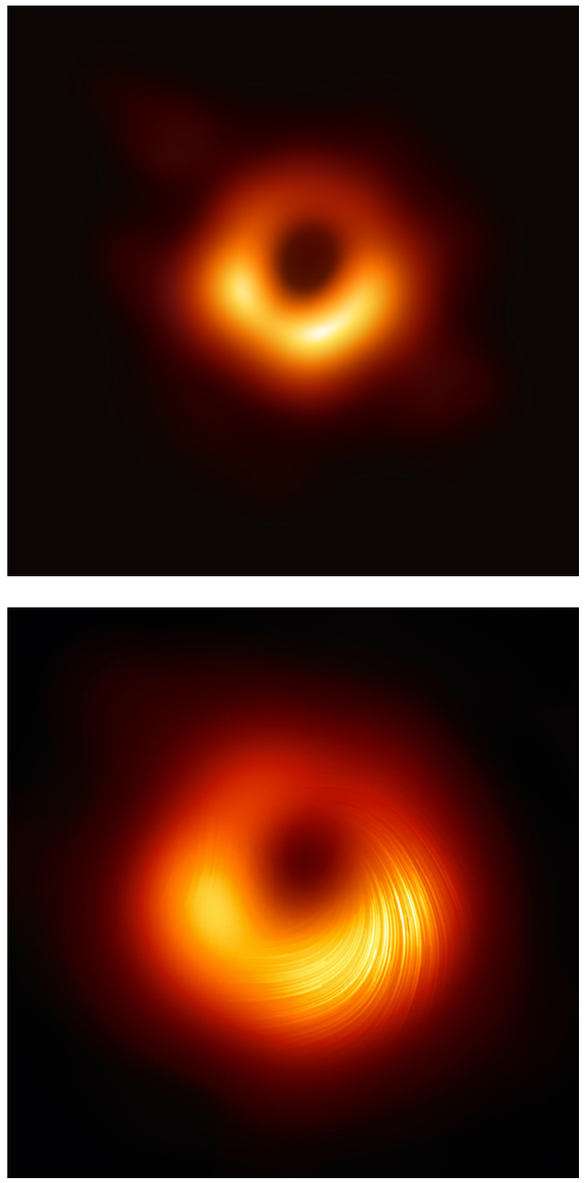
EVENT HORIZON TELESCOPE

- Event horizon telescope (EHT) used NRPT+HMC to achieve better performance within 2% of computation budget of original photo of blackhole M87 without NRPT (Tiede, 2021)

EVENT HORIZON TELESCOPE

- Event horizon telescope (EHT) used NRPT+HMC to achieve better performance within 2% of computation budget of original photo of blackhole M87 without NRPT (Tiede, 2021)

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

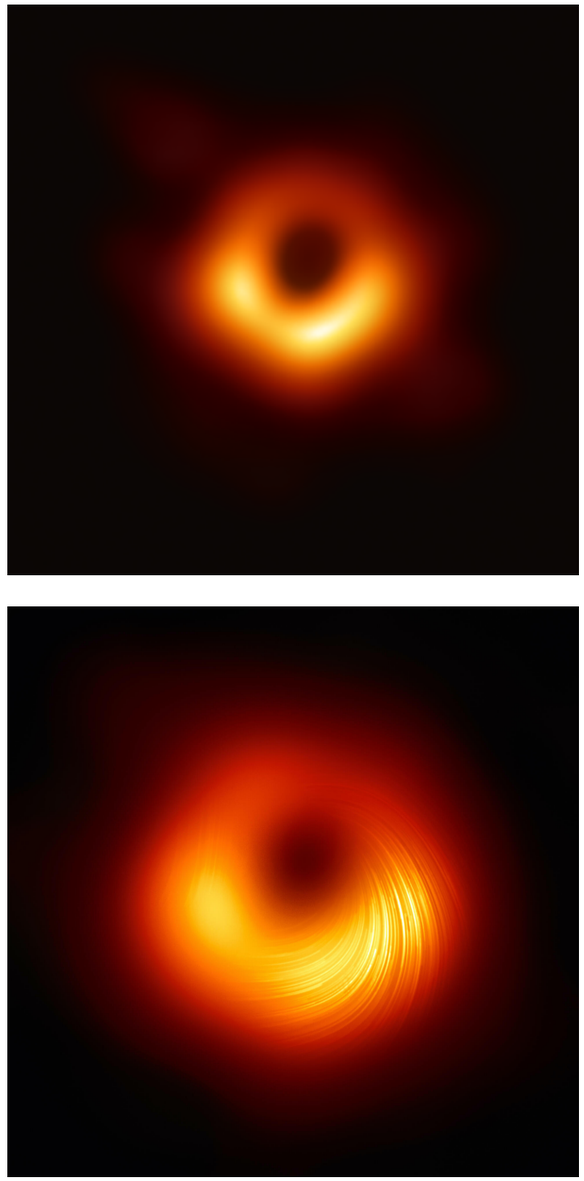


EVENT HORIZON TELESCOPE

- Event horizon telescope (EHT) used NRPT+HMC to achieve better performance within 2% of computation budget of original photo of blackhole M87 without NRPT (Tiede, 2021)

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

- 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



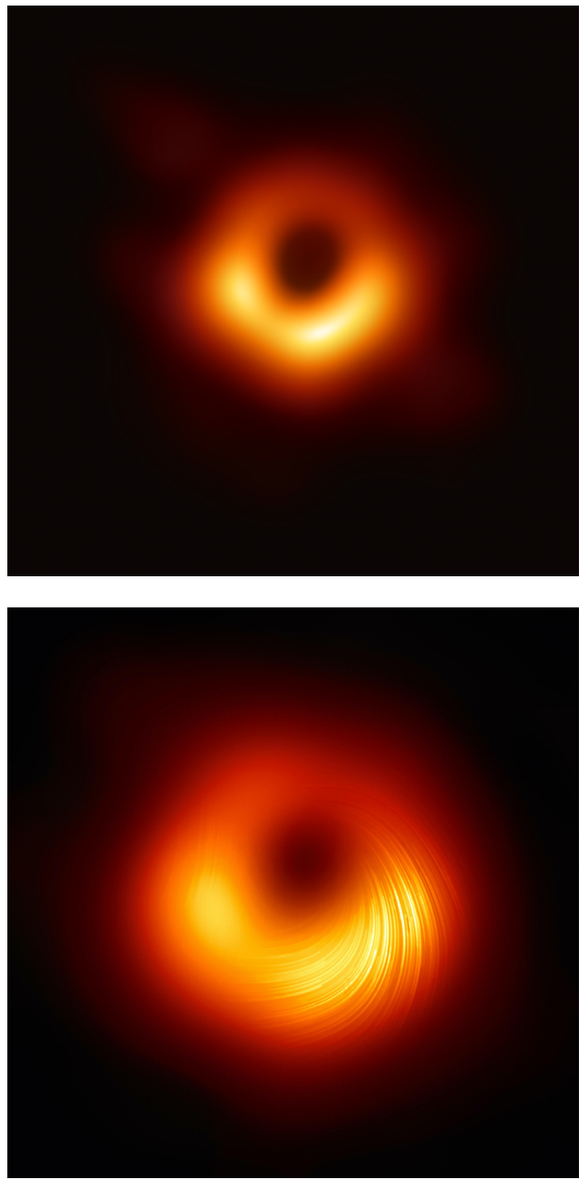
EVENT HORIZON TELESCOPE

- Event horizon telescope (EHT) used NRPT+HMC to achieve better performance within 2% of computation budget of original photo of blackhole M87 without NRPT (Tiede, 2021)

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

- 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

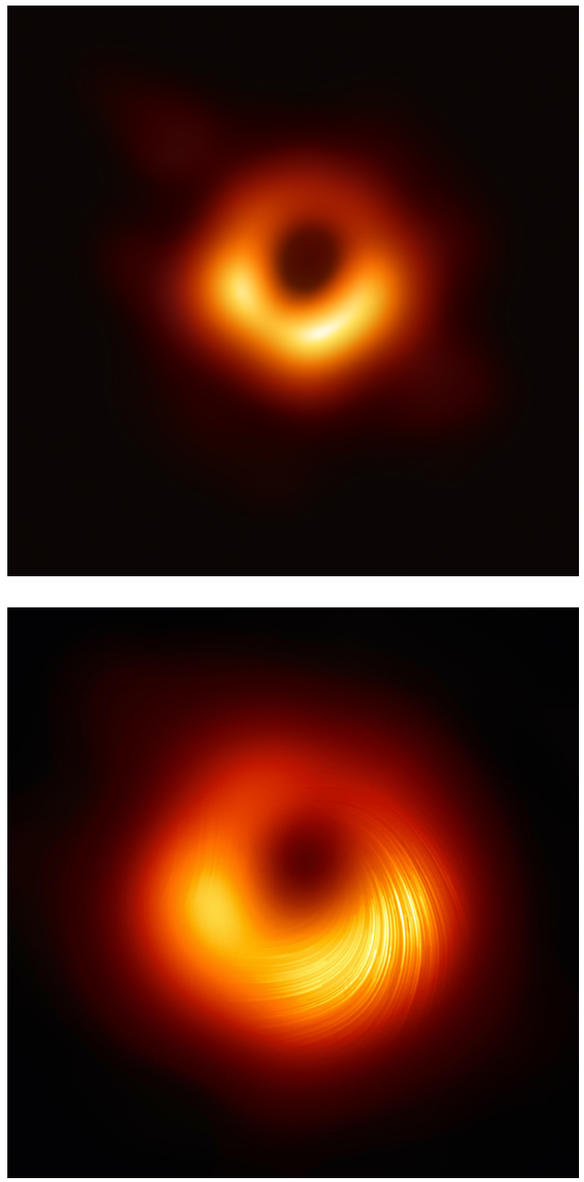
- Google research used NRPT for tackle ill-conditioned Bayesian inverse problems applied to nuclear fusion (Langmore, et al 2021)



EVENT HORIZON TELESCOPE

- Event horizon telescope (EHT) used NRPT+HMC to achieve better performance within 2% of computation budget of original photo of blackhole M87 without NRPT (Tiede, 2021)

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

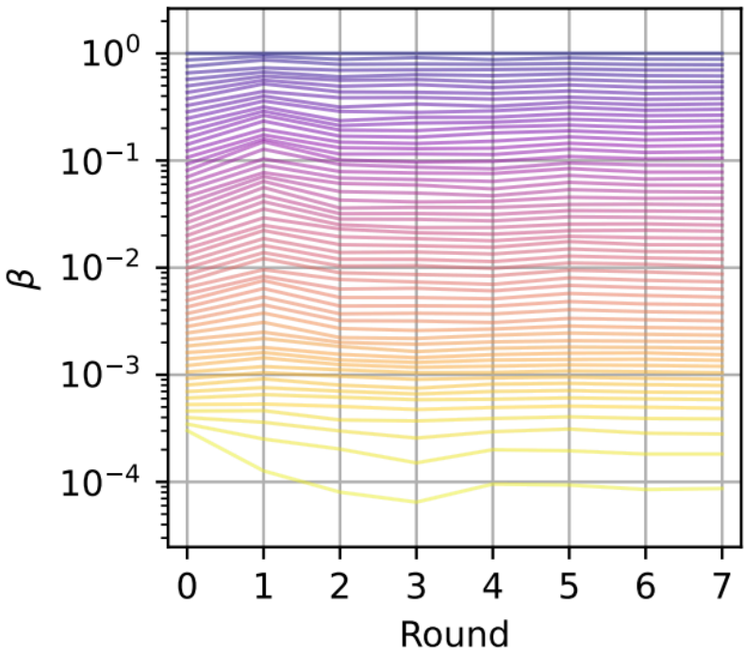


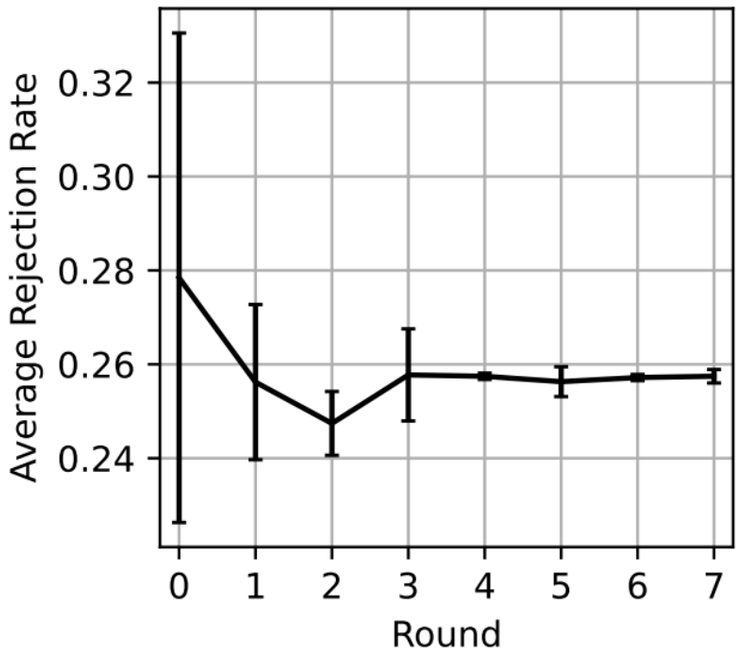
- BC Cancer Research Center and MSK used NRPT to model time-series of single-cell cancer genomes (Dorri et al 2020, Salehi et al, 2021) in Nature

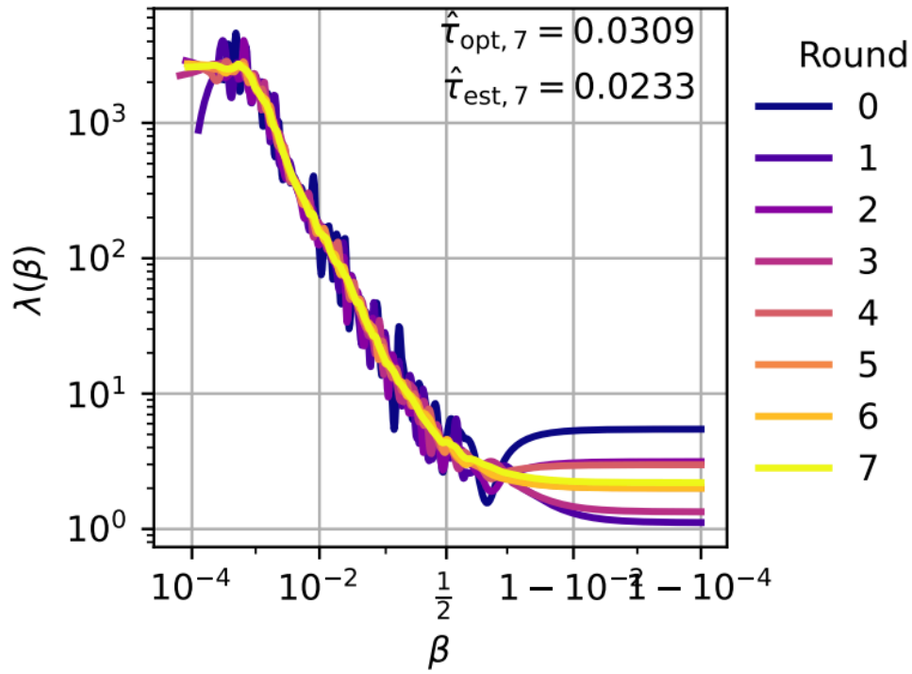
- Google research used NRPT for tackle ill-conditioned Bayesian inverse problems applied to nuclear fusion (Langmore, et al 2021)

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

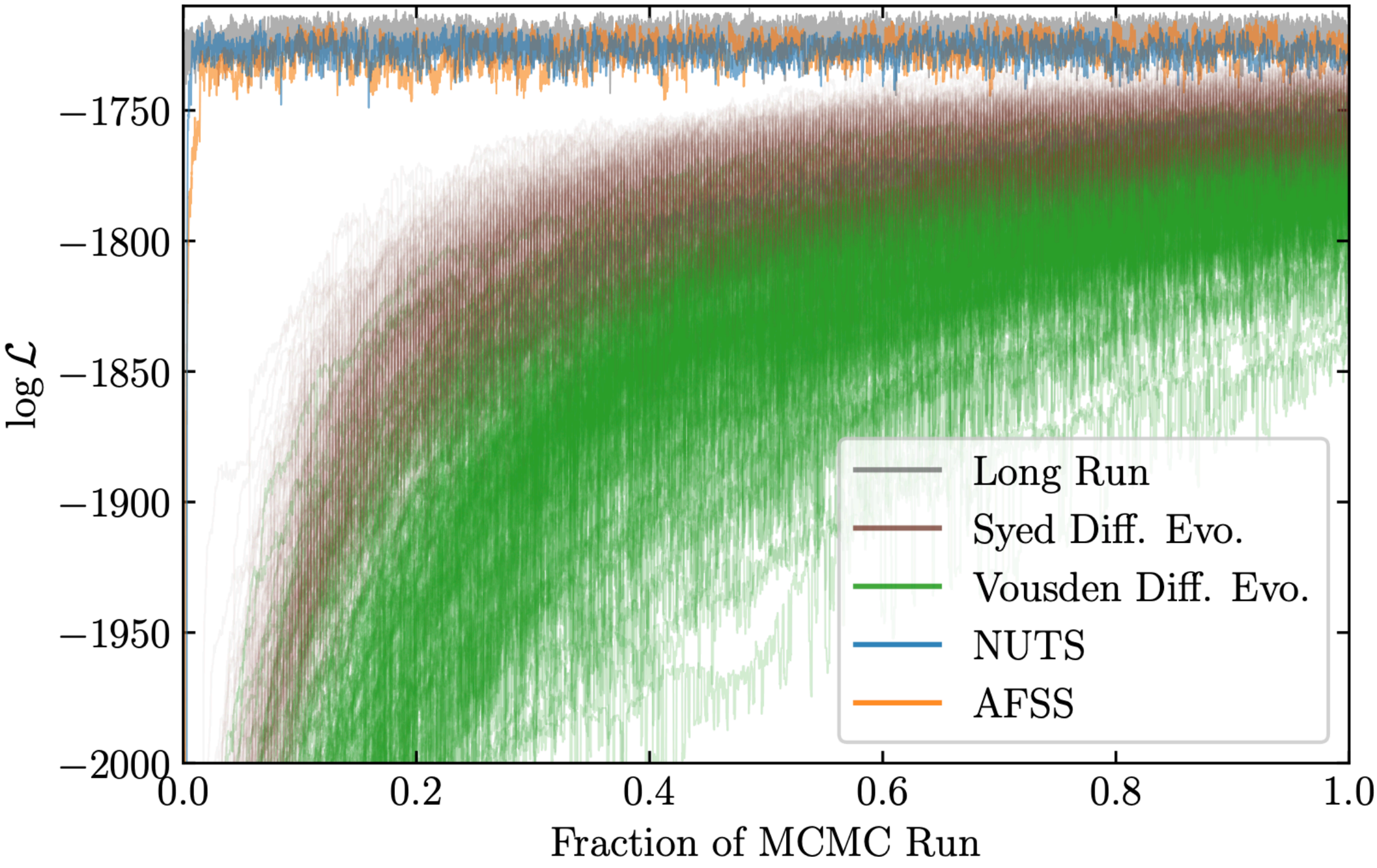
APPLICATIONS OF NRPT







BENCHMARK (TIEDE, 2021)



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