3rd Symposium on Advances in Approximate Bayesian Inference, 2020 1–14

# Optimal Transport Couplings of Gibbs Samplers on Partitions for Unbiased Estimation

**Brian L. Trippe*** BTRIPPE@MIT.EDU
**Tin D. Nguyen*** TDN@MIT.EDU
**Tamara Broderick** TBRODERICK@CSAIL.MIT.EDU
*MIT CSAIL*

## Abstract

Computational couplings of Markov chains provide a practical route to unbiased Monte Carlo estimation that can utilize parallel computation. However, these approaches depend crucially on chains meeting after a small number of transitions. For models that assign data into groups, e.g. mixture models, the obvious approaches to couple Gibbs samplers fail to meet quickly. This failure owes to the so-called 'label-switching' problem; semantically equivalent relabelings of the groups contribute well-separated posterior modes that impede fast mixing and cause large meeting times. We here demonstrate how to avoid label switching by considering chains as exploring the space of partitions rather than labelings. Using a metric on this space, we employ an optimal transport coupling of the Gibbs conditionals. This coupling outperforms alternative couplings that rely on labelings and, on a real dataset, provides estimates more precise than usual ergodic averages in the limited time regime. Code is available at github.com/tinnguyen96/coupling-Gibbs-partition.

## 1. Introduction

**Couplings for unbiased Markov chain Monte Carlo.** Consider estimating an analytically intractable expectation of a function $h$ of a random variable $X$ distributed according to $p$, $H^* \coloneqq \int h(X)p(X)dX$. Given a Markov chain $\{X_t\}_{t=0}^{\infty}$ with initial distribution $X_0 \sim p_0$ and evolving according to a transition kernel $X_t \sim T(X_{t-1}, \cdot)$ stationary with respect to $p$, one option is to approximate $H^*$ with the empirical average of samples $\{h(X_t)\}$. However, while ergodic averages are asymptotically consistent, they are in general biased when computed from finite simulations. As such, one cannot effectively utilize parallelism to reduce error to any desired tolerance.

Computational couplings provide a route to unbiased estimation in finite simulation (Glynn and Rhee, 2014); in this work we build on the framework of Jacob et al. (2020). One designs an additional Markov chain $\{Y_t\}$ with two properties. First, $Y_t|Y_{t-1}$ also evolves using the transition $T(\cdot, \cdot)$, so that $\{Y_t\}$ is equal in distribution to $\{X_t\}$. Secondly, there exists a random *meeting time* $\tau < \infty$ such that the two chains meet exactly at some time $\tau$, $X_\tau = Y_{\tau-1}$, and remain faithful afterwards: for all $t \geq \tau$, $X_t = Y_{t-1}$. Then, one can compute

* These authors contributed equally

an unbiased estimate of $H^*$ as

$$H_{k:m}(X,Y) \coloneqq \underbrace{\frac{1}{m-k+1}\sum_{l=k}^{m} h(X_l)}_{\text{Usual MCMC average}} + \underbrace{\sum_{l=k+1}^{\tau-1} \min\left(1, \frac{l-k}{m-k+1}\right)\{h(X_l) - h(Y_{l-1})\}}_{\text{Bias correction}} \quad (1)$$

where $k$ is the burn-in length, and $m$ sets a minimum number of iterations (Jacob et al., 2020, Equation 2). One interpretation of this estimator is as the usual MCMC estimate plus a bias correction. Since $H_{k:m}$ is unbiased, we can make the squared error (for estimating $H^*$) arbitrarily small by simply averaging many estimates computed in parallel. However, the practicality of Equation (1) relies on a coupling that provides sufficiently small meeting times. Large meeting times are doubly problematic: they lead to greater computational cost and higher variance due to the additional terms.

**Gibbs samplers over discrete structures and their couplings.** Gibbs sampling is a standard inference method for models with discrete structures and tractable conditional distributions. Numerous applications include Bayesian nonparametric clustering using Dirichlet process mixture models (Antoniak, 1974; Neal, 2000), graph coloring for randomized approximation algorithms (Jerrum, 1998), community detection using stochastic block models (Holland et al., 1983; Geng et al., 2019) and computational redistricting (DeFord et al., 2019). In these cases, the discrete structure is the partition of data into components.

While some earlier works have described couplings of Gibbs samplers, they have not sought to address computational approaches applicable in these settings. For example, Jerrum (1998) uses maximal couplings on labelings to prove convergence rates for graph coloring, and Gibbs (2004) uses a common random number coupling for two-state Ising models. Notably, these approaches rely on explicit labelings and, in our experiments, suffer from large meeting times. We attribute this issue to the label-switching problem (Jasra et al., 2005); heuristically, many different labelings imply the same partition, and two chains may nearly agree on the partition but require many iterations to change label assignments.

**Our contribution.** We view the Gibbs sampler as exploring a state-space of partitions rather than labelings (as, for example, in Tosh and Dasgupta (2014)), and define an optimal transport (OT) coupling in this space. We show that our algorithm has a fast run time and empirically validate it in the context of Dirichlet process mixture models (Antoniak, 1974; Prabhakaran et al., 2016) and graph coloring (Jerrum, 1998), where it provides smaller meeting times than the label-based couplings of Jerrum (1998); Gibbs (2004). We demonstrate the benefits of unbiasedness by reporting estimates of the posterior predictive density and cluster proportions. Our implementation is publicly available at github.com/tinnguyen96/coupling-Gibbs-partition.

## 2. Our Method

### 2.1. Gibbs samplers over partitions

For a natural number $N$, a *partition* of $[N] \coloneqq \{1, 2, \ldots, N\}$ is a collection of non-empty disjoint sets $\{A_1, A_2, \ldots, A_k\}$, whose union is $[N]$ (Pitman, 2006, Section 1.2). We use $\mathcal{P}_N$ to denote the set of all partitions of $[N]$. Throughout, we use $\pi$ to denote elements of $\mathcal{P}_N$ and

**Algorithm 1:** Gibbs Sweep with Optimal Transport Coupling

**Input:** Target probability mass function (PMF) $p_\Pi$. Current partitions $\pi$ and $\nu$.

**for** $n \leftarrow 1$ **to** $N$ **do**

  // Compute Gibbs marginals (PMFs over partitions)

  $q, r \leftarrow p_{\Pi|\Pi_{-n}}(\cdot|\pi_{-n}), p_{\Pi|\Pi_{-n}}(\cdot|\nu_{-n})$

  // Compute and sample from optimal transport coupling

  $[\pi^1, \pi^2, \ldots, \pi^K], [\nu^1, \nu^2, \ldots, \nu^{K'}] \leftarrow \text{support}(q), \text{support}(r)$

  $\gamma^* = \arg\min_{\gamma \in \Gamma(q,r)} \sum_{k=1}^{K} \sum_{k'=1}^{K'} \gamma(\pi^k, \nu^{k'}) \mathrm{d}(\pi^k, \nu^{k'})$

  $\pi, \nu \sim \gamma^*$

**end**

Return $\pi, \nu$

$\Pi$ for a random partition (i.e. a $\mathcal{P}_N$-valued random variable) with probability mass function (PMF) $p_\Pi$. Finally $\pi_{-n}$ and $\Pi_{-n}$ denote these partitions with data-point $n$ removed. For example, if $\pi = \{\{1,3\},\{2\}\}$, then $\pi_{-1} = \{\{3\},\{2\}\}$.

Drawing direct Monte Carlo samples $\Pi \sim p_\Pi$ is often impossible. However, the conditional distributions $p_{\Pi|\Pi_{-n}}$ are supported on at most $N$ partitions. Hence, when $p_\Pi$ is available up to a proportionality constant, computing and sampling from $p_{\Pi|\Pi_{-n}}$ are tractable operations. A Gibbs sampler exploiting this tractability proceeds as follows. First, a partition $\pi$ is drawn from an initial distribution $p_0$ on $\mathcal{P}_N$. For each iteration, we *sweep* through each data-point $n \in [N]$, temporarily remove it from $\pi$, and then randomly reassign it to one of the sets within $\pi_{-n}$ or add it as singleton (that is, as a new group) according the conditional PMF $p_{\Pi|\Pi_{-n}}(\cdot|\pi_{-n})$.

### 2.2. Our approach: optimal coupling of Gibbs conditionals

Our coupling encourages the chains to become 'closer' while maintaining the correct marginal evolution. To quantify closeness we use a metric on $\mathcal{P}_N$. While a number of metrics exist (Meilă, 2007, Section 2), for simplicity we chose a classical metric introduced by Mirkin and Chernyi (1970); Rand (1971),

$$\mathrm{d}(\pi, \nu) = \sum_{A \in \pi} |A|^2 + \sum_{B \in \nu} |B|^2 - 2 \sum_{A \in \pi, B \in \nu} |A \cap B|^2, \tag{2}$$

which is equivalent to Hamming distance on the adjacency matrices implied by partitions (Mirkin and Chernyi, 1970, Theorems 2-3). We leave investigation of the impact of metric choice on meeting time distribution to future work.

With the metric in Equation (2), we can formalize an *optimal transport coupling* of two Gibbs conditionals, i.e. the coupling that minimizes the expected distances between the updates. In particular, we let $q := p_{\Pi|\Pi_{-n}}(\cdot|\pi_{-n})$ and $r := p_{\Pi|\Pi_{-n}}(\cdot|\nu_{-n})$ with supports $[\pi^1, \pi^2, \ldots, \pi^K] := \text{support}(q)$ and $[\nu^1, \nu^2, \ldots, \nu^{K'}] := \text{support}(r)$ and define the OT coupling as

$$\gamma^* := \arg\min_{\gamma \in \Gamma(q,r)} \sum_{k=1}^{K} \sum_{k'=1}^{K'} \gamma(\pi^k, \nu^{k'}) \mathrm{d}(\pi^k, \nu^{k'}), \tag{3}$$

where $\Gamma(q, r)$ is the set of all couplings of $q$ and $r$. Algorithm 1 summarizes this approach.

### 2.3. Efficient computation of optimal couplings

The practicality of our OT coupling depends both on successfully encouraging chains to meet in a small number of steps and on an implementation with computational cost comparable to running single chains. If Algorithm 1 required orders of magnitude more time than the Gibbs sweep of single chains, the extent of parallelism required to place the unbiased estimates from coupled chains on an even footing with standard MCMC could be prohibitive.

In many applications, including those in our experiments, for partitions of size $K$, the Gibbs conditionals may be computed in $\Theta(K)$ time, and a full sweep through the $N$ data-points takes $\Theta(NK)$ time for a single chain. At first consideration, an implementation of Algorithm 1 with comparable efficiency might seem infeasible. In particular, when $\pi$ and $\nu$ are of size $O(K)$, Equation (3) requires computing $O(K^2)$ pairwise distances, each of which naively might seem to require at least $O(KN)$ operations — let alone the OT problem.

The following result shows that we can in fact compute this coupling efficiently.

**Theorem 1 (Gibbs Sweep Time Complexity)** *Let $p_\Pi$ be the law of a random $N$-partition. If for any $\pi \in \mathcal{P}_N, p_{\Pi|\Pi_{-n}}(\cdot|\pi_{-n})$ is computed in constant time, the Gibbs sweep in Algorithm 1 has $O(N\widetilde{K}^3 \log \widetilde{K})$ run time, where $\widetilde{K}$ is the max partition size encountered.*

As a proof of Theorem 1, we detail an $O(N\widetilde{K}^3 \log \widetilde{K})$ implementation in Appendix A.

Theorem 1 guarantees that the run time of a coupled-sweep is no more than a $O(\widetilde{K}^2 \log \widetilde{K})$ factor slower than a single-sweep. The relative magnitude of $\widetilde{K}$ versus $N$ depends on the target distribution. For the graph coloring distribution, $\widetilde{K}$ is upper bounded by the numbers of available colors. Under the Dirichlet process mixture model prior (DPMM), with high probability, the size of partition of $N$ data points is within multiplicative factors of $\ln N$ (Arratia et al., 2003, Section 5.2). We *conjecture* that under most initializations of the Gibbs sampler (such as from the DPMM prior), $\widetilde{K} = O(\ln N)$ with high probability.

**Remark 2** *The worst-case run time of Theorem 1 is attained with Orlin's algorithm (Orlin, 1993) to solve Equation (3) in $O(\widetilde{K}^3 \log \widetilde{K})$ time. However, our implementation uses the simpler network simplex algorithm (Kelly and O'Neill, 1991) as implemented by Flamary and Courty (2017). Although Kelly and O'Neill (1991, Section 3.6) upper bound the worst-case complexity of the network simplex as $O(\widetilde{K}^5)$, the algorithm's average-case performance may be as good as $O(\widetilde{K}^2)$ (Bonneel et al., 2011, Figure 6).*

Although Orlin's algorithm (Orlin, 1993) has a better worst-case runtime, convenient public implementations are not available. In addition, our main contribution is the formulation of the coupling as an OT problem — in principle, the dependence on $\widetilde{K}$ of the runtime in Theorem 1 inherits from the best OT solver used.

## 3. Empirical Results

In Section 3.2, we compare the distribution of meeting times between our partition-based coupling and two label-based couplings: under our coupling, chains meet earlier. In Section 3.3, we report unbiased estimates of two estimands of common interest: posterior predictive

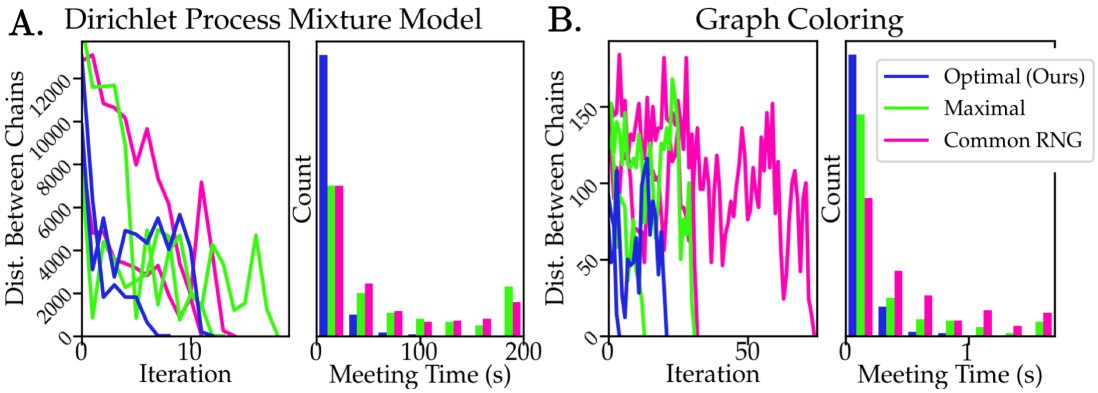



Figure 1: Reduced meeting times are achieved by OT couplings of Gibbs conditionals relative to maximal and common random number couplings in applications to (A) DPMM and (B) graph coloring. (A) Left and (B) left show two representative traces of the distance between coupled chains by iteration. (A) Right and (B) right show histograms of meeting times 250 replicate coupled chains.

densities and the posterior mean proportion of data assigned to the largest clusters. But first, we describe the applications and the target distributions under consideration in Section 3.1.

### 3.1. Applications

**Dirichlet process mixture models.** Clustering is a core task for understanding structure in data and density estimation. When the number of latent clusters is a priori unknown, DPMMs (Antoniak, 1974) are a useful tool. Part of DPMM generative process is the *Chinese restaurant process*, or $\mathrm{CRP}(\alpha, N)$, which is a probability distribution over $\mathcal{P}_N$ with mass $\Pr(\Pi = \pi) = \frac{\alpha^K \prod_{A\in\pi}(|A|-1)!}{\alpha(\alpha+1)\ldots(\alpha+N-1)}$ where $K$ is the number of clusters in $\pi$, and $\prod_{A\in\pi}$ iterates through the clusters. We consider fully conjugate DPMM (N.MacEachern, 1994),

$$\Pi \sim \mathrm{CRP}(\alpha, N), \quad \mu_A \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu_0, \Sigma_0) \text{ for } A \in \Pi, \quad W_j|\mu_A \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu_A, \Sigma_1) \text{ for } j \in A. \tag{4}$$

The hyper-parameters of Equation (4) are concentration $\alpha$, cluster prior mean $\mu_0$, observational covariance $\Sigma_1$ and cluster covariance $\Sigma_0$. For this application, the distribution is the Bayesian posterior, $p_\Pi(\pi) := \Pr(\Pi = \pi|W)$. The Gibbs conditionals of the posterior $p_{\Pi|\Pi_{-n}}$ can be computed in closed form, using simple formulas for conditioning of jointly Gaussian random variables and the well-known Polya urn scheme (Neal, 2000, Equation 3.7).

**Graph coloring.** Uniform sampling of graph colorings is a problem of fundamental interest in theoretical computer science for its role as a subroutine within fully polynomial randomized approximation algorithms, where samples from the uniform distribution on graph colorings are used to estimate the number of unique colorings (Jerrum, 1998).

Notably, this sampling problem reduces to sampling from the induced distribution on partitions, by choosing an ordering of the sets in the partition and associating it with a random permutation of the set of colors. Accordingly, estimates are just as easily constructed for a Markov chain defined on partitions. See Appendix B for additional details.

### 3.2. Reduced meeting times with OT couplings

Figure 1 demonstrates that our approach yields faster couplings than the classical maximal coupling approach (Jerrum, 1998, Section 5), or an analogous coupling using shared common random numbers (see e.g. Gibbs (2004)). In applications to both Bayesian clustering and graph coloring, the distance between coupled chains stochastically decreases to 0 (Figure 1 left panels), with our approach leading to meetings after fewer sweeps. Despite the larger per-sweep computational cost, our OT coupled chains typically meet after a shorter wall-clock time as well. We suspect this improvement comes from avoiding label-switching, which hinders mixing of the maximal and common-RNG coupled chains.

The tightest bounds for mixing time for Gibbs samplers on graph colorings to date (Chen et al., 2019) rely on couplings on labeled representations. Our results suggest better bounds may be attainable by considering convergence of partitions rather than labelings. Reducing the mixing time for Gibbs samplers of DPMM has been a motivation behind collapsed samplers (N.MacEachern, 1994), but the literature lacks upper bounds on the mixing time.

### 3.3. Unbiased estimation with parallel computation

We adapt the setup from Jacob et al. (2020, Section 3.3). Fixing a time budget, we run a single chain until time runs out and report the ergodic average. For coupled chains, we attempt as many meetings as possible in this time, and report the average across attempts.

**Posterior mean predictive density.** The posterior predictive is a key quantity used in model selection (Görür and Rasmussen, 2010), and is of particular interest for DPMMs as it is known to be consistent for the underlying data distribution in total variation distance (Ghosal et al., 1999). As a proof of concept, we computed unbiased estimates of the posterior predictive distribution of a DPMM (Figure 2 A).

We generated $N = 100$ data points from a 10-component Gaussian mixture model in one dimension, with the variance around cluster means equal to 4. We used a DPMM with $\alpha = 1$, $\mu_0 = 0$, $\Sigma_1 = 4.0$, $\Sigma_0 = 9.0$ to analyze the $N$ observations. The solid curve is an unbiased estimate of the posterior predictive density, while the dashed curve is the true density. Because of the finite sample size, the predictive density is not equal to the true density. In Appendix C, the difference between the model's predictive density and the true density decreases as sample size $N$ increases.

**Posterior mean component proportions.** A second key quantity of interest in DPMMs is the posterior mean of the proportion of data-points in the largest cluster(s) (e.g. as reported by Liverani et al. (2015)). We lastly explored parallel computation for unbiased estimation of this quantity on a real dataset (Figure 2 B). Specifically, we use a subset of the data used by Prabhakaran et al. (2016), who used a DPMM to analyse single-cell RNA-sequencing data obtained from Zeisel et al. (2015) (see Appendix B for details).

Figure 2 B presents a series of estimates of the proportion of cells in the largest component, and approximate frequentist confidence intervals. For each number of processes $M$, we aggregated $M$ independent single and coupled chain estimates, each from a single processor with a 250 second limit. We compare to the 'ground-truth' proportion obtained by MCMC run for 10,000 sweeps. Our results demonstrate the advantage of unbiased estimates in the high-parallelism, time-limited regime; while single-chain estimates have lower variance,

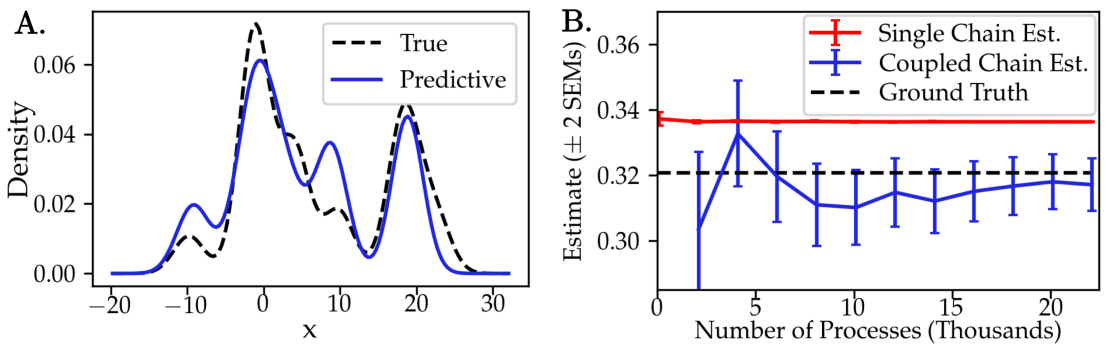



Figure 2: Unbiased estimates for Dirichlet process mixture model are obtained using OT coupled chains. (A) Unbiased estimate of the posterior predictive density for a toy problem. (B) Parallelism/accuracy trade-off for single and coupled chain estimators of the posterior mean portion of cells in the largest cluster. Each process is allocated 250 seconds, error bars indicate ±2SEM. Ground truth denotes estimates from very long MCMC chains.

coupled chains yield smaller error when aggregated across many processes. In addition, as result of unbiasedness, standard frequentist intervals may be expected to have good coverage. By contrast, we cannot expect such intervals from single chains to be calibrated; indeed, the true value is many standard errors from the single chain estimates (Figure 2 B).

However, due to the variance of the unbiased estimates we require a degree of parallelism that may be impractical for most practitioners ($\approx$ 5,000 pairs of chains to attain error comparable to that of as many single chains). Indeed, in our experiments, we simulated this high parallelism by sequentially running batches of 100 processes in parallel. Additionally, the estimation strategy can be finicky: unbiasedness requires coupled chains to meet exactly, and for some models & experiments not shown, we found that some pairs of coupled chains failed to meet quickly. This difficulty is expected for problems where single chains mix slowly, as slow mixing precludes the existence of fast couplings (Jacob, 2020, Chapter 3). Looking forward, we expect that our work will naturally benefit from advances in parallel-computation software and hardware, such as GPU implementations. Reducing the variance of the unbiased estimates is an open question, and is the target of ongoing work.

## Acknowledgements

The authors thank the anonymous reviewers for constructive feedback. BLT is supported by NSF GRFP.

## References

Charles E Antoniak. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174, 1974.

Richard Arratia, Andrew D Barbour, and Simon Tavaré. *Logarithmic combinatorial structures: a probabilistic approach*, volume 1. European Mathematical Society, 2003.

Nicolas Bonneel, Michiel Van De Panne, Sylvain Paris, and Wolfgang Heidrich. Displacement interpolation using Lagrangian mass transport. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, pages 1–12, 2011.

Sitan Chen, Michelle Delcourt, Ankur Moitra, Guillem Perarnau, and Luke Postle. Improved bounds for randomly sampling colorings via linear programming. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2216–2234. SIAM, 2019.

Daryl DeFord, Moon Duchin, and Justin Solomon. Recombination: A family of Markov chains for redistricting. *arXiv preprint arXiv:1911.05725*, 2019.

Rémi Flamary and Nicolas Courty. POT python optimal transport library. *GitHub: https://github. com/rflamary/POT*, 2017.

Junxian Geng, Anirban Bhattacharya, and Debdeep Pati. Probabilistic community detection with unknown number of communities. *Journal of the American Statistical Association*, 114(526):893–905, 2019.

Subhashis Ghosal, Jayanta K Ghosh, and RV Ramamoorthi. Posterior consistency of Dirichlet mixtures in density estimation. *Annals of Statistics*, 27(1):143–158, 1999.

Jayanta K Ghosh and RV Ramamoorthi. *Bayesian nonparametrics*. Springer Series in Statistics, 2003.

Alison L Gibbs. Convergence in the Wasserstein metric for Markov chain Monte Carlo algorithms with applications to image restoration. 2004.

Peter W Glynn and Chang-han Rhee. Exact estimation for Markov chain equilibrium expectations. *Journal of Applied Probability*, 51(A):377–389, 2014.

Dilan Görür and Carl Edward Rasmussen. Dirichlet process Gaussian mixture models: choice of the base distribution. *Journal of Computer Science and Technology*, 25(4):653–664, 2010.

Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

Pierre Jacob. Couplings and Monte Carlo. Course Lecture Notes, 2020.

Pierre E Jacob, John OLeary, Yves F Atchadé, et al. Unbiased Markov chain Monte Carlo methods with couplings. *Journal of the Royal Statistical Society Series B*, 82(3):543–600, 2020.

Ajay Jasra, Chris C Holmes, and David A Stephens. Markov chain Monte Carlo methods and the label switching problem in bayesian mixture modeling. *Statistical Science*, pages 50–67, 2005.

Mark Jerrum. Mathematical foundations of the Markov chain Monte Carlo method. In *Probabilistic methods for algorithmic discrete mathematics*, pages 116–165. Springer, 1998.

Damian J. Kelly and Garrett M. O'Neill. *The minimum cost flow problem and the network simplex solution method*. PhD thesis, Citeseer, 1991.

Antonio Lijoi, Igor Prünster, and Stephen G Walker. On consistency of nonparametric normal mixtures for Bayesian density estimation. *Journal of the American Statistical Association*, 100(472):1292–1296, 2005.

Silvia Liverani, David I Hastie, Lamiae Azizi, Michail Papathomas, and Sylvia Richardson. PReMiuM: An R package for profile regression mixture models using Dirichlet processes. *Journal of Statistical Software*, 64(7):1, 2015.

Marina Meilă. Comparing clusteringsan information based distance. *Journal of multivariate analysis*, 98(5):873–895, 2007.

BG Mirkin and LB Chernyi. Measurement of the distance between distinct partitions of a finite set of objects. *Autom Tel*, 5:120–127, 1970.

Radford Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9, 01 2000. doi: 10.2307/1390653.

Steven N.MacEachern. Estimating normal means with a conjugate style dirichlet process prior. *Communications in Statistics - Simulation and Computation*, 23(3):727–741, 1994. doi: 10.1080/03610919408813196. URL https://doi.org/10.1080/03610919408813196.

James B. Orlin. A faster strongly polynomial minimum cost flow algorithm. *Operations Research*, 41(2):338–350, 1993.

Jim Pitman. *Combinatorial Stochastic Processes: Ecole d'Eté de Probabilités de Saint-Flour XXXII-2002*. Springer, 2006.

Sandhya Prabhakaran, Elham Azizi, Ambrose Carr, and Dana Peer. Dirichlet process mixture model for correcting technical variation in single-cell gene expression data. In *International Conference on Machine Learning*, pages 1070–1079, 2016.

William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

Christopher Tosh and Sanjoy Dasgupta. Lower bounds for the Gibbs sampler over mixtures of Gaussians. In *International Conference on Machine Learning*, pages 1467–1475, 2014.

Amit Zeisel, Ana B Muñoz-Manchado, Simone Codeluppi, Peter Lönnerberg, Gioele La Manno, Anna Juréus, Sueli Marques, Hermany Munguba, Liqun He, Christer Betsholtz, et al. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science*, 347(6226):1138–1142, 2015.

## Appendix A. Proof of Gibbs Sweep Time Complexity

We here detail our $O(N\widetilde{K}^3 \log \widetilde{K})$ implementation of Algorithm 1. This serves as proof of Theorem 1.

Note that work in Algorithm 1 may be separated into 2 computationally demanding stages for each of the $N$ data-points, $n \in [N]$; computing the distances between each pair of

partitions in the Cartesian product of supports of the Gibbs conditionals $p_{\Pi|\Pi_{-n}}(\cdot|\pi_{-n})$ and $p_{\Pi|\Pi_{-n}}(\cdot|\nu_{-n})$ and solving the optimal transport problem in line 7. As discussed in Remark 2, the optimal transport problem may be solved in $O(K^3 \log K)$ time, and is the bottleneck step. As such it remains only to show that for each $n \in [N]$, the pairwise distances may also be computed in $O(K^3 \log K)$ time.

Recall that for two partitions $\pi, \nu \in \mathcal{P}_N$ the metric of interest is

$$\mathrm{d}(\pi, \nu) = \sum_{A \in \pi} |A|^2 + \sum_{B \in \nu} |B|^2 - 2 \sum_{(A,B) \in \pi \times \nu} |A \cap B|^2. \tag{5}$$

However, it is not obvious from this expression alone that fast computation of pairwise distances should be possible. We make this explicit in the following remark.

**Remark 3** *Given constant $O(1)$ time for querying set membership (e.g. as provided by a standard hash-table set implementation), for $\pi, \nu \in \mathcal{P}_N$, $d(\pi, \nu)$ in Equation (2) may be computed in $O\left(N \min(|\pi|, |\nu|)\right)$ time. If we let $\widetilde{K}$ be the number of groups, so that $\widetilde{K} \approx \min(|\pi|, |\nu|)$, this gives $O(N\widetilde{K})$ time.*

While this is certainly faster than a naive approach relying on the formulation of this metric based on adjacency matrices, it is still not sufficient, as it is a factor of $N\widetilde{K}^2$ slower than the original (recall that we will need to do this for $\widetilde{K}^2$ pairs of clusters assignments).

However we can do better for the Gibbs update by making two observations. First, if we use $A_n$ and $B_n$ to denote the elements of $\pi$ and $\nu$, respectively, containing data-point $n$, then for any $n$ we may write

$$\begin{aligned} d(\pi, \nu) &= d(\pi_{-n}, \nu_{-n}) + \left[|A_n|^2 - (|A_n| - n)^2\right] + \left[|B_n|^2 - (|B_n| - n)^2\right] && (6) \\ &\quad -2\left[|A_n \cap B_n|^2 - (|A_n \cap B_n| - 1)^2\right] && (7) \\ &= d(\pi_{-n}, \nu_{-n}) + 2\left[|A_n| + |B_n| - 2|A_n \cap B_n|\right]. && (8) \end{aligned}$$

Second, the solution to the optimisation problem in Equation (3) is unchanged when we add a constant value to every distance: Using again the notation of Algorithm 1 we let $q := p_{\Pi|\Pi_{-n}}(\cdot|\pi_{-n})$ and $r := p_{\Pi|\Pi_{-n}}(\cdot|\nu_{-n})$ with supports $(\pi_1, \pi_2, \ldots, \pi_K) = \mathrm{support}(q)$ and $(\nu_1, \nu_2, \ldots, \nu_{K'}) = \mathrm{support}(r)$. and rewrite

$$\begin{aligned} \gamma^* &:= \arg\min_{\gamma \in \Gamma(q,r)} \sum_{\in \mathcal{P}_N} \sum_{y \in \mathcal{P}_N} d(x, y)\gamma(x, y) && (9) \\ &= \arg\min_{\gamma \in \Gamma(q,r)} \sum_{x \in \mathcal{P}_N} \sum_{y \in \mathcal{P}_N} (d(x, y) - c)\gamma(x, y) && (10) \end{aligned}$$

for any constant $c$; taking $c = d(\pi_{-n}, \nu_{-n})$ reveals that we need only compute the second term in Equation (6).

At first it may seem that this still does not solve the problem, as directly computing the size of the set intersections is $O(N)$ (if cluster sizes scale as $O(N)$). However, Equation (9) is just our final stepping stone. If we additionally keep track of sizes of intersections at every step, updating them as we adapt the partitions will take constant time for each update.

As such, we are able to form the matrix of pairwise distances in $O(\widetilde{K}^2)$ time. Regardless of $N$, this moves the bottleneck step to solving the OT problem which, as discussed in Theorem 2, may be computed in $O(\widetilde{K}^3 \log \widetilde{K})$ time with Orlin's algorithm (Orlin, 1993). We provide a practical implementation of this approach in our code; see `pairwise_dists()` in `modules/utils.py`.

## Appendix B. Additional Experimental Details

### B.1. Meeting time distributions

**DP mixtures.** For each replicate, we simulated $N = 150$ data-points from a $K = 4$ component, 2 dimensional Gaussian mixture model. The target distribution was the posterior of the probabilistic model Equation (4), with $\Sigma_0 = 2I_2$, $\Sigma_1 = 2.5I_2$ and $\alpha = 0.2$. For each replicate true means for the finite mixture were sampled as $\mu_k \sim \mathcal{N}(0, \Sigma_0)$, mixing proportions as $\theta \sim \text{Dir}(\alpha 1_K)$, and each of the $n \in [N]$ observations as $z_n \sim \text{Cat}(\theta)$, $W_n \sim \mathcal{N}(\mu_{z_n}, \Sigma_1)$. See `notebooks/Coupled_CRP_sampler.ipynb` for complete implementation and details. This code is adapted from github.com/tbroderick/mlss2015_bnp_tutorial/blob/master/ex5_dpmm.R

**Graph coloring** Let $G$ be an undirected graph with vertices $V = [N]$ and edges $E \subset V \otimes V$, and let $Q = [q]$ be set of $q$ colors. A graph coloring is an assignment of a color in $Q$ to each vertex satisfying that the endpoints of each edge have different colors. We here demonstrate an application of our method to a Gibbs sampler which explores the uniform distribution over valid $q-$colorings of $G$, i.e. the distribution which places equal mass on ever proper coloring of $G$.

To employ Algorithm 1, for this problem we need only to characterise the PMF on partitions of the vertices implied by the uniform distribution on its colorings.

A partition corresponds to a proper coloring only if no two adjacent vertices are in the element of the partition. As such, we can write

$$p_{\Pi_N}(\pi) \propto \mathbb{1}\{|\pi| \leq q \text{ and } A(\pi)_{i,j} = 1 \to (i,j) \notin E, \ \forall i \neq j\} \binom{q}{|\pi|} |\pi|!,$$

where the indicator term checks that $\pi$ can correspond to a proper coloring and the second term accounts for the number of unique colorings which induce the partition $\pi$. In particular it is the product of the number of ways to choose $|\pi|$ unique colors from $Q$ ( $\binom{q}{|\pi|} := \frac{q!}{|\pi|!(q-|\pi|)!}$ ) and the number of ways to assign those colors to the groups of vertices in $\pi$.

For the experiments in Figure 1, we simulated Erdős-Rényi random graphs with $N = 25$ vertices, and including each possible edge with probability 0.2. We chose a maximum number of colors $Q$ by first initializing a coloring greedily and setting $Q$ as the number of colors used in this initial coloring plus two. See `notebooks/coloring_OT.ipynb` for complete implementation and results. This code is adapted from:
github.com/pierrejacob/couplingsmontecarlo/inst/chapter3/3graphcolourings.R

### B.2. Unbiased estimation

**Predictive density.** The true density is a 10-component Gaussian mixture model with known observational noise variance $\sigma = 2.0$. The cluster proportions were generated from a

symmetric Dirichlet distribution with mass 1 for all 10-coordinates. The cluster means were randomly generated from $\mathcal{N}(0, 10^2)$. The target DP mixture model had $\alpha = 1$, standard deviation over cluster means 3.0 and standard deviation over observations 2.0. We ran 10,000 replicates of the time-budgeted estimator using coupled chains, each replicate given a sufficient time budget so that all 10,000 replicates had at least one successful meeting in the allotted time.

**Top component proportion in single-cell RNAseq.** We extracted $D = 50$ genes with the most variation of $N = 200$ cells. We then take the log of the features, and normalize so that each feature has mean 0 and variance 1. We as our target the posterior of the probabilistic model in Eq. (4) with $\alpha = 1.0$, $\mu_0 = 0$, $\Sigma_0 = 0.5$, $\Sigma_1 = 1.3 I_D$. Notably, this is a simplification of the set-up considered by Prabhakaran et al. (2016), who work with a larger dataset and additionally perform fully Bayesian inference over these hyper-parameters. In our experiments, the function of interest is the posterior expected of the proportion of cells in the largest cluster i.e. $\mathbb{E}[\max_{A \in \pi} |A|/N | W]$.

## Appendix C. More plots of predictive density

### C.1. Posterior concentration implies convergence in total variation of predictive density

Some references on posterior concentration are Ghosal et al. (1999); Lijoi et al. (2005). The true data generating process is that there exists some density $f_0$ w.r.t. Lebesgue measure that generates the data in an iid manner $X_1, X_2, \ldots, X_n$. We use the notation $P_{f_0}$ to denote the probability measure with density $f_0$. The probabilistic model is that we have a prior $\Pi$ over densities $f$, and observations $X_i$ are conditionally iid given $f$. Let $\mathcal{F}$ be the set of all densities on $\mathbb{R}$. For any measurable subset $A$ of $\mathcal{F}$, the posterior of $A$ given the observations $X_i$ is denoted $\Pi(A | X_{1:N})$. A strong neighborhood around $f_0$ is any subset of $\mathcal{F}$ containing a set of the form $V = \{f \in \mathcal{F} : \int |f - f_0| < \epsilon\}$ according to Ghosal et al. (1999). The prior $\Pi$ is strongly consistent at $f_0$ if for any strong neighborhood $U$,

$$\lim_{n \to \infty} \Pi(U | X_{1:n}) = 1, \tag{11}$$

holds almost surely for $X_{1:\infty}$ distributed according to $P_{f_0}^{\infty}$.

**Theorem 4 (Ghosh and Ramamoorthi (2003, Proposition 4.2.1))** *If a prior $\Pi$ is strongly consistent at $f_0$ then the predictive distribution, defined as*

$$\widehat{P}_n(A \mid X_{1:n}) = \int_f P_f(A) \Pi(f | X_{1:n}) \tag{12}$$

*also converges to $f_0$ in total variation in a.s. $P_{f_0}^{\infty}$*

$$d_{TV}\left(\widehat{P}_n, P_{f_0}\right) \to 0.$$

**Theorem 5 (DP mixtures prior is consistent for finite mixture models)** *Let the true density be a finite mixture model $f_0(x) := \sum_{i=1}^{m} p_i \mathcal{N}(x | \theta_i, \sigma_1^2)$. Consider the following*

*probabilistic model*

$$\begin{aligned} \widehat{P} &\sim \mathrm{DP}(\alpha, \mathcal{N}(0, \sigma_0^2)) \\ \theta_i \mid \widehat{P} &\overset{iid}{\sim} \widehat{P} && i = 1, 2, \ldots, n \\ X_i \mid \theta_i &\overset{indep}{\sim} \mathcal{N}(\theta_i, \sigma_1^2) && i = 1, 2, \ldots, n \end{aligned}$$

*Let $\widehat{P}_n$ be the posterior predictive distribution of this generative process. Then with a.s. $P_{f_0}$*

$$d_{TV}\left(\widehat{P}_n, P_{f_0}\right) \xrightarrow{n\to\infty} 0.$$

**Proof** [Proof of Theorem 5] First, we can rewrite the DP mixture model as a generative model over continuous densities $f$

$$\begin{aligned} \widehat{P} &\sim \mathrm{DP}(\alpha, \mathcal{N}(0, \sigma_0^2)) \\ f &= \mathcal{N}(0, \sigma_1^2) * \widehat{P} \\ X_i \mid f &\overset{iid}{\sim} f && i = 1, 2, \ldots, n \end{aligned} \tag{13}$$

where $\mathcal{N}(0, \sigma_1^2) * \widehat{P}$ is a convolution, with density $f(x) := \int_\theta \mathcal{N}(x - \theta | 0, \sigma_1^2) d\widehat{P}(\theta)$.

The main idea is showing that the posterior $\Pi(f | X_{1:n})$ is strongly consistent and then leveraging Theorem 4. For the former, we verify the conditions of Lijoi et al. (2005, Theorem 1).

The first condition of Lijoi et al. (2005, Theorem 1) is that $f_0$ is in the K-L support of the prior over $f$ in Equation (13). We use Ghosal et al. (1999, Theorem 3). Clearly $f_0$ is the convolution of the normal density $\mathcal{N}(0, \sigma_1^2)$ with the distribution $P(.) = \sum_{i=1}^m p_i \delta_{\theta_i}$. $P(.)$ is compactly supported since $m$ is finite. Since the support of $P(.)$ is the set $\{\theta_i\}_{i=1}^m$ which belongs in $\mathbb{R}$, the support of $\mathcal{N}(0, \sigma_0^2)$, by Ghosh and Ramamoorthi (2003, Theorem 3.2.4), the conditions on $P$ are satisfied. The condition that the prior over bandwidths cover the true bandwidth is trivially satisfied since we perfectly specified $\sigma_1$.

The second condition of Lijoi et al. (2005, Theorem 1) is simple: because the prior over $\widehat{P}$ is a DP, it reduces to checking that

$$\int_{\mathbb{R}} |\theta| \mathcal{N}(\theta \mid 0, \sigma_0^2) < \infty$$

which is true.

The final condition trivial holds because we have perfectly specified $\sigma_1$: there is actually zero probability that $\sigma_1$ becomes too small, and we never need to worry about setting $\gamma$ or the sequence $\sigma_k$. ■

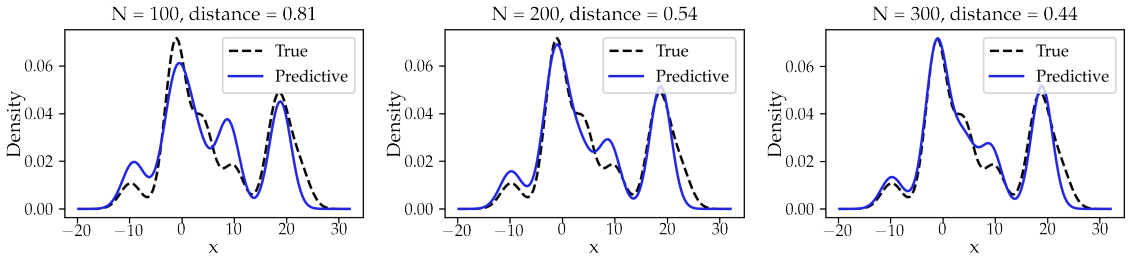
### C.2. Predictive density plots for varying $N$

In Figure 3, the distance between the posterior predictive density and the underlying density decreases as $N$ increases. We sampled a grid $\{u_j\}$ of 150 evenly-spaced points in the domain $[-20, 30]$, and evaluated both the true density and the posterior predictive density on this

grid. The distance in question sums over the absolute differences between the evaluations over the grid

$$\text{dist} := \sum_j |f_N(u_j) - f_0(u_j)|.$$

where $f_N(u_j)$ is the posterior predictive density of the $N$ observations under the DPMM at $u_j$. The distance is meant to illustrate *pointwise* rather than total variation convergence. Although the predictive density converges in total variation to the underlying density, it is only guaranteed that a subsequence of the predictive density converges pointwise to the underlying density.




(*a*) $N = 100$. (*b*) $N = 200$. (*c*) $N = 300$.

Figure 3: Posterior predictive density for different $N$. The time budget for each replicate when $N = 100, 200, 300$ is respectively $50, 300, 800$ seconds.

In Figure 3, each $N$ has a different time budget because for larger $N$, in general per-sweep time increases and number of sweeps until coupled chains meet also increase.