Policy Gradients for Optimal Parallel Tempering MCMC

Daniel Zhao [1] **Natesh S. Pillai** [1]

Abstract

Parallel tempering is a meta-algorithm for Markov Chain Monte Carlo (MCMC) methods which uses multiple chains to sample from tempered versions of the target distribution, improving mixing on multi-modal distributions that are difficult to explore for traditional methods. The success of this technique depends critically on the choice of chain temperatures. We introduce an adaptive temperature selection algorithm which adjusts temperatures during sampling using a policy gradient method. Experimental results using techniques from Reinforcement Learning show that it can outperform traditional geometrically-spaced temperatures and uniform acceptance rate temperature ladders in terms of integrated autocorrelation time on test distributions.

1. Introduction

The core of many Bayesian inference problems is estimating the parameters of a model whose underlying probability distribution is too complex to solve or sample from directly. MCMC methods address this challenge by generating correlated random samples to represent the distribution. However, traditional MCMC algorithms often struggle to effectively sample target distributions that are strongly multimodal or have rugged energy landscapes, as the chains may become trapped in local modes and fail to explore the entirety of the space, resulting in slow convergence.

To mitigate this problem, parallel tempering MCMC introduces auxillary tempered chains which exchange information by periodically swapping states with their neighbors. While low temperature chains are prone to becoming trapped into local energy minima, hot chains can more easily traverse entropic barriers in the flattened energy landscape to jump between modes. Having adjacent chains swap states allows the colder chain to explore new modes which would otherwise take much longer to reach. However, determining the optimal temperatures to maximize the efficiency of parallel tempering remains an open problem since the temperatures at which a chain can effectively cross entropic barriers (e.g. near a phase transition) varies widely according to the target distribution. Adaptive approaches which can automatically adjust the temperature ladder to the features of the target distribution have thus become popular in recent works (Katzgraber et al., 2006; Miasojedow et al., 2012; Vousden et al., 2015).

In this paper, we introduce a novel reinforcement learning approach to optimizing parallel tempering Markov Chain Monte Carlo (MCMC). We conceptualize the selection of the temperature ladder as a stateless policy optimization problem with an associated reward function. The term "stateless" reflects that updating the temperature ladder does not alter the chain's positions. The reward function is designed to measure the efficiency of the sampler, such that optimizing this reward enhances the mixing of the Markov chain by minimizing sample autocorrelation. While most adaptive algorithms in the literature assume that uniform acceptance rates between chains optimize sampler efficiency (Miasojedow et al., 2012; Vousden et al., 2015), we explore several alternative reward formulations in our approach.

Our contributions in this paper are:

Novel algorithm for optimizing parallel tempering MCMC. We introduce a policy gradient-based approach for the temperature ladder selection problem. Our algorithm gradually shapes the temperatures to maximize the long-term average reward. We show that, with diminishing update magnitudes, it satisfies the necessary ergodic properties to ensure convergence to the target distribution. Furthermore, our algorithm does not require separated train/sampling phases, due to its statelessness.

Distance metric for estimating the efficiency of swaps. We propose the use of a swap mean distance metric as the primary component of the reward function. Intuitively, for each swap attempt with the coldest chain, this metric measures how surprising the new swapped state is compared to its most recent states, measured by its mean distance. We show that the swap mean-distance metric is strongly correlated with ACT.

[1]Department of Statistics, Harvard University, Cambridge, USA. Correspondence to: Daniel Zhao <danielzhao@alumni.harvard.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

Experimental findings. We run experiments to demonstrate the effectiveness of our method compared to benchmarks in the literature. Our results show that the algorithm is stable and roughly replicates state-of-the-art results (Vousden et al., 2015). Additionally, when optimizing for the swap mean-distance metric, we are able to achieve lower ACT than both geometrically spaced temperatures and uniform acceptance rate chains on test distributions. This suggests that incorporating both metrics in the objective function is better than achieving uniform acceptance rates alone.

2. Related Work

Many authors have advocated choosing a temperature ladder which achieves a fixed, uniform acceptance rate across all chains (Roberts & Rosenthal, 1997; Sugita & Okamoto, 1999; Kofke, 2002; Kone & Kofke, 2005). To achieve this, the prevailing heuristic set forth by Kofke et al. (2002) is geometrically spacing the temperatures so that the ratio of the temperatures of any two adjacent chains is constant. It has been shown that geometric spacing is optimal for the Gaussian distribution but not sufficient to achieve uniform acceptance rates in general (Katzgraber et al., 2006; Vousden et al., 2015).

Adaptive Affine-Invariant Sampler

Vousden et al. (2015) proposed an adaptive algorithm which aims to achieve uniform acceptance rates across chains (Vousden et al., 2015). Their algorithm periodically updates the temperatures based on adjacent acceptance rates so that neighbors with low acceptance rates are moved closer together in temperature, while high acceptance rate neighbors are pushed apart. The maximum temperature is fixed to infinity so that the target distribution becomes flat in the hottest chain, bypassing the need to manually select a maximum temperature which could be insufficient to overcome pathological features in the distribution. Conseqeuntly, every update only modulates the temperatures of intermediate chains. The eventual acceptance rate, then, is dependent only on the chosen number of chains and the target distribution. The authors show that their method converges and empirically achieves lower ACT than the geometric temperature ladder in non-Gaussian cases. In this work, we use their implementation of parallel tempering MCMC, which uses as its base the ensemble affine invariant sampler (Goodman & Weare, 2010). See Appendix A for details.

Conceptual RL for MCMC

Optimization problems in MCMC often display properties which make a reinforcement learning approach appealing. For example, Markovian sequential decision-making mirrors the typical settings in which RL excels in, and agent-based learning methods are usually well-suited to deal with the curse of dimensionality and the exploration-exploitation trade-off inherent in optimization problems. Some authors have begun creating conceptual frameworks for using reinforcement learning to improve MCMC sampling (Bojesen, 2018; Chung et al., 2020; Wang et al., 2024). For Metropolis-Hastings, the state of the MDP is identified with the state of the Markov chain, while the action is identified with the proposal mechanism. This naturally leads to a policy learning scenario where the objective is to learn a stochastic policy that represents the optimal proposal distribution for maximizing sampler efficiency. Empirically, a DDPG implementation of a policy gradient adaptive Metropolis-Hastings algorithm was shown to outperform existing benchmarks (Wang et al., 2024), demonstrating the potential of RL for optimizing MCMC.

3. Problem Setup

3.1. Background

Let $\mathcal{X}$ be the phase space and f the target distribution. In parallel tempering MCMC, M replica chains $\{(X_t^{(1)})_{t\in\mathbb{N}}, \dots (X_t^{(M)})_{t\in\mathbb{N}}\}$ run simultaneously at different temperatures $T = \{T_1, ..., T_N\}$ where $T_1 < T_2 < ... < T_M$ and $T_1 = 1$. The inverse of the temperatures are the betas: $\beta_i = 1/T_i$, and the stationary distribution f^{β_i} of $(X^{(i)})$ is defined as

$$f^{\beta_i}(x^{(i)}) := (f(x^{(i)}))^{\beta_i} \quad \forall x^{(i)} \in \mathcal{X}.$$

The joint chain $(X_t^{(1)} \dots X_t^{(M)})_{t\in\mathbb{N}}$ is also a Markov chain, whose canonical distribution is

$$\tilde{f}(x^{(1)}, ..., x^{(M)}) = f^{\beta_1}(x^{(1)}) \dots f^{\beta_M}(x^{(M)}),$$

from which the coldest chain marginally has stationary distribution f. While the chains run independently, they will periodically attempt a Metropolis swap move where two adjacent chains $X_t^{(i)}$ and $X_t^{(i+1)}$ propose to switch states $(x, y) \mapsto (y, x)$. This move is accepted with probability

$$A(x, y) = \min\left(1, \left(\frac{\pi(y)}{\pi(x)}\right)^{\beta_i} \left(\frac{\pi(x)}{\pi(y)}\right)^{\beta_{i+1}}\right) \quad (1)$$

to preserve the detailed balance of the joint chain, where π is the unnormalized density.

Denote the acceptance rate A_i of a chain as the proportion of attempted swaps between itself and any adjacent chains which are successful.

3.2. Maximizing Average Reward

Suppose that each temperature ladder T is associated with a reward distribution R_T which represents the average reward obtained after running all chains some fixed N steps, assuming stationarity. We consider the problem of selecting

$$T^* = \text{argmax}_T \; E[R_T].$$

In this setup, the action is identified with the temperature ladder T, and the state is null. The lack of a state presents a unique challenge compared to previous works in RL for adaptive Metropolis-Hastings.

This problem is also known as the infinite-horizon Lipschitz bandits problem, for which the finite-time case has been studied (Agrawal, 1995; Magureanu et al., 2014). However, our goal is not to maximize the cumulative reward in a fixed number of samples, but to achieve convergence towards the optimal temperature configuration with the highest mean reward over time. This motivates gradient-based methods, which are guaranteed to converge to local minima. They are especially powerful since certain reward functions may further be convex. For example, the variance of all acceptance rates is convex, since the acceptance rate between two chains increases monotonically with the difference in their temperatures.

3.3. Policy Gradient Methods

Policy gradient methods are popular for problems where the action space is continuous and high-dimensional—such as temperature selection—because parameterizing the policy allows us to output a continuous action directly, rather than relying on action-value estimation as in value-based methods. In addition, far fewer parameters are needed to be learned for the policy, compared to the value function, which is often complex to approximate.

Given an MDP (S, A, P_a, R_a) with states S, actions A, transition kernel P_a, and reward function R_a associated with action a, policy gradient methods attempt to learn a parameterized policy mapping $\pi_\theta : S \times A \to [0, 1]$ which maximizes the cumulative reward G_L where $t = 1, \cdots, L$.

In the infinite time horizon context, we want π_θ which maximizes the expected reward at stationarity: define the objective function as $J(\theta) := E_{\pi_\theta}[R_a]$. Notice that in the context of parallel tempering MCMC, the Markov chain we refer to in this section corresponds to the sequence of temperatures $(T_i)_{i \le L}$, and not the Markov chain of the sampler.

4. Policy Gradients for Temperature Selection

The temperature selection problem can be modeled as a single-state RL problem in which the temperatures a_t are sampled from the stochastic policy π_θ, parameterized by some θ. If the density π_θ is centered around θ, we have a natural interpretation of θ as the best estimate of the optimal action.

The action space is taken to be the space of log temperature differences. That is, for $D_i = \log \beta_i - \log \beta_{i+1}$, the space of configurations $A = (D_1, \ldots, D_{M-1})$ is $[0, \infty)^{M-1}$, which can further be made compact (and thus the reward function is Lipschitz continuous) by choosing reasonable bounds $A = [D_{\text{min}}, D_{\text{max}}]^{M-1}$ for constants $0 < D_{\text{min}} < D_{\text{max}}$. Considering the log-differences between temperatures ensures that the scale of all parameters are similar, which contributes to learning stability, and allows us to easily fix $\beta_1 = 1$ and $\beta_M = 0$. In our experiments, we take $D_{\text{min}} = 0.01$, $D_{\text{max}} = 10$, and $N = 500$.

Note that in the single-state case, the gradient of the objective function $J(\theta)$ reduces to

$$\begin{aligned}\nabla_\theta J(\theta) &= \int_A \nabla_\theta \pi_\theta(a) E_\pi[R_t | A_t = a]\, da \\ &= \int_A \pi_\theta(a) \nabla_\theta \log \pi_\theta(a) E_\pi[R_t | A_t = a]\, da.\end{aligned}$$

via the Policy Gradient Theorem (See Appendix C). Thus, we may obtain an unbiased Monte Carlo estimate of $\nabla_\theta J(\theta)$ by calculating $\nabla_\theta \log \pi_\theta(a)\, E_\pi[R_t | A_t = a]$ for every action of the current policy π_θ. The update is performed as the last step:

$$\theta_{t+1} = \theta_t + \alpha \nabla_\theta J(\theta)|_{\theta=\theta_n}$$

for learning rate α.

In Algorithm 1, we describe our method generally, so that it remains applicable to other optimization problems in MCMC which may have similar setups.

Algorithm 1 Single-State Policy Gradient for Maximizing Average Reward

input Initialize policy parameters θ_0, initial positions of walkers $\{\mathbf{x}_0^{(i)}\}_{i=1}^M$, policy distribution $\pi_\theta(\cdot)$

1: **for** $t = 1$ to L **do**
2: Generate a set of sampler parameters a_t by drawing from policy distribution π_θ
3: Run sampler with parameters a_t, observe reward r_t
4: Normalize r_t and use it to estimate the average reward $\bar{R}$ based on observed rewards
5: Calculate the gradient of the policy $\nabla_\theta \log \pi_\theta(a_t)$ w.r.t. θ and clip it
6: Update the policy parameters: $\theta \leftarrow \theta + \alpha(r_t - \bar{R}) \nabla_\theta \log \pi_\theta(a_t)$
7: **end for**

output Samples $\{\mathbf{x}_j^{(1)}\}_{j=1}^L$, Policy parameters θ_L

In our experiments, we take the policy function to be normal centered at θ: $\pi_\theta(\cdot) \sim \mathcal{N}(\theta, \sigma I)$ so that the gradient is simply

$$\nabla_\theta \log \pi_\theta(a_t) = -\sigma^{-1}(a_t - \theta).$$

The average reward $\bar{R}$ is estimated as the average of the last 500 rewards. Initial θ_0 values are spaced such that $\Delta\theta_i = 1$.

4.1. Convergence Analysis

In general, if the temperature ladder is changed during sampling, there is no guarantee that the coldest chain converges to the target distribution. However, it has been shown (Roberts & Rosenthal, 2007; Saksman & Vihola, 2010) that the convergence of an adaptive MCMC method is preserved if the following conditions hold.

Theorem 4.1. *Let $\{X_t\}_{t\in\mathbb{N}}$ be Markov chain and $P_{\Gamma_t}(x,\cdot)$ the tth adapted conditional density for X_{t+1} given $X_t = x$. Then the Strong Law of Large Numbers holds for $\{X_t\}$ if two conditions are met:*

(a) (Diminishing Adaptation) For every starting $x \in \mathcal{X}$,

$$\lim_{t\to\infty} \sup_{x\in\mathcal{X}} \|P_{\Gamma_{t+1}}(x,\cdot) - P_{\Gamma_t}(x,\cdot)\| = 0.$$

(b) (Containment) For any $\epsilon > 0$, the sequence

$$\{M_\epsilon(X_t, \Gamma_t)\}_{t\in\mathbb{N}}$$

is bounded in probability, where

$$M_\epsilon(x, \Gamma_t) := \inf\{n \geq 1 : \|P^n_{\Gamma_t}(x,\cdot) - \pi(\cdot)\| \leq \epsilon\}.$$

The Diminishing Adaptation condition requires that changes to the transition kernel decay to zero, and Containment requires the convergence time of the chain to be bounded in probability—note that this is trivially satisfied with a compact phase space. Subsequent works extended Theorem 4.1 for weaker forms of the Containment condition (Bai et al., 2011; Rosenthal & Yang, 2018).

Diminishing Adaptation. To modify our algorithm to satisfy the Diminishing Adaptation condition, we artificially dampen the variance of the sampling from the policy $\pi_{\theta_t}(\cdot) \sim \mathcal{N}(\theta_t, \epsilon_t \sigma I)$ by decaying $\epsilon_t \to 0$. The gradient $\nabla_\theta \log \pi_\theta(T) = -\sigma^{-1}(T - \theta)$ is not scaled with ϵ_t, so it also diminishes to 0 as $t \to \infty$.

This ensures that adaptations vanish over sufficiently long time scales, and all temperatures approach a fixed value. In fact, exponential decay of ϵ_t also ensures that the policy π_{θ_t} converges fairly quickly to θ_t, so the optimal temperature ladder T^* is solved for by estimating θ^*. Not scaling the gradient does bias the estimates of $\nabla_\theta J(\theta)$, but our experiments show that it does not significantly affect the convergence of the algorithm.

Containment. We are only concerned with the convergence of the coldest chain to the target distribution, so a sufficient condition for Containment is that the convergence time of the coldest chain is bounded in probability. Since its temperature is invariant, convergence fails only if the independent chain fails to converge. This is a very weak condition for most practical problems. Ergodicity of the coldest chain is typically a concern for the design of underlying sampler rather than the adaptive scheme.

5. Swap Mean Distance

With guarantees on our algorithm's convergence, the next step is choosing a reward function which approximates sampler efficiency. Efficiency is generally measured in terms of the autocorrelation between successive samples—highly autocorrelated Markov chains are slow to mix. Frequently swapping states with hotter chains in parallel tempering helps reduce autocorrelation. However, developing a feedback mechanism for the swap mechanism that effectively captures this goal remains an open challenge. In this section, we review some potential reward functions and their implications for mixing. We propose a new metric based on the distance between the states of a swap.

Integrated Autocorrelation Time

The most direct metric for measuring sampler efficiency is the integrated autocorrelation time (ACT). For any function h suppose $\langle h^N \rangle$ is a Monte Carlo estimator for $E[h(X)]$. Its variance when the samples are correlated converges:

$$Var[\langle h^N \rangle] \longrightarrow \frac{\tau_h}{N} Var[h^\pi]$$

where τ_h is the integrated autocorrelation time of the Markov chain $(X_t)_{t\in\mathbb{N}}$, given by

$$\tau_h = \sum_{i=-\infty}^{\infty} \rho_i. \tag{2}$$

and $\rho_i = Cov[h(X_1), h(X_{1+i})]$ is the autocorrelation of the ith lag.

As N is quite large in most MCMC applications, the asymptotic variance is usually approximated quite well by the ACT. Optimizing the efficiency of the sampler is then achieved by minimizing the integrated autocorrelation time.

In practice, estimating the autocorrelation time of a sampler is inconsistent with limited samples, so it is not necessarily a good feedback mechanism to use for distributions whose densities are not known a-priori.

Uniform Acceptance Rate

Achieving uniform acceptance rates across chains is a rather intuitive objective, as consistent swaps between chains allows information to be exchanged effectively down the entire temperature ladder. Empirical implementations has been shown to be more efficient than geometrically spaced temperatures (Vousden et al., 2015; Sugita & Okamoto, 1999). Various reward functions can be employed to optimize for uniform acceptance rates. For example, in our experiments, we find that maximizing the negative standard deviation of the acceptance rates achieves this goal.

Expected Squared Jumping Distance

Another metric for assessing sampler efficiency proposed by (Atchadé et al., 2011) is the expected squared jumping

distance (ESJD). Suppose we have some fixed chain at temperature β, and we propose a swap with some other chain at temperature $\beta + \epsilon$. Let $\gamma = \beta + \epsilon$ if the swap is accepted, or $\gamma = \beta$ if the swap is rejected. Then, define

$$\begin{aligned} ESJD_\beta &:= E_\pi[(\gamma - \beta)^2] \\ &= \epsilon^2 E_\pi[A]. \end{aligned}$$

The acceptance rate serves as a swap-dependent metric, primarily assessing the efficiency of the swap mechanism without addressing the impact of each swap in enhancing mixing. In contrast, information-theoretic metrics, such as the Kullback-Leibler (KL) divergence between adjacent chains, are swap-independent. The greater the divergence, the more information the colder chain is expected to gain per swap (Vousden et al., 2015). However, estimating KL-divergence becomes increasingly computationally expensive as $t \to \infty$. The Expected Squared Jump Distance (ESJD) scales the expected acceptance rate by the squared difference in temperatures, ϵ^2. This approach aims to balance swap efficiency with swap effectiveness, as a large ϵ indicates significant differences between neighboring chains, thus assigning greater weight to successful swaps.

Swap Mean Distance

We propose a metric which is sensitive to both the underlying dynamics of tempered chains and the efficiency of the swap mechanism. For the first, a tempered chain which is sufficiently different from the colder chain should tend to propose states which are far from previously well-explored regions. Thus, an accepted swap to a distant state should be weighted more heavily than to one that is close. Secondly, a rejected proposal should be measured proportionate to its actual impact. Accepting a swap right next to the current state is not much better than rejecting it. This induces a rather natural continuous metric based on distance.

The m-mean swap distance ω_m (referred to as the swap mean-distance) is defined as

$$\omega_m^{(i)}(t) := \frac{1}{m} \sum_{j=1}^{m} (d(x_t^{(i+1)}, x_{t-j}^{(i)}))$$

where $d(x, y)$ denotes the Euclidean distance between points $x, y \in \mathbb{R}^n$.

The finite $m > 1$ term represents the "memory" of the chain which persists through swaps. Consider a distribution with three separated modes, for which two chains are initialized at separate modes. If neither have the sufficient energy to hop to the third mode, consistent swapping would still only explore two of the three modes. However, neither the acceptance rate nor the 1-swap mean-distance metrics would capture this behavior, since immediate swap distances are high and likely. Choosing some m greater than the swap period of $X_t^{(i)}$ and $X_t^{(i+1)}$ ensures that at least some of the states prior to a successful swap are remembered, so that $\omega_m^{(i)}(t)$ is controlled in such scenarios.

We tested several choices of m, and found that larger m is typically more strongly correlated with the ACT. This can be explained by the fact that an extended memory captures a larger subset of previously explored states. However, too large an m devalues the influence of the most immediate states. Additionally, it is essential to maintain $m << N$ to control the variance of $\overline{\omega_m^{(i)}}$ estimates. We use $m = 50$ in our experiments. Some alternative formulations of the swap mean-distance are discussed in Appendix **??**.

Empirically, we find strong correlations between the swap mean-distance and ACT. Figure 5 presents scatterplots for three distinct distributions, namely the multimodal Gaussian, egg-box, and Rosenbrock distributions. The Spearman correlation coefficients for these distributions are found to be are -0.997, -0.998, -0.856, respectively, with p-values less than $1e - 100$. These findings strongly indicate that swap mean-distance is an effective proxy for ACT. Consequently, the application of swap mean-distance as a reward function is substantiated, with the assumption that ACT generally exhibits non-decreasing behavior in relation to it.

6. Results

To evaluate the performance of our algorithm, we ran it on three distributions characterized by pronounced multimodality or rugged features: a mixture of ten 8-dimensional Gaussians, a 5-dimensional egg-box distribution with 243 modes, and the Rosenbrock distribution. Detailed specifications can be found in Appendix B. For each distribution, the algorithm was run on 15 temperatures for 4000 iterations. The reward functions tested were the swap mean-distance, expected squared jumping distance, and the negative standard deviation of acceptance rates. Each combination of distribution and reward function was tested over 10 independent trials. We derived an average ACT estimate for each trial by sampling $N = 10000$ iterations from the final temperature configuration after the algorithm terminated. These values are reported in Table 6.1.

6.1. ACT Comparison

Notably, our policy gradient method consistently outperformed geometric spacing across all reward functions, with the swap mean-distance yielding the lowest ACT for every test distribution. This is consistent with our earlier observation that the swap mean-distance is highly correlated with ACT.

In Figure 1, we see that for the egg-box distribution, both the swap mean-distance and acceptance rate standard deviation

Table 1. ACT estimates for policy gradient algorithm and associated reward functions, and geometric spacing.

DISTRIBUTION	SWAP MEAN-DISTANCE	ACCEPTANCE RATE STD	ESJD	GEOMETRIC
MULTIMODAL GAUSSIAN	**1.138** $\pm$ **0.06**	1.222 $\pm$ 0.03	2.717 $\pm$ 0.09	4.670 $\pm$ 0.05
EGG-BOX	**1.097** $\pm$ **0.08**	2.772 $\pm$ 0.22	4.817 $\pm$ 0.46	9.670 $\pm$ 0.09
ROSENBROCK	**24.08** $\pm$ **4.32**	43.55 $\pm$ 3.66	41.21 $\pm$ 8.54	81.32 $\pm$ 8.21

achieves near-optimal performance after only 800 iterations. Comparatively, ESJD takes longer to converge and exhibits greater variance across its trials throughout all stages of sampling.

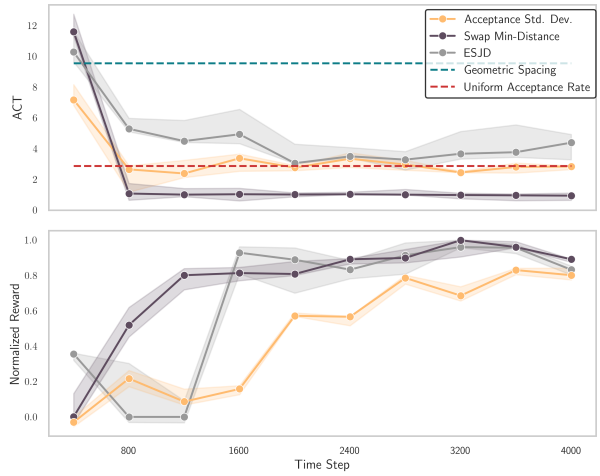

Figure 1. Plot of the ACT over time for geometrically spaced temperatures, uniform acceptance rate (Vousden et al., 2015), and policy gradient algorithm with different reward functions. Target is the egg-box distribution. Each step represents 400 iterations of the policy gradient update.

6.2. Improving on Uniform Acceptance Rates

The findings from Section 6.1 suggest a need for a critical review of the longstanding assumption regarding the optimality of uniform acceptance rates. While analogies to physical systems provide and intuitive basis to this assumption (Sugita & Okamoto, 1999), from a statistical perspective, it lacks a rigorous foundation to justify its presumed optimality across general distribution. The evidence suggests that more nuanced reward shaping strategies may offer superior performance.

As a baseline, we demonstrate that uniform acceptance rates is achievable via policy gradients by setting our reward function to the scaled negative standard deviation of the acceptance rates. As illustrated in Fig. 2, the acceptance rates of all chains converge uniformly.

We emphasize that improvement over the uniform acceptance rate paradigm was achieved simply by modifying the reward function of the policy gradient algorithm. This method is applicable to nearly any desired temperature configuration for which a respective reward can be shaped, opening doors for understanding of the efficiency of its swapping mechanism.

Note, however, that because the swap mean-distance metric is based only on the coldest chain, it is not particularly sensitive to the dynamics of hot chains. For example, figure C.3 demonstrates that across all the trials for the egg-box distribution first four temperature When M is large, it is possible that hotter chains are slow to converge to optimal temperatures. It may be advantageous to use a weighted combination of the swap mean-distance metric and a function of acceptance rates (e.g. minimum, harmonic mean, standard deviation, etc.) as the reward function, or imposing penalties on bottleneck pairs of adjacent chains whose acceptance rate is close to 0.

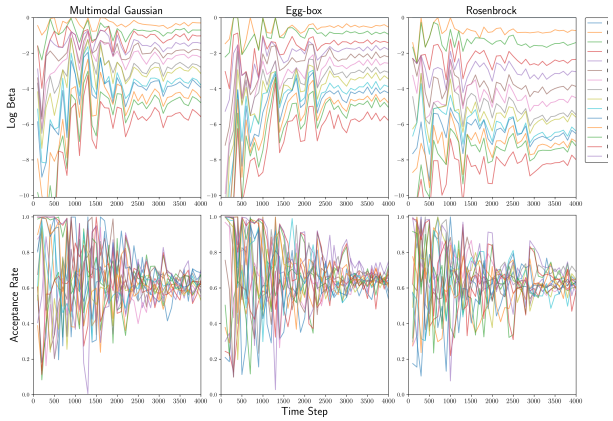

Figure 2. Evolution of log β and acceptance rates over 4000 update steps for a . Data is thinned by a factor of 100.

Conclusion

In this paper, we propose a policy gradient-based optimization method for the temperature ladder of parallel tempering MCMC. The algorithm is validated with theoretical results on convergence and experiments to illustrate its effectiveness. To design an objective function which best captures the dynamics of the swap mechanic, we introduce a new swap distance metric and show that it is strongly associated

with the sampler ACT. Empirical experiments show that the swap mean-distance metric can be used to achieve lower sampler ACT than uniform acceptance rate temperature ladders.

Acknowledgements

We thank Prof. Eric Laber and Dr. Zachary Moore for their generous input into this project and help with ideas from Reinforcement Learning. Some of this work was inspired by similar work with the above two researchers on other MCMC algorithms.

References

Agrawal, R. The continuum-armed bandit problem. *SIAM Journal on Control and Optimization*, 33:1926–1951, 1995. ISSN 0363-0129,1095-7138. doi: 10.1137/s0363012992237273. URL http://doi.org/10.1137/s0363012992237273.

Atchadé, Y., Roberts, G., and Rosenthal, J. Towards optimal scaling of metropolis-coupled markov chain monte carlo. *Statistics and Computing*, 21:555–568, 10 2011. doi: 10.1007/s11222-010-9192-1.

Bai, Y., Roberts, G., and Rosenthal, J. On the containment condition for adaptive markov chain monte carlo algorithms. *Adv. Appl. Stat.*, 21, 01 2011.

Bojesen, T. A. Policy-guided monte carlo: Reinforcement-learning markov chain dynamics. *Physical Review E*, 98 (6), December 2018. ISSN 2470-0053. doi: 10.1103/physreve.98.063303. URL http://dx.doi.org/10.1103/PhysRevE.98.063303.

Chung, E. T., Efendiev, Y. R., Leung, W. T., Pun, S.-M., and Zhang, Z. Multi-agent reinforcement learning accelerated mcmc on multiscale inversion problem. *ArXiv*, abs/2011.08954, 2020. URL https://api.semanticscholar.org/CorpusID:227012696.

Goodman, J. and Weare, J. Ensemble samplers with affine invariance. *Communications in Applied Mathematics and Computational Science*, 5, 01 2010. doi: 10.2140/camcos.2010.5.65.

Katzgraber, H. G., Trebst, S., Huse, D. A., and Troyer, M. Feedback-optimized parallel tempering monte carlo. *Journal of Statistical Mechanics: Theory and Experiment*, 2006(03):P03018–P03018, March 2006. ISSN 1742-5468. doi: 10.1088/1742-5468/2006/03/p03018. URL http://dx.doi.org/10.1088/1742-5468/2006/03/P03018.

Kofke, D. On the acceptance probability of replica-exchange monte carlo trials. *jcph*, 117:6911–, 10 2002. doi: 10.1063/1.1507776.

Kone, A. and Kofke, D. Selection of temperature intervals for parallel-tempering simulations. *The Journal of chemical physics*, 122:206101, 06 2005. doi: 10.1063/1.1917749.

Magureanu, S., Combes, R., and Proutiere, A. Lipschitz bandits: Regret lower bounds and optimal algorithms, 2014.

Miasojedow, B., Moulines, E., and Vihola, M. Adaptive parallel tempering algorithm, 2012.

Roberts, G. and Rosenthal, J. Markov-chain monte carlo: Some practical implications of theoretical results. *Canadian Journal of Statistics*, 26:5 – 20, 12 1997. doi: 10.2307/3315667.

Roberts, G. O. and Rosenthal, J. S. Coupling and ergodicity of adaptive markov chain monte carlo algorithms. *Journal of Applied Probability*, 44:458 – 475, 2007. URL https://api.semanticscholar.org/CorpusID:21227921.

Rosenthal, J. S. and Yang, J. Ergodicity of combocontinuous adaptive mcmc algorithms. *Method. Comput. Appl. Prob.*, 20(2):535–551, jun 2018. ISSN 1387-5841. doi: 10.1007/s11009-017-9574-3. URL https://doi.org/10.1007/s11009-017-9574-3.

Saksman, E. and Vihola, M. On the ergodicity of the adaptive metropolis algorithm on unbounded domains. *The Annals of Applied Probability*, 20(6):2178–2203, 2010. ISSN 10505164. URL http://www.jstor.org/stable/20799809.

Sugita, Y. and Okamoto, Y. Replica-exchange molecular dynamics method for protein folding. *Chemical Physics Letters*, 314(1):141 – 151, 1999. ISSN 0009-2614. doi: http://dx.doi.org/10.1016/S0009-2614(99)01123-9. URL http://www.sciencedirect.com/science/article/pii/S0009261499011239.

Vousden, W. D., Farr, W. M., and Mandel, I. Dynamic temperature selection for parallel tempering in markov chain monte carlo simulations. *Monthly Notices of the Royal Astronomical Society*, 455(2):1919–1937, November 2015. ISSN 1365-2966. doi: 10.1093/mnras/stv2422. URL http://dx.doi.org/10.1093/mnras/stv2422.

Wang, C., Chen, W., Kanagawa, H., and Oates, C. J. Reinforcement learning for adaptive mcmc, 2024.

A. Adaptive Affine Algorithm Details.

In Sections 2 and 6.1, we reference an adaptive parallel tempering algorithm (Vousden et al., 2015) that results in uniform acceptance rates, and the ensemble-based affine invariant sampler (Goodman & Weare, 2010) on which it is built. We provide details on these algorithms.

A.1. Affine Invariant Ensemble Sampler.

The core idea behind the affine invariant sampler is that it can use context information from other chains in an ensemble of chains to perform well under affine transformations of the parameter space. An affine transformation involves linear transformations such as scaling, translation, rotation, and shearing. Metropolis-Hastings proposal distributions have trouble navigating these features, but the ensemble sampler makes use of the "stretch move" to overcome this issue.

Algorithm 2 Affine Invariant Ensemble Sampler

input Initial positions of walkers $\{\mathbf{x}_i^{(0)}\}_{i=1}^N$, target distribution $\pi(\mathbf{x})$, stretch move parameter a
Initialize the ensemble of walkers $\{\mathbf{x}_i^{(0)}\}_{i=1}^N$
for $t = 0$ to $T-1$ **do**
  for each walker i **do**
    Randomly select another walker $j \neq i$ from the ensemble
    Draw z from the distribution $g(z) \propto \frac{1}{\sqrt{z}}$ defined over $\left[\frac{1}{a}, a\right]$
    Propose a new position $\mathbf{x}_i' = \mathbf{x}_j + z(\mathbf{x}_i^{(t)} - \mathbf{x}_j)$
    Compute acceptance probability $\alpha = \min\left(1, z^{d-1}\frac{\pi(\mathbf{x}_i')}{\pi(\mathbf{x}_i^{(t)})}\right)$
    Draw a uniform random number $u \sim \mathcal{U}(0,1)$
    if $u < \alpha$ **then**
      Accept the proposal: $\mathbf{x}_i^{(t+1)} = \mathbf{x}_i'$
    else
      Reject the proposal: $\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)}$
    end if
  end for
end for
output $\{\mathbf{x}_i^{(t)}\}_{i=1}^N$

In context of parallel tempering, this method is adapted so that an ensemble sampler is constructed at each temperature level. Each ensemble only considers the samplers in its own temperature level when performing the stretch move. Independently, after each stretch move, a random permutation of swaps are proposed between every sampler of ensembles in adjacent temperature levels. In our experiments, each ensemble is composed of 16 samplers.

A.2. Adaptive Parallel Tempering Ensemble.

The adaptive parallel tempering algorithm constructs an ensemble sampler for each temperature in the ladder and updates the log differences $S_i := \log(T_{i+1} - T_i)$ between temperatures based on neighboring acceptance rates:

$$S_i(t+1) = S_i(t) + \kappa(t)[A_i(t) - A_{i+1}(t)].$$

The function $\kappa(t)$ is a hyperbolic decay function which dampens the amplitude of adaptations over time to ensure convergence. This updating mechanism is quite intuitive—each S_i is adjusted to close the gap between two chains with relatively low acceptance rates, or repel if they have high acceptance rates.

B. Toy Distributions.

We provide specifics on the toy distributions we used to test our algorithm. The following three distributions all exhibit multimodality to varying degrees. The Rosenbrock function additionally possesses a narrow, curved valley between modes, testing the algorithm's capability to navigate extreme topographies.

Multimodal Gaussian Distribution. Given parameters $\{w_i, \mu_i, \Sigma_i\}_{1 \leq i \leq n}$, the Multimodal Gaussian is a mixture of Gaussian distributions, which has likelihood

$$L(\theta) \propto \sum_{i=1}^{n} w_i \cdot p(\theta | \mu_i, \Sigma_i)$$

where

$$p(\mathbf{x}|\mu, \mathbf{\Sigma}) = \frac{1}{(2\pi)^{k/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^\top \mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)\right).$$

In our experiments, $n = 10$ and each μ_i is chosen uniformly at random in the interval $[-1, 1]$. Furthermore, $\Sigma_i = \sigma_i I$ where $\sigma_i \sim [0.01, 0.3]$ uniformly. The parameter space is restricted to $[-2, 2]^8$.

Egg-box Distribution. The egg-box distribution is a popular test distribution in machine learning and optimization because it presents several isolated modes. Its density is

$$L(\theta) \propto \left(\frac{1}{2}\prod_{i=1}^{d} \cos\theta_i + \frac{1}{2}\right)^{\beta}.$$

Under large β, these modes look locally Gaussian and do not overlap unless the distribution is sufficiently tempered. We take $\beta = 1000$ in our experiments and restrict the parameter space to $[-3\pi/2, 3\pi/2]^5$.

Rosenbrock Distribution. The Rosenbrock distribution contains two modes with extremely narrow, curved valleys which are difficult to traverse. Its density is given by

$$L(x, y) \propto \left(\frac{1}{c + f(x, y)} + \frac{1}{c + f(-x, y)}\right)^{\beta}$$

where

$$f(x, y) = (a - x^2)^2 + b(y - x^2)^2.$$

In our experiments, we take $a = 4, b = 1, c = 0.1, \beta = 1000$.

C. Experiments Extended.

C.1. Markov Chain Convergence in Log-Likelihood.

A possibility when sampling from complex distributions (ie. Rosenbrock) is that the Markov chain does not converge to the target within the given finite number of time steps. Stationarity is required to make estimates of the ACT. To check that our algorithm converges, we plot the negative log likelihood of the samples over the 4000 iterations.

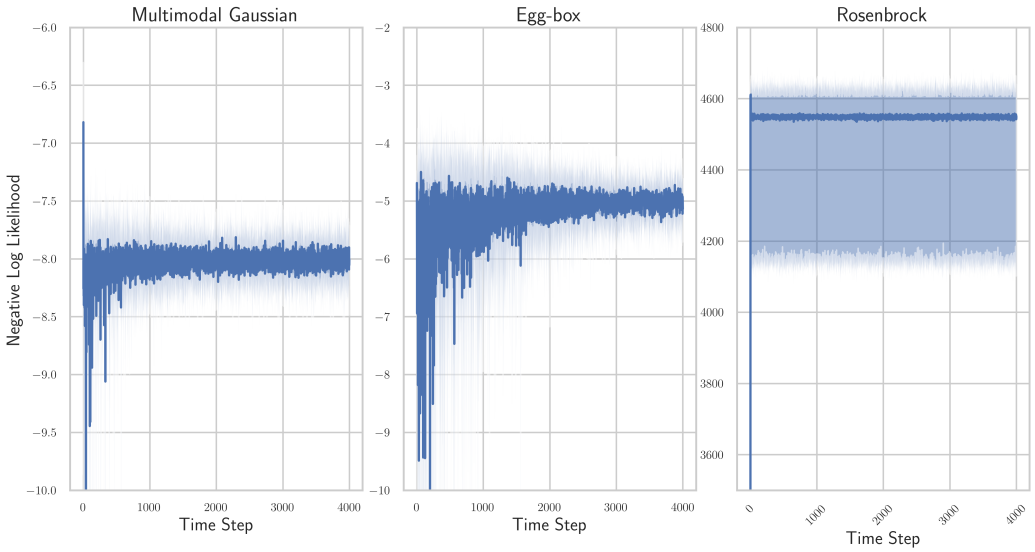

Figure 3. Plotted negative log likelihoods of the samples each time step of the algorithm. The shaded region indicates a 95% confidence interval over 10 trials.

C.2. Swap Mean Distance Correlation with ACT.

We plot the scatterplot between swap mean-distance and estimated ACT for three distributions in Figure 5. Data is generated by randomly selecting 1000 temperature ladders for each m value and estimating the average swap mean-distance and ACT with $N = 1000$ MCMC iterations.

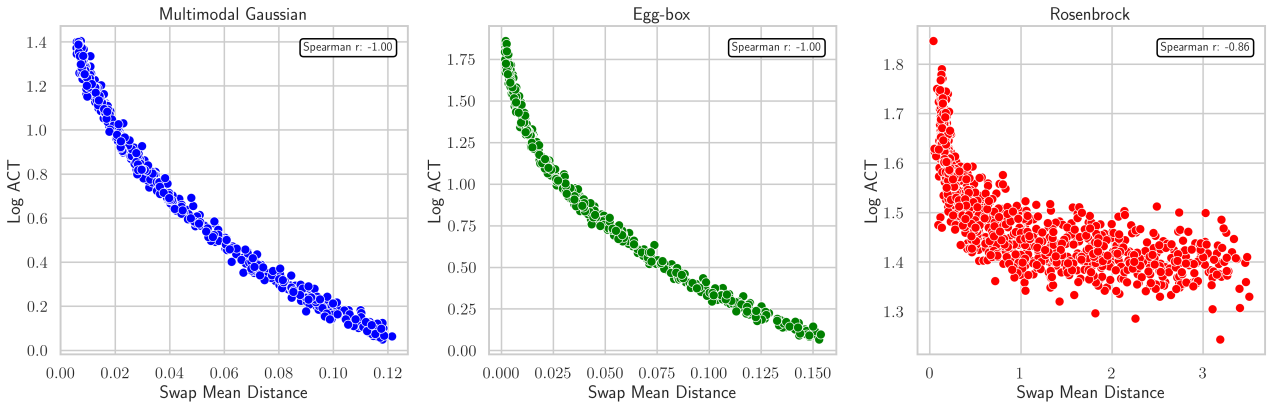

Figure 4. Scatter plot of swap mean-distance against ACT.

C.3. Distribution of Final Betas.

Although we present guarantees on the ergodicity of our adaptive algorithm, the temperature ladder to which it converges is not fixed. This is especially true for non-convex reward functions such as the swap mean-distance. Different random seeds have led to significantly different outcomes in. Figure C.3 is a box plot showing the distribution of final parameters in terms of $\Delta \log \beta_i$ using the swap mean-distance reward function.

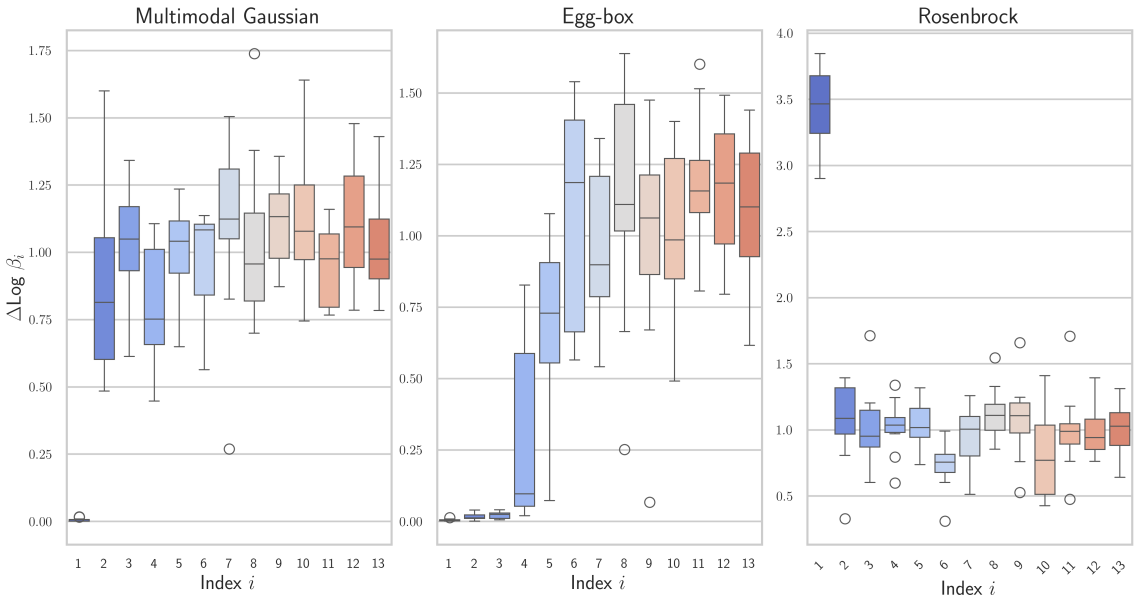

Figure 5. Box plot of final temperatures after 4000 iterations with the swap mean-distance reward function.