TRANSPORT MAP ACCELERATED-PAIS, AND APPLICATION TO INVERSE PROBLEMS ARISING FROM MULTISCALE STOCHASTIC REACTION NETWORKS

SIMON COTTER, YANNIS KEVREKIDIS, PAUL RUSSELL

Abstract. In many applications, inverse problems arise where where there are complex correlations between the different parameters which we wish to infer from data. The correlations often manifest themselves as lower dimensional manifolds on which the likelihood function is invariant, or varies very little. This can be due to trying to infer unobservable parameters, or due to sloppiness in the model which is being used to describe the data. In such a situation, standard sampling methods for characterising the posterior distribution which do not incorporate information about this structure will be highly inefficient. Moreover, most methods are inherently serial in nature, and as such are not expoiting the parallelised nature of modern computer infrastructure. In this paper, we seek to develop a method to tackle this problem, using optimal transport maps to simplify posterior distributions which are concentrated on lower dimensional manifolds.

We demonstrate the approach by considering inverse problems arising from partially observed stochastic reaction networks. In particular, we consider systems which exhibit multiscale behaviour, but for which only the slow variables in the system are observable. We demonstrate that certain multiscale approximations lead to more consistent approximations of the posterior than others.

1. Introduction. In Section 2 we show how an appropriate transport map can be constructed from importance samples which maps the posterior close to a reference Gaussian measure. In Section 6 we show how such a map can be incorporated into a sophisticated parallel MCMC infrastructure in order to accelerate mixing. In Section 7 we consider how likelihoods can be approximated using multiscale methodologies in order to carry out inference for multiscale and/or partially observed stochastic reaction networks. In Section 8 we present some numerical examples, which serve to demonstrate the increased efficiency of the described sampling methodologies, as well as investigating the posterior approximations discussed in the previous section. We conclude with a discussion in Section 9.

2. Construction of transport maps in importance sampling. In [1] the transport map was introduced to provide a transformation from the prior distribution to the posterior distribution, the idea being that one could draw a moderately sized sample from the prior distribution and use this sample to approximate a map onto the target space. Once this map was known to the desired accuracy a larger sample from the prior could be used to investigate the posterior distribution. This methodology was adapted in [4] to form a new proposal method for MH algorithms. In this case, rather than transforming a sample from the prior into a sample from the target distribution, the map transforms a sample from the posterior onto a reference space. The reference density is chosen to allow efficient proposals using a simple proposal distribution such as a Gaussian centred at the previous state. Proposed states can then be mapped back into a sample from the posterior by applying the inverse of the transport map. Proposing new states in this way allows us to make large steps around complex probability distributions. It is also feasible in this framework to assume that the reference density is close enough to a standard Gaussian that we can efficiently propose moves using a proposal distribution which is independent of the current state, e.g. choose $q(\theta) = \mathcal{N}(0, I_n)$.

In this Section we outline the methodology in [4] for coupling the target, μ_θ, with the reference distribution, μ_r, and show how the map can be constructed using a weighted sample and hence how we can incorporate the map into importance sampling schemes.

DEFINITION 2.1 ((Exact) Transport Map T). *An (exact) transport map T is a function $T\colon \mathcal{X} \to \mathbb{R}^d$ such that the pullback of the reference measure with density $\phi(\cdot)$,*

$$\tilde{\pi}(\theta) = \phi(T(\theta))|J_T(\theta)|, \tag{2.1}$$

is equal to the target density $\pi(\theta)$ for all $\theta \in \mathcal{X}$. The pullback is defined in terms of the determinant of the Jacobian of T,

$$|J_T(\theta)| = det \begin{bmatrix} \partial_{\theta_1} T_1(\theta) & \dots & \partial_{\theta_d} T_1(\theta) \\ \vdots & \ddots & \vdots \\ \partial_{\theta_1} T_d(\theta) & \dots & \partial_{\theta_d} T_d(\theta) \end{bmatrix}.$$

There are infinitely many such maps, a subset of which will be invertible. In the case that we have an exact invertible map, $T \in \mathcal{T}$ where $\mathcal{T}$ is the space of all invertible maps, we are able to draw a sample from $\pi_r = \phi$, the density of the reference distribution, which could be picked to be the standardised Gaussian distribution for example, and map these samples back onto target space using T^{-1}. These proposed samples are then distributed according to the target distribution.

DEFINITION 2.2 (Target and Reference Space). *The transport map pushes a particle from a target space $\mathcal{X}$, that is a subset of $\mathbb{R}^d$ equipped with a target measure μ_θ, onto a reference space, R, again a subset of $\mathbb{R}^d$ equipped with the reference measure μ_r.*

On target space, the proposal density is induced by the pullback of ϕ through T^{-1}. Clearly the pullback only exists when T is monotonic, i.e. has a positive definite Jacobian, and has continuous first derivatives. Not all maps satisfy these conditions, so we define a smaller space of maps, $\mathcal{T}^\uparrow \subset \mathcal{T}$ which contains all feasible maps. An exact map T is not necessarily in $\mathcal{T}^\uparrow$, so we are motivated to formulate an optimisation problem to find the map $T \in \mathcal{T}^\uparrow$ which most closely maps the target density to the reference density.

We attempt to find the deterministic coupling of two continuous probability distributions, $(\mu_\theta, \hat{\mu_r})$, such that $\hat{\mu_r} = T\mu_\theta$ where the distance between μ_r (the desired reference measure) and $\hat{\mu_r}$ (the achieved reference measure) is minimised, for $T \in \mathcal{T}^\uparrow$. As in [4] we aim to minimise the Kullback-Liebler (KL) divergence between the density of μ_θ and the pullback of the density of μ_r, i.e. the distance between $\pi(\theta)$ and $\tilde{\pi}(\theta)$. For two absolutely continuous measures with densities π_1 and π_2 respectively, the KL divergence is given by

$$D_{\mathrm{KL}}(\pi_1 \| \pi_2) = \mathbb{E}_{\pi_1}\left[\log\left(\frac{\pi_1(\theta)}{\pi_2(\theta)}\right)\right].$$

The KL divergence is not itself a norm, since it is not symmetric, i.e. $D_{\mathrm{KL}}(\pi_1\|\pi_2) \neq D_{\mathrm{KL}}(\pi_2\|\pi_1)$ in general. However, it is still a useful measure of the similarity of two probability distributions, not least since the square root of the KL-divergence is an upperbound to the Hellinger distance metric.

As in previous work in [4], when we optimise the cost function

$$C(T) = D_{\mathrm{KL}}(\pi\|\tilde{\pi}),$$

to ensure invertibility we restrict the map to be lower triangular, i.e. $\tilde{T} \in \mathcal{T}^\triangleright \subset \mathcal{T}^\uparrow$.

This lower triangular map has the form,

$$T(\theta_1, \ldots, \theta_n) = \begin{bmatrix} T_1(\theta_1) \\ T_2(\theta_1, \theta_2) \\ \vdots \\ T_n(\theta_1, \ldots, \theta_n) \end{bmatrix},$$

where $T_i \colon \mathbb{R}^i \to \mathbb{R}$. We assume that the target and reference probability densities are absolutely continuous on $\mathbb{R}^d$. Under this formulation we are guaranteed a unique invertible map $\tilde{T}$ with the property that

$$\tilde{T}\mu_\theta \approx \mu_r.$$

Relaxing the equality constraint to finding the approximate map $\tilde{T} \in \mathcal{T}^\triangle$ which minimises $C(T)$, gives us a practical route to finding a good candidate map.

2.1. The optimisation problem. With these constraints in mind we formulate the optimisation problem explicitly. The cost function is chosen to be the Kullback-Leibler divergence between the posterior density and the pullback density,

$$D_{\mathrm{KL}}(\pi \| \tilde{\pi}) = \mathbb{E}_\pi \left[\log \left(\frac{\pi(\theta)}{\tilde{\pi}(\theta)} \right) \right].$$

This divergence results in some nice properties which we will explore in the following derivation. The KL divergence is not a true metric since it is not symmetric, however it is commonly used to measure the distance between probability distributions due to it's relatively simple form, and because it provides a bound for the square of the Hellinger distance by Pinsker's inequality [5],

$$D_{KL}(p \| q) \geq D_H^2(p, q),$$

which is a true metric between probability distributions p and q. Given the form of the pullback in Equation (2.1), now taken through an approximate map $\tilde{T}$, the divergence becomes

$$D_{\mathrm{KL}}(\pi \| \tilde{\pi}) = \mathbb{E}_\pi \left[\log \pi(\theta) - \log \pi_r(\tilde{T}(\theta)) - \log |J_{\tilde{T}}(\theta)| \right].$$

We note the posterior density is independent of $\tilde{T}$, and so it is not necessary for us to compute it when optimising this cost function. This expression is a complicated integral with respect to the target distribution, for which the normalisation constant is unknown. However this is exactly the scenario for which we would turn to MCMC methods for a solution.
To find the best coupling, $\tilde{T} \in \mathcal{T}^\triangle$, we solve the optimisation problem,

$$\tilde{T} = \arg\min_{T \in \mathcal{T}^\triangle} \mathbb{E}_\pi \left[-\log \pi_r(T(\theta)) - \log |J_T(\theta)| \right]$$

which has a unique solution since the cost function is convex.
We also include a regularisation term which is required for reasons which will become clear later. The optimisation problem now takes the form

$$\tilde{T} = \arg\min_{T \in \mathcal{T}^\triangle} \left[\mathbb{E}_\pi \left[-\log \pi_r(T(\theta)) - \log |J_T(\theta)| \right] + \beta \mathbb{E}(T(\theta) - \theta)^2 \right]. \tag{2.2}$$

This parameter β does not need to be tuned, experimentation has shown that the choice $\beta = 1$ is sufficient for most problems. The form of the penalisation term promotes maps which are closer to the identity.

2.2. The structure of the map. Before we continue with the derivation of the optimisation problem, we consider the structure of the map in more detail. The lower triangular structure of the map not only guarantees monotonicity, it also allows for efficient calculation of the pullback density, as well as the inverse of the map, $\tilde{T}^{-1}$. The Jacobian of $\tilde{T}$ is a lower triangular matrix,

$$J_T(\theta) = \begin{bmatrix} \partial_{\theta_1}\tilde{T}_1(\theta) & \dots & \partial_{\theta_d}\tilde{T}_1(\theta) \\ \vdots & \ddots & \vdots \\ \partial_{\theta_1}\tilde{T}_d(\theta) & \dots & \partial_{\theta_d}\tilde{T}_d(\theta) \end{bmatrix} = \begin{bmatrix} \partial_{\theta_1}\tilde{T}_1(\theta) & \dots & 0 \\ \vdots & \ddots & \vdots \\ \partial_{\theta_1}\tilde{T}_d(\theta) & \dots & \partial_{\theta_d}\tilde{T}_d(\theta) \end{bmatrix}$$

since $\partial_{\theta_n}\tilde{T}_k(\theta) = 0$ for all $n > k$. This lower triangular structure means that the determinant of the Jacobian is a product of the diagonal elements which, when we take logs, becomes

$$\log |J_{\tilde{T}}(\theta)| = \sum_{i=1}^{d} \log \partial_{\theta_i}\tilde{T}_i(\theta). \tag{2.3}$$

Here we note that this term is separable in terms of the dimension i.
Inverting $\tilde{T}$ at a point r is simplified by the lower triangular structure of the map. The map component $\tilde{T}_1(\theta)$ is a univariate polynomial in θ_1, so we can find the inverse of this function by solving the equation $T_1(\theta_1) = r_1$. This inversion tells us the value of θ_1, which means the next component is again a univariate polynomial, $T_2(\theta_2; \theta_1) = r_2$. We can then perform d root finding problems instead of a full d dimensional non-linear solve.
We require that the first derivatives of the map are continuous, which is easy to enforce by the choice of basis functions. Here we assume that the map will be built from a family of orthogonal polynomials, $\mathcal{P}(\theta)$, not necessarily orthogonal with respect to the target distribution. Each component of the map is defined as a multivariate polynomial expansion,

$$\tilde{T}_i(\theta; \gamma_i) = \sum_{\mathbf{j} \in \mathcal{J}_i} \gamma_{i,\mathbf{j}} \psi_{\mathbf{j}}(\theta). \tag{2.4}$$

The parameter γ_i is a vector of coefficients in $\mathbb{R}^{M_i}$. Each component of γ_i corresponds to a basis function $\psi_{\mathbf{j}}$, indexed by the multi-index $\mathbf{j} \in \mathbb{N}_0^d$. These multi-indices are elements of the multi-index set $\mathcal{J}_i$. A multi-index defines a product of univariate polynomials in θ_k,

$$\psi_{\mathbf{j}}(\theta) = \prod_{k=1}^{i} \varphi_{j_k}(\theta_k), \quad \text{for} \quad \mathbf{j} \in \mathcal{J}_i,$$

and where $\varphi_{j_k}(\theta_k) \in \mathcal{P}(\theta_k)$. Since $\tilde{T}$ is lower triangular, a multi-index $\mathbf{j} \in \mathcal{J}_i$ only contains entries for univariate polynomials in θ_k for $k \leq i$.
The cardinalities of the multi-index sets, $M_i = \text{card}(\mathcal{J}_i)$, give the number of unknowns in our optimisation problem, and so we would like to keep this number as small as possible. One option is to use polynomials of total order p,

$$\mathcal{J}_i^{\text{TO}} = \{\mathbf{j} : \|\mathbf{j}\|_1 \leq p, j_k = 0 \ \forall k > i\},$$

which is optimal in terms of the amount of information captured by the map about the target. The cardinality of $\mathcal{J}_i^{\text{TO}}$ is $M_i = \binom{i+p}{p}$ which increases rapidly in d and p,

where $i = 1, \ldots, d$. Smaller optimisation problems can be produced by constructing subsets of $\mathcal{J}_i^{\mathrm{TO}}$. These index sets are discussed in [4]. Increased information with a slower increase in the number of map parameters can be achieved with the composition of maps discussed in [3]. Here we stick with polynomials of total order p since we work with low dimensional problems with the PAIS algorithm.

2.3. Implementation of the optimisation problem. We now discuss how we can evaluate Equation (2.2) using a sample from the target distribution. We first reformulate the expectation in the cost functional in terms of a MC estimator,

$$\begin{aligned} C(T) &= \mathbb{E}_\pi \left[-\log \pi_r(T(\theta)) - \log |J_T(\theta)| \right] + \beta \mathbb{E}(T(\theta) - \theta)^2 \\ &\approx \frac{1}{K} \sum_{i=1}^{d} \sum_{k=1}^{K} \left[-\log \pi_r(T_i(\theta^{(k)})) - \log \left| \frac{\partial T_i}{\partial \theta_i}(\theta^{(k)}) \right| + \beta (T_i(\theta^{(k)}) - \theta^{(k)})^2 \right]. \quad (2.5) \end{aligned}$$

Optimisation of this cost function results in a map from π to some reference density π_r. By choosing the reference density to be a Gaussian density, we can simplify this expression greatly. Substitution of the Gaussian density into Equation (2.5) leads to

$$C(T) = \frac{1}{K} \sum_{i=1}^{d} \sum_{k=1}^{K} \left[\frac{1}{2} T_i^2(\theta^{(k)}) - \log \frac{\partial T_i}{\partial \theta_i}(\theta^{(k)}) + \beta (T_i(\theta^{(k)}) - \theta^{(k)})^2 \right]. \quad (2.6)$$

Note that since we assume that the map is monotonic, the derivatives of each component are positive and so this functional is always finite. In practice it is infeasible to enforce this condition across the whole parameter space. We instead enforce this condition by ensuring that the derivatives are positive at each sample point. This means that when we sample away from these support points while in reference space, it is possible to enter a region of space where the map is not monotonic.

We now return to the structure of the map components given in Equation (2.4). Since the basis functions are fixed, the optimisation problem in (2.2) is really over the map components $\bar{\gamma} = (\gamma_1, \ldots, \gamma_d)$ where $\gamma_i \in \mathbb{R}^{M_i}$. Note that $C(T)$ is the sum of d expectations, and these expectations each only concern one dimension. Therefore we can rewrite (2.2) as d separable optimisation problems.

$$\begin{aligned} &\arg \min_{\gamma_i \in \mathbb{R}^{M_i}} \frac{1}{K} \sum_{k=1}^{K} \left[\frac{1}{2} T_i^2(\theta^{(k)}; \gamma_i) - \log \frac{\partial T_i}{\partial \theta_i}(\theta^{(k)}; \gamma_i) + \beta (T_i(\theta^{(k)}; \gamma_i) - \theta^{(k)})^2 \right], \quad (2.7) \\ &\text{subject to} \quad \frac{\partial T_i}{\partial \theta_i}(\theta^{(k)}; \gamma_i) > 0 \text{ for all } k = 1, \ldots, K, \ i = 1, \ldots, d. \end{aligned}$$

The sum in Equation (2.4) is an inner product between the vector of map coefficients, and the evaluations of the basis function at a particular $\theta^{(k)}$. If we organise our basis evaluations into two matrices,

$$(F_i)_{k,\mathbf{j}} = \psi_{\mathbf{j}}(\theta^{(k)}), \quad \text{and} \quad (G_i)_{k,\mathbf{j}} = \frac{\partial \psi_{\mathbf{j}}}{\partial \theta_i}(\theta^{(k)}),$$

for all $\mathbf{j} \in \mathcal{J}_i^{\mathrm{TO}}$, and $k = 1, \ldots, K$, then we have that

$$T_i(\theta^{(k)}) = (F_i)_{k\cdot} \gamma_i \quad \text{and} \quad \frac{\partial T_i}{\partial \theta_i}(\theta^{(k)}; \gamma_i) = (G_i)_{k\cdot} \gamma_i,$$

so (2.7) becomes

$$\arg\min_{\gamma_i\in\mathbb{R}^{M_i}} \frac{1}{2}(F_i\gamma_i)^\top(F_i\gamma_i) - c^\top\log(G_i\gamma_i) + \beta\sum_{k=1}^{K}(F_i\gamma_i-\theta^{(k)})^\top(F_i\gamma_i-\theta^{(k)}), \quad (2.8)$$
$$\text{subject to} \quad G_i\gamma_i > 0.$$

In this expression, the vector c is a $K\times 1$ vector of ones, and $\log(G_i\gamma_i)$ is to be evaluated element-wise. As the Monte Carlo simulations advance, new rows can be appended to the F_i and G_i matrices, and $F_i^\top F_i$ can be efficiently updated via the addition of rank-1 matrices.
The regularisation term in Equation (2.8) can be approximated using Parseval's identity,

$$\sum_{k=1}^{K}(F_i\gamma_i-\theta^{(k)})^\top(F_i\gamma_i-\theta^{(k)}) \approx \int_{\mathbb{R}^n}|T(\theta)-\theta|^2\mathrm{d}\mu_\theta = \sum_{\mathbf{j}\in\mathcal{J}_i^{\mathrm{TO}}}(\gamma_{i,\mathbf{j}}-\iota_{\mathbf{j}})^2,$$

where ι is the vector of coefficients for the identity map. This is of course only true when the polynomial family $\mathcal{P}(\theta)$ is chosen to be orthonormal with respect to μ_θ; however this approximation prevents the map from collapsing onto a Dirac when the expectation is badly approximated by a small number of samples. If we do not normalise the MC estimator by K, we can allow this regularisation term to be dominated by the rest of the cost function as K increases.
These simplifications result in the efficiently implementable, regularised optimisation problem for computing the map coefficients,

$$\arg\min_{\gamma_i\in\mathbb{R}^{M_i}} \frac{1}{2}\gamma_i^\top F_i^\top F_i\gamma_i - c^\top\log(G_i\gamma_i) + \beta\|\gamma_i-\iota\|^2, \quad (2.9)$$
$$\text{subject to} \quad G_i\gamma_i > 0.$$

This optimisation problem can be efficiently solved using Newton iterations. It is suggested in [4] that this method usually converges in around 10-15 iterations, and we have seen no evidence that this is not a reasonable estimate. When calculating the map several times during a Monte Carlo run, using previous guesses of the optimal map to seed the Newton algorithm results in much faster convergence, usually taking only a couple of iterations to satisfy the stopping criteria.

2.4. Implementation of the optimisation problem in PAIS. In the PAIS algorithm, we use weighted samples to approximate the posterior rather than equally weighted samples. Fortunately, the majority of the derivation of this cost function follows unchanged. We look at the importance sampling Monte Carlo estimate of $C(T)$, compared with Equation (2.6),

$$C(T) = \frac{1}{\bar{w}}\sum_{i=1}^{d}\sum_{k=1}^{K} w_k\left[\frac{1}{2}T_i^2(\theta^{(k)}) - \log\frac{\partial T_i}{\partial\theta_i}(\theta^{(k)}) + \beta(T_i(\theta^{(k)})-\theta^{(k)})^2\right], \quad (2.10)$$

here w_k are the weights associated with each sample $\theta^{(k)}$, and $\bar{w}$ is the sum of all these weights. This necessitates a minor alteration to the optimisation problem in Equation (2.9),

$$\arg\min_{\gamma_i\in\mathbb{R}^{M_i}} \frac{1}{2\bar{w}}\gamma_i^\top F_i^\top W F_i\gamma_i - \frac{w^\top}{\bar{w}}\log(G_i\gamma_i) + \frac{\beta}{K}\|\gamma_i-\iota\|^2, \quad (2.11)$$
$$\text{subject to} \quad G_i\gamma_i > 0.$$

We introduce a diagonal matrix $W = \text{diag}(w)$, where $w = (w_1, \ldots, w_K)^\top$ into the log of the reference density, and replace the ones vector, c, with w. Analogously to the unweighted case, we include a $1/K$ to the regularisation term so that it has less influence as we obtain more samples. If we allowed this regularisation term to exert influence at all updates we would converge to a suboptimal map. We should converge to the same optimal map with a weighted sample as we do with an unweighted sample since we are trying to approximate the same expectation. The weights are strictly positive resulting in a positive definite matrix W. This means that the Hessian of the objective function is still positive definite, and so the optimisation problem remains convex. It is important in implementation to ensure that any weights which are numerically zero are dealt with to ensure this positive definiteness.
This optimisation problem can be efficiently solved using the Newton optimisation algorithm. The Hessian takes the form

$$HC_i(\gamma_i) = \frac{1}{\bar{w}} \left[F_i^\top W F_i + G_i^\top W \text{diag}([G_i\gamma_i]^{-2}) G_i \right] + \beta I, \tag{2.12}$$

where $[G_i\gamma_i]^{-2}$ is to be taken element-wise, and I is the $M_i \times M_i$ identity matrix. The first derivative of $C_i(T)$ is

$$\nabla C_i(\gamma_i) = \frac{1}{\bar{w}} \left[F_i^\top W F_i \gamma_i - G_i^\top W [G_i\gamma_i]^{-1} \right] + \beta(\gamma_i - \iota),$$

again $[G_i\gamma_i]^{-1}$ is taken element-wise.

3. Using the transport map to propose states in MCMC algorithms. Given samples from the target distribution, we have demonstrated how to construct an approximate transport map from the target measure to a reference measure. We now consider how to implement an MCMC algorithm which uses these maps to propose new states. These algorithms fit into the adaptive framework described in Section **??** since we use the full history of the chain when updating the map, and this update changes the shape of our proposal distribution. Convergence of this adaptation is shown in [4].

3.1. Transport map MH algorithms. As introduced in [4], a MH algorithm using the transport map to propose new states can be designed, as given in Algorithm 1. These algorithms map a sample from the target onto the reference space, a new state is then efficiently proposed using a Gaussian kernel and is mapped back onto target space. This can result in rapid mixing, where step size and direction are now a function of the current particles location in state space, similar to the ideas of Riemann manifold Hamiltonian Monte Carlo [2].
Initially we begin with the identity map, ι, while we build enough samples to produce an approximate map. For simplicity, we update the map at uniform intervals of K_U iterations, given an initial burn-in phase. In order to satisfy the conditions for adaptive algorithms to be ergodic, discussed in Section **??**, this adaptive phase must come to an end. It is sensible to stop adaptation once the map has sufficiently converged, e.g. when updates do not change the coefficients, γ, above some tolerance, or as suggested by [4] when the variance of $D_{\text{KL}}(\pi \| \tilde{\pi})$ is sufficiently close to zero.
Once the map is sufficiently converged, we can decide whether we would like to continue using a local proposal distribution, or whether to use a distribution which is independent of the current state. An independent proposal is more efficient if the reference space is close to Gaussian; however, if it is not then certain regions of the

Algorithm 1: MH algorithm with adaptive transport map [4]

1 Initialise state $\theta^{(1)} = \theta_0$.
2 Initialise map $\bar{\gamma}^{(1)} = \iota$.
3 **for** $k \leftarrow 1, \dots, L-1$ **do**
4 Compute $r = \tilde{T}(\theta^{(k)}; \bar{\gamma}^{(k)})$.
5 Sample $r' \sim q_r(\cdot; r)$.
6 Invert $\theta' = \tilde{T}^{-1}(r'; \bar{\gamma}^{(k)})$.
7 Calculate:

$$\alpha = 1 \wedge \frac{\pi(\theta')}{\pi(\theta^{(k)})} \frac{q_r(r|r')|J_{\tilde{T}}(\theta^{(k)}; \bar{\gamma}^{(k)})|}{q_r(r'|r)|J_{\tilde{T}}(\theta'; \bar{\gamma}^{(k)})|}.$$

8 Sample $u \sim U[0,1]$.
9 Set $\theta^{(k+1)}$ to θ' with probability α, otherwise $\theta^{(k+1)} = \theta^{(k)}$.
10 **if** $k \ mod \ K_U = 0$ *and* $k < K_{stop}$ **then**
11 **for** $i \leftarrow 1, \dots, n$ **do**
12 Solve (2.9) with $\{\theta^{(1)}, \dots, \theta^{(k+1)}\}$ and update $\gamma_i^{(k+1)}$.
13 **else**
14 $\bar{\gamma}^{(k+1)} = \bar{\gamma}^{(k)}$.

target space could be less likely to be visited than in a standard MH algorithm. A compromise can be found by randomly selecting which kernel to sample from, i.e. the proposal distribution

$$q(x, \cdot) = p\mathcal{N}(x, \Sigma) + (1-p)\mathcal{N}(0, \mathrm{I}), \quad p \in [0,1].$$

3.2. Transport map PAIS algorithms. In a similar way, we can use the Transport map derived in Equation (2.11) to design a proposal scheme for the PAIS algorithm. In this case we have a choice in how to proceed; we propose new samples on reference space and resample on target space, or we both propose and resample on reference space, mapping onto target space to output the samples. The first option allows us to reuse much of the framework from the standard PAIS algorithm and in the numerics later we see that this performs better than both the Transport MH algorithm, and the standard PAIS algorithm. The second option requires some restructuring but results in improved performance from the resampler.
The first option is given in Algorithm 2. We denote the ensembles of states in target space $\theta^{(k)} = \{\theta_1^{(k)}, \dots, \theta_M^{(k)}\}$, and the states in the reference space, $r = \{r_1, \dots, r_M\}$, where M is the ensemble size. Similarly, the proposal states are denoted $r' = \{r'_1, \dots, r'_M\}$ and $(w^{(k)}, \hat{\theta}^{(k)}) = \{(w_1^{(k)}, \hat{\theta}_1^{(k)}), \dots, (w_M^{(k)}, \hat{\theta}_M^{(k)})\}$, where these pairs are the states which we consider to be our sample from the target distribution. As in the standard version of the PAIS algorithm we use the deterministic mixture weights.
The second option, Algorithm 3, is similar to the first except on Line 8 where rather than resampling in target space we resample in reference space. In reference space the dimensions are roughly uncorrelated, and the Gaussian marginals are easy to approximate with fewer ensemble members. This means that the resampling step will be more efficient in higher dimensions, which we discuss in Section 5.

Algorithm 2: PAIS algorithm with adaptive transport map. Option 1.

Initialise state $\theta_i^{(1)} = \theta_0, \quad i = 1, \dots, M$.
Initialise map $\bar{\gamma}^{(1)} = \iota$.
for $k \leftarrow 1, \dots, L-1$ **do**
  Compute $r_i = \tilde{T}(\theta_i^{(k)}; \bar{\gamma}^{(k)}), \quad i = 1, \dots, M$.
  Sample $r_i' \sim q_r(\cdot; r_i)$.
  Invert $\hat{\theta}_i^{(k)} = \tilde{T}^{-1}(r_i'; \bar{\gamma}^{(k)})$.
  Calculate:

$$w_i^{(k)} = \frac{\pi(\hat{\theta}_i^{(k)})}{\left(\sum_{j=1}^{M} q_r(r_i'; r_j)\right) |J_{\tilde{T}}(\hat{\theta}_i^{(k)}; \bar{\gamma}^{(k)})|}.$$

  Resample $\theta^{(k+1)} \leftarrow \|w^{(k)}\|^{-1} \sum_{j=1}^{M} w_j^{(k)} \delta_{\hat{\theta}_j^{(k)}}(\cdot)$.
  if $k \ mod \ K_U = 0 \ and \ k < K_{stop}$ **then**
    for $i \leftarrow 1, \dots, n$ **do**
      Solve (2.11) with $\{(w^{(1)}, \hat{\theta}^{(1)}), \dots, (w^{(k+1)}, \hat{\theta}^{(k+1)})\}$ and update $\gamma_i^{(k+1)}$.
  else
    $\bar{\gamma}^{(k+1)} = \bar{\gamma}^{(k)}$.

4. Convergence of the transport proposal based MCMC algorithms. In this section we study the convergence of the transport based proposal distributions which we have described in Section 3. We follow a similar strategy to that in Chapter **??**. We begin by finding optimal values for the scaling parameters by performing a series of simulations with differing scaling parameters. We then produce 1 million samples from the target distribution with the optimal scaling parameter and an ensemble size of $M = 150$. We take 32 repeats of this optimal simulation and present the geometric average of the convergence rates. This process is then repeated for each of the algorithms we have discussed, Algorithms 1, 2, 3.
In this section we return to the Rosenbrock banana-shaped density which in Chapter **??** we labelled R_1. This target density is

$$\pi(\theta) = \frac{\sqrt{10}}{\pi} \exp\left\{-(1-\theta_1)^2 - 10(\theta_2 - \theta_1^2)^2\right\}. \tag{4.1}$$

A contour plot of the target density is given in Figure 4.1.

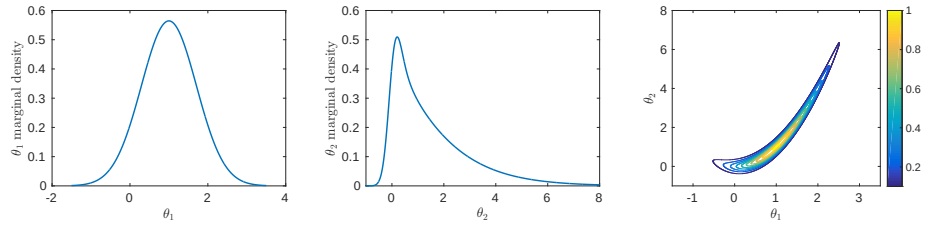
4.1. Implementation details. We now demonstrate some properties of the transport maps we will be using in our MCMC algorithms. We draw 1 million samples from the density in (4.1), and use this sample in the framework of Section 2.3 to build a transport map. We use this map to push forward the original sample onto the reference space, where we will be able to see how well the map has performed at converting the original sample to a standard Gaussian. We then pull the sample back on to target space using the inverse map to check that our map is invertible and well behaved.

Algorithm 3: PAIS algorithm with adaptive transport map. Option 2.

Initialise state $\theta_i^{(1)} = \theta_0, \quad i = 1, \dots, M$.
Initialise map $\bar{\gamma}^{(1)} = \iota$.
for $k \leftarrow 1, \dots, N-1$ **do**
  Compute $r_i = \tilde{T}(\theta_i^{(k)}; \bar{\gamma}^{(k)}), \quad i = 1, \dots, M$.
  Sample $r_i' \sim q_r(\cdot; r_i)$.
  Invert $\hat{\theta}_i^{(k)} = \tilde{T}^{-1}(r_i'; \bar{\gamma}^{(k)})$.
  Calculate:

$$w_i^{(k)} = \frac{\pi(\hat{\theta}_i^{(k)})}{\left(\sum_{j=1}^{M} q_r(r_i'; r_j)\right) |J_{\tilde{T}}(\hat{\theta}_i^{(k)}; \bar{\gamma}^{(k)})|}.$$

  Resample $r^* \leftarrow \|w^{(k)}\|^{-1} \sum_{j=1}^{M} w_j^{(k)} \delta_{r_j'}(\cdot)$.
  Invert $\theta_i^{(k+1)} = \tilde{T}^{-1}(r_i^*)$.
  if $k \ mod \ K_U = 0 \ and \ k < K_{stop}$ **then**
    for $i \leftarrow 1, \dots, n$ **do**
      Solve (2.11) with $\{(w^{(1)}, \hat{\theta}^{(1)}), \dots, (w^{(k+1)}, \hat{\theta}^{(k+1)})\}$ and update $\gamma_i^{(k+1)}$.
  else
    $\bar{\gamma}^{(k+1)} = \bar{\gamma}^{(k)}$.



(a) Marginal density function for θ_1. (b) Marginal density function for θ_2. (c) Contour plot for Rosenbrock density.

FIG. 4.1. *Visualisation of the density of example R_1, as given in Equation* (4.1).

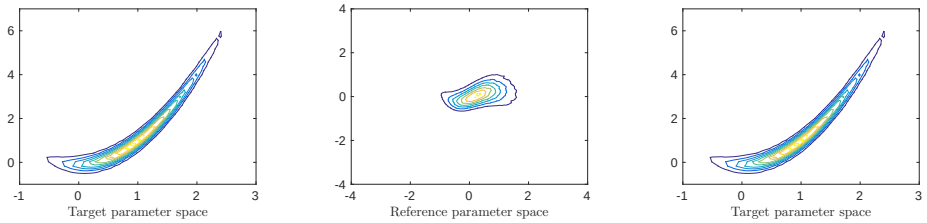
For this example, we use an index set of total order 3 with monomial basis functions. This results in a map of the form

$$T(\theta_1, \theta_2) = \begin{bmatrix} T_1(\theta_1) \\ T_2(\theta_1, \theta_2) \end{bmatrix},$$

where

$$\begin{aligned} T_1(\theta_1) &= \gamma_{1,1} + \gamma_{1,2}\theta_1 + \gamma_{1,3}\theta_1^2 + \gamma_{1,4}\theta_1^3, \\ T_2(\theta_1, \theta_2) &= \gamma_{2,1} + \gamma_{2,2}\theta_1 + \gamma_{2,3}\theta_1^2 + \gamma_{2,4}\theta_1^3 + \gamma_{2,5}\theta_2 + \gamma_{2,6}\theta_1\theta_2 \\ &\quad + \gamma_{2,7}\theta_1^2\theta_2 + \gamma_{2,8}\theta_2^2 + \gamma_{2,9}\theta_1\theta_2^2 + \gamma_{2,10}\theta_2^3. \end{aligned}$$

Clearly even with only basis functions of total order 3, we have a large number of unknowns in our optimisation problem, $\bar{\gamma} \in \mathbb{R}^{14}$. If we were to increase the dimension of θ further we would need to reduce the number of terms we include in the expansion by, for example, removing all the cross terms. This reduces the quality of our map but since we only require an approximate map we can afford to reduce the accuracy.



(a) Original sample θ from RWMH algorithm.

(b) Push forward of θ onto reference space.

(c) Pull back of reference sample onto target space.

FIG. 4.2. *Rosenbrock target density as described in Equation* (4.1).

Figure 4.2 shows the efficiency of the transport map. Even though we have truncated the infinite expansion in the monomial basis down to 4 and 10 terms in respective dimensions, the push forward of the sample is still a unimodal distribution centred at the origin with standard deviation 1. As you move out into the tails of the reference density more non-Gaussian features form, but these are not too much of a problem when using a suitable proposal distribution. The pullback from reference space, in Figure 4.2, is an exact match of the original sample since we have not perturbed the sample in reference space. This inversion is well defined in the sampling region, although not necessarily outside.

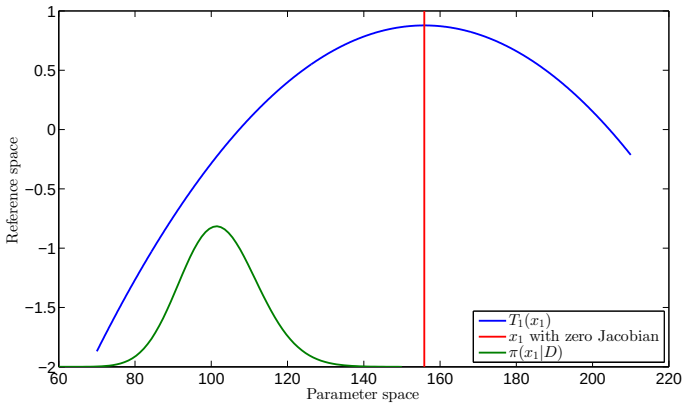


FIG. 4.3. *An example transport map which does not allow exploration of the full target distribution, and has a non-invertible point.*

If we consider a map built from polynomials of total order $p = 2$, it is clear that the map $\tilde{T}$ does not map onto $\mathbb{R}^d$, but on to a subset of it. Figure 4.3 illustrates this point. We see that in this dimension, the $\tilde{T}^{-1}$ will not map any proposal $r > 0.8$ to a point in the target space. The point at which no inverse exists for a point r_1 can be made

arbitrarily far out in the tails of the posterior distribution, but will be finite. The point at which the Jacobian is zero is also a non-invertible point since the logarithm in the pullback becomes infinite. This means that (1) we will never sample from the tail of the target distribution which lies beyond this turning point, and (2) all points in reference space $r < 0.8$ have the possibility of being mapped to two distinct points in target space. We can choose to truncate our parameter space at this turning point which allows our map to be bijective, however we will not be able to sample from the true posterior. For this reason it is best to chose an index set where the maximum order is an odd number.

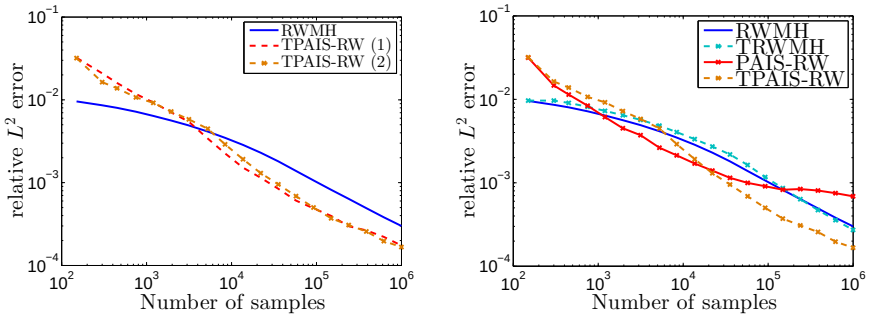
4.2. Numerical results for convergence of transport map based algorithms. We first find the optimal scaling parameters for the individual algorithms. This is done as before by optimising the effective sample size in the PAIS algorithm, and by tuning the relative L^2 error in the MH algorithm. There is currently no guidance on the best way of tuning the MH algorithm with transport map proposals although one might expect results similar to the standard MH results. As in the PAIS algorithm, the effective sample size might be the best option.

Statistic / Algorithm	1	2	3
δ_{L^2}	1.0e-0	1.1e-1	3.5e-1
δ_{ESS}	-	1.0e-1	5.2e-1
Acc. rate	0.23	-	-
ESS ratio	-	0.62	0.71

TABLE 4.1
Optimal scaling parameters for the transport map based algorithms applied to R_1.

The optimal scaling parameters are given in Table 4.1. Here we see that the effective sample size is much lower than we see in the one-dimensional examples with the PAIS algorithms. However, in R_1 we are dealing with a much more complicated correlation structure, as well as a very slowly decaying tail in θ_2. We have seen in Section **??** that the standard PAIS-RW required an ensemble size of $M = 500$ to overcome the problems in this density, however the transport map transforms the tails to be more like those of a Gaussian which can be approximated well by a smaller ensemble size of $M = 150$.
The convergence of the three algorithms is displayed in Figure 4.4. Figure (a) shows that the two variations of the transport based PAIS algorithms converge with similar rates. The second version, which performs the resampling stage in reference space rather than target space, has a slightly higher ESS, and is more stable than option (1). This version also has a property that we can exploit in Section 5.

5. Sampling in higher dimensions. Algorithm 3 allows us to decorrelate the dimensions of our random parameter on reference space, where we then can resample and map the resulting ensemble back onto target space. Since, on reference space, the dimensions are uncorrelated, we are able to resample in each dimension separately. Resampling in a single dimension allows for optimisations in resampling code, and also means that the resampler is not affected by the curse of dimensionality.
If we can approximate the posterior well with our mixture and with the transport map, we should not be affected by the increase in dimension to the extent we have been with the standard PAIS-RW algorithm. In one dimension the ETPF algorithm



(a) Comparison of the two Transport PAIS options.

(b) Comparison of Transport PAIS option (2) with the standard algorithms.

FIG. 4.4. *Convergence of Algorithms 1, 2, 3 for R_1. Ensemble size $M = 150$, resampling performed using the AMR algorithm.*

can be implemented very efficiently. As described in [6], the coupling matrix has all non-zero entries in a staircase pattern when the state space is ordered. We can exploit this knowledge to produce Algorithm 4. Which is much faster than using the simplex algorithm to minimise the associated cost function, and faster than the AMR algorithm.

Algorithm 4: ETPF algorithm in one dimension.

Sort the states, $\{(w_i, x_i)\}_{i=1}^M$, into ascending order.
Normalise the weights $p_i = w_i / \|w\|_1$.
Set $y_i \leftarrow 0$ for all $i = 1, \ldots, M$.
Set $c \leftarrow 0$
for $i \leftarrow 1, \ldots, M$ **do**
    Set $t \leftarrow p_i$
    while $j \leq M$ *and* $t > 0$ **do**
        Set $s \leftarrow \left(M^{-1} - c\right) \wedge t$
        Increase y_j by $M \times s \times x_i$.
        Decrease t by s.
        Increase c by s.
        if $t > 0$ **then**
            Increase j by 1.
            Set $c \leftarrow 0$.
Return y.

6. Transport map-accelerated Parallel Adaptive Importance Sampling (TPAIS).

7. Multiscale Methods for Stochastic Chemical Reaction Networks.

8. Numerical Examples.

9. Discussion.

REFERENCES

[1] T. El Moselhy and Y. Marzouk, *Bayesian inference with optimal maps*, Journal of Computational Physics, 231 (2012), pp. 7815–7850.

[2] M. Girolami and B. Calderhead, *Riemann manifold langevin and hamiltonian monte carlo methods*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 73 (2011), pp. 123–214.

[3] M. Parno, *Transport maps for accelerated Bayesian computation*, PhD thesis, Massachusetts Institute of Technology, 2015.

[4] M. Parno and Y. Marzouk, *Transport map accelerated Markov chain Monte Carlo*, arXiv preprint arXiv:1412.5492, (2014).

[5] M. Pinsker, *Information and information stability of random variables and processes*, (1960).

[6] S. Reich, *A nonparametric ensemble transform method for Bayesian inference*, SIAM Journal on Scientific Computing, 35 (2013), pp. A2013–A2024.