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 will briefly reintroduce Parallel Adaptive Importance Sampling (PAIS), a variant of Population Monte Carlo (PMC) which incorporates state of the art resamplers. In Section 3 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 4 we show how such a map can be incorporated into a sophisticated parallel MCMC infrastructure in order to accelerate mixing. In Section 5 we seek to show the advantages of this approach through the analysis of test problems. In Section 6 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 7 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 8.

2. Parallel Adaptive Importance Sampling.

PAIS****CITE**** is a variant of PMC****CITE****, which is a family of methods which are based on importance sampling. In importance sampling, we attempt to characterise a target density through sampling from that density. However, the target density π itself is often too complex to sample from directly, so we instead sample from a proposal density χ. Each sample $\theta^{(k)} \sim \chi$ is then weighted by $w_k = \frac{\pi(\theta^{(k)})}{\chi(\theta^{(k)})}$, to take account of the bias of sampling from a different distribution to π. Monte Carlo estimates using a sample of size N of a function f with respect to π can then be made through the formula

$$\mathbb{E}_\pi(f) \approx \frac{1}{\bar{w}} \sum_{k=1}^{N} w_k f(\theta^(k)).$$

This method works well when π and χ are close, but can be excruciatingly slow when they are not. The idea behind PMC methods is to construct a good proposal distribution, either from the entire history of the algorithm up to the current point, or to use the current state of a whole ensemble of M particles in the system.

In PAIS, the proposal distribution χ is chosen to be the equally weighted mixture of any choice of MCMC proposal kernel, evaluated at each of the current particles in the system. If $\theta^{(k)} = [\theta_1^{(k)}, \theta_2^{(k)}, \dots, \theta_M^{(k)}]^\top$ is the current state of the ensemble, and we wish to use an MCMC proposal density $q(\cdot;\cdot,\beta)$ then

$$\chi^{(k)} = \frac{1}{M}\sum_{i=1}^{M} q(\cdot;\theta_i^{(k)}.$$

Often the variance of the MCMC proposal kernels can be tuned using their respective algorithmic parameters β. Good values for these algorithmic parameters can be found by optimising for the effective sample size of the importance sample that is produced (see ***CITE*** for more details).
If the ensemble is large enough, and the chain has entered probabilistic stationarity, then the current state of the ensemble is a good rough discrete approximation of the target density, and in turn $\chi^{(k)}$ is close enough to π to produce an efficient importance sample $\{\hat{\theta}^{(k)}, w^{(k)}\}$. It can be advantageous to use stratified sampling of the mixture in order to ensure that the sample made is as representative as possible of the target density, i.e.

$$\hat{\theta}_i^{(k)} \sim q(\cdot;\theta_i^{(k)}$$

for each $i = 1, 2, \dots, M$. We now have a weighted sample, and it would be inadvisable to use an equally weighted mixture of proposal distributions from each of these points. Therefore, before starting the next iteration, the importance sample $\{\hat{\theta}^{(k)}, w^{(k)}\}$ is resampled to produce an equally weighted sample, ready for the next iteration of the algorithm. In PAIS, a state-of-the-art resampler is used, which uses optimal transport methods to find the equally weighted discrete sample which best represents the statistics of the importance sample***CITE****. For larger ensemble sizes M this can become expensive, in which case a greedy approximation of this algorithm, the Approximate Multinomial Resampler (AMR) can be implemented***CITE****. The output of the resampler is then denoted $\theta^{(k+1)}$ and the algorithm is ready for the next iteration. The importance samples $\{\hat{\theta}^{(k)}, w^{(k)}\}$ The PAIS is summarised in Algorithm 1.

Algorithm 1: The PAIS Algorithm.

```
Initialise $\theta^{(1)} \sim \mu_0$.
for $k = 1, \dots, N$ do
    Sample $\hat{\theta}_i^{(k)} \sim q(\cdot;\theta_j^{(k)}, \beta)$, for $i = 1, \dots, M$.
    Calculate $w^{(k)} = (w_1^{(k)}, \dots, w_M^{(k)})^\top$, where
        $w_i^{(k)} = \frac{\mu(\hat{\theta}_i^{(k)})}{\chi^{(k)}(\hat{\theta}_i^{(k)};\theta^{(k)},\beta)}.$
    Resample $\theta^{(k+1)} \leftarrow \|w^{(k)}\|_1^{-1} \sum_{j=1}^{M} w_j^{(k)} \delta_{\hat{\theta}_j^{(k)}}(\cdot)$.
Output $\{(w^{(n)}, \hat{\theta}^{(n)}\}_{n=1}^{N}$.
```

One problem with the PAIS and other PMC methods can become apparent if the target density has very strong correlations in its structure, in particular if that correlation is not global but only local. In this case, unless the proposal densities q are informed by this local structure, the mixture distribution proposal may not well approximate π without a very large large ensemble size M, which can become inhibitively expensive. Some methods have been proposed ****CITE**** which use samples local to each particle to inform local covariance structure.

In this paper, we investigate the use of transport maps to learn local covariances across the whole of the domain, in order to stabilise PMC-type methods, and make these methods more applicable to a wider range of more challenging inference problems.

3. Construction of transport maps in importance sampling.

In this Section, we describe the construction of transport maps which allow for the simplification of complex posterior distributions in order to allow for improved sampling, in particular for methods based on importance sampling. In [5] 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 [12] 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 [12] for approximately 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 3.1 ((Exact) Transport Map T). *A transport map T is a function $T\colon \mathcal{X} \to \mathbb{R}^d$ such that the pullback of the reference measure with density $\phi(\cdot)$,*

$$\tilde{\pi}(\theta) = \phi(T(\theta))|J_T(\theta)|, \tag{3.1}$$

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,

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

Definition 3.2 (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.*

Armed with such a map, independent samples can be made of the target measure, using the pullback of the reference density 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. This space does not necessarily contain an exact coupling between target and reference space, and so we are motivated to formulate an optimisation problem to find the map $\tilde{T} \in \mathcal{T}^{\uparrow}$ which most closely maps the target density to the reference density.

As in previous work in [12], we can ensure invertibility if we restrict the map to be lower triangular, i.e. $\tilde{T} \in \mathcal{T}^{\triangle} \subset \mathcal{T}^{\uparrow}$. This lower triangular map has the form,

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

where $T_i \colon \mathbb{R}^i \to \mathbb{R}$.

3.1. The optimisation problem. Our aim is now to find the lower triangular map $\tilde{T} \in \mathcal{T}^{\triangle}$ such that the difference between target density and the pullback of the reference density is minimised. As in [12], we choose the cost function to be the Kullback-Leibler (KL) 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 [13],

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

which is a true metric between probability distributions p and q. Given the form of the pullback in Equation (3.1), 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

$$\tilde{T} = \arg\min_{T \in \mathcal{T}^{\triangle}} \left[\mathbb{E}_{\pi}\left[-\log \pi_r(T(\theta)) - \log |J_T(\theta)|\right] + \beta \mathbb{E}(T(\theta) - \theta)^2\right]. \tag{3.2}$$

The parameter $\beta > 0$ does not need to be tuned, as experimentation has shown that the choice $\beta = 1$ is sufficient for most problems. This expectation can be approximated by using an MCMC approximation. The form of the penalisation term promotes maps which are closer to the identity, and so prevents overfitting when the quality or size of the current sample from the posterior is not sufficient.

3.2. 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

$$\log |J_{\tilde{T}}(\theta)| = \sum_{i=1}^{d} \log \partial_{\theta_i}\tilde{T}_i(\theta), \tag{3.3}$$

where 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,

$$\tilde{T}_i(\theta; \gamma_i) = \sum_{\mathbf{j} \in \mathcal{J}_i} \gamma_{i,\mathbf{j}} \psi_{\mathbf{j}}(\theta). \tag{3.4}$$

The parameter $\gamma_i \in \mathbb{R}^{M_i}$ is a vector of coefficients. 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 $k \le i$.

The cardinalities of the multi-index sets, $M_i = \mathrm{card}(\mathcal{J}_i)$, 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,

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

which is optimal in terms of the amount of information captured by the map about the target. The cardinality of $\mathcal{J}_i^{\mathrm{TO}}$ is $M_i = \binom{i+p}{p}$ which increases rapidly in d and p, where $i = 1, \dots, d$. Smaller optimisation problems can be produced by constructing subsets of $\mathcal{J}_i^{\mathrm{TO}}$. These index sets are discussed in [12]. Increased information with a slower increase in the number of map parameters can be achieved with the composition of maps discussed in [11]. Here we stick with polynomials of total order p since we work with low dimensional problems with the PAIS algorithm.

3.3. Implementation of the optimisation problem. We now discuss how we can evaluate the cost function in Equation (3.2). In [12], this expectation is approximated using an MCMC estimator, such that

$$\begin{aligned} C(T) &= \mathbb{E}_\pi\left[-\log \pi_r(T(\theta)) - \log |J_T(\theta)|\right] + \beta \mathbb{E}(T(\theta) - \theta)^2 \\ &\approx \frac{1}{K}\sum_{i=1}^{d}\sum_{k=1}^{K}\left[-\log \pi_r(T_i(\theta^{(k)})) - \log\left|\frac{\partial T_i}{\partial \theta_i}(\theta^{(k)})\right| + \beta(T_i(\theta^{(k)}) - \theta^{(k)})^2\right]. \end{aligned} \tag{3.5}$$

Here we diverge from previous work, as we aim to build a map from samples from an importance sampling scheme. Such samples no longer carry equal weight, and as such the Monte Carlo estimator becomes

$$C(T) = \frac{1}{\bar{w}}\sum_{i=1}^{d}\sum_{k=1}^{K} w_k \left[-\log \pi_r(T_i(\theta^{(k)})) - \log\left|\frac{\partial T_i}{\partial \theta_i}(\theta^{(k)})\right| + \beta(T_i(\theta^{(k)}) - \theta^{(k)})^2\right], \tag{3.6}$$

where w_k are the weights associated with each sample $\theta^{(k)}$, and $\bar{w}$ is the sum of all these weights. 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 (3.6) leads to

$$C(T) = \frac{1}{\bar{w}}\sum_{i=1}^{d}\sum_{k=1}^{K} w_k \left[\frac{1}{2}T_i^2(\theta^{(k)}) - \log \frac{\partial T_i}{\partial \theta_i}(\theta^{(k)}) + \beta(T_i(\theta^{(k)}) - \theta^{(k)})^2\right], \tag{3.7}$$

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 (3.4). Since the basis functions are fixed, the optimisation problem in (3.2) is really over the map components $\bar{\gamma} = (\gamma_1, \dots, \gamma_d)$ 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 (3.2) as d separable optimisation problems.

$$\arg\min_{\gamma_i\in\mathbb{R}^{M_i}} \frac{1}{\bar{w}}\sum_{k=1}^{K} w_k\left[\frac{1}{2}T_i^2(\theta^{(k)};\gamma_i)-\log\frac{\partial T_i}{\partial\theta_i}(\theta^{(k)};\gamma_i)+\beta(T_i(\theta^{(k)};\gamma_i)-\theta^{(k)})^2\right], \tag{3.8}$$

$$\text{subject to}\quad \frac{\partial T_i}{\partial\theta_i}(\theta^{(k)};\gamma_i)>0 \text{ for all } k=1,\dots,K,\ i=1,\dots,d.$$

The sum in Equation (3.4) 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 $k=1,\dots,K$, 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 (3.8) becomes

$$\arg\min_{\gamma_i\in\mathbb{R}^{M_i}} \frac{1}{2}(F_i\gamma_i)^\top W(F_i\gamma_i)-\mathbf{w}^\top\log(G_i\gamma_i)+\frac{\beta}{\bar{w}}\sum_{k=1}^{K}w_k(F_i\gamma_i-\theta^{(k)})^\top(F_i\gamma_i-\theta^{(k)}), \tag{3.9}$$

$$\text{subject to}\quad G_i\gamma_i>0.$$

In this expression, the vector $\mathbf{w}=[w_1,w_2,\dots,w_K]^\top$ is the vector of the weights, W is the diagonal matrix $W=\mathrm{diag}(w)$ and $\log(G_i\gamma_i)$ is to be evaluated element-wise. As more importance samples are made, new rows can be appended to the F_i and G_i matrices, and $F_i^\top WF_i$ can be efficiently updated via the addition of rank-1 matrices. The regularisation term in Equation (3.9) can be approximated using Parseval's identity,

$$\frac{1}{\bar{w}}\sum_{k=1}^{K}w_k(F_i\gamma_i-\theta^{(k)})^\top(F_i\gamma_i-\theta^{(k)})\xrightarrow[K\to\infty]{}\int_{\mathbb{R}^n}|T(\theta)-\theta|^2\mathrm{d}\mu_\theta=\sum_{\mathbf{j}\in\mathcal{J}_i^{\mathrm{TO}}}(\gamma_{i,\mathbf{j}}-\iota_{\mathbf{j}})^2,$$

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.
These simplifications result in the efficiently implementable, regularised optimisation problem for computing the map coefficients,

$$\arg\min_{\gamma_i\in\mathbb{R}^{M_i}} \frac{1}{2\bar{w}}\gamma_i^\top F_i^\top WF_i\gamma_i-\frac{w^\top}{\bar{w}}\log(G_i\gamma_i)+\beta\|\gamma_i-\iota\|^2, \tag{3.10}$$

$$\text{subject to}\quad G_i\gamma_i>0,$$

This optimisation problem can be efficiently solved using Newton iterations. It is suggested in [12] 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.
The Hessian takes the form

$$HC_i(\gamma_i) = \frac{1}{\bar{w}} \left[F_i^\top W F_i + G_i^\top W \mathrm{diag}([G_i\gamma_i]^{-2}) G_i\right] + \beta I, \tag{3.11}$$

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

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

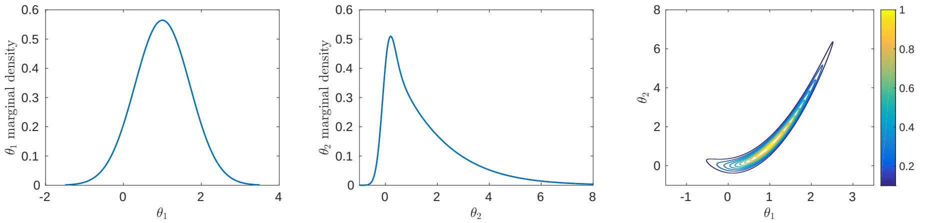
again $[G_i\gamma_i]^{-1}$ is taken element-wise.

4. Transport map usage in PAIS and other PMC algorithms. Given importance 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 importance sampling-based MCMC algorithm which uses these maps to propose new states. In [12] it was shown how approximate transport maps can be used to accelerate Metropolis-Hastings methods, with the map being periodically updated with the samples produced from the target measure. Convergence of this adaptation is shown in [12]. In this Section, we will show how similarly, these maps can be used to construct highly efficient importance sampling schemes.
In particular, we will show how we can use the transport map derived in Equation (3.10) 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 together form 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 moderately higher dimensions, which we discuss in Section 8.1.

5. 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 4. We take as a test problem the ubiquitous

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 $k \bmod K_U = 0$ and $k < K_{stop}$ then
        for $i \leftarrow 1, \dots, n$ do
            Solve (3.10) 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. 5.1. *Visualisation of the Rosenbrock density as given in Equation* (5.1).

Rosenbrock banana-shaped density. This target density is given by

$$\pi(\theta) = \frac{\sqrt{10}}{\pi} \exp\left\{-(1-\theta_1)^2 - 10(\theta_2 - \theta_1^2)^2\right\}. \tag{5.1}$$

A contour plot of the target density is given in Figure 5.1. This problem is challenging to sample from since it has a highly peaked and curved ridge, and is often used as a test problem in optimisation and MCMC communities.

5.1. Implementation details. Before looking at the performance of the MCMC algorithms, we demonstrate some properties of the transport maps we will be using in our MCMC algorithms. We draw 1 million samples from the density in (5.1), and use this sample in the framework of Section 3 to build a transport map. We use this map

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 (3.10) with $\{(w^{(1)}, \hat{\theta}^{(1)}), \dots, (w^{(k+1)}, \hat{\theta}^{(k+1)})\}$ and update $\gamma_i^{(k+1)}$.
    else
        $\bar{\gamma}^{(k+1)} = \bar{\gamma}^{(k)}$.
```

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.

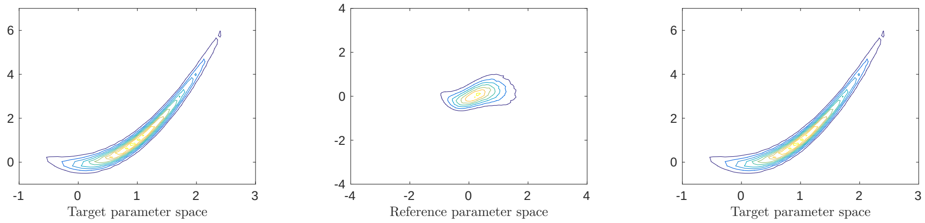
For this example, we use an index set of total order 3 with monomial basis functions. It is important that total order is an odd number, since otherwise the map will not be surjective. This results in a map of the form

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

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. 5.2. *Rosenbrock target density as described in Equation* (5.1).

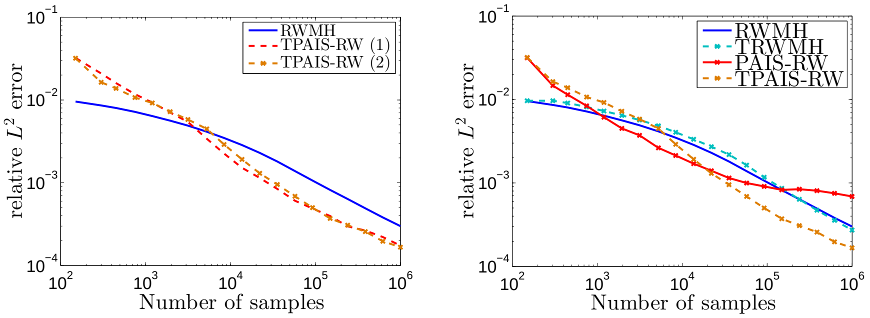
Figure 5.2 shows the output of the approximate transport map. Even though we have truncated the infinite expansion in the monomial basis down to 4 and 10 terms in respective dimensions, the push forward of the sample is still a unimodal distribution centred at the origin with standard deviation 1. As you move out into the tails of the reference density more non-Gaussian features are clearly visible. However, overall, the push forward of the target density does not look a challenging one to sample from, with even relatively simple MCMC methods such as RWMH. The pullback from reference space, in Figure 5.2, 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 [12].

5.2. Numerical results for convergence of transport map based algorithms on the Rosenbrock density. We first find the optimal scaling parameters for the individual algorithms. This is done, as in ****CITE US**** by optimising for 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, especially if adaptation of the map is stopped after a given point. As in the PAIS algorithm, optimising for the effective sample size might be the best option.

Statistic / Algorithm	Transport M-H	Alg. 2	Alg. 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 5.1
Optimal scaling parameters for the transport map based algorithms applied to R_1.

The optimal scaling parameters are given in Table 5.1. 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 the Rosenbrock density (5.1) we are dealing with a much more complicated correlation structure, as well as a very slowly decaying tail in θ_2. From our experiments, we have observed 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 $M = 150$.



(a) Comparison of the two Transport PAIS options.

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

FIG. 5.3. *Convergence of Algorithms Transport M-H, 2, 3 for density* (5.1). *Ensemble size $M = 150$, resampling performed using the AMR algorithm.*

The convergence of the three algorithms is displayed in Figure 5.3. 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 8.1.

6. Multiscale Methods for Stochastic Chemical Reaction Networks.

In this Section, we discuss some recent advances in multiscale methods for stochastic reaction networks. Inverse problems arising in this area often lead to highly correlated and complex posterior distributions, which traditional MCMC methods can struggle to sample from. We will then go on to solve some inverse problems related to this in Section 7, using the transport map versions of the PAIS algorithm, as described in Section 4.

We consider chemical reaction networks of N_s chemical species $\{S_j\}_{j=1}^{N_s}$, with population numbers given by $X(t) = [X_1(t), X_2(t), \dots, X_{N_s}(t)]^\top \in \mathbb{N}_0^{N_s}$ reacting in a small reactor, through N_r different reaction channels. When population numbers of one or more of the chemical species in the reactor is small, as is the case with chemical reactions occurring within living cells, the sporadic and discrete way in which reactions occur can not be well modelled by deterministic continuous models, such as ODEs. In such a situation, we may wish to model the dynamics through a discrete stochastic model, such as a continuous time Markov chain.

For each reaction R_j, for $j = 1, 2, \dots N_r$, there is a propensity or hazard function $\alpha_j(X(t)$ which indicates how likely that reaction is to fire, defined by

$$\alpha_j(X(t)) = \lim_{dt \to 0} \mathbb{P}(\text{Reaction } R_j \text{ in the time interval} \quad s \in [t, t + dt]).$$

If a system satsifies what is called *mass action kinetics*, then the form of the function α_j is determined, up to a rate constant, by the reactants involved in that reaction:

$$\alpha_j(X) = k_j \prod_{m=1}^{N_s} \prod_{n=0}^{\nu_{j,m}-1} (X_m - n), \tag{6.1}$$

where $\nu_{j,m}$ is the mth component of the stoichiometric vector ν_j, X_m is the mth component of the state vector X, and the k_j are rate constants.

Following each reaction R_j there is an instantaneous change in the current state, as the reactants of the reaction are consumed, and the products produced. This is modelled by a change in the state vector $X(t) = X(t) + \nu_j$ where ν_j is that stochiometric vector for reaction R_j.

The model can be represented as the following expression involving N_r different unit rate Poisson processes****CITE ANDERSON ET AL**** Y_j, given by:

$$X(t) = X(0) + \sum_{j=1}^{N_r} \nu_j \int_0^t \alpha_j(X(s))ds. \tag{6.2}$$

The master equation for this model is only solvable in certain situations, for example monomolecular networks****CITE****, or for steady-state solutions certain deficiency zero networks****CITE*****. Trajectories for this system can be sampled exactly, for instance using the Gillespie SSA***CITE***, or its variants [6]****CITE****. However, if the system is stiff, i.e. there are some reactions which are firing many times on a timescale for which others are unlikely to fire at all, then trajectories can become prohibitively expensive to simulate, since these methods simulate every single reaction event with the same cost. In such a system, one might employ multiscale methods in order to approximate trajectories at a lower cost than the exact algorithms.

One common approach is to induce the quasi-steady-state approximation QSSA. This approximation makes the assumption that fast reactions enter quasi-equillibrium on a timescale which is negligible with respect to the timescale on which the slow dynamics in the system are evolving. This amounts to taking the assymptotic limit that the propensities of the "slow" reaction channels are equal to zero. This allows us to sample approximate trajectories of the slow reactions without the need to compute many fast reaction events.

This approach can work well when there is a large timescale gap between the fast and slow reactions in a system, but where the timescale gap is less pronounced, it can lead to significant errors***cite cotter erban errors paper***. Another approach is the constrained multiscale method (CMA)***CITE***, based in part on the equation-free approach to multiscale computations***CITE YANNIS****. This approach also assumes quasi-equillibrium in the system, but takes into account the effect that the slow reactions can have on the invariant distribution of the fast variables in the system. For a more detailed description of this method, please refer to the literature***CITATIONS***.

Multiscale methods are not only of use when forward evaluations of the dynamics are costly, but can also be of use where we attempt to solve an inverse problem where the fast variables are unobservable. One approach in this situation would be to integrate over all possible trajectories of the fast variables, but this would almost always be prohibitively expensive. Another approach would be to use an appropriate multiscale approximation, so that the effective dynamics of the slow variables can be approximate without the need to consider the rapid fluctuations of the fast variables in the system.

6.1. Likelihoods arising from stochastic reaction networks. Suppose that we are able to exactly observe the number of molecules of each chemical species in a system which satisfies mass action kinetics, and which can be well approximated by (6.2). Suppose that we wish to be able to infer the value of the rate constants of each reaction from these observations. Even with no observational noise, since the dynamics of the system are stochastic, this still leads to a Bayesian inverse problem where we can only infer a joint probability distributions on the reaction parameters. Suppose that we are in state $X(t) = X_0$. There are two independent univariate random variables which decide when and what the next reaction in the system is. There is the jth waiting time τ_j to the next reaction, which is given by an exponential random variable

$$\tau_j \sim \exp\left(\frac{1}{\alpha_0(X(t))}\right),$$

where $\alpha_0(X(t)) = \sum_i^{N_r} \alpha_i(X(t))$ is the total propensity in the system. The second is a multinomial random variable r_j which dictates which reaction has occurred during the jth reaction, which takes the value $r \in \{1, 2, \ldots, N_r\}$ with probability

$$\mathbb{P}(r = i) = \frac{\alpha_i(X(t))}{\alpha_0(X(t))}.$$

As such, in order to compute the likelihood of a particular trajectory given a possible realisation of the reaction parameters, it is sufficient to have access to the total time spent in each state, and the frequency of each reaction which led to leaving each state. From this formulation, we see that the random variables (τ_j, r_j) only depend on the states $\mathbf{X}(t_{j-1})$ and so are Markovian. This conditional independence means that we can group events together by what state the system was in when the event happened. We define two new random variables which depend on a state $\mathbf{Y} \in \mathcal{S}$, first the total time spent in state $\mathbf{Y}$,

$$T(\mathbf{Y}) = \sum_{j=1}^{M} \tau_j \; \mathrm{I}(\mathbf{X}(t_{j-1}) = \mathbf{Y}).$$

This random variable, $T(\mathbf{Y})$, is a sum of exponentially distributed random variable, each with the rate $\alpha_0(\mathbf{Y}; \mathbf{k})$, and hence follows the Gamma distribution,

$$T(\mathbf{Y}) \sim \text{Gamma}\left(\alpha = K(\mathbf{Y}),\ \beta = \alpha_0(\mathbf{Y}; \mathbf{k})\right), \tag{6.3}$$

where $K(\mathbf{Y}) = \sum\limits_{j=1}^{M} \mathrm{I}(\mathbf{X}(t_{j-1}) = \mathbf{Y})$.

Similarly, we can define the reactions which occurred when the system was in state $\mathbf{Y}$ as $\mathbf{r}(\mathbf{Y}) = [r_1(\mathbf{Y}), \ldots, r_{N_r}(\mathbf{Y})]^\top$ where

$$r_i(\mathbf{Y}) = \sum_{j=1}^{M} \mathrm{I}(r_j = i \textbf{ and } \mathbf{X}(t_{j-1}) = \mathbf{Y}).$$

Here all the random variables r_j follow the same multinomial distribution, and so

$$\mathbf{r}(\mathbf{Y}) \sim \text{Multinomial}(K(\mathbf{Y}),\ \mathbf{p}(\mathbf{Y})). \tag{6.4}$$

The random variables defined in Equations (6.3) and (6.4) are sufficient statistics for the posterior distribution $\pi(\mathbf{k}|D)$. With these definitions we define two new structures

$$\mathbf{T} = [T(\mathbf{Y}_1), \ldots, T(\mathbf{Y}_K)]^\top, \quad \text{and} \quad \mathbf{R} = [\mathbf{r}(\mathbf{Y}_1), \ldots, \mathbf{r}(\mathbf{Y}_K)]^\top,$$

where $K = |\mathcal{S}|$, the number of states in $\mathcal{S}$, and each state $\mathbf{Y}_i \in \mathcal{S}$ has been enumerated. We use these structures to define shorter notation,

$$\mathbf{T}_i = T(\mathbf{Y}_i), \quad \mathbf{R}_{ij} = r_j(\mathbf{Y}_i), \quad \text{and} \quad \mathbf{K}_i = K(\mathbf{Y}_i).$$

To construct the posterior distribution for the reaction rates, $\mathbf{k}$, in the chemical system, we formulate the likelihood using the sufficient statistics derived in the previous section. Due to the non-negativity of these reaction rates, we assign a Gamma prior distribution to each rate. Given the distributions in Equations (6.3) and (6.4) for our data, the likelihood of observing the data $\mathbf{R}$ and $\mathbf{T}$ is

$$\ell(\mathbf{R}, \mathbf{T}|\mathbf{k}) \propto \prod_{i=1}^{K} \text{Multi}(\mathbf{r}(\mathbf{Y}_i); \mathbf{K}_i, \mathbf{p}(\mathbf{Y}_i))\text{Gamma}(\mathbf{T}_i; \mathbf{K}_i, \alpha_0(\mathbf{Y}_i)),$$

where again $\mathbf{Y}_i \in \mathcal{S}$ and the propensities α_i and probabilities p_i depend on the reaction rates $\mathbf{k} = [k_1, k_2, \dots k_{N_r}]^\top$ through the concept of mass action kinetics given in (6.1).

Our choice of Gamma prior distributions with hyper-parameters (a_i, b_i) results in the posterior distribution of the form

$$\begin{aligned} \pi(\mathbf{k}|\mathbf{R}, \mathbf{T}) &\propto \ell(\mathbf{R}, \mathbf{T}|\mathbf{k}) \prod_{i=1}^{N_r} \text{Gamma}(k_i; a_i, b_i) \\ &\propto \exp\left\{ \sum_{i=1}^{K} \left[\mathbf{K}_i \log \alpha_0(\mathbf{Y}_i; \mathbf{k}) - \mathbf{T}_i \alpha_0(\mathbf{Y}_i; \mathbf{k}) + \sum_{j=1}^{N_r} \mathbf{R}_{ij} \log \mathbf{p}_j(\mathbf{Y}_i; \mathbf{k}) \right] \right. \\ &\qquad \left. + \sum_{i=1}^{N_r} ((a_i - 1) \log k_i - b_i k_i) \right\}. \end{aligned} \tag{6.5}$$

6.2. Approximation of likelihoods in multiscale chemical networks. Suppose now that we are only observing the slower variables in a multiscale chemical system of this type. Suppose that $n_r < N_r$ is the number of slow reactions in the system, and the slow reactions are given by $\{R_{s_1}, R_{s_2}, \dots R_{s_{n_r}}\} \subset \{R_1, R_2, \dots, R_{N_r}\}$. The propensities of the slow reactions α_{s_i} may depend on the value of the fast variables which is unknown. However, the invariant distribution of the fast variables conditioned on the slow variables in the system can be approximated using a multiscale method, such as the QSSA or the CMA, as described briefly in Section 6. Once the approximations have been made, we arrive at approximate *effective* propensities $\bar{\alpha}_{s_i}$, which have been averaged over the computed invariant distributions of the fast variables. Then the approximate effective dynamics on the slow variable $S(t)$ are given by

$$S(t) = S(0) + \sum_{j=1}^{n_r} \nu_{s_j} \int_0^t \bar{\alpha}_{s_j}(S(s)) ds. \tag{6.6}$$

In turn, the approximate posterior distribution on the reaction parameters $\mathbf{k}$ is then given by

$$\begin{aligned}\pi(\mathbf{k}|\mathbf{R},\mathbf{T}) &\propto \ell(\mathbf{R},\mathbf{T}|\mathbf{k})\prod_{i=1}^{N_r}\mathrm{Gamma}(k_i;a_i,b_i)\\ &\propto \exp\left\{\sum_{i=1}^{K}\left[\mathbf{K}_i\log\alpha_0(\mathbf{Y}_i;\mathbf{k})-\mathbf{T}_i\alpha_0(\mathbf{Y}_i;\mathbf{k})+\sum_{j=1}^{N_r}\mathbf{R}_{ij}\log\mathbf{p}_j(\mathbf{Y}_i;\mathbf{k})\right]\right.\\ &\qquad\left.+\sum_{i=1}^{N_r}((a_i-1)\log k_i-b_ik_i)\right\}. \qquad (6.7)\end{aligned}$$

7. Numerical Examples.

In this section we look at two examples of chemical systems to demonstrate the effectiveness of the Bayesian approach we have proposed, along with the transport PAIS algorithms for sampling the challenging posterior distribution on the reaction parameters that arise.

7.1. A Multiscale Chemical System.

First we consider a simple multiscale example involving two chemical species S_1 and S_2 involved in only zeroth and first order reactions:

$$\emptyset \xrightarrow{k_1} S_1 \underset{k_3}{\overset{k_2}{\rightleftarrows}} S_2 \xrightarrow{k_4} \emptyset. \qquad (7.1)$$

Each arrow represents a reaction from a reactant to a product, with some rate constant k_i, and where mass action kinetics is assumed. The parameters k_i are non-negative, and $\mathbf{k} = [k_1,\dots,k_4]^\top \in \mathbb{R}^4_+ = \mathcal{X}$. We denote the population count of species S_i by $X_i \in \mathbb{N}_0$. We assume that we are in a parameter regime such that the reactions $R_2\colon S_1 \to S_2$ and $R_3\colon S_2 \to S_1$ occur more frequently than the other reactions, $R_1\colon \emptyset \to S_1$, and $R_4\colon S_2 \to \emptyset$. Notice that both chemical species are involved in fast reactions. However, the quantity $S = X_1 + X_2$, is conserved by both of the fast reactions, and as such, this is the slowly changing quantity in this system.
We assume that we are only able to observe the slow variable $S(t)$. The effective dynamics of S can be approximated by a system of only two reactions:

$$\emptyset \xrightarrow{\alpha_1(S)} S \xrightarrow{\bar{\alpha}_4(S)} \emptyset. \qquad (7.2)$$

The effective propensity $\bar{\alpha}_4(S)$ can be approximated through application of a multiscale approximation, as detailed in Section 6, and in more detail in ****CITE****. Under the QEA, this is given by

$$\bar{\alpha}_4^{\mathrm{QEA}}(s) = \frac{k_2k_4s}{k_2+k_3}.$$

Similarly, the analysis of the constrained system as discussed in [4] yields the effective propensity

$$\bar{\alpha}_4^{\mathrm{CMA}}(s) = \mathbb{E}_{\mathrm{CMA}}\left[k_4X_2|S=s\right] = \frac{k_2k_4s}{k_2+k_3+k_4}. \qquad (7.3)$$

Our observations are uninformative about the reaction rates k_2, k_3, and k_4, as there is a sub-manifold $\mathcal{M} \subset \mathcal{X}$ along which the effective rate $\hat{k}_4$ is invariant, leading to a highly ill-posed inverse problem. This type of problem is notoriously difficult

parameter/dimension	1	2	3	4
α_i	150	5	5	3
β_i	15/9	5/12	5/12	1

TABLE 7.1
Hyper-parameters in the prior distributions for the multiscale problem described in Section 7.1.

to sample from using standard MH algorithms, as the algorithms quickly gravitate towards the manifold $\mathcal{M}$ but then exploration around $\mathcal{M}$ is slow.

We aim to recover three different posterior distributions. In the first, which is of the form (6.5) we assume that we can fully and exactly observe the whole state vector for the whole of the observation time window. In the second and third, the observations are restricted to observations of the slow variable $S = X_1+X_2$. In these two examples, we approximate the posterior using (6.7) with QEA and CMA approximations of the effective dynamics respectively.

In all three cases, we find the posterior distribution for $\mathbf{k} \in \mathcal{X}$. These four parameters are assigned Gamma prior distributions,

$$k_i \sim \text{Gamma}(\cdot; \alpha_i, \beta_i), \quad \text{for} \quad i = 1, \ldots, 4.$$

The hyper-parameters corresponding to each of these prior distributions are given in Table 7.1. These priors are the same for each of the three posterior distributions.

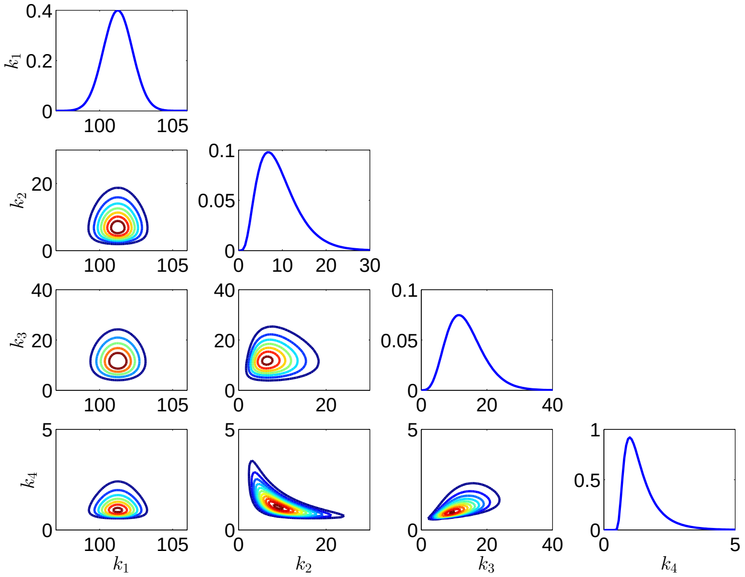


FIG. 7.1. *Posterior for the constrained multiscale problem outlined in Section 7.1.*

Figure 7.1 shows how this posterior looks when we use the CMA to model the effective degradation propensity $\bar{\alpha}_4$.

We consider several different algorithms for sampling from these distributions. First we implement both the PAIS and MH algorithms with a Gaussian proposal distribu-

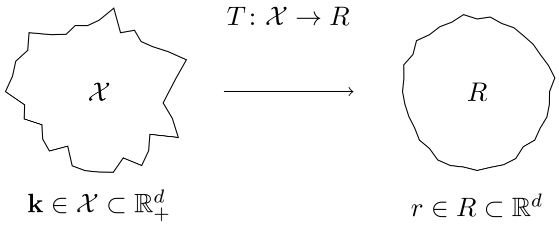

$T\colon \mathcal{X} \to R$
$\mathcal{X}$
R
$\mathbf{k} \in \mathcal{X} \subset \mathbb{R}^d_+$
$r \in R \subset \mathbb{R}^d$

Fig. 7.2. *Couplings between parameter space and reference when using the transport map proposal method to propose moves on parameter space.*

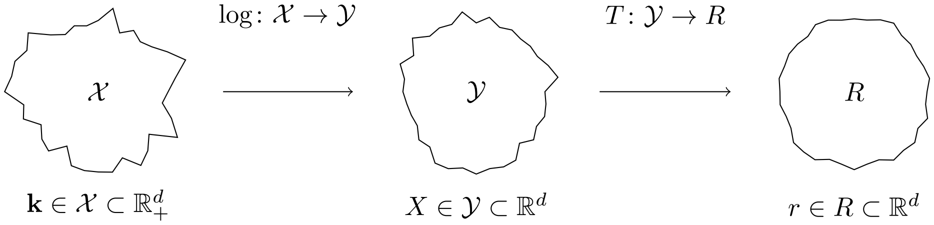

$\log\colon \mathcal{X} \to \mathcal{Y}$
$T\colon \mathcal{Y} \to R$
$\mathcal{X}$
$\mathcal{Y}$
R
$\mathbf{k} \in \mathcal{X} \subset \mathbb{R}^d_+$
$X \in \mathcal{Y} \subset \mathbb{R}^d$
$r \in R \subset \mathbb{R}^d$

Fig. 7.3. *Couplings between parameter space, an intermediate space and reference space when using the transport map proposal method to propose moves on a log space.*

tion. In the case of the PAIS algorithm, this is a Gaussian mixture proposal distribution, and in the MH algorithm this is equivalent to a vanilla random walk. The proposal distribution uses a covariance matrix which has been constructed using the sample covariance of a sample produced using a standard RWMH algorithm. We also compare the PAIS and MH algorithms when using a transport map proposal distribution. This proposal method was discussed in detail in Chapter 4.

Figure 7.2 shows how we define a bijective map, $\hat{T}$, between parameter space $\mathcal{X}$ and a reference space R. When using the transport map proposal distribution, we prefix the proposal method with a T, e.g. MH-RW (RWMH) and MH-TRW, as well as PAIS-RW and PAIS-TRW. In practice we cannot ensure that the approximate map $\tilde{T}$ is uniquely invertible over the whole of R and so $\tilde{T}$ is not truly bijective. This leads to problems for our strictly positive state space, $\mathcal{X} \subset \mathbb{R}^d_+$, since proposals in R do not necessarily map back onto $\mathcal{X}$. This motivates the use of an intermediate space.

We also consider how these algorithms perform when they are applied to the log of the reaction rates. We choose to use this log transformation since it converts our strictly positive parameter space, $\mathcal{X}$, into an intermediate space $\mathcal{Y} \subset \mathbb{R}^d$. This allows us to define T between two subsets of $\mathbb{R}^d$, which means that even if $\tilde{T}$ is not uniquely invertible in some region of R, all possibilities are valid proposals. It also reduces the work required of the map, which since we use only a finite dimensional approximation, can improve performance and stability. As before, the proposal distributions are labelled with a T for transport map and when using the intermediate space we prepend 'log' to the proposal method, e.g. MH-logRW and MH-logTRW, with, PAIS-logRW and PAIS-logTRW.

Figure 7.3 displays the composition of maps when we include this intermediate space $\mathcal{Y}$. Every proposal in R results in a valid proposal on $\mathcal{X}$. The inclusion of this additional map means that we must again alter our importance weight definition to reflect the pullback from R through $\mathcal{Y}$. The weight is now

$$w_i(\theta) = \frac{\pi(\theta|\mathbf{R}, \mathbf{T})}{\chi(\tilde{T} \circ \log(\theta)|\tilde{T} \circ \log(\theta^{(i-1)}))|J_{\tilde{T}\circ\log}(\theta)|},$$

where θ is a proposed new state, $\theta^{(i-1)}$ is the ensemble of states from the previous iteration, and $J_{\tilde{T}\circ\log}(\theta)$ is the Jacobian of the composition of the two maps. This Jacobian is straightforward to calculate,

$$|J_{\tilde{T}\circ\log}(\theta')| = |J_{\tilde{T}}(\log(\theta'))||J_{\log}(\theta')|,$$

where the first determinant is as we saw in the previous chapter, and the second is

$$|J_{\log}(\theta')| = \prod_{i=1}^{d} \frac{1}{\theta'_i}.$$

For this problem, we continue to use monomials in each dimension in our transport map construction. We use polynomials of total order $p = 3$ as the basis functions, i.e.

$$T_i(\theta) = \sum_{\mathbf{j}\in\mathcal{J}_i^{\mathrm{TO}}(p)} \gamma_{i,\mathbf{j}}\psi_{\mathbf{j}}(\theta) \quad \text{where} \quad \psi_{\mathbf{j}}(\theta) = \prod_{k=1}^{i} \theta_k^{j_k},$$

and

$$\mathcal{J}_i^{\mathrm{TO}}(p) = \{\mathbf{j} \in \mathbb{N}_0^d \mid \|\mathbf{j}\|_1 \leq p, \text{ and } j_k = 0 \; \forall k > i\}.$$

We will use the AMR***CITE*** resampler with an ensemble size of $M = 500$. This increase in ensemble size in comparison with previous example in Section 5 is to allow for the increase in parameter dimension.

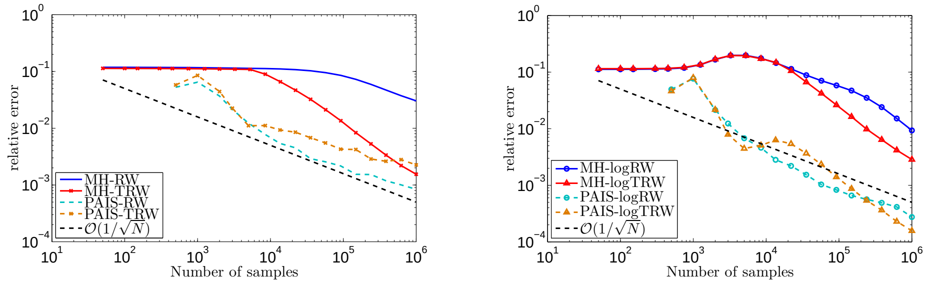
To measure the convergence of the sampling methods in this section, we will compare the convergence of the mean of each parameter. We approximate $\mathbb{E}(\mathbf{k})$ using 2.4 billion samples from the MH-RW algorithm. We do this for each of the eight algorithms we have so far discussed. Convergence is shown only for the CMA approximation of the posterior (6.7) with effective rate for the degradation of S given by (7.3), but we expect very similar results for the other posterior distributions discussed.

Algorithm	MH	PAIS	
	$\delta_{\%}$	δ_{ESS}	ESS
RW	5.5e-3	1.0e-0	9.0e-3
TRW	1.2e-0	4.0e-1	1.2e-3
logRW	1.7e-1	1.2e-0	6.0e-2
logTRW	2.7e-2	1.5e-1	3.5e-1

Table 7.2

Optimal scaling parameters for the MH and PAIS algorithms applied to the constrained multiscale problem in Section 7.1. MH parameters optimised by acceptance rate, and PAIS parameters optimised using effective sample size.

The optimal scaling parameters for the proposal distributions are given in Table 7.2. We note that for MH-TRW and PAIS-TRW the scaling parameter is near to 1, particularly for MH-RW, this is what we should expect since the proposal is made on a reference space which should be near to $\mathcal{N}_d(0, \mathrm{I})$. The same should be true for the MH-logTRW and PAIS-logTRW algorithms since these have the same target reference space, however the optimal scaling parameters here are much smaller. We see that the ESS is higher for the algorithms which sample on $\mathcal{Y}$, and we expect that convergence will be fastest for the PAIS-logTRW algorithm.



(a) Sampling algorithms on $\mathcal{X}$.

(b) Sampling algorithms on $\mathcal{Y}$.

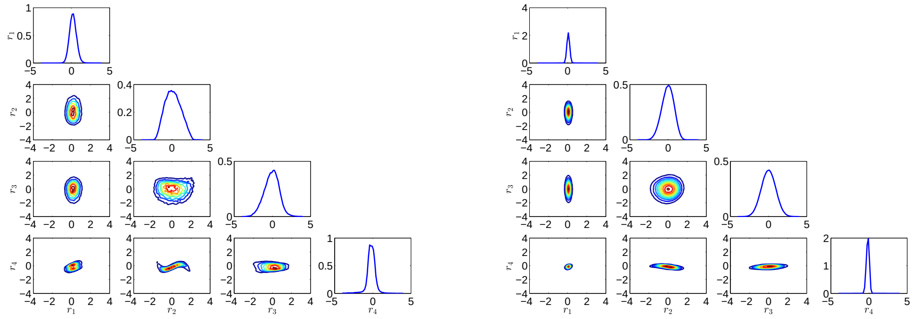
FIG. 7.4. *Convergence of the constrained multiscale example described in Section 7.1.*

Convergence of the eight algorithms for this example is shown in Figure 7.4. We first note the poor performance of the MH based algorithms, each of them taking roughly 10,000 samples to begin converging. Only the MH-TRW is at all competitive with the PAIS algorithms. During the simulation interval, the MH-TRW algorithm has not settled down to the expected $\mathcal{O}(1/\sqrt{N})$ rate which means that the estimate is still biased by the long burn-in time. As we have seen in previous examples, the burn-in time for the PAIS algorithm is negligible.

The PAIS variants with RW and TRW proposals perform similarly on both sample spaces. When sampling on $\mathcal{X}$ the transport map is not quite as efficient, largely due to the difficulties discussed in the previous section i.e. many proposals are made which do result in negative reaction rates. Sampling on $\mathcal{Y}$ leads to more comparable Monte Carlo errors, with the logTRW being apparently slightly less stable. This proposal method becomes more stable as we increase either the ensemble size, or the number of iterations between updates of the transport map, T. Overall we see the smallest Monte Carlo errors for a given amount of computational effort coming from the PAIS-logTRW algorithm.

We now look at the proposal distributions of the transport map accelerated algorithms. In Figure 7.5, we see the reference spaces found by; in (a) mapping the posterior through the map $\hat{T}$, and in (b) by mapping the posterior through $T \circ \log$. For the most part, each of these marginal distributions can be recognised as a Gaussian. However, with the exception of $\mathbb{P}(r_2, r_3)$, we would not consider them to be close to a standardised $\mathcal{N}(0, \mathrm{I})$ distribution. Before thinking that the transport map has not helped us to find a 'nicer' space on which to propose new values we should consider that the dimensions are now (1) largely uncorrelated, and (2) the variances in each dimension are much more similar than they are in Figure 7.1.

Particularly in Figure 7.5 (b) we see that var(r_1) and var(r_4) are much smaller than var(r_2) and var(r_3). To combat this we have a number of choices, we might wish to use two different scaling parameters to match these scales, which would require knowledge of the reference space before beginning sampling. We could alternatively use a proposal distribution such as in the LPAIS algorithm to adaptively learn this at each iteration. Another option is to increase the total order of our index set. For



(a) Reference space mapped to by $\hat{T}$.

(b) Reference space mapped to by $T \circ \log$.

Fig. 7.5. *Reference space for the problem described in Section* **??** *found using the TRW and logTRW proposal distributions. Components are linked by the relation $r_i = T_i'(k_i)$ in (a) and $r_i = T_i \circ \log(k_i)$ in (b).*

these numerics we have chosen $p = 3$, but we know that we can obtain reference spaces which are closer to $\mathcal{N}_d(0, \mathrm{I})$ by choosing a larger p.

7.1.1. Comparison of the Constrained and QEA approaches. The convergence analysis has been performed for the constrained approach to this multiscale system. We now look at the differences between the constrained and QEA posterior distributions. Recall that the approaches differed only in the form of the effective degradation rate $\hat{k}_4$,

$$\hat{k}_4^{\mathrm{QEA}}(s) = \frac{k_2 k_4 s}{k_2 + k_3} \quad \text{and} \quad \hat{k}_4^{\mathrm{CMA}}(s) = \frac{k_2 k_4 s}{k_2 + k_3 + k_4}.$$

This difference in the denominator causes a shift in the parameters as can be seen in Figure 7.6. The figure shows the difference in posteriors,

$$\mathrm{diff}(\mu^{\mathrm{CMA}}, \mu^{\mathrm{QEA}}) = \pi^{\mathrm{CMA}}(\mathbf{k}|\mathbf{R}, \mathbf{T}) - \pi^{\mathrm{QEA}}(\mathbf{k}|\mathbf{R}, \mathbf{T}). \tag{7.4}$$

Since the two posteriors have been approximated using an MCMC sample, there is a significant amount of noise, particularly in the tails of the distributions We can see that the differences in the marginals for k_1, k_2, and k_3 are relatively small and the differences are largely positive. These positive differences tell us that the constrained approach claims to be more informative about the values of these first three parameters. However the marginal for k_4 varies by a significant amount and is more negative. This means that the mean estimates for k_4 differ by a large amount, the posterior for the QEA is most peaked.

We now consider how we should interpret the information given by these models. The QEA assumption tells us that we cannot observe the parameters k_2, k_3 and k_4 independently from the reduced data set, but we can observe the quantity $\hat{k}_4^{\mathrm{QEA}} = k_2 k_4/(k_2 + k_3)$. Similarly, the constrained approach tells us that we are able to observe the quantity $\hat{k}_4^{\mathrm{CMA}} = k_2 k_4/(k_2 + k_3 + k_4)$. To validate our inferences on k_2, k_3 and k_4 we would like to discover which model is most accurate and informative about these parameters, and which model gives us results which are most similar to what we can obtain from the full model, with all reactions observed.

A conventional way to compare two models under a Bayesian framework is to calculate the Bayes factors [3]. The Bayes factor, $B_{1,2}$, between two models, $\mathcal{M}_1$ and $\mathcal{M}_2$, can

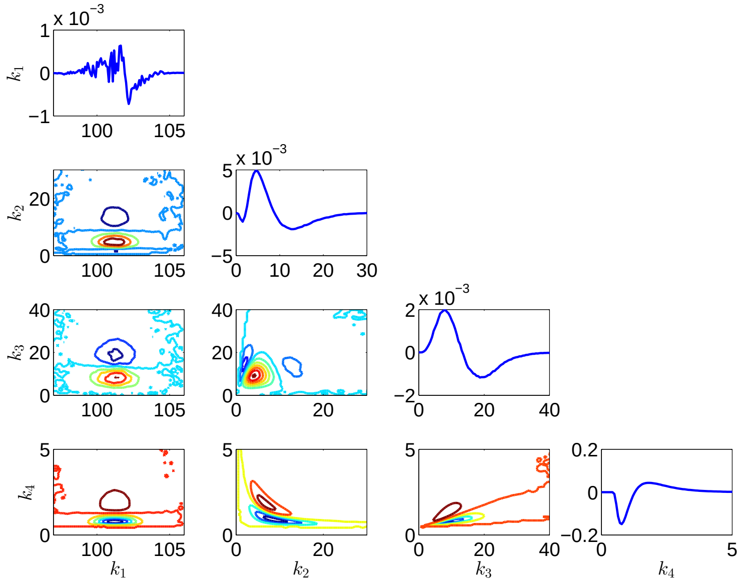


FIG. 7.6. *Difference between the CMA and QEA posteriors as defined in Equation* (7.4).

be interpreted as a ratio of the normalisation constants of the posterior distributions given each model,

$$B_{1,2} = \frac{\mathbb{P}(D|\mathcal{M}_1)}{\mathbb{P}(D|\mathcal{M}_2)}, \quad \text{where} \quad \mathbb{P}(D|\mathcal{M}_k) = \int_{\mathcal{X}} \mathbb{P}(D|\theta_k, \mathcal{M}_k)\mathbb{P}(\theta_k|\mathcal{M}_k)\,\mathrm{d}\theta_k.$$

Under the PAIS framework, it is straightforward to calculate these factors using the Monte Carlo estimator for the normalisation constants. From [15] these normalisation constants take the form

$$Z_k \approx \hat{Z}_k \frac{1}{NM} \sum_{i=1}^{N} \sum_{j=1}^{M} w_j^{(i)}(\mathcal{M}_k),$$

where $w_j^{(i)}(\mathcal{M}_k)$ is the weight under model $\mathcal{M}_k$ corresponding to the j-th ensemble member on the i-th iteration. Hence, $B_{1,2} \approx \hat{Z}_1/\hat{Z}_2$. We can compare more than two models in this way by selecting the model with the largest marginal distribution as the best model.

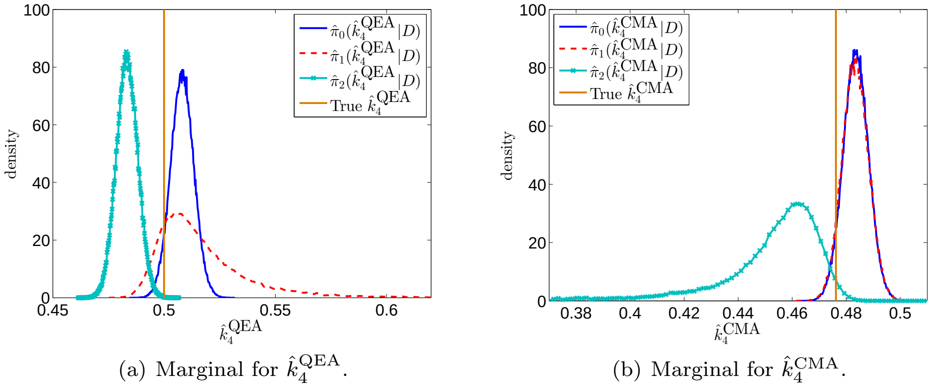
We now label the constrained model as $\mathcal{M}_1$, the model arising from the QEA as $\mathcal{M}_2$, and the full data model as $\mathcal{M}_0$. Under $\mathcal{M}_1$, the parameter $\theta_1 = (k_1, \hat{k}_4^{\mathrm{CMA}})^\top$, and under $\mathcal{M}_2$, the parameter $\theta_2 = (k_1, \hat{k}_4^{\mathrm{QEA}})^\top$. The full model $\mathcal{M}_0$ observes all four parameters $\theta_0 = (k_1, k_2, k_3, k_4)^\top$. The marginal densities for the data, evaluated at the observed data, given the models are displayed in Table 7.3.

Computing the Bayes factors from Table 7.3 we see that we should of course prefer the model in which we observe all reactions perfectly, however this model is not significantly better than the CMA model ($B_{0,1} = 2.09 < 3.2$, [10]). Again the constrained model is not substantially more attractive than the QEA model when we

k	$\mathbb{P}(D = (\mathbf{R}, \mathbf{T})^\top \vert \mathcal{M}_k)$
0	6.8e-3
1	3.2e-3
2	1.7e-3

Table 7.3
Marginal distributions for the data $(\mathbf{R}, \mathbf{T})^\top$ *for each model considered in Section* **??**.

consider the Bayes factor $B_{1,2} = 1.96 < 3.2$. The Bayes factor $B_{0,2} = 4.1 > 3.2$ does tell us that we should significantly prefer the full model to the QEA approximation of the posterior. These Bayes factors present a weak argument that that the constrained model provides us with a better description of the data than the QEA model. We can also present this information graphically, which might provide us with a greater justification for preferring the constrained approximation of the posterior.



(a) Marginal for $\hat{k}_4^{\text{QEA}}$.

(b) Marginal for $\hat{k}_4^{\text{CMA}}$.

Fig. 7.7. *Comparison of the approximate marginal densities for the 'observable parameter'* $\hat{k}_4$ *under models* $\mathcal{M}_1$ *and* $\mathcal{M}_2$. *Marginals for these two parameters are approximated using samples which have been produced by targeting the three densities* $\pi_i = \mathbb{P}(\theta_i|D, \mathcal{M}_i)$ *for* $i = 0, 1, 2$.

Figure 7.7 displays the marginal distributions for the parameters $\hat{k}_4^{\text{QEA}}$ and $\hat{k}_4^{\text{CMA}}$. For each of these two observable parameters we obtain approximations for three marginal distributions, $\pi_i(\cdot|D, \mathcal{M}_i)$, $i = 0, 1, 2$. Each of these approximations has been produced by marginalising a sample drawn from the full posterior $\pi_i(\mathbf{k}|D, \mathcal{M}_i)$ defined in terms of the respective model $\mathcal{M}_i$. In other words, to calculate $\hat{\pi}_1(\hat{k}_4^{\text{QEA}}|D)$, we first draw a sample using the PAIS algorithm targeting the posterior density $\pi_1(\mathbf{k}|D, \mathcal{M}_1)$. We then calculate $\theta^{(j)} = k_2^{(j)} k_4^{(j)}/(k_2^{(j)} + k_3^{(j)})$ for each sample produced. Finally we produce a histogram from this sample $\theta^{(j)}$.

The first thing to note is that in subfigure (a) when we approximate the marginals for $\hat{k}_4^{\text{QEA}}$ the density $\hat{\pi}_2(\cdot|D, \mathcal{M}_2)$ is much more peaked than $\hat{\pi}_1(\cdot|D, \mathcal{M}_1)$. As we might expect the same is true in reverse in subfigure (b). What is interesting here is that for both parameters, the marginal distribution which assumes $\mathcal{M}_2$ assigns a small density value to the true value of the parameter, while the marginals which assume $\mathcal{M}_1$ assign a relatively high density to the truth. We also draw attention to the similarities between the constrained model and the full data model. In subfigure (a) these two marginals are peaked around a similar value, and in subfigure (b) the full data and

NEW FIGURE HERE

constrained models almost exactly coincide. This supports the hypothesis that the constrained approach provides a more accurate approximation to the dynamics of the full system.

7.2. Gene Regulatory Networks. In this example, we look at a model of a gene regulatory network (GRN) mechanism [2,7,9] used by a particular cell to regulate the amount of a protein, P, present. In this model, we again have reactions occurring at particular rates, k_i, between four different species. Here there is a special species, G, representing the gene responsible for building the protein, of which there is only one at a time, and this species can either be turned on or off. When the gene is off we denote it as G'.

$$\begin{aligned} P+P &\underset{k_2}{\overset{k_1}{\rightleftarrows}} D & \qquad G+D &\underset{k_4}{\overset{k_3}{\rightleftarrows}} G' \\ G &\xrightarrow{k_5} G+M & \qquad M &\xrightarrow{k_6} M+P \\ P &\xrightarrow{k_7} \emptyset & \qquad M &\xrightarrow{k_8} \emptyset \end{aligned} \tag{7.5}$$

The eight reactions are given in Equation (7.5). In addition to the gene, G, and protein, P, the system also involves molecules of mRNA, M, and dimers, D, which are two proteins joined together.

In this example protein is created by a molecule of mRNA at a rate k_6. The amount of protein in the cell is then governed by the rate of decay of P, and the number of mRNA molecules in the cell creating P. In turn the number of mRNA molecules is controlled by its own decay rate and whether or not the gene G, which produces the mRNA, is active. The gene can be turned on and off depending on how many dimers are present in the system, which is related to how many protein molecules there are. This completes a feedback cycle whereby if the protein population gets too high then the gene will turn off and the rate of new protein formation will decrease, causing the level to drop and the gene to turn back on. Figure **??** shows what happens to the molecules in the system as the gene turns on and off.

Due to the small scale of the system and the speed of the reactions, we are only able to measure the total amount of protein, T, and mRNA, M, while the number of dimers and whether or not the gene is switched on must be considered to be missing data. For this scenario, the likelihood function is intractable and so we must use a multiscale approximation of the system. Here we choose the constrained multiscale approximation due to its success in the previous example.

The formulation here is slightly more complex. We define a fast subsystem which acts on the fast variables $\mathcal{F} = [P, D, G]$ and incorporates reactions R_1-R_4. For the true parameter values we have chosen, 90% of reactions which occur in a particular time frame are one of these four reactions.

$$\text{Fast subsystem:} \quad P+P \underset{k_2}{\overset{k_1}{\rightleftarrows}} D, \quad G+D \underset{k_4}{\overset{k_3}{\rightleftarrows}} G'.$$

We also define the slow variables, $T = P + 2D + 2(1-G)$ and M which are held constant by these fast reactions.

When we observe a cell, we cannot distinguish between the fast species and so don't know which of the first four reactions are happening. Due to this, we observe how many of reactions R_5-R_8 occur for each combination of the two slow variables $\mathcal{S} = [T, M]$. Since we don't observe P and G for every reaction, the propensities must be altered to take this into account,

$$\begin{aligned} \hat{\alpha}_5 &= k_5 \mathbb{E}[G|T = t], \qquad & \alpha_6 &= k_6 M, \\ \hat{\alpha}_7 &= k_7 \mathbb{E}[P|T = t], \qquad & \alpha_8 &= k_8 M. \end{aligned} \tag{7.6}$$

To calculate these expectations, we note that the fast subsystem is a deficiency zero system [1], and is also reversible. i.e. The system has four complexes,

$$C_1 = P + P, \quad C_2 = D, \quad C_3 = G + D \quad \text{and} \quad C_4 = G',$$

there are two linkage classes (or two sets of reversible reactions), and the space spanned by the stoichiometries is two dimensional,

$$\text{span}\left\{\pm[-2, 1, 0, 0]^\top, \pm[0, -1, 0, -1]^\top\right\},$$

hence the deficiency is $d = 4 - 2 - 2 = 0$. This means that, by [1], the conditional probabilities have the closed form equation,

$$\mathbb{P}[\mathcal{F}|T = t] \propto \text{Poisson}(P_t; \bar{p})\text{Poisson}(D_t; \bar{d})\text{Poisson}(G_t; \bar{g})\text{Poisson}(1 - G_t; 1 - \bar{g}),$$

from which we can calculate the expected values of P_t, D_t, and G_t. The fast variables P_t, D_t and G_t are constrained to the line defined by $T = P + 2D + 2(1 - G) = t$. The concentrations $\bar{p}$, $\bar{d}$ and $\bar{g}$, must be calculated as the solution to the nonlinear system produced by examining the steady state behaviour of the fast subsystem,

$$\begin{aligned} T &= p + 2d + 2(1 - g), \\ \frac{\mathrm{d}p}{\mathrm{d}t} &= -2k_1 p^2 + 2k_2 d = 0, \\ \frac{\mathrm{d}d}{\mathrm{d}t} &= k_1 p^2 - k_2 d - k_3 g d + k_4(1 - g) = 0, \\ \frac{\mathrm{d}g}{\mathrm{d}t} &= -k_3 g d + k_4(1 - g) = 0, \end{aligned} \tag{7.7}$$

subject to $p \in [0, T]$, $g \in [0, 1]$, and $0 \leq d \leq \text{floor}(T/2)$.
With these effective propensities, given in Equation (7.6), we can calculate the posterior distribution as given in Equation (6.5), again assigning Gamma priors to the eight reaction rates. We note that the reaction rates for the fast variables do not appear in the posterior density explicitly which only uses the propensities α_5-α_8, however they appear through the solution to the nonlinear system in Equation (7.7).

7.2.1. Target Distribution. As in the previous chemical system example we use the target distribution as defined in Equation (6.5). The application here is slightly more involved as discussed previously in Section 7.2. We have an eight dimensional parameter space, with one reaction rate corresponding to one of eight reactions. The first four reactions, R_1-R_4, are combined into a fast subsystem for which we do not observe the time or identifier for which reaction has fired. We instead use the constrained multiscale approach to approximate what has happened in this subsystem between occurrences of the slow reactions firing (reactions R_5-R_8). For this reason, we apply the form in Equation (6.5) to reactions R_5-R_8, and allow the rates for reactions R_1-R_4 to implicitly influence the posterior density through the effective propensities, $\hat{\alpha}_5$ and $\hat{\alpha}_7$.

As we mentioned earlier, for this problem we only consider the CMA's approximation to the system dynamics.

The priors for this problem were chosen to be fairly uninformative with respect to the likelihood function for each reaction rate. A list of the hyper-parameters corresponding to each Gamma prior can be found in Table 7.4.

Dimension i	1	2	3	4	5	6	7	8
α_i	2	100	100	3	3	3	2	2
β_i	50	0.02	1	1	0.6	1	50	50

Table 7.4

Hyper parameters for the Gamma priors on each of the reaction rates in the GRN example in Section 7.2.

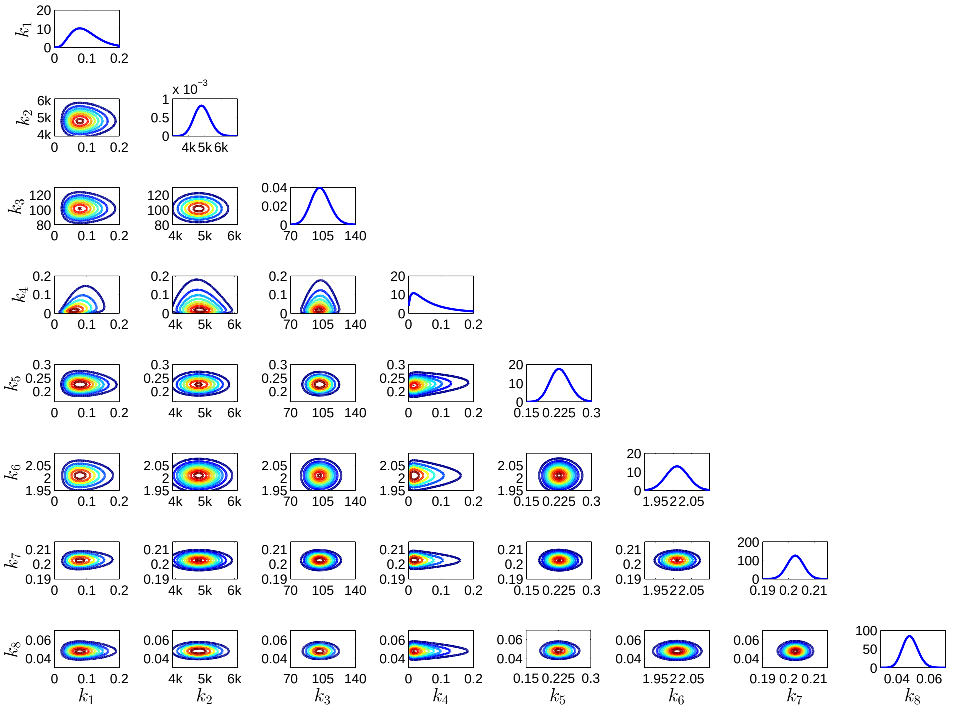


Fig. 7.8. *Marginal densities of the posterior distribution for the GRN example in Section 7.2.*

The one- and two-dimensional marginal distributions for the full posterior distribution are displayed in Figure 7.8. The posterior does not contain many interesting correlation structures, however several dimensions have leptokurtic tails which are difficult for the standard PAIS algorithm to sample from. The marginal densities also vary over very different scales, which might require us to use a variant of the LPAIS algorithm.

7.2.2. Implementation. We apply the eight RW and logRW proposal distributions, with and without transport map acceleration, which were discussed in Section 7.1 to this posterior. Again the intermediate log space is required so that our transport map is a function $T\colon \mathbb{R}^d \to \mathbb{R}^d$, rather than a map from $\mathbb{R}^d_+$ to $\mathbb{R}^d$.

The transport map setup has not changed from the previous example. However, due to the scaling of the weight we have had to be more careful about which samples we use to build our map. We require that the objective function $C(T)$, from Equation (3.7) is convex, which requires that the second derivative, given in Equation (3.11), is positive definite. This positive definite property depends on all weights in our sample being strictly positive, which is not always possible on a computer. For this reason we filter out any samples from the optimisation problem where the weight is a numeric zero. This does not affect the validity of the map since a weight zero would not contribute to the expectation, and we do not require (and could not enforce) an exact map.

In all eight algorithms we adapt the proposal variances during the burn-in phase of the algorithms to avoid behaviour like that seen in Example **??** where we needed a huge ensemble size to approximate the posterior with our mixture of isotropic Gaussian kernels.

We again use the AMR resampler, this time with an ensemble size of $M = 2500$. The increase in ensemble size again compensates for the increase in dimension (from four to eight parameters.)

As in the previous chemical reaction example, we measure convergence of the algorithms only through the convergence of the mean, for computational ease. We use the sample Mahalanobis distance, with 'true' covariance and 'true' mean built from a sample of size 2.4 billion using the MH-RW algorithm.

7.2.3. Convergence results. The scaling parameter selection is done by optimising the acceptance rate for the MH algorithms, and optimising the effective sample size for the PAIS algorithms. The optimal parameters are given in Table 7.5.

	MH				PAIS			
	RW	logRW	TRW	log-TRW	RW	logRW	TRW	log-TRW
$\beta^*_{\%}$	1.0e-1	3.9e-1	1.0e-0	6.0e-1	-	-	-	-
β^*_{ESS}	-	-	-	-	1.0e-0	1.3e-0	8.0e-1	5.0e-1
ESS/M	-	-	-	-	3.5e-4	4.7e-2	4.8e-2	1.6e-1

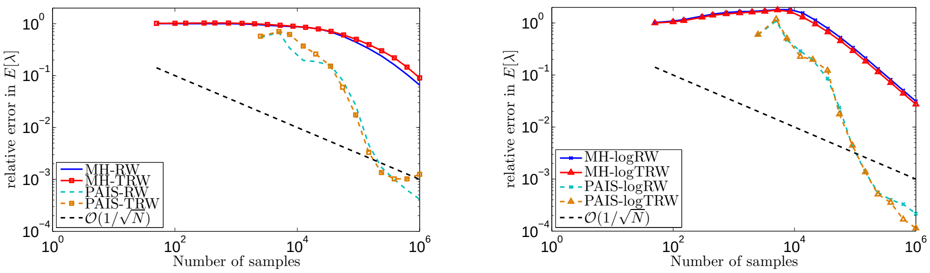
Table 7.5

Scaling parameters for the sampling algorithms applied to the GRN example in Section 7.2.

From the effective sample sizes shown in Table 7.5 we can see the improvement to the efficiency of the algorithm both by proposing on the log space, and by transforming the parameter space into something closer to Gaussian.

Due to the numerical cost of calculating the full relative L^2 errors for this posterior, we quantify the convergence speeds of these algorithms using the sample Mahalanobis distance between the sample mean and the 'true' mean. This convergence analysis is shown in Figure 7.9.

The first thing to note is that the first PAIS measurement happens after 2500 samples, whereas the first MH measurement occurs after 50 samples. This is due to the much larger ensemble size required to sample in higher dimensions. We believe that an ensemble size of 2500 is over-cautious in this example and we could have used a smaller ensemble size. We also think that the required ensemble size to sample from



(a) Sampling algorithms on $\mathcal{X}$.

(b) Sampling algorithms on $\mathcal{Y}$.

FIG. 7.9. *Convergence of the GRN example described in Section 7.2.*

these posteriors is reduced by the use of the transport map. This is due to the way the tail behaviour is improved by the transformation.

The second obvious feature of these convergence plots is that the PAIS algorithms outperform the MH algorithms by a large margin - roughly a reduction of two orders of magnitude in the relative error over the same number of samples. A less positive result is that for this example the PAIS-TRW algorithm has found a map which has prevented further convergence of the mean. This is likely because one dimension of the transport map is sending proposals onto the negative real line and so all samples are being assigned a zero weight. This is behaviour which was anticipated, and it is always suggested that the transport map is constructed as a map $T\colon \mathbb{R}^d \to \mathbb{R}^d$ rather than $T\colon \mathcal{X} \to \mathbb{R}^d$ where $\mathcal{X} \subset \mathbb{R}^d$. The large increase in the effective sample size observed between the PAIS-logRW and PAIS-logTRW algorithms is converted into an estimate which is twice as accurate after 1 million samples.

A similar pattern is seen in the MH algorithms, where the MH-TRW performs the worst, and the MH-logTRW algorithm performs the best, although only marginally.

8. Discussion.

8.1. Sampling in (moderately) higher dimensions.

One major problem with importance sampling schemes is the curse of dimensionality, which means that methods such as PAIS, and other PMC methods, can only be used for relatively low dimensional problems. Here, we will briefly discuss how transport maps could aid with making moderately higher dimensional problems accessible to this family of methods. 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 can be implemented very efficiently. As described in [14], 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****CITE OUR OTHER PAPER****.

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 $i = 1, \dots, M$.
Set $c \leftarrow 0$
for $i \leftarrow 1, \dots, M$ 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 $y$.
```

REFERENCES

[1] D. Anderson, G. Craciun, and T. Kurtz, *Product-form stationary distributions for deficiency zero chemical reaction networks*, Bulletin of mathematical biology, 72 (2010), pp. 1947–1970.

[2] A. Becskei and L. Serrano, *Engineering stability in gene networks by autoregulation*, Nature, 405 (2000), p. 590.

[3] M. Chen, Q. Shao, and J. Ibrahim, *Monte Carlo methods in Bayesian computation*, Springer Science & Business Media, 2012.

[4] S. Cotter, *Constrained approximation of effective generators for multiscale stochastic reaction networks and application to conditioned path sampling*, Journal of Computational Physics, 323 (2016), pp. 265–282.

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

[6] D. Gillespie, *Stochastic simulation of chemical kinetics*, Annu. Rev. Phys. Chem., 58 (2007), pp. 35–55.

[7] N. Guido, X. Wang, D. Adalsteinsson, D. McMillen, J. Hasty, C. Cantor, T. Elston, and J. Collins, *A bottom-up approach to gene regulation*, Nature, 439 (2006), p. 856.

[8] T. Jahnke and W. Huisinga, *Solving the chemical master equation for monomolecular reaction systems analytically*, Journal of Mathematical Biology, 54 (2007), pp. 1–26.

[9] M. Kaern, T. Elston, W. Blake, and J. Collins, *Stochasticity in gene expression: from theories to phenotypes*, Nature reviews. Genetics, 6 (2005), p. 451.

[10] R. Kass and A. Raftery, *Bayes factors*, Journal of the American Statistical Association, 90 (1995), pp. 773–795.

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

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

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

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

[15] C. Robert and G. Casella, *Monte Carlo statistical methods*, Springer New York, 2004.