
Transport Reversible Jump Proposals

Laurence Davies^{1,3}

Robert Salomone^{2,3}

Matthew Sutton^{1,3}

Christopher Drovandi^{1,3}

¹ School of Mathematical Sciences, Queensland University of Technology

² School of Computer Science, Queensland University of Technology

³ Centre for Data Science, Queensland University of Technology

Abstract

Reversible jump Markov chain Monte Carlo (RJMCMC) proposals that achieve reasonable acceptance rates and mixing are notoriously difficult to design in most applications. Inspired by recent advances in deep neural network-based normalizing flows and density estimation, we demonstrate an approach to enhance the efficiency of RJMCMC sampling by performing transdimensional jumps involving reference distributions. In contrast to other RJMCMC proposals, the proposed method is the first to apply a non-linear transport-based approach to construct efficient proposals between models with complicated dependency structures. It is shown that, in the setting where exact transports are used, our RJMCMC proposals have the desirable property that the acceptance probability depends only on the model probabilities. Numerical experiments demonstrate the efficacy of the approach.

1. INTRODUCTION

The problem of interest is sampling from a probability distribution π with support

$$\mathcal{X} = \bigcup_{k \in \mathcal{K}} (\{k\} \times \Theta_k), \quad (1)$$

where \mathcal{K} is a discrete *index* set, and $\Theta_k \subseteq \mathbb{R}^{n_k}$, where the n_k may differ, and hence \mathcal{X} is a *transdimensional* space. The choice of notation Θ_k is made as the problem typically arises in Bayesian model selection, where such sets correspond to the space of model *parameters*, and $k \in \mathcal{K}$ is a *model index* or indicator. The reversible jump Markov chain Monte Carlo (RJMCMC) algorithm, formally introduced by [Green \(1995\)](#), generalizes the Metropolis–Hastings algorithm ([Hastings, 1970](#)) via the introduction

of user-specified diffeomorphisms¹ $h_{k,k'} : \Theta_k \times \mathcal{U}_k \rightarrow \Theta_{k'} \times \mathcal{U}_{k'}$ where $\mathcal{U}_k, \mathcal{U}_{k'}$ are (possibly empty) sets chosen to ensure dimensions are matched. Due to this additional complexity, RJMCMC proposals that perform well in practice are generally challenging to design.

RJMCMC methods are a frequently revisited research topic for which many approaches exist. [Brooks et al. \(2003\)](#) outline several approaches to improve the efficiency of jump proposals on various problem types, including variable selection and nested models. [Green and Mira \(2001\)](#) identify the efficiency issues associated with naïve transdimensional proposals between non-overlapping targets and propose a delayed-rejection auxiliary-proposal mechanism. [Al-Awadhi et al. \(2004\)](#) employ an auxiliary target density instead of an auxiliary proposal. [Hastie \(2005\)](#) proposes an approach for (potentially) multi-modal conditional target densities, achieved by fitting a Gaussian mixture model to each individual conditional target, and using a shift and whitening transformation corresponding to a randomly selected mixture component. [Fan et al. \(2008\)](#) propose a conditional factorization of a differentiable target density to sequentially construct a proposal density. [Karagiannis and Andrieu \(2013\)](#) propose the construction of a Markov chain through an annealed sequence of intermediate distributions between models to encourage higher acceptance rates between model jumps. [Farr et al. \(2015\)](#) propose a KD-tree approximation of the target density for auxiliary variable draws to improve the efficiency of RJMCMC proposals. [Gagnon \(2021\)](#) uses the locally-informed MCMC approach for discrete spaces developed by [Zanella \(2020\)](#) to improve the exploration efficiency of the model space when global jump proposals are unavailable. However, there appears to be no *general* strategy for the design of across-model proposals that is widely applicable, theoretically justified, and of practical use.

The use of measure transport to enhance sampling methods is an area of recent interest. This was first formalized and demonstrated in [Parno and Marzouk \(2018\)](#), where *ap-*

¹bijective functions that are differentiable and have a differentiable inverse

proximate transport maps (TMs) are used for accelerating MCMC. The form of the approximate transports as applied to MCMC sampling is described in general terms, but their choice of transports uses systems of orthogonal multivariate polynomials. The application of approximate TMs to enhance sampling methods includes mapping a deterministic step within sequential Monte Carlo (Arbel et al., 2021); the transformation of a continuous target distribution to an easier-to-sample distribution via a map learned using stochastic variational inference (Hoffman et al., 2018); and the use of approximate TMs for the construction of independence proposals in an adaptive MCMC algorithm (Gabrié et al., 2022).

However, despite the considerable promise of incorporating approximate TMs into sampling methodology, to our knowledge, such ideas have not been considered in the *transdimensional* sampling setting. Such an omission is somewhat surprising, as RJMCMC samplers are constructed using a form of invertible maps involving distributions, and hence it would intuitively appear natural that distributional transport could feature in a useful capacity. This work aims to develop such an approach and to demonstrate its benefits.

Contribution. The primary contributions of this work are as follows:

I. A new class of RJMCMC proposals for across-model moves, called *transport reversible jump* (TRJ) proposals are developed. In the idealized case where exact transports are used, the proposals are shown to have a desirable property (Proposition 1).

II. A numerical study is conducted on challenging examples demonstrating the efficacy of the proposed approach in the setting where *approximate* transport maps are used.

III. An alternative “all-in-one” approach to training approximate TMs is developed, which involves combining a saturated state space formulation of the target distribution with *conditional* TMs.

Code for the numerical experiments is made available at <https://github.com/davies1/trjp>.

Structure of this Article. The remainder of this article is structured as follows: Section 2 discusses the required background concepts regarding RJMCMC, transport maps, and flow-based models. Section 3 introduces a general strategy for using transport maps within RJMCMC and discusses its properties. Section 4 conducts a numerical study to demonstrate the efficacy of the strategy in the case where *approximate* transports are used. Section 5 explores an alternative “all-in-one” approach to training transport maps and provides an associated numerical example. Section 6 concludes the paper.

Notation. For a function T and distribution ν , $T\#\nu$ denotes the *pushforward* of ν under T . That is, if $Z \sim \nu$, then $T\#\nu$ is the probability distribution of $T(Z)$. For a univariate probability distribution ν , define $\otimes_n \nu$ as $\underbrace{\nu \otimes \cdots \otimes \nu}_{n \text{ times}}$.

Throughout, univariate functions are to be interpreted as applied element-wise when provided with a vector as input. The symbol \odot denotes the Hadamard (element-wise) product. The univariate standard normal probability density function is denoted as ϕ , ϕ_d is the d -dimensional multivariate standard normal probability density function, and $\phi_{\Sigma_{d \times d}}$ is the d -dimensional multivariate normal probability density function centered at $\mathbf{0}_d$ with covariance $\Sigma_{d \times d}$. For a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, the notation $|J_f(\boldsymbol{\theta})|$ denotes the absolute value of the determinant of the Jacobian matrix of f evaluated at some $\boldsymbol{\theta} \in \mathbb{R}^n$. For distributions π defined on sets of the form in (1), we write π_k for the distribution conditional on k , and $\pi(k)$ for its k -marginal distribution.

2. BACKGROUND

2.1 Reversible Jump Markov Chain Monte Carlo

For a distribution π defined on a space of the form in (1), with associated probability density function $\pi(\mathbf{x})$, the standard method to construct a π -invariant Metropolis-Hastings kernel (and thus an associated MCMC sampler) is the *reversible jump* approach introduced in the seminal work by Green (1995). The proposal mechanism is constructed to take $\mathbf{x} = (k, \boldsymbol{\theta}_k)$ to $\mathbf{x}' = (k', \boldsymbol{\theta}'_{k'})$ where the dimensions of $\boldsymbol{\theta}_k$ and $\boldsymbol{\theta}'_{k'}$ are n_k and $n_{k'}$, respectively. The approach employs *dimension matching*, introducing auxiliary random variables $\mathbf{u}_k \sim g_{k,k'}$ and $\mathbf{u}'_{k'} \sim g_{k',k}$ of dimensions w_k and $w_{k'}$, which are arbitrary provided that $n_k + w_k = n_{k'} + w_{k'}$. A proposal is then made using these auxiliary random variables and a chosen diffeomorphism $h_{k,k'}$ defined so that $(\boldsymbol{\theta}'_{k'}, \mathbf{u}'_{k'}) = h_{k,k'}(\boldsymbol{\theta}_k, \mathbf{u}_k)$. A discrete proposal distribution j_k is also specified for each $k \in \mathcal{K}$, where $j_k(k')$ defines the probability of proposing to model k' from model k . More generally, the distributions j_k may also depend on $\boldsymbol{\theta}_k$, but we do not consider this case. With the above formulation of the proposal, the RJMCMC acceptance probability is

$$\alpha(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{\pi(\mathbf{x}') j_{k'}(k) g_{k',k}(\mathbf{u}'_{k'})}{\pi(\mathbf{x}) j_k(k') g_{k,k'}(\mathbf{u}_k)} |J_{h_{k,k'}}(\boldsymbol{\theta}_k, \mathbf{u}_k)|. \quad (2)$$

2.2 Transport Maps and Flow-Based Models

Consider two random vectors $\boldsymbol{\theta} \sim \mu_\theta$ and $Z \sim \mu_z$, such that their distributions μ_θ and μ_z are absolutely continuous with respect to n -dimensional Lebesgue measure. A function $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is called a *transport map* (TM) from μ_θ to μ_z if $\mu_z = T\#\mu_\theta$. In this setting, we refer to μ_θ as the *target* distribution and μ_z as the *reference* distribution. Transport maps between two prescribed distributions

are known to exist under mild conditions (see e.g., [Parno and Marzouk \(2018\)](#) and the references therein).

One strategy for obtaining approximate TMs from samples of a target π is via density estimation with a family of distributions arising from transformations. Let $\{T_\psi\}$ be a family of diffeomorphisms parameterized by $\psi \in \Psi$ with domain on the support of some arbitrary *base* distribution μ_z . Then, for fixed ψ , the probability density function of the random vector $\zeta = T_\psi(\mathbf{Z})$ is

$$\mu_\zeta(\zeta; \psi) = \mu_z(T_\psi^{-1}(\zeta)) |J_{T_\psi^{-1}}(\zeta)|, \quad \zeta \in \mathbb{R}^n. \quad (3)$$

An approximate TM from μ_z to some prescribed target distribution π can be learned as a result of fitting a model of the above form to samples from π by minimizing KL divergence between the model and an empirical distribution of possibly-approximate samples from π , which is equivalent to maximum likelihood estimation.

Families of distributions arising from the specification of highly flexible and scalable $\{T_\psi\}$ have been an area of intense recent interest in the machine learning literature, often referred to as *flow-based models*, with the families $\{T_\psi\}$ often referred to as *flows* or *normalizing flows* (NF). A general class of transform is *autoregressive flows*, briefly described as follows. Let $\tau(\cdot; \omega)$, called the *transformer*, be a univariate diffeomorphism parametrized by $\omega \in \Omega$, where Ω is the set of admissible parameters. Then, the transformation is defined elementwise via

$$T(\mathbf{Z})_i = \tau(Z_i; \zeta_i(z_{<i}; \psi)), \quad i = 1, \dots, n, \quad (4)$$

where the functions $\{\zeta_i : \mathbb{R}^{i-1} \rightarrow \Omega\}$ are called the *conditioners*. In practice, the $\{\zeta_i\}$ are each individually subsets of the output of a single neural network that takes \mathbf{Z} as input and is designed to respect the autoregressive structure. When fitting approximate transports to samples in our experiments in Section 4, for additional flexibility the flow-based model used arises from *several* chained transformations of the form in (4), each with different parameters, and where the transformer is a piecewise function involving monotonic rational-quadratic splines ([Durkan et al., 2019](#)). For further details and a comprehensive overview of other flows, see [Papamakarios et al. \(2021\)](#).

3. TRANSPORT REVERSIBLE JUMP PROPOSALS

We introduce a special form of general proposal between two models, corresponding to indices k and k' . Let ν be some *univariate* reference distribution, and let $\{T_k : \mathbb{R}^{n_k} \rightarrow \mathcal{Z}^{n_k}\}$ be a collection of diffeomorphisms such that $T_k \# \pi_k \approx \otimes_{n_k} \nu$, $k \in \mathcal{K}$, i.e., T_k is an approximate TM from the conditional target corresponding to index k , to an appropriately-dimensioned product of independent reference distributions. Similarly, T_k^{-1} is an ap-

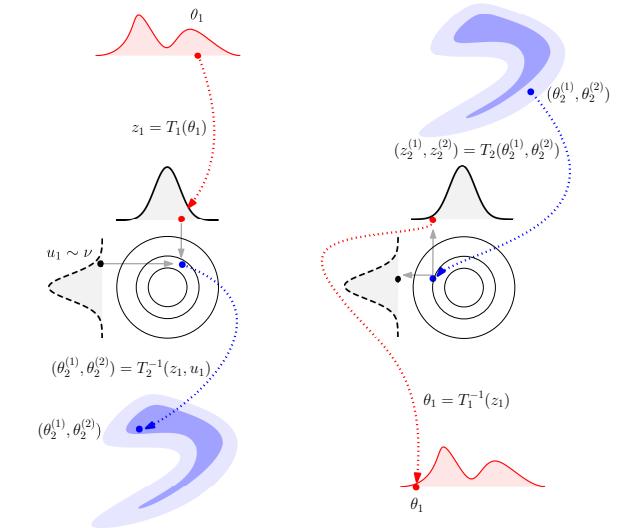


Figure 1: Illustration of the proposal class. Here, the reference ν is Gaussian, and the functions $\bar{h}_{k,k'}$ are the identity map.

proximate TM from the reference to the conditional target for index k . The general idea of the proposal scheme from model k to model k' is to first apply the transformation that would (approximately) transport π_k to its product reference, drawing auxiliary variables (if proposing to a higher-dimensional model), optionally applying a product reference measure preserving diffeomorphism, discarding auxiliary variables (if proposing to a lower dimensional model), and then applying the transformation that would (approximately) transport the augmented vector to the new conditional target. Figure 1 illustrates this idea for jumps between targets of one and two dimensions.

Formally, we first restrict the dimension of proposed auxiliary variables \mathbf{u} (if any) to w_k , which is defined to be the dimension difference between θ_k and $\theta'_{k'}$, i.e., $n_k + w_k = n_{k'}$. Then, assuming $w_k \geq 0$ (a jump is proposed to a model of equal or higher dimension), the proposal is obtained via

$$\begin{aligned} \mathbf{z}_k &= T_k(\theta_k), \\ \mathbf{z}_{k'} &= \bar{h}_{k,k'}(\mathbf{z}_k, \mathbf{u}_k), \quad \text{where } \mathbf{u}_k \sim \otimes_{w_k} \nu, \\ \theta'_{k'} &= T_{k'}^{-1}(\mathbf{z}_{k'}), \end{aligned} \quad (5)$$

where each $\bar{h}_{k,k'} : \mathbb{R}^{n_k} \times \mathbb{R}^{w_k} \rightarrow \mathbb{R}^{n_{k'}}$, is a diffeomorphism that both satisfies the pushforward-invariance condition

$$\bar{h}_{k,k'} \# \otimes_{\max\{n_k, n'_{k'}\}} \nu = \otimes_{\max\{n_k, n'_{k'}\}} \nu, \quad (6)$$

and is volume-preserving (i.e., the absolute value of the Jacobian determinant is always equal to one). The default

choice is to set each $\bar{h}_{k,k'}$ equal to the identity (resulting in the concatenation of the two inputs), but other choices are possible. For example, as $\otimes_{\max\{n_k, n_{k'}\}} \nu$ is exchangeable, permutations are also suitable. Proposal moves to lower dimensions are similar to those in (5), but auxiliary variables of appropriate dimension are discarded as opposed to being drawn. The procedure of using our TRJ proposals within RJMCMC is given in Algorithm 1.

Algorithm 1: TRJ: Transport Reversible Jump MCMC

```

input :  $x = (k, \theta_k), \{T_k\}, \pi, \{j_k\}, \nu, \{P_k(\theta, d\theta')\}$  such
       that each  $P_k$  is  $\pi_k$ -reversible.
Draw  $k' \sim j_k$ 
if  $k' = k$  then
| Draw  $\theta'_k \sim P_k(\theta_k, d\theta'_k)$  and return  $(k, \theta'_k)$ 
else
|  $z_k \leftarrow T_k(\theta_k)$ 
| if  $n_{k'} > n_k$  then
| |  $w_k \leftarrow n_{k'} - n_k$ 
| | Draw  $u \sim \otimes_{w_k} \nu$ 
| |  $g_u \leftarrow (\otimes_{w_k} \nu)(u), g'_u \leftarrow 1$ 
| |  $z_{k'} \leftarrow \bar{h}_{k,k'}(z_k, u)$ 
| else if  $n_{k'} < n_k$  then
| |  $w_{k'} \leftarrow n_k - n_{k'}$ 
| |  $(z_{k'}, u) \leftarrow \bar{h}_{k,k'}^{-1}(z_k)$ 
| |  $g_u \leftarrow 1, g'_u \leftarrow (\otimes_{w_{k'}} \nu)(u)$ 
| else
| |  $z_{k'} \leftarrow \bar{h}_{k,k'}^{-1}(z_k)$ 
| |  $g_u \leftarrow 1, g'_u \leftarrow 1$ 
| |  $\theta'_{k'} \leftarrow T_{k'}^{-1}(z_{k'})$ 
| |  $x' \leftarrow (k', \theta'_{k'})$ 
| |  $\alpha \leftarrow 1 \wedge \frac{\pi(x')}{\pi(x)} \frac{j_{k'}(k)}{j_k(k')} \frac{g'_u}{g_u} \left| J_{T_k}(\theta_k) \right| \left| J_{T_{k'}}(\theta'_{k'}) \right|^{-1}$ 
| Draw  $V \sim \mathcal{U}(0, 1)$ 
| if  $\alpha > V$  then return  $x'$ 
| else return  $x$ 

```

The following proposition provides a formal justification for the proposed approach, showing that when exact transports are used, (2) reduces to a form depending only on the marginal posterior model probabilities (i.e., there is a marginalizing effect).

Proposition 1. Suppose that RJMCMC proposals are of the form described in (5), and for each $k \in \mathcal{K}$, satisfy $T_k \# \pi_k = \otimes_{n_k} \nu$. Then, (2) reduces to

$$\alpha(x, x') = 1 \wedge \frac{\pi(k')}{\pi(k)} \frac{j_{k'}(k)}{j_k(k')}. \quad (7)$$

Corollary 1. Provided the conditions of Proposition 1 are satisfied, choosing $\{j_k\}$ such that

$$\pi(k') j_{k'}(k) = \pi(k) j_k(k'), \quad \forall k, k' \in \mathcal{K}, \quad (8)$$

leads to a rejection-free proposal.

A natural choice when allowing *global* moves, i.e., $j_k(k') > 0$ for all $k, k' \in \mathcal{K}$ that solves the above detailed balance condition (8), is simply to choose $j_k(k') = \pi(k')$ for all $k \in \mathcal{K}$. Of course, in practice the marginal probabilities $\pi(k)$ are typically unknown. However, the above observation is potentially instructive as to what a good choice may be if some approximation of model probabilities is available.

Transdimensional RJMCMC proposals that employ *exact* transport maps to and from a reference distribution result in the same acceptance probabilities as performing *marginal* MCMC on the model indicator space and thus are in some sense an optimal proposal for general RJMCMC settings.

4. NUMERICAL EXPERIMENTS

For all experiments, we use masked autoregressive transforms with rational quadratic splines (RQMA) (Durkan et al., 2020). Prior to the RQMA transforms, a fixed (i.e., non-trainable) element-wise standardization is applied, which we find has a positive effect on the speed and stability of training. We also consider simpler NFs in the form of affine transformations, which are constructed using the sample mean vector and Cholesky factorization of the sample covariance matrix of the training set (Affine NFs). We highlight that in general, the choice of normalizing flow family is essentially arbitrary. However, as the proposed TRJ approach requires that the flows be evaluated in both directions (i.e., both the original flow and its inverse are needed in order to transform both to and from the reference distribution), it is computationally convenient to employ flows that are analytically tractable in both directions (as is the case for RQMA and Affine flows). Such tractability does not hold for all flows. For example, Invertible Residual Networks (Behrmann et al., 2019) require numerical fixed-point iteration to evaluate in the inverse direction. In all experiments, flows are fit for each individual k -conditional target using a set of N training samples obtained via a pilot run of either MCMC, Sequential Monte Carlo samplers (Del Moral et al., 2006), or exact sampling. The performance of transdimensional proposals is benchmarked in two ways, discussed below.

Model Probability (Running Estimates). One way to benchmark the performance of an RJMCMC chain is to visualize the running estimates of the marginal posterior probability of a given model k as the RJMCMC chain progresses. On average, a chain that mixes more rapidly is expected to produce more accurate results with fewer iterations.

Bartolucci Bridge Estimator. To assess the effect of different transdimensional proposals alone (as opposed to in combination with within-model moves), we employ the

Bayes factor bridge-sampling estimator of [Bartolucci et al. \(2006\)](#), Eq.16). The estimator relies on constructing a Rao-Blackwellized bridge sampling approach. An advantage of the approach as originally introduced is that the estimators can be computed from stored values obtained from a standard RJMCMC run. Specifically, the Bayes factor $B_{k,k'}$ (ratio of marginal likelihoods) is estimated via

$$\hat{B}_{k,k'} = \frac{N_{k'}^{-1} \sum_{i=1}^{N_{k'}} \alpha'_i}{N_k^{-1} \sum_{i=1}^{N_k} \alpha_i}, \quad (9)$$

where $N_{k'}$ and N_k are the number of proposed moves from model k' to k , and from k to k' , respectively in the run of the chain. The α and α' terms are simply the acceptance probabilities corresponding to those proposals, respectively. Rather than use the output of an RJMCMC run, a straightforward alternative when one has samples from the individual conditionals π_k (not necessarily obtained via RJMCMC) is the following. First, for each sample from each target, make one proposal of RJMCMC (inclusive of the random choice of model to jump to). Then, treat each of these as if they were a proposal obtained from a standard RJMCMC chain, used together to compute estimators of the form (9). In the special case when prior model probabilities are uniform, it is straightforward to convert estimators of Bayes factors to estimators of model probabilities ([Bartolucci et al., 2006](#)) via

$$\hat{\pi}(k) = \hat{B}_{j,k}^{-1} \left(1 + \sum_{i \in \mathcal{K} \setminus \{j\}} \hat{B}_{i,j} \right)^{-1}, \quad (10)$$

for arbitrary $j \in \mathcal{K}$. In the sequel, the combination of multiple estimators of the form (9) within (10) is referred to as the *Bartolucci bridge estimator* (BBE) of model probabilities. We expect the BBE to exhibit the lowest variance for the best across-model proposal type when computed using the same input set of samples from all models (as well as the same model jump probabilities $\{j_k\}$). When evaluating performance using the BBE, we generate an *additional* set of N samples (for each model k) that are independent of the training samples. This set is referred to as the *evaluation* set. One proposal of RJMCMC is made using each of these evaluation samples to obtain the BBE (9) terms within the estimator of model probabilities (10).

4.1 Illustrative Example

To construct an illustrative example where the exact TMs are known, we use the (element-wise) inverse sinh-arcsinh (SAS) transformation ([Jones and Pewsey, 2009](#)),

$$S_{\epsilon, \delta}(\cdot) = \sinh(\delta^{-1} \odot (\sinh^{-1}(\cdot) + \epsilon)),$$

where $\epsilon \in \mathbb{R}^n$, and $\delta \in \mathbb{R}_+^n$. For an $n \times n$ matrix L, define a transform $T(\mathbf{Z})$ where

$$T(\mathbf{Z}) = S_{\epsilon, \delta}(L\mathbf{Z}), \quad (11)$$

and $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}_n, I_{n \times n})$. The probability density function for the transformed variable $\boldsymbol{\theta} = T(\mathbf{Z})$ is

$$p_{\epsilon, \delta, L}(\boldsymbol{\theta}) = \phi_{LL^\top} \left(S_{\epsilon, \delta}^{-1}(\boldsymbol{\theta}) \right) |J_{S_{\epsilon, \delta}^{-1}}(\boldsymbol{\theta})|,$$

where $S_{\epsilon, \delta}^{-1}(\cdot) = \sinh(\delta \odot \sinh^{-1}(\cdot) - \epsilon)$. We consider a transdimensional target consisting of two models of dimension $n_1 = 1$ and $n_2 = 2$. We assign a probability of 1/4 for model $k = 1$ with parameters $\boldsymbol{\theta}_1 = (\theta_1^{(1)})$ and model probability of 3/4 for $k = 2$ with parameters $\boldsymbol{\theta}_2 = (\theta_2^{(1)}, \theta_2^{(2)})$. The target of interest for this example is

$$\pi(k, \boldsymbol{\theta}_k) = \begin{cases} \frac{1}{4} p_{\epsilon_1, \delta_1, 1}(\boldsymbol{\theta}_1), & k = 1, \\ \frac{3}{4} p_{\epsilon_2, \delta_2, 2}(\boldsymbol{\theta}_2), & k = 2, \end{cases} \quad (12)$$

where

$$\begin{aligned} \epsilon_1 &= -2, & \delta_1 &= 1, \\ \epsilon_2 &= (1.5, -2), & \text{and } \delta_2 &= (1, 1.5), \end{aligned} \quad (13)$$

and L is a lower-triangular matrix such that

$$LL^\top = \begin{bmatrix} 1 & 0.99 \\ 0.99 & 1 \end{bmatrix}.$$

By construction, for a reference distribution chosen to be a multidimensional standard Gaussian of appropriate dimension, an exact transport is given by the function

$$T^{-1}(\cdot) = L^{-1} S_{\epsilon, \delta}^{-1}(\cdot). \quad (14)$$

Note that the choice of $j_k(k') = \frac{1}{4}\mathbb{I}_{\{k'=1\}} + \frac{3}{4}\mathbb{I}_{\{k'=2\}}$, for $k \in \{1, 2\}$, i.e., the mixture weights, correspond to a rejection-free proposal when the exact transports of the form in (14) are used. Both the affine and RQMA NF are trained on a set of $N = 5 \cdot 10^4$ samples obtained via exact simulation for each of the two targets. Figure 2 depicts a visualization of the proposal distributions (including those obtained using an exact transport map), and Figure 3 visualizes the running estimate of model probabilities for each proposal type, where $j_k(k')$ is set to the mixture weights as above. On average, the accuracy of the running model probability estimates is best for the exact TM, followed by RQMC NF, which provides a better approximation of the exact TM compared to Affine.

4.2 Bayesian Factor Analysis Example

Here we consider the factor analysis (FA) model from [Lopes and West \(2004\)](#). Factor analysis assumes the model $\mathbf{Y} \sim \mathcal{N}(\mathbf{0}_6, \Sigma_{6 \times 6})$ where the covariance matrix takes the form $\beta_k \beta_k^\top + \Lambda$, where Λ is a 6×6 positive diagonal matrix, β_k is a $6 \times k$ lower-triangular matrix with a positive diagonal, k is the number of factors, and the total number of parameters is $6(k+1) - k(k-1)/2$. We model monthly exchange rates of six currencies relative to the British pound, spanning January 1975 to December 1986 ([West and Harrison, 1997](#)) as 143 realizations, denoted as $\mathbf{y}_i \in \mathbb{R}^6$ for

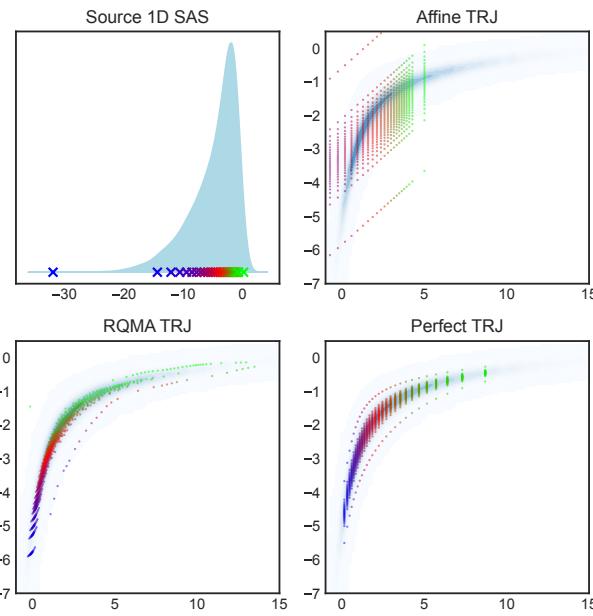


Figure 2: Systematic draws from conditional target $\pi(x_1|k = 1)$ defined in (12) are transported from $(1, \theta_1) \in \mathcal{K} \times \mathbb{R}^1$ (top left) to $(2, (\theta_1, \theta_2)) \in \mathcal{K} \times \mathbb{R}^2$ via proposals using: (1) approximate affine (top right), (2) approximate RQMA-NF (bottom left), and (3) perfect (bottom right) TMs. The auxiliary variables in the proposals are also drawn systematically (30 for each point in the source distribution).

$i = 1, \dots, 143$. For this example, we are interested in $k = 2$ or 3 factors, yielding parameter spaces of dimensions 17 and 21 respectively. Following Lopes and West (2004), for each $\beta_k = [\beta_{ij}]$ with $i = 1, \dots, 6$, $j = 1, \dots, k$, the priors are

$$\begin{aligned} \beta_{ij} &\sim \mathcal{N}(0, 1), \quad i < j, \\ \beta_{ii} &\sim \mathcal{N}_+(0, 1), \\ \Lambda_{ii} &\sim \mathcal{IG}(1.1, 0.05), \end{aligned} \quad (15)$$

where \mathcal{N}_+ denotes the positive half-normal distribution and \mathcal{IG} the inverse gamma distribution. We are interested in the posterior of $\theta_k = (\beta_k, \Lambda)$ for $k = 2$ or 3 factors. Via Bayes' Theorem, the posterior is

$$\pi(k, \theta_k | \mathbf{y}) \propto p(k)p(\beta_k | k)p(\Lambda) \prod_{i=1}^{143} \phi_{\beta\beta^\top + \Lambda}(\mathbf{y}_i), \quad (16)$$

where $\mathbf{y} = (\mathbf{y}_1, \dots, \mathbf{y}_{143})$. In this example, we compare TRJ proposals to the original RJMCMC proposal from Lopes and West (2004). The form of the Lopes and West (2004) RJMCMC proposal is an independence proposal trained on the conditional posterior of each model. Write μ_{β_k} and B_k as the posterior mean and covariance matrix

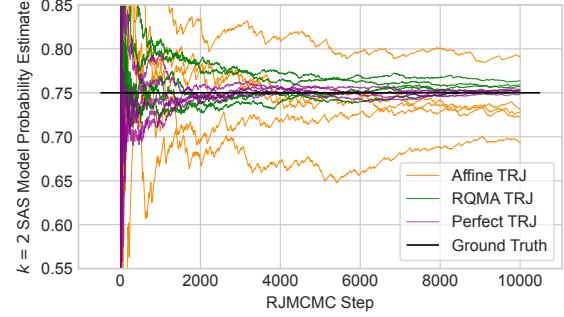


Figure 3: Running estimates of the model probabilities for the $k = 2$ component of the Sinh-Arcsinh target (12). Proposals are all Algorithm 1 with input approximate TMs (1) Affine, (2) RQMA-NF, (3) Perfect (14). Five chains on each proposal type are depicted, where alternating within-model proposals are a simple normal random walk.

of β_k , respectively. Defining $\theta_k = (\beta_k, \Lambda)$, the independence proposal is $q_k(\theta_k) = q_k(\beta_k) \prod_{i=1}^6 q_k(\Lambda_{ii})$, where for $k \in \mathcal{K}$, $q_k(\beta_k) = \mathcal{N}(\mu_{\beta_k}, 2B_k)$, and $q_k(\Lambda_{ii}) = \mathcal{IG}(18, 18v_{k,i}^2)$ where each $v_{k,i}^2$ is chosen to be an estimate of the marginal mode of Λ_{ii} for model k obtained from the training samples.

To obtain training and evaluation sets of samples from (16), we use the No-U-Turn Sampler (NUTS) (Hoffman and Gelman, 2014) as implemented in PyMC (Salvatier et al., 2016) for the three-factor model, and static temperature-annealed sequential Monte Carlo (SMC) (Del Moral et al., 2006) for the two-factor model. The number of training and evaluation samples is kept equal, and the experiments consider both $N = 2 \cdot 10^3$ and $N = 1.6 \cdot 10^4$ for each. Ten pairs of training and evaluation sample sets are generated. Each evaluation process produces a single BBE estimate, and this is conducted 100 times per training/evaluation pair, yielding a total of 10^3 BBE estimates.

Figure 4 visualizes a comparison of the running estimate of the two-factor posterior model probability for RJMCMC chains that use the independence proposal and the RQMA-NF TRJ proposal against the ground truth value of 0.88 as reported in Lopes and West (2004). Each chain was run for 10^5 steps with an RJMCMC proposal and a within-model random walk proposal used at each step in sequence. Figure 5 visualizes the variability of model probability estimates obtained from the BBE. Both figures demonstrate the superior performance of the TRJ sampler approach with RQMA flows compared to the standard approach of Lopes and West (2004). Figure 6 visualizes the transdimensional proposals with respect to selected bivariate marginals.

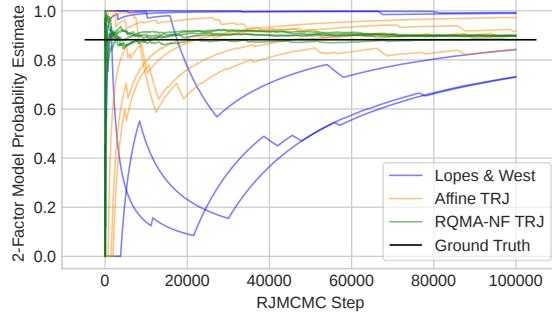


Figure 4: Running estimates of the model probabilities for the two-factor model. Proposal types are the [Lopes and West \(2004\)](#) independence proposal, and Affine and RQMA-NF proposals using Algorithm 1. Five chains on each proposal type are depicted, where alternating within-model proposals are not shown.

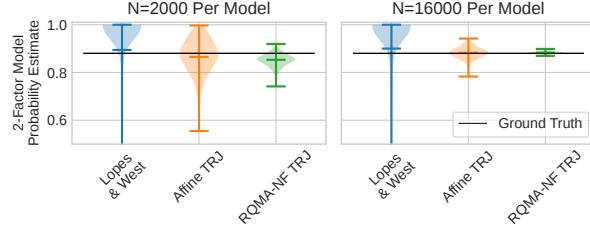


Figure 5: Violin plot for the Factor Analysis example, showing the variability of the two-factor model probability estimates in the case where only the two-factor and three-factor models are compared. Model probability estimates are obtained via the BBE.

5. CONDITIONAL TRANSPORT PROPOSALS

So far, we have introduced a proposition for rejection-free RJMCMC moves based on perfect TMs and model probabilities, and an approach to approximate both of the unknown quantities. However, there is one clear drawback to this approach: larger model spaces, such as those encountered in variable selection problems, would still require k individual approximate TMs to be trained. This section explores how to overcome the above problem by modifying the sampling problem appropriately via a *saturated space* approach, so one can obtain the approximate TMs by training a single *conditional* NF (e.g., [Winkler et al. \(2019\)](#)). The general idea of the approach is that the model indicator is given as an input (context) to a function that outputs the parameters of a flow (or sequence thereof). Such an approach implicitly defines a family of flows across *all* models, yet only requires training once. The benefit is thus primarily computational. We employ the notation $\tilde{T}(\cdot|k)$ for the flow obtained by providing the conditional normalizing

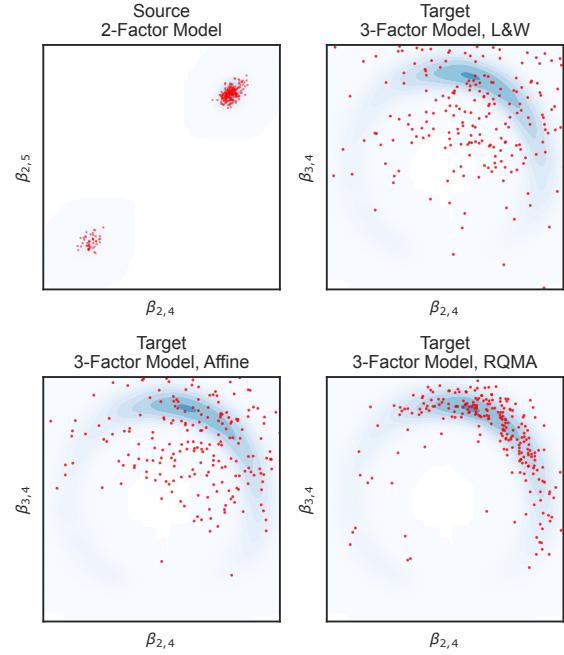


Figure 6: A visualization (using selected bivariate plots) of the proposal from points on the 2-factor model (*top-left*) to proposed points on the 3-factor model for each proposal type: [Lopes and West \(2004\)](#) independence RJMCMC proposal (*top right*); TRJ with Affine map (*bottom left*); TRJ with RQMA-NF map (*bottom right*). Ground truth kernel density estimate was made using $5 \cdot 10^4$ SMC particles from [South et al. \(2019\)](#).

flow model with context $k \in \mathcal{K}$.

The dimension-saturation approach, originally considered in [Brooks et al. \(2003\)](#), is an equivalent formulation of RJMCMC that involves an augmented target. Writing the maximum model dimension as n_{\max} (assumed to be finite) and recalling that $\mathbf{x} = (\boldsymbol{\theta}_k, k)$, we define our augmented target to be

$$\tilde{\pi}(\tilde{\mathbf{x}}) = \pi(\mathbf{x})(\otimes_{n_{\max}-n_k} \nu)(\mathbf{u}_{\sim k}), \quad (17)$$

where $\tilde{\mathbf{x}} = (k, \boldsymbol{\theta}, \mathbf{u}_{\sim k})$, and “ $\sim k$ ” identifies that the auxiliary variable is of dimension $n_{\max} - n_k$. In this setting, one can obtain approximate TMs by training a single conditional NF with the conditioning vector being the model index $k \in \mathcal{K}$ (further details are provided in the supplement). The associated proposals analogous to those in (5) are constructed as $(\boldsymbol{\theta}'_{k'}, \mathbf{u}_{\sim k'}) = c_{k'}^{-1} \circ \tilde{T}^{-1}(\cdot|k') \circ \tilde{T}(\cdot|k) \circ c_k(\boldsymbol{\theta}_k, \mathbf{u}_{\sim k})$, where $k' \sim j_k$, and c_k is simply concatenation. The above yields Algorithm 2.

5.1 Variable Selection in Robust Regression Example

Here, a variable selection example is provided as a proof of concept for the conditional transport approach just described. For this example, we seek a challenging Bayesian

Algorithm 2: CTRJ: Conditional Transport RJMCMC

```

input :  $\tilde{x} = (k, \theta_k, \mathbf{u}_{\sim k}), \tilde{T}, \tilde{\pi}, \{j_k\}, \nu, \{P_k(\theta, d\theta')\}$ 
Draw  $k' \sim j_k$ 
if  $k' = k$  then
    Draw  $\theta'_k \sim P_k(\theta_k, d\theta'_k)$ 
    return  $(k, \theta'_k, \mathbf{u}_{\sim k})$ 
else
     $\xi_k \leftarrow c_{k,k'}(\theta_k, \mathbf{u}_{\sim k})$ 
     $z \leftarrow \tilde{T}(\xi_k | k)$ 
     $\xi_{k'} \leftarrow \tilde{T}^{-1}(z | k')$ 
     $(\theta'_{k'}, \mathbf{u}_{\sim k'}) \leftarrow c_{k',k}^{-1}(\xi_{k'})$ 
     $\tilde{x}' \leftarrow (k', \theta'_{k'}, \mathbf{u}_{\sim k'})$ 
     $\alpha \leftarrow 1 \wedge \frac{\tilde{\pi}(\tilde{x}') j_{k'}(k)}{\tilde{\pi}(\tilde{x}) j_k(k')} \left| J_{\tilde{T}}(\theta_k | k) \right| \left| J_{\tilde{T}}(\theta'_{k'} | k') \right|^{-1}$ 
    Draw  $V \sim \mathcal{U}(0, 1)$ 
    if  $\alpha > V$  then return  $\tilde{x}'$ 
    else return  $\tilde{x}$ 

```

model choice problem where conventional methods do not perform well. The interest is in realizations of a random response variable $\mathbf{Y} = (Y_1, \dots, Y_{80})^\top$, where for each Y_i , we have associated covariates $(x_{i0}, x_{i1}, x_{i2}, x_{i3})$ where $x_{i0} = 1$. Writing \mathbf{X} for the associated 80×4 design matrix, our model is $\mathbf{Y} = \mathbf{X}\beta + \epsilon$, where $\beta = (\beta_0, \dots, \beta_3)$ is the vector of unknown regression parameters and $\epsilon = (\epsilon_1, \dots, \epsilon_{80})$ is the residual error term. We model ϵ as a vector of independently and identically distributed random variables, each distributed as a mixture of a standard normal variable and a zero-mean normal variable with variance 10^2 . To ease the exposition, we adjust the notation slightly to accommodate the Bayesian variable selection literature. Let k be an indicator vector where the i th position of k equals 1 if the variable β_i is included in model k and 0 otherwise — e.g., the model indexed by $k = (1, 0, 1, 1)$ corresponds to the model with parameters $\theta_k = (\beta_0, \beta_2, \beta_3)$. We consider a model space of cardinality of 4, composed of all models of the form $k = (1, k_1, k_2, k_2)$ where $k_i \in \{0, 1\}$ for $i = 1, 2$. These models will include or exclude the parameters β_2 and β_3 jointly as commonly considered in group variable selection (e.g. Xu and Ghosh (2015)). The prior distributions are specified as

$$\begin{aligned} k_i &\sim \text{Bernoulli}(1/2), \quad i \in \{1, 2\}, \quad \text{and} \\ \beta_i &\sim \mathcal{N}(0, 10^2), \quad i \in \{0, 1, 2, 3\}. \end{aligned} \quad (18)$$

The target distribution π is then the posterior distribution over the set of models and regression coefficients, obtained via Bayes' Theorem.

The dataset for the example was simulated from the model $k = (1, 1, 0, 0)$, which has corresponding parameters $\theta_k = (\beta_0, \beta_1)$. We simulated half of the observations of the response using $(\beta_0, \beta_1) = (1, 1)$ and half using $(\beta_0, \beta_1) = (6, 1)$. This choice was made to induce further challenging

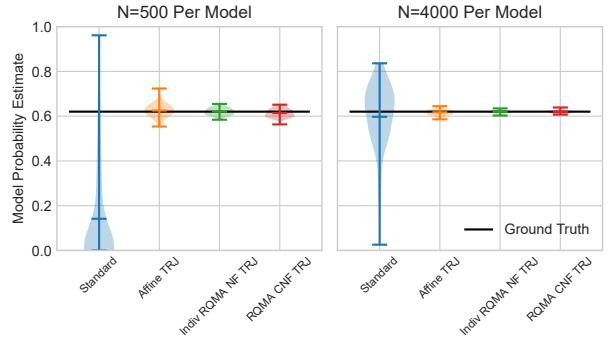


Figure 7: Violin plot (for the example in Section 5.1) showing the variability of the $k = (1, 1, 1, 1)$ model probability estimate for each proposal type using the Bartolucci Estimator vs a ground truth given by individual SMC ($N = 5 \cdot 10^4$). Here, Indiv RQMA-NF TRJ corresponds to Algorithm 1 with four *individual* normalizing flows, whereas RQMA-CNF TRJ corresponds to Algorithm 2 using a single *conditional* normalizing flow.

multi-modal features in the resulting posterior. The elements of the matrix \mathbf{X} were simulated independently from a standard normal distribution, and the elements of the residual vector ϵ were simulated from a normal distribution with a mean of zero and a variance of 25. See the supplementary material for visualizations of the multimodal posterior behavior. The design of each proposal is as follows. The “standard” proposal adopts the *independent auxiliary variable* method of Brooks et al. (2003), which is equivalent to using the augmented target (17) where ν is the prior distribution for β_i , i.e., $\mathcal{N}(0, 10^2)$. The standard proposals are deterministic in the sense that no auxiliary variables are drawn, but rather the i th block of parameters is reinstated to its previous value when $k_i = 1$. The TRJ affine and RQMA-NF proposals apply Algorithm 1 with approximate affine, and RQMA-NF TMs respectively trained on samples from the conditional target distributions. The CTRJ proposal adopts the augmented target (17) where ν is set to the reference univariate density as described in Section 5, and proceeds as per Algorithm 2. The empirical performance of each proposal is examined for training and evaluation sets via the BBE. The number of training and evaluation samples was kept the same, and we considered both $N = 500$ and $N = 4000$ sampled from a single SMC run. The process of training the flow and estimation of model probability via BBE was repeated eighty times.

Figure 7 visualizes the variability of the $k = (1, 1, 1, 1)$ model probability estimate using the same approach as in Section 4.2. The figure shows the vastly superior accuracy and variability of model probability estimates using the TRJ proposals relative to the independent auxiliary method. The performance of the conditional RQMA flow appears

similar to that of the individual RQMA and affine flows but only requires fitting a single conditional flow as opposed to a flow per model.

Numerical examples demonstrate a clear benefit to using transport RJMCMC proposals in cases where the transport maps are *approximate*, which is typically the best one can hope for in practice.

6. DISCUSSION

The work introduces a method to design across-model RJMCMC proposals using ideas from measure transport. The approach was shown to have desirable properties in the case where exact TMs are used, while still providing good performance in the case where the TMs are approximate. Although some form of initial effort to obtain samples from each conditional target is somewhat common when calibrating proposals (e.g., as in Lopes and West (2004); Hastie (2005)), this is still a weakness of TRJ proposals, along with the computational effort required to train the approximate transport maps. The use of conditional NFs may mitigate the latter somewhat, but a promising avenue for future research would be some form of amortized approach in model spaces where there is some shared structure (and thus perhaps samples from only *some* models would suffice). Moreover, the effort may be justified in settings where likelihoods are particularly expensive, gradient-based methods are unavailable (e.g., implicit models), or by and large the gain in performance appropriately justifies the effort in obtaining the approximate TMs. Finally, while we mainly repurposed the BBE to benchmark cross-model proposal quality, the results seem promising, and it may be interesting to explore its use in lieu of standard RJMCMC if efforts to obtain approximate TMs have already occurred.

Acknowledgements

Christopher Drovandi acknowledges support from an Australian Research Council Discovery Project (DP200102101), which Laurence Davies was also partially supported by.

References

- Al-Awadhi, F., Hurn, M., and Jennison, C. (2004). Improving the acceptance rate of reversible jump MCMC proposals. *Statistics & Probability Letters*, 69(2):189–198.
- Arbel, M., Matthews, A., and Doucet, A. (2021). Annealed flow transport Monte Carlo. In *International Conference on Machine Learning*, pages 318–330. PMLR.
- Bartolucci, F., Scaccia, L., and Mira, A. (2006). Efficient Bayes factor estimation from the reversible jump output. *Biometrika*, 93(1):41–52.
- Behrman, J., Grathwohl, W., Chen, R. T. Q., Duvenaud, D., and Jacobsen, J.-H. (2019). Invertible residual networks. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 573–582. PMLR.
- Brooks, S. P., Giudici, P., and Roberts, G. O. (2003). Efficient construction of reversible jump Markov chain Monte Carlo proposal distributions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(1):3–39.
- Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436.
- Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. (2019). Neural Spline Flows. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. (2020). nflows: normalizing flows in PyTorch. <https://zenodo.org/record/4296287>.
- Fan, Y., Peters, G. W., and Sisson, S. A. (2008). Automating and evaluating reversible jump MCMC proposal distributions. *Statistics and Computing*, 19(4):409.
- Farr, W. M., Mandel, I., and Stevens, D. (2015). An efficient interpolation technique for jump proposals in reversible-jump Markov chain Monte Carlo calculations. *Royal Society Open Science*, 2(6):150030.
- Gabrié, M., Rotkoff, G. M., and Vanden-Eijnden, E. (2022). Adaptive Monte Carlo augmented with normalizing flows. *Proceedings of the National Academy of Sciences*, 119(10):e2109420119.
- Gagnon, P. (2021). Informed reversible jump algorithms. *Electronic Journal of Statistics*, 15(2):3951–3995.
- Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732.
- Green, P. J. and Mira, A. (2001). Delayed rejection in reversible jump Metropolis-Hastings. *Biometrika*, 88(4):1035–1053.
- Hastie, D. (2005). *Towards Automatic Reversible Jump Markov Chain Monte Carlo*. PhD thesis, University of Bristol.
- Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109.
- Hoffman, M., Sountsov, P., Dillon, J. V., Langmore, I., Tran, D., and Vasudevan, S. (2018). NeuTra-lizing

bad geometry in Hamiltonian Monte Carlo using neural transport. In *1st Symposium on Advances in Approximate Bayesian Inference*.

Hoffman, M. D. and Gelman, A. (2014). The No-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *The Journal of Machine Learning Research*, 15(1):1593–1623.

Jones, M. C. and Pewsey, A. (2009). Sinh-arcsinh distributions. *Biometrika*, 96(4):761–780.

Karagiannis, G. and Andrieu, C. (2013). Annealed importance sampling reversible jump MCMC algorithms. *Journal of Computational and Graphical Statistics*, 22(3):623–648.

Lopes, H. F. and West, M. (2004). Bayesian model assessment in factor analysis. *Statistica Sinica*, 14(1):41–67.

Papamakarios, G., Nalisnick, E. T., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64.

Parno, M. D. and Marzouk, Y. M. (2018). Transport map accelerated Markov chain Monte Carlo. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):645–682.

Salvatier, J., Wiecki, T. V., and Fonnesbeck, C. (2016). Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, 2:e55.

South, L. F., Pettitt, A. N., and Drovandi, C. (2019). Sequential Monte Carlo samplers with independent Markov chain Monte Carlo proposals. *Bayesian Analysis*, 14(3):753–776.

West, M. and Harrison, J. (1997). *Bayesian forecasting and dynamic models*. Springer series in statistics. Springer, New York, 2nd ed edition.

Winkler, C., Worrall, D., Hoogeboom, E., and Welling, M. (2019). Learning likelihoods with conditional normalizing flows. *arXiv preprint arXiv:1912.00042*.

Xu, X. and Ghosh, M. (2015). Bayesian variable selection and estimation for group lasso. *Bayesian Analysis*, 10(4):909–936.

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

SUPPLEMENTARY MATERIAL

A. PROOF OF PROPOSITION 1

Proof. Here, we employ the shorthand notation $\nu_n := \otimes_n \nu$ and $\nu_n(\cdot) := (\otimes_n \nu)(\cdot)$ for measure and probability density functions, respectively. For a diffeomorphism $(\boldsymbol{\theta}'_{k'}, \mathbf{u}_{k'}) = h_{k,k'}(\boldsymbol{\theta}_k, \mathbf{u}_k)$, auxiliary variables $\mathbf{u}_k \sim g_{k,k'}$, $\mathbf{u}_{k'} \sim g_{k',k}$ of respective dimensions w_k and $w_{k'}$, if the dimension matching requirement $n_{k'} + w_{k'} = n_k + w_k$ is satisfied, then the standard choice (Green, 1995) for the acceptance probability that satisfies detailed balance is

$$\alpha(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{\pi(\mathbf{x}') j_{k'}(k) g_{k',k}(\mathbf{u}_{k'})}{\pi(\mathbf{x}) j_k(k') g_{k,k'}(\mathbf{u}_k)} |J_{h_{k,k'}}(\boldsymbol{\theta}_k, \mathbf{u}_k)|. \quad (19)$$

First, consider the case that $n_{k'} > n_k$. The proposed move from index k to k' is

$$\boldsymbol{\theta}'_{k'} = h_{k,k'}(\boldsymbol{\theta}_k, \mathbf{u}_k) = T_{k'}^{-1}(\bar{h}_{k,k'}(T_k(\boldsymbol{\theta}_k), \mathbf{u}_k)), \quad \text{where } \mathbf{u}_k \sim \nu_{w_k}. \quad (20)$$

In this setting, we have $\mathbf{u}_{k'} = \emptyset$, $w_{k'} = 0$, and $g_{k,k'} = \nu_{w_k}$. Note that $g_{k,k'} = \nu_0$ (i.e., an “empty” distribution), so $g_{k',k}(\mathbf{u}_{k'}) = 1$. Further, the dimension-matching requirement reduces to $n_{k'} = n_k + w_k$. Using the properties that $|J_{f_1 \circ f_2}| = |J_{f_1}| \cdot |J_{f_2}|$ and $|J_{f^{-1}}| = |J_f|^{-1}$, along with the assumption that $\bar{h}_{k,k'}$ is a volume-preserving map, i.e., $|J_{\bar{h}_{k,k'}}| = 1$, we have that

$$|J_{h_{k,k'}}(\boldsymbol{\theta}'_{k'}, \mathbf{u}_{k'})| = |J_{T_k}(\boldsymbol{\theta}_k)| \cdot |J_{\bar{h}_{k,k'}}(T_k(\boldsymbol{\theta}'_{k'}), \mathbf{u}_{k'})| \cdot |J_{T_{k'}}(\boldsymbol{\theta}'_{k'})|^{-1} = \frac{|J_{T_k}(\boldsymbol{\theta}_k)|}{|J_{T_{k'}}(\boldsymbol{\theta}'_{k'})|}. \quad (21)$$

Making the above substitutions into (19), and employing the conditional factorization identity $\pi(\mathbf{x}) = \pi(k)\pi_k(\boldsymbol{\theta}_k)$ to both $\pi(\mathbf{x})$ and $\pi(\mathbf{x}')$ yields

$$\alpha(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{j_{k'}(k) \pi(k')}{j_k(k') \pi(k)} \left(\frac{\pi_{k'}(\boldsymbol{\theta}'_{k'}) |J_{T_k}(\boldsymbol{\theta}_k)|}{\nu_{w_k}(\mathbf{u}_k) \pi_k(\boldsymbol{\theta}_k) |J_{T_{k'}}(\boldsymbol{\theta}'_{k'})|} \right). \quad (22)$$

Hence, showing that the bracketed term above must always be equal to one establishes the desired result. To accomplish this, we make use of several identities. First, note that by hypothesis, T_k and $T_{k'}$ are diffeomorphic and satisfy $T_k \sharp \pi_k = \nu_{n_k}$ and $T_{k'} \sharp \pi_{k'} = \nu_{n_{k'}}$. Thus, by the standard change of variables theorem, we obtain the identities

$$\pi_k(\boldsymbol{\theta}_k) = \nu_{n_k}(T_k(\boldsymbol{\theta}_k)) |J_{T_k}(\boldsymbol{\theta}_k)|, \quad \text{and} \quad \pi_{k'}(\boldsymbol{\theta}'_{k'}) = \nu_{n_{k'}}(T_{k'}(\boldsymbol{\theta}'_{k'})) |J_{T_{k'}}(\boldsymbol{\theta}'_{k'})|. \quad (23)$$

Next, by hypothesis $\bar{h}_{k,k'} \sharp \nu_{n_k+w_k} = \nu_{n_k+w_k}$, and since $\bar{h}_{k,k'}$ is a volume-preserving diffeomorphism we obtain

$$\nu_{n'_k}(\mathbf{z}) = \nu_{n'_k}(\bar{h}_{k,k'}^{-1}(\mathbf{z})). \quad (24)$$

Finally, observe

$$\bar{h}_{k,k'}^{-1} \circ T_k(\boldsymbol{\theta}'_{k'}) \underset{(20)}{=} \bar{h}_{k,k'}^{-1} \circ T_{k'} \circ T_{k'}^{-1} \circ \bar{h}_{k,k'}(T_k(\boldsymbol{\theta}_k), \mathbf{u}_k) = (T_k(\boldsymbol{\theta}_k), \mathbf{u}_k), \quad (25)$$

where the final equality above uses associativity of function composition. Using the above identities, and that $\nu_{n_{k'}} = \nu_{n_k+w_k}$, we obtain

$$\frac{\pi_{k'}(\boldsymbol{\theta}'_{k'}) |J_{T_k}(\boldsymbol{\theta}_k)|}{\nu_{w_k}(\mathbf{u}_k) \pi_k(\boldsymbol{\theta}_k) |J_{T_{k'}}(\boldsymbol{\theta}'_{k'})|} \underset{(23)}{=} \frac{\nu_{n'_k}(T_{k'}(\boldsymbol{\theta}'_{k'}))}{\nu_{n_k+w_k}(T_k(\boldsymbol{\theta}_k), \mathbf{u}_k)} \underset{(24)}{=} \frac{\nu_{n'_k}(\bar{h}_{k,k'}^{-1}(T_{k'}(\boldsymbol{\theta}'_{k'})))}{\nu_{n_k+w_k}(T_k(\boldsymbol{\theta}_k), \mathbf{u}_k)} \underset{(25)}{=} \frac{\nu_{n_k+w_k}(T_k(\boldsymbol{\theta}_k), \mathbf{u}_k)}{\nu_{n_k+w_k}(T_k(\boldsymbol{\theta}_k), \mathbf{u}_k)} = 1,$$

and the result is established for the stated case. The result is established for the case $n_k = n_{k'}$ by repeating the above argument but setting $\nu_{w_k}(\cdot) = 1$ because $w_k = 0$ in that case. Similarly, for the case where $n_{k'} < n_k$, repeat the above argument by applying the reverse move (replacing $h_{k,k'}$ with $h_{k',k} := h_{k,k'}^{-1}$), and using $\bar{h}_{k',k}^{-1} \sharp \nu_{n_k+w_k} = \nu_{n_k+w_k}$. \square

B. NORMALIZING FLOW CONFIGURATIONS

This section gives details regarding the flows used in the experiments. Write $h : \mathbb{R}^n \rightarrow [0, 1]^n$ to denote the sigmoid function (applied element-wise). Define

$$s(\mathbf{z}; \mathbf{a}, \mathbf{b}) = \mathbf{b} \odot (\mathbf{z} - \mathbf{a}), \quad \mathbf{a} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}_+^n. \quad (26)$$

The flow-based models used are of the form $T = h^{-1} \circ F \circ h \circ s$, where $F : [0, 1]^n \rightarrow [0, 1]^n$ is the composition of three (3) *masked autoregressive rational quadratic spline* transforms, implemented in the Python package `nflows`. Parameters used for each of the three transforms were 10 bins (corresponding to 11 “knot” points), the autoregressive neural networks used in each transform had two hidden layers, each with $32 \times n$ hidden features. A Stochastic Gradient Descent (SGD) algorithm is used to train the parameters of the sequence of transforms F . The transform s is fixed at the beginning of training and remains so during training, with the values of \mathbf{a} and \mathbf{b} taken to be the marginal means and reciprocal standard deviations of the data (samples), respectively, from a given π_k . It is beneficial to fix s as described above as it essentially standardizes the data prior to model training.

B.1 Conditional Normalizing Flow

The conditional approach uses the conditional transform $\tilde{T}(\cdot | k) = h^{-1} \circ F(\cdot | k) \circ h \circ s(\cdot | k)$. In the previous equation, $F(\cdot; k)$ is now a conditional normalizing flow model that, given as input a context (conditioning) variable k , outputs a flow of the same form as F in the non-conditional case. Additionally, the CNF base distribution is now a *conditional* base — an independent Gaussian whose mean and scale parameters are determined via a context k (implemented as `ConditionalDiagonalNormal` in `nflows`). Finally, $s(\cdot | k)$ is a *conditional standardization* that takes into account the known distribution of the auxiliary variables as well as when they feature for different models. Note that under the augmented target distribution, conditional on $k \in \mathcal{K}$, θ_k is distributed according to the conditional target π_k , and the auxiliary variables $\mathbf{u}_{\setminus k}$ are distributed according to $\nu_{n_{\max} - n_k}$. Rather than simply standardize the auxilliary variables (which are known to be marginally distributed according to ν) for each model, it is often straightforward to apply a perfect *Gaussianizing* transform. Write F_ν for the cumulative density function (cdf) of the reference distribution ν , and Φ^{-1} for the inverse cdf of the univariate standard normal distribution (i.e., the probit function). The generalization of the standardizing transform to the saturated state space (defined for output i) is then

$$s(\mathbf{v}|k; \{\mathbf{a}_k, \mathbf{b}_k\}_{k \in \mathcal{K}})_i = \begin{cases} b_{ki}(v_i - a_{ki}), & v_i \text{ is not an auxilliary variable for model } k, \\ \Phi^{-1} \circ F_\nu(v_i), & \text{otherwise.} \end{cases} \quad (27)$$

Similar to the unconditional case, the vectors \mathbf{b}_k and \mathbf{a}_k are fixed to be the reciprocal standard deviation and mean vectors of the data (samples) from each π_k . Optionally, one can set $\mathbf{a}_k = \mathbf{a}$ and $\mathbf{b}_k = \mathbf{b}$ for all k using the same quantities obtained from pooling the samples (this is the approach implemented). In the considered example, the reference distribution ν was taken to be a standard Gaussian, and thus the second case above reduces simply to returning v_i . We highlight that the check for which case to apply in the function above is implemented via a *masking* function that, for input $k \in \mathcal{K}, i \in \{1, \dots, n_{\max}\}$, returns one if the condition is satisfied, and zero otherwise.

C. BLOCK VARIABLE SELECTION EXAMPLE

Figure 8 depicts the distributions for each conditional target in the example. Note that each is multi-modal. However, there are overlaps in some regions of high probability mass, which could possibly assist the performance of the CTRJ proposal. Figure 9 is an extended version of Figure 7 in the paper, showing results of the Bartolucci estimator across varying number of samples N for each of the four estimated model probabilities.

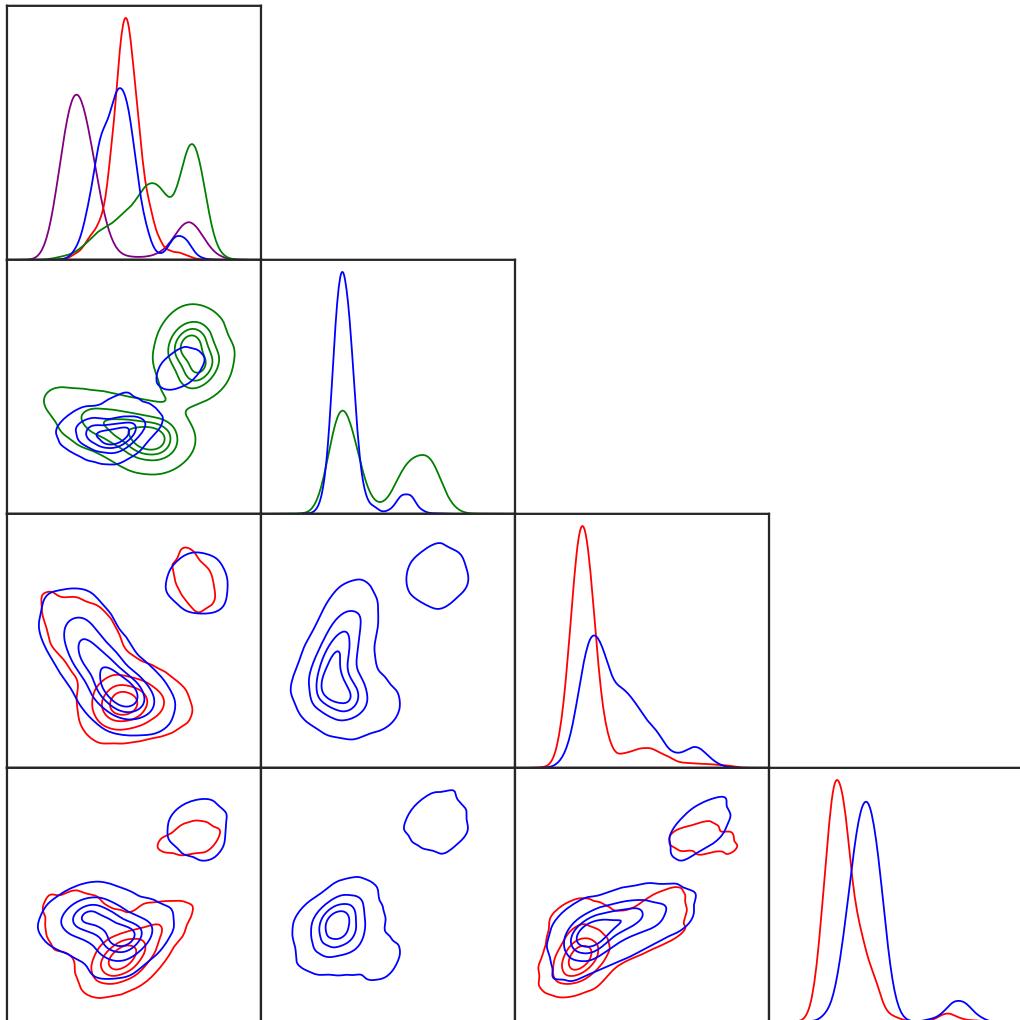


Figure 8: Pairwise plot of the conditional bivariate posterior densities in the Bayesian variable selection example. All four models feature: $k = (1, 0, 0, 0)$ (Purple), $k = (1, 1, 0, 0)$ (Green), $k = (1, 0, 1, 1)$ (Red), and $k = (1, 1, 1, 1)$ (Blue).

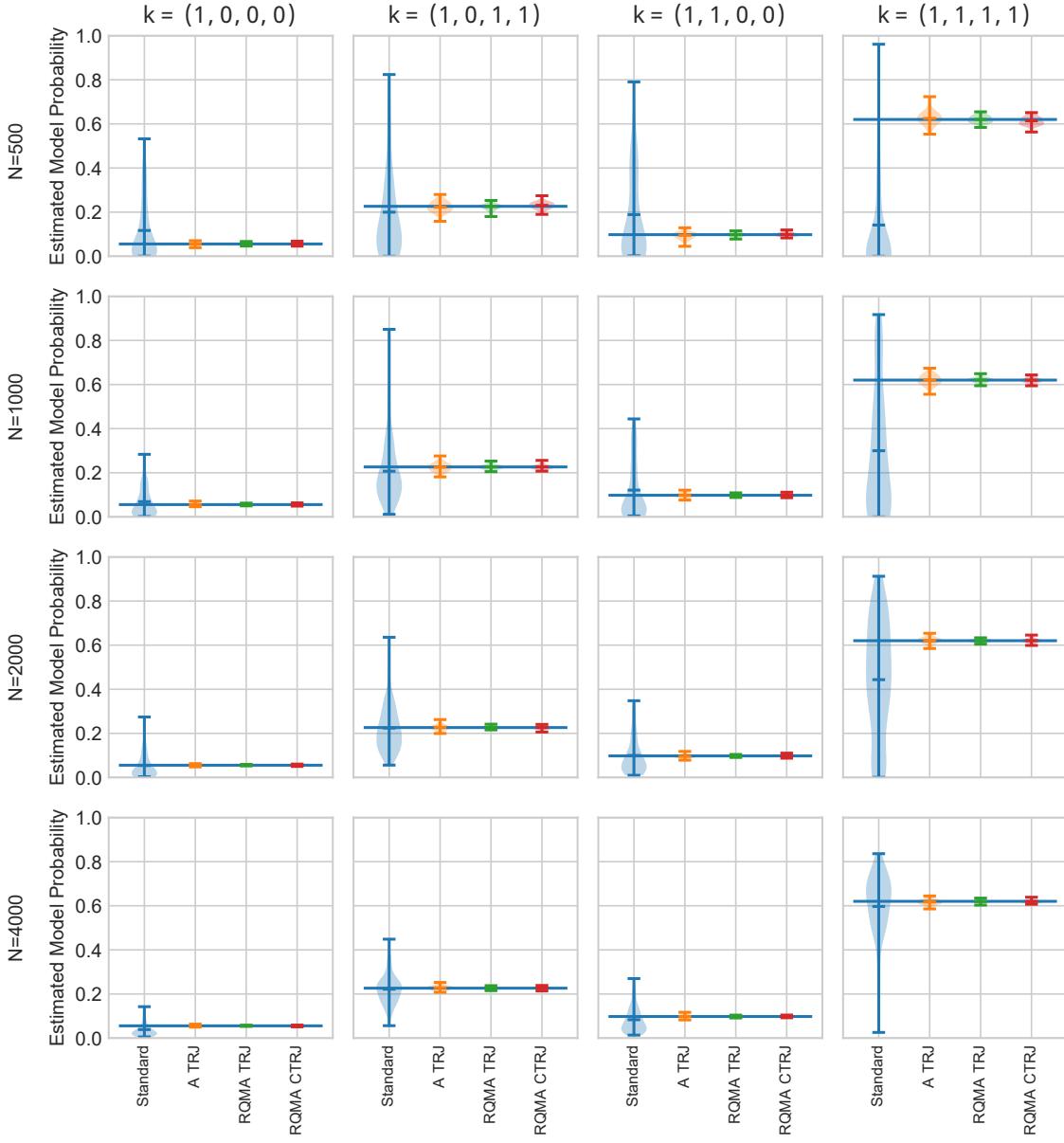


Figure 9: Violin plot showing variability of each proposal type using the Bartolucci Bridge Estimator for the Bayesian variable selection example. To obtain samples to train the approximate transport maps, each model posterior was sampled using individual SMC samplers with $N \in \{500, 1000, 2000, 8000\}$ particles with a random walk kernel. Horizontal lines indicate ground-truth values obtained using large-sample runs of individual SMC ($N = 5 \cdot 10^4$).