# Sequential Monte Carlo with active subspaces

Leonardo Ripoli<sup>1</sup> and Richard G. Everitt<sup>2,3</sup>

<sup>1</sup>Department of Mathematics and Statistics, University of Reading
<sup>2</sup>Department of Statistics, University of Warwick
<sup>3</sup>The Zeeman Institute for Systems Biology & Infectious Disease Epidemiology Research,

University of Warwick

<sup>2</sup>Email: richard.everitt@warwick.ac.uk

12th November 2024

#### Abstract

Monte Carlo methods, such as Markov chain Monte Carlo (MCMC), remain the most regularly-used approach for implementing Bayesian inference. However, the computational cost of these approaches usually scales worse than linearly with the dimension of the parameter space, limiting their use for models with a large number of parameters. However, it is not uncommon for models to have identifiability problems. In this case, the likelihood is not informative about some subspaces of the parameter space, and hence the model effectively has a dimension that is lower than the number of parameters. Constantine et al. (2016); Schuster et al. (2017) introduced the concept of directly using a Metropolis-Hastings (MH) MCMC on the subspaces of the parameter space that are informed by the likelihood as a means to reduce the dimension of the parameter space that needs to be explored. The same paper introduces an approach for identifying such a subspace in the case where it is a linear transformation of the parameter space: this subspace is known as the active subspace. This paper introduces sequential Monte Carlo (SMC) methods that make use of an active subspace. As well as an SMC counterpart to MH approach of Schuster et al. (2017), we introduce an approach to learn the active subspace adaptively, and an SMC<sup>2</sup> approach that is more robust to the linearity assumptions made when using active subspaces.

# 1 Introduction

#### 1.1 Motivation

This paper considers the Bayesian estimation of the random vector parameter  $\theta$  (with range  $\mathbb{R}^d$ ) from data  $y \in \mathcal{Y}$ . We let p denote the prior distribution on  $\theta$ , and f denote the model for p given p. Let p denote the posterior distribution on p, with p denoting the unnormalised version of this distribution arising from multiplying the prior and likelihood, i.e. p (p (p (p )) p (p ) with p (p ) with p (p ) p (p ). From hereon we denote the posterior as p (p ):= p (p ) likelihood as p (p ):= p (p ) p (p ). In this paper we require that p is differentiable in p. We let p denote the normalising constant of p.

It is not uncommon for models to have identifiability problems (Constantine et al., 2016). In this case, the likelihood is not informative about some subspaces of the parameter space, and hence the model has an 'intrinsic' dimension that is lower than the number of parameters. Constantine et al. (2016) introduces the idea of identifying the subspace of the parameter space that is informed by the likelihood (the 'active' subspace (AS)), and using MCMC on this space only. The advantage of this approach, when it is possible, is that the dimension of the AS is smaller than the dimension of the whole space, therefore we expect that our MCMC will not need to be run for as many iterations. This paper makes three contributions:

- We introduce the *active subspace sequential Monte Carlo* (AS-SMC) algorithm, which makes use of active subspaces in sequential Monte Carlo (SMC).
- In most previous work the identification of an AS must be performed in advance of running an inference algorithm. A typical approach would be to simulate points from the prior, and to use these to infer an AS. However, as we outline in the next section, were they available, one would use points from the posterior to infer the AS. This paper introduces the *adaptive AS-SMC* approach to allow the adaptation of the AS as the algorithm runs, using the points simulated at each step of the algorithm.

• We further develop the AS-SMC algorithm, introducing a nested SMC approach we call AS-SMC<sup>2</sup>.

### 1.2 Active subspace Metropolis-Hastings

### 1.2.1 Active subspaces

We follow the approach of Constantine et al. (2016) and Schuster et al. (2017), which we now describe. This approach considers only subspaces that are linear projections of the parameter space, and this is also the approach used in this paper. We denote the active variables by a (in  $\mathbb{R}^{d_a}$ ) and inactive variables by i (in  $\mathbb{R}^{d_i}$ ). Our aim is to find a reparameterisation of  $\theta$  of the form

$$\theta = Aa + Ii,\tag{1}$$

where  $A \in \mathbb{R}^{d \times d_a}$  and  $I \in \mathbb{R}^{d \times d_i}$  are such that [A, I] is an orthonormal basis of  $\mathbb{R}^d$ . We wish to choose the reparameterisation such that performing MCMC on the marginal distribution of a is sufficient for simulation from the posterior  $\pi$ . Constantine *et al.* (2016) also describes an approach for finding such a reparameterisation. The key idea is to find an estimate of the eigendecomposition of the matrix

$$\int_{\theta} \nabla \log l(\theta) \nabla \log l(\theta)^{T} \phi(\theta) d\theta, \tag{2}$$

for some distribution  $\phi$  (the choice of which will be discussed in section 1.3), using an eigendecomposition of the Monte Carlo approximation

$$\frac{1}{N} \sum_{m=1}^{N} \nabla \log l \left(\theta^{m}\right) \nabla \log l \left(\theta^{m}\right)^{T} \tag{3}$$

for  $\theta^i \sim \phi$  for m=1:N. This is equivalent to performing a uncentred and unscaled principal component analysis (PCA) to find the directions in which the score function  $\nabla \log l(\theta)$  is most variable under  $\phi$ : the eigenvectors of this decomposition with the largest eigenvalues will indicate the directions in which there is most variation in  $\nabla \log l(\theta)$ . The directions in which there is little variation in  $\nabla \log l(\theta)$  - when the likelihood is flat - are those that we may designate as inactive variables. Constantine et al. (2016); Schuster et al. (2017) propose to identify a spectral gap in the eigendecomposition to choose which directions are designated as active and inactive variables: identifying a group of dominant eigenvalues that are well separated from the remainder, and choosing the corresponding directions to give the active variables. This procedure is similar to the approach used in PCA for dimensionality reduction, where the directions that explain a large proportion (e.g. 90%) of the variance are selected.

#### 1.2.2 Active subspace Metropolis-Hastings

To see how to use active subspaces within MCMC, we first write the posterior on the new parameterisation.

$$\pi_{a,i}(a,i) \propto p_{a,i}(a,i) l (Aa + Ii),$$
 (4)

where  $p_{a,i}\left(a,i\right)=p\left(Aa+Ii\right)$ . Since  $B_a$  and  $B_i$  are orthonormal there is no Jacobian associated with this change in parameterisation. Throughout, as in Constantine *et al.* (2016), we assume that we have available the distributions  $p_a$  and  $p_i$  arising from the factorisation  $p_{a,i}\left(a,i\right)=p_a\left(a\right)p_i\left(i\mid a\right)$ : specifically, that we can evaluate  $p_a$  and  $p_i$  pointwise, and that we can simulate from  $p_i\left(\cdot\mid a\right)$ . Constantine *et al.* (2016) describes how to construct an approximate MCMC algorithm (in the style of Alquier *et al.* (2016)) on the active variables through using a numerical estimate of the (unnormalised) marginal distribution

$$\tilde{\pi}_{a}(a) = \int_{i} p(Aa + Ii) l(Aa + Ii) di$$

$$= p_{a}(a) \int_{i} p_{i}(i \mid a) l(Aa + Ii) di$$

at every iteration of the algorithm. The second term is the marginal likelihood

$$l_a(a) := \int_i p_i(i \mid a) l(Aa + Ii) di.$$

$$(5)$$

Schuster et al. (2017) made the observation that this approximate method can be made exact by formulating it as a pseudo-marginal algorithm (Beaumont, 2003; Andrieu and Roberts, 2009), using the unbiased importance sampling (IS) estimator

$$\bar{l}_{a}(a) = \frac{1}{N_{i}} \sum_{n=1}^{N_{i}} \frac{p_{i}(i^{n} \mid a) l(Aa + Ii^{n})}{q_{i}(i^{n} \mid a)},$$
(6)

where  $i^n \sim q_i(\cdot \mid a)$  for  $n = 1 : N_i$ , at each step of the MCMC. Algorithm 1 gives the active subspace MH (AS-MH) algorithm from Schuster *et al.* (2017)).

## Algorithm 1: Active subspace Metropolis-Hastings

```
1 Initialise a^0;
  2 for n = 1 : N_i do
              i^{n,0} \sim q_i \left( \cdot \mid a^0 \right);
                                                                                                               \tilde{w}^{n,0} = \frac{p_i \left( i^{n,0} \mid a^0 \right) l \left( A a^0 + I i^{n,0} \right)}{q_i \left( i^{n,0} \mid a^0 \right)};
  6 u^0 \sim \mathcal{M}\left(\left(w^{1,0},...,w^{N_i,0}\right)\right), where for n=1:N_i
                                                                                                                               w^{n,0} = \frac{\tilde{w}^{n,0}}{\sum_{p=1}^{N_i} \tilde{w}^{p,0}};
 7 Let \bar{l}_{a}^{0} = \frac{1}{N_{i}} \sum_{n=1}^{N_{i}} \tilde{w}^{n,0};

8 for m = 1 : N_{a} do

9 | a^{*m} \sim q_{a} \left( \cdot \mid a^{m-1} \right);

10 | for n = 1 : N_{i} do

11 | i^{*n,m} \sim q_{i} \left( \cdot \mid a^{*m} \right);
10
11
12
                                                                                                         \tilde{w}^{*n,m} = \frac{p_i \left( i^{*n,m} \mid a^{*m} \right) l \left( A a^{*m} + I i^{*n,m} \right)}{q_i \left( i^{*n,m} \mid a^{*m} \right)};
13
                 u^{*m} \sim \mathcal{M}\left(\left(w^{*1,m},...,w^{*N_i,m}\right)\right), where for n=1:N_i
14
                                                                                                                                w^{*n,m} = \frac{\tilde{w}^{*n,m}}{\sum_{i=1}^{N_i} \tilde{w}^{*p,m}};
             Let \bar{l}_a(a^{*m}) = \frac{1}{N_i} \sum_{n=1}^{N_i} \tilde{w}^{*n,m};

Set \left(a^m, \{i^{n,m}, w^{n,m}\}_{n=1}^{N_i}, u^m, \bar{l}_a^m\right) = \left(a^{*m}, \{i^{*n,m}, w^{*n,m}\}_{n=1}^{N_i}, u^{*m}, \bar{l}_a(a^{*m})\right) with probability
                                                                                                    \alpha_a^m = 1 \wedge \frac{p_a(a^{*m}) \bar{l}_a(a^{*m})}{p_a(a^{m-1}) \bar{l}_a^{m-1}} \frac{q_a(a^{m-1} \mid a^{*m})}{q_a(a^{*m} \mid a^{m-1})};
                 Else let \left(a^m, \{i^{n,m}, w^{n,m}\}_{n=1}^{N_i}, u^m, \bar{l}_a^m\right) = \left(a^{m-1}, \left\{i^{n,m-1}, w^{n,m-1}\right\}_{n=1}^{N_i}, u^{m-1}, \bar{l}_a^{m-1}\right);
17
18 end
```

#### 1.3 Efficiency of AS-MH

The statistical efficiency of AS-MH depends on the choice of  $q_a, q_i$  and the AS defined by A and I. For a pseudomarginal approach to offer better performance than an MH algorithm that directly proposes a move on the vector  $\theta$ , we require that the estimator in equation (6) has a low variance, so that the pseudo-marginal resembles the marginal MH, where the integral in equation (5) is available exactly. The optimal choice of  $q_i$  ( $i \mid a$ ), in terms of minimising the IS variance, would be for it to be proportional to  $p_i(i \mid a) l(Aa + Ii)$ . We now describe how the use of an AS allows us to use a proposal close to this optimal choice.

The AS is chosen so that, in an ideal case, the inactive variables are not at all influenced by the likelihood. In this case, for any a, l  $(Aa + Ii) \propto 1$  as  $\theta_i$  varies, thus the optimal choice of  $q_i$   $(i \mid a)$  is the prior  $p_i$   $(i \mid a)$ . Since this distribution is available to us, in this ideal case we can easily implement an efficient pseudo-marginal method. We can see the AS as offering a criterion through which we may determine when the pseudo-marginal approach, with the prior  $p_i$  as the proposal, might be an effective sampler.

Outside of the ideal case, the likelihood may only be approximately flat as i varies, thus  $p_i$  will not be optimal, with the distance from optimality depending on a. We now consider two ways in which the AS-MH algorithm may be improved, each of which motivates a method in the paper.

#### 1.3.1 Prior and posterior active subspaces

For the AS-MH algorithm to be effective, we require that  $p_i$  is close to optimal for the values of a that we might visit when running the algorithm: i.e. those that are the region of the posterior  $\pi_a$ . Recall equation (2), used to find the AS: here we aimed to find the directions in which the score function  $\nabla \log l$  ( $\theta$ ) is most variable under some distribution  $\phi$ . We now see the effect that the choice of  $\phi$  (and hence  $B_a$  and  $B_i$ ) might have on the efficiency of AS-MH: if  $\phi$  is close to the posterior  $\pi$ ,  $p_i$  is likely to be a good proposal for the inactive variables conditional on the a we are likely to visit. If  $\phi$  is far from  $\pi$ , then although the inactive variables will be chosen to be those directions in which  $\nabla \log l$  ( $\theta$ ) is least variable over  $\phi$ , the directions may not correspond to the directions in which  $\nabla \log l$  ( $\theta$ ) is least variable over  $\pi$ . This suggests  $\phi = \pi$  as an appropriate choice for determining an AS for used in AS-MH: we refer to this as the posterior active subspace.

Recall now equation (3), the Monte Carlo approximation used to estimate equation (2). In Constantine *et al.* (2016),  $\phi$  is taken to be equal to the prior p: we call this the *prior active subspace*. Since it is usually possible to easily sample from p, it is straightforward to implement equation (3). However, the prior AS may not be close to the posterior AS. An alternative to using equation (3) for estimating the posterior AS would be to use an IS estimate with the prior as the proposal, but this will only be accurate when the prior is close to the posterior, in which case the prior and posterior active subspaces are unlikely to differ significantly.

In this section we introduce a toy example that illustrates a case when the prior and posterior active subspaces are very different. Let the prior on  $\theta$  be multivariate Gaussian  $\theta \sim \mathcal{MVN}(\mathbf{0}, 5000\mathbf{I})$  and the likelihood have the form

$$l\left(\theta\right) \propto \prod_{j=1}^{d} \left[ \exp\left(-\left(\frac{\theta_{j}}{\sigma_{j}}\right)^{2}\right) \frac{1}{1 + \left(\frac{\theta_{j}}{\gamma_{j}}\right)^{2}} \right],$$

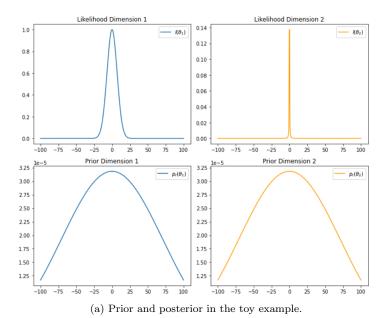
where for the purposes of this definition only we have altered the meaning of the subscript of  $\theta$  to denote the different dimensions of  $\theta$ . Each dimension of  $\theta$  has an independent likelihood term, each composed of a Gaussian and a Cauchy term. Its gradient is

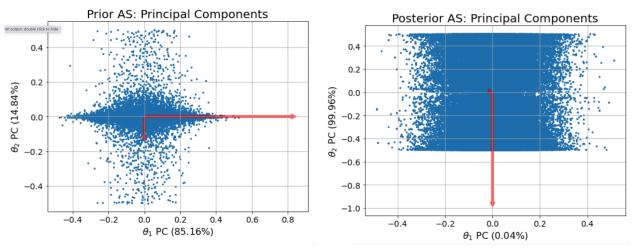
$$\nabla \log l\left(\theta\right) = \begin{pmatrix} -2\theta_1 \left(\frac{1}{\sigma_1^2} + \frac{1}{\theta_1^2 + \gamma_1^2}\right) \\ \dots \\ -2\theta_d \left(\frac{1}{\sigma_d^2} + \frac{1}{\theta_d^2 + \gamma_d^2}\right) \end{pmatrix}.$$

As each  $\theta_j$  gets further from 0, the Gaussian term  $(1/\sigma_1^2)$  dominates, and so for  $\theta_j$  far from zero this term will dominate the gradient. However, when  $\theta_j$  is near zero, with appropriately set parameters, we will see the Cauchy term  $(1/(\theta_d^2 + \gamma_d^2))$  dominate.

We examine a two dimensional (d = 2) example with  $\sigma_1 = 10$ ,  $\gamma_1 = 10^{12}$ ,  $\sigma_2 = 50$  and  $\gamma_2 = 0.1$ . Figure 1a shows the prior and posterior for these parameters, with figures 1b and 1c showing uncentred and unscaled PCA performed on the gradient of points simulated from the prior and posterior respectively. Across the range of the prior, the gradient varies more in the first dimension, therefore in the prior AS the second variable will be chosen to be inactive through examining the spectral gap. However, in the posterior AS, we see the first variable being chosen to be inactive.

We see from this example that the approach in Constantine *et al.* (2016) and Schuster *et al.* (2017) of using the prior AS is not always desirable. Whilst the example in this section is very artificial, real applications can also exhibit significant differences between the prior and posterior active subspaces (see Parente (2020) and Zahm *et al.* (2022), for example).





(b) PCA performed on the gradient of points simulated from the(c) PCA performed on the gradient of points simulated from the prior.

Figure 1: An illustration of the toy example.

#### 1.3.2 Importance sampling for the inactive variables

AS-MH uses the IS proposal  $q_i$  to simulate the inactive variables at every step of the MCMC. The prior  $p_i$  is a good choice for this in the case of 'ideal' inactive variables. Outside of this special case, it is less clear that using IS with this proposal should be an efficient means for sampling these variables. active subspaces will be of most use when the dimension of the inactive variables,  $d_i$ , is large. Since IS requires an exponential number of points in the dimension to obtain a fixed variance (Agapiou et al., 2017), it is precisely the case of  $d_i$  being large when IS will be most inefficient. For  $d_i$  large, the variance of the IS estimator of the marginal likelihood is likely to be large, and hence AS-MH is likely to be inefficient.

## 1.4 Proposed approach and overview of paper

In the paper we introduce AS-SMC as an SMC analogue of AS-MH, allowing the use of active subspaces in SMC. This SMC sampler moves a population of weighted importance points ('particles') through a sequence of distributions, using an AS-MH move at each iteration. Typically this approach would use the easy to estimate prior AS, which is useful in a number of applications.

In some cases (such as in Parente (2020) and Zahm et al. (2022)), an estimate of the posterior AS is required in order for AS-MH or AS-SMC to be efficient. For this purpose we extend AS-SMC to, at each iteration of the SMC, use an AS tailored to the current step of the algorithm by estimating the AS using the current set of weighted particles (section 2.3). If a useful AS is not found at any given SMC iteration, a standard MH move will be used instead of AS-MH. Adaptive sequential importance sampling approaches such as this have been considered previously in Parente (2020) and Zahm et al. (2022) These previous approaches adaptively construct an AS in a similar way: our main contribution is to combine such an approach with the exact pseudo-marginal framework introduced by Schuster et al. (2017).

Separately we consider the case where the inactive variables have some (small) influence on the likelihood, where IS is not an effective method for estimating the marginal likelihood. For this situation we introduce an SMC<sup>2</sup> algorithm, using an SMC algorithm to estimate the marginal likelihood for each particle.

In section 2 we introduce the new methods, followed by presenting empirical results in section 3 and conclusions in section 4.

# 2 SMC with active subspaces

In section 2.1 we recap the fundamentals of SMC samplers, before introducing SMC approaches using a fixed AS in section 2.2, an SMC sampler using an adaptive AS in section 2.3, followed by the SMC<sup>2</sup> approach in section 2.4.

## 2.1 SMC sampler recap

An SMC sampler (Del Moral et al., 2006) operates by using a sequence of distributions  $\pi_0, ..., \pi_T$  such that  $\pi_T = \pi$ ,  $\pi_0$  is 'easy' to simulate from, and where the sequence can be seen as a path of distributions bridging between  $\pi_0$  and  $\pi_T$ . The algorithm is devised so as to guide a set of weighted importance points ('particles') from  $\pi_0$  to  $\pi_T$ . The formal construction of the algorithm uses a sequence  $\overline{\pi}_0, ..., \overline{\pi}_T$  of extended distributions, where for each  $t, \overline{\pi}_t$  has  $\pi_t$  as a marginal distribution. Specifically,

$$\overline{\pi}_t(\theta_{0:t}) = \pi(\theta_t) \prod_{s=0}^{t-1} L_s(\theta_s \mid \theta_{s+1}), \qquad (7)$$

where each  $L_s$  is known as a 'backwards' kernel. The SMC sampler is a sequential importance sampling approach, which uses as a proposal

$$\overline{q}_t(\theta_{0:t}) = q_0(\theta_0) \prod_{s=0}^t K_s(\theta_s \mid \theta_{s-1})$$
(8)

at target t, where each  $K_s$  is known as a 'forwards' kernel. The algorithm proceeds at iteration t by moving each particle using  $K_t$ , reweighting each particle using the ratio of equations (7) and (8), and resampling the weighted population of particles so that particles with a small weight tend not to remain in the population for the following iteration of the algorithm. The marginal distribution of the weighted  $\theta_t$  values of the particles can then be used as an approximation to  $\pi_t$ .

The implemention of this algorithm only requires the evaluation of each  $\pi_t$  up to a normalising constant; we let  $\tilde{\pi}_t$  denote an unnormalised version of  $\pi_t$ . The weights calculated at each step then contain information about the normalising constant of each unnormalised distribution  $\tilde{\pi}_t$ , and can be used to additionally calculate an estimate its normalising constant  $Z_t$ . The choice of the sequence of distributions significantly affect the performance of the algorithm. We restrict our attention to the choice of a factorised likelihood. Suppose that

$$l\left(\theta\right) = \prod_{s=1}^{T} l_s\left(\theta\right),\,$$

for some sequence  $l_s(\theta)$ . We use the notation

$$l_{1:t}\left(\theta\right) = \prod_{s=1}^{t} l_{s}\left(\theta\right),$$

so that  $l_{1:T}(\theta) = l(\theta)$ . We choose  $\pi_t(\theta) = p(\theta) l_{1:t}(\theta) / Z_t$ , and  $\pi_0(\theta) = p(\theta)$ , which encompasses the data point tempering of Chopin (2002) (where the factors each depend on a different data point), and also the widely-used annealing approach (where  $l(\theta) = \prod_{s=1}^{T} l^{\eta_t - \eta_{t-1}}(\theta)$  for  $\eta_0 = 0 < \dots < \eta_T = 1$ ).

The efficiency of the algorithms is also affected by the choice of  $K_t$  and  $L_{t-1}$ : here we restrict our choice to  $K_t$  being an MCMC move with invariant distribution  $\pi_t$  and  $L_{t-1}$  to be the reversed Markov kernel associated with  $K_t$ . When the MCMC move mixes well, this can be an effective choice for  $K_t$ , and its reverse is close to the optimal choice of  $L_{t-1}$  when  $\pi_{t-1}$  is close to  $\pi_t$  (Del Moral *et al.*, 2006). When using an MCMC move, each iteration of the SMC algorithm will consist of a reweighting step, a resampling step and a move step (where the MCMC move is used). Throughout the paper we use stratified resampling (Douc *et al.*, 2005).

In all of the SMC samplers in this section, we may use the current set of weighted particles to inform subsequent steps of the algorithm. In particular we describe how to use the current particles to adapt the proposal on the active variables  $q_a$  and the proposal on the active variables  $q_i$ . In full generality, we may: use an adaptive approach to resampling where the particle population is resampled and moved when the effective sample size (Kong et al., 1994) falls below  $0.5 \times N_a$ ; adapt the sequence of distributions, as originally described in Del Moral et al. (2012) (see Kerama et al. (2022) for a description of how to implement this approach in the SMC<sup>2</sup> setting); and adapt the number of MCMC moves (as in South et al. (2016)). However we do not make use of these approaches in this paper when implementing the new approaches.

#### 2.2 SMC sampler with a fixed active subspace

#### 2.2.1 Construction

In this section we describe the SMC counterpart of the AS-MH algorithm; we embed the AS-MH move within an SMC sampler with an annealed sequence of distributions. The structure of this algorithm has some elements that are similar to SMC<sup>2</sup> (Chopin *et al.*, 2013). We use an AS, given by A and I that is fixed across the SMC iterations. Our aim is to use an SMC sampler to simulate from the target distribution  $\pi_{a,i}$  in equation (4). From hereon we abuse notation and denote this simply by  $\pi$  to avoid additional subscripts. The target distribution  $\pi_t$  used at the tth iteration is

$$\pi_{t}\left(a,\left\{i^{n}\right\}_{n=1}^{N_{i}}\right) := \frac{1}{Z_{t}}p_{a}\left(a\right)\prod_{j=1}^{N_{i}}q_{t,i}\left(i^{j}\mid a\right)\left(\frac{1}{N_{i}}\sum_{n=1}^{N_{i}}\frac{p_{i}\left(i^{n}\mid a\right)l_{1:t}\left(Aa+Ii^{n}\right)}{q_{t,i}\left(i^{n}\mid a\right)}\right) \tag{9}$$

where  $q_{t,i}$  is the proposal used for the inactive variables at iteration t and where  $Z_t$  is a normalising constant, which by the unbiasedness of the estimator in equation (6) we know to be the same as the  $Z_t$  defined in section 2.1. We may rearrange this target as follows:

$$\pi_{t}\left(a,\left\{i^{n}\right\}_{n=1}^{N_{i}}\right) = \frac{1}{Z_{t}}p_{a}\left(a\right)\prod_{j=1}^{N_{i}}q_{t,i}\left(i^{j}\mid a\right)\left(\frac{1}{N_{i}}\sum_{n=1}^{N_{i}}\frac{p_{i}\left(i^{n}\mid a\right)l_{1:t}\left(Aa+Ii^{n}\right)}{q_{t,i}\left(i^{n}\mid a\right)}\right)$$

$$= \frac{1}{N_{i}}\sum_{n=1}^{N_{i}}\frac{p_{a}\left(a\right)p_{i}\left(i^{n}\mid a\right)l_{1:t}\left(Aa+Ii^{n}\right)}{Z_{t}}\left(\prod_{\substack{j=1\\j\neq n}}^{N_{i}}q_{t,i}\left(i^{j}\mid a\right)\right)$$

$$= \frac{\pi_{t,a}\left(a\right)}{N_{i}}\sum_{n=1}^{N_{i}}\pi_{t,i}\left(i^{n}\mid a\right)\left(\prod_{\substack{j=1\\j\neq n}}^{N_{i}}q_{t,i}\left(i^{j}\mid a\right)\right)$$

$$(10)$$

where we used the result

$$p_a(a) p_i(i \mid a) l_{1:t}(Aa + Ii^n) = Z_t \pi_{t,a}(a) \pi_{t,i}(i \mid a),$$

where  $\pi_{t,i}$  is the conditional distribution defined by  $\pi_{t,i}$  ( $i \mid a$ ) =  $\pi_t$  (a,i) / $\pi_{t,a}$  (a). Equation (10) includes in its marginals the target at iteration t,  $\pi_t$  (a,i) (here notation is abused since we use  $\pi_t$  to denote both the target in equation (9) and this marginal). The sequence of likelihoods chosen in the previous section has the result that at iteration T of the SMC, the marginal  $\pi_T$  (a,i) is equal to the desired posterior in equation 4.

### 2.2.2 Algorithm

In this section we use  $a_t^m$  to denote the *m*th active variable *a*-particle at the *t*th iteration of the SMC, and  $i_t^{m,n}$  to denote the *n*th inactive variable *i*-particle for the *m*th *a*-particle at the *t*th iteration of the SMC. Using this sequence of targets, taking the backwards kernel to be the reverse of  $K_t$ , following Del Moral *et al.* (2006) we obtain the weight update at iteration *t* to be

$$\begin{split} \tilde{\omega}_{t}^{m} &= \omega_{t-1}^{m} \frac{\pi_{t} \left( a_{t-1}^{m}, \left\{ i_{t-1}^{n,m} \right\}_{n=1}^{N_{i}} \right)}{\pi_{t-1} \left( a_{t-1}^{m}, \left\{ i_{t-1}^{n,m} \right\}_{n=1}^{N_{i}} \right)} \\ &= \omega_{t-1}^{m} \frac{p_{a} \left( a_{t-1}^{m} \right) \prod_{j=1}^{N_{i}} q_{t,i} \left( i_{t-1}^{n,m} \mid a_{t-1}^{m} \right) \frac{1}{N_{i}} \sum_{n=1}^{N_{i}} \tilde{w}_{t}^{n,m} \left( a_{t-1}^{m}, i_{t-1}^{n,m} \right)}{p_{a} \left( a_{t-1}^{m} \right) \prod_{j=1}^{N_{i}} q_{t-1,i} \left( i_{t-1}^{n,m} \mid a_{t-1}^{m} \right) \frac{1}{N_{i}} \sum_{n=1}^{N_{i}} \tilde{w}_{t-1}^{n,m} \left( a_{t-1}^{m}, i_{t-1}^{n,m} \right)} \\ &= \omega_{t-1}^{m} \frac{\prod_{j=1}^{N_{i}} q_{t,i} \left( i_{t-1}^{n,m} \mid a_{t-1}^{m} \right) \sum_{n=1}^{N_{i}} \tilde{w}_{t}^{n,m} \left( a_{t-1}^{m}, i_{t-1}^{n,m} \right)}{\prod_{j=1}^{N_{i}} q_{t-1,i} \left( i_{t-1}^{n,m} \mid a_{t-1}^{m} \right) \sum_{n=1}^{N_{i}} \tilde{w}_{t-1}^{n,m} \left( a_{t-1}^{m}, i_{t-1}^{n,m} \right)}, \end{split}$$

where

$$\tilde{w}_{t}^{n,m}\left(a_{t-1}^{m}, i_{t-1}^{n,m}\right) = \frac{p_{i}\left(i_{t-1}^{n,m} \mid a_{t-1}^{m}\right) l_{1:t}\left(A a_{t-1}^{m} + I i_{t-1}^{n}\right)}{q_{t.i}\left(i_{t-1}^{n,m} \mid a_{t-1}^{m}\right)},\tag{11}$$

with the t subscript for the unnormalised weight  $\tilde{w}_t^{n,m}$  denoting that it uses the likelihood  $l_{1:t}$  and proposal  $q_{t,i}$  (so that  $\tilde{w}_{t-1}^{n,m}$  follows equation (11), but uses  $l_{1:t-1}$  and  $q_{t-1,i}$ ). At iteration t, for each particle we use an MCMC move with invariant distribution  $\pi_t\left(a,\{i^n\}_{n=1}^{N_i}\right)$ ; for this we may use an AS-MH move with likelihood  $l_{1:t}$ . The active subspace SMC (AS-SMC) algorithm is given in algorithm 2. Discussion of how to tune the proposals  $q_{t,a}$  and  $q_{t,i}$  is found in sections 2.3.5 and 2.3.6 respectively.

Chopin et al. (2013) shows, for estimating the expectation of g with respect to  $\pi_t$ , we may use either of the following

1. Using one *i*-point for each *a*-point:

$$\sum_{m=1}^{N_a} \omega_t^m g\left(A a_t^m + I i_t^{u_t^m, m}\right). \tag{12}$$

2. Using all of the accepted *i*-points for each *a*-point:

$$\sum_{m=1}^{N_a} \omega_t^m \sum_{n=1}^{N_i} w_t^{n,m} g\left(A a_t^m + I i_t^{n,m}\right). \tag{13}$$

### **Algorithm 2:** AS-SMC

```
1 Simulate N_a points, \{\theta_0^m\}_{m=1}^{N_a} \sim p and set each weight \omega_0^m = 1/N_a;
  2 Find AS using points \{\theta_0^m\}_{m=1}^{N_a} in equation (3), yielding A and I;
  3 for m = 1 : N_a do
                 Set a_0^m = A^T(\theta_0^m), i_0^{1,m} = I^T(\theta_0^m) and u_0^m = 1;
                 for n=2:N_i do
                    | i_0^{n,m} \sim q_{0,i} \left( \cdot \mid a_0^m \right) := p_i \left( \cdot \mid a_0^m \right); 
   6
   7
                 Set w_0^{n,m} = 1/N_i for n = 1 : N_i;
  8
  9 end
10 for t = 1 : T do
                 for m = 1 : N_a \text{ do // reweight}
11
                          for n = 1 : N_i do
12
                                 \tilde{w}_{t}^{n,m}\left(a_{t-1}^{m},i_{t-1}^{n,m}\right) = \frac{p_{i}\left(i_{t-1}^{n,m}|a_{t-1}^{m}\right)l_{1:t}\left(B_{a}a_{t-1}^{m}+B_{i}i_{t-1}^{n,m}\right)}{q_{t,i}\left(i_{t-1}^{n,m}|a_{t-1}^{m}\right)};
13
                          end
14
15
                                                                                     \tilde{\omega}_{t}^{m} = \omega_{t-1}^{m} \frac{\prod_{j=1}^{N_{i}} q_{t,i} \left(i_{t-1}^{n,m} \mid a_{t-1}^{m}\right) \sum_{n=1}^{N_{i}} \tilde{w}_{t}^{n,m} \left(a_{t-1}^{m}, i_{t-1}^{n,m}\right)}{\prod_{i=1}^{N_{i}} q_{t-1,i} \left(i_{t-1}^{n,m} \mid a_{t-1}^{m}\right) \sum_{n=1}^{N_{i}} \tilde{w}_{t}^{n,m} \left(a_{t-1}^{m}, i_{t-1}^{n,m}\right)};
16
                 \{\omega_t^m\}_{m=1}^{N_a} \leftarrow \text{normalise}\left(\{\tilde{\omega}_t^m\}_{m=1}^{N_a}\right);
17
                 for m = 1 : N_a do
18
                        \{w_t^{n,m}\}_{n=1}^{N_i} \leftarrow \text{ normalise } \left(\{\tilde{w}_t^{n,m}\}_{n=1}^{N_i}\right);
19
20
                 for m = 1 : N_a do
21
                         u_t^m \sim \mathcal{M}\left(\left(w_t^{1,m},...,w_t^{N_i,m}\right)\right);
22
23
                 if some degeneracy condition is met then // resample and move
24
                          for m = 1 : N_a do
25
                                   Simulate (a_t^m, i_t^{1:N_i,m}) from the mixture distribution
26
                                                                                                                           \sum_{i=1}^{N_a} \omega_t^j K_{t,a} \left\{ \cdot \mid \left( a_{t-1}^j, i_{t-1}^{1:N_i, j} \right) \right\},\,
                                  where K_{t,a} is an AS-MH move, i.e.: j^* \sim \mathcal{M}\left(\left\{\omega_t^j\right\}_{j=1}^{N_a}\right); a_t^{*m} \sim q_a\left(\cdot \mid a_t^{j^*}\right);
27
                                   for n = 1 : N_i do \begin{vmatrix} i_t^{*n,m} \sim q_{t,i} \left( \cdot \mid a_t^{*m} \right); \end{vmatrix}
28
 29
 30
                                                                                                 \tilde{w}_{t}^{n,m}\left(a_{t}^{*m}, i_{t,i}^{*n,m}\right) = \frac{p_{i}\left(i_{t}^{*n,m} \mid a_{t}^{*m}\right) l_{1:t}\left(Aa_{t}^{*m} + Ii_{t}^{*n,m}\right)}{q_{t,i}\left(i_{t}^{*n,m} \mid a_{t}^{*m}\right)};
31
                                  u_t^{*m} \sim \mathcal{M}\left(\left(w_t^{*1,m}, ..., w_t^{*N_i,m}\right)\right), where for n = 1: N_i
32
                                                                                                                         w_t^{*n,m} = \frac{\tilde{w}_t^{n,m} \left( a_t^{*m}, i_t^{*n,m} \right)}{\sum_{t=1}^{N_t} \tilde{w}_t^{p,m} \left( a_t^{*m}, i_t^{*p,m} \right)};
                                  Set \left(a_t^m, \{i_t^{n,m}, \tilde{w}_t^{n,m}\}_{n=1}^{N_i}, u_t^m\right) = \left(a_t^{*m}, \{i_t^{*n,m}, \tilde{w}_t^{*n,m}\}_{n=1}^{N_i}, u_t^{*m}\right) with probability
33
                                                                                       \alpha_{t,a}^{m} = 1 \wedge \frac{p_{a}\left(a_{a}^{*m}\right) \sum_{n=1}^{N_{i}} \tilde{w}_{t}^{n,m}\left(a_{t}^{*m}, i_{t}^{*n,m}\right)}{p_{a}\left(a_{t}^{j^{*}}\right) \sum_{n=1}^{N_{i}} \tilde{w}_{t}^{n,j^{*}}\left(a_{t}^{j^{*}}, i_{t-1}^{n,j^{*}}\right)} \frac{q_{t,a}\left(a_{t}^{j^{*}} \mid a_{t}^{*m}\right)}{q_{t,a}\left(a_{t}^{*m} \mid a_{t}^{j^{*}}\right)};
                                  Else let \left(a_t^m, \left\{i_t^{n,m}, \tilde{w}_t^{n,m}\right\}_{n=1}^{N_i}, u_t^m\right) = \left(a_t^{j^*}, \left\{i_t^{n,j^*}, \tilde{w}_t^{n,j^*}\right\}_{n=1}^{N_i}, u_t^{j^*}\right);
34
35
                         \omega_t^m = 1/N_a for m = 1: N_a;
36
                 end
37
                                                                                                                                                  9
38 end
```

To see that the first of these holds, we consider an extended version of the target distribution in equation 10, as we will see in section 2.3.

### 2.3 SMC sampler with an adaptive active subspace

#### 2.3.1 Adapting the active subspace

At the beginning of each SMC iteration, we find an estimate of the AS that we will use at that iteration. At the beginning of the tth iteration we estimate the matrix in equation (2) for the score function  $\nabla \log l_{1:t}$  using  $\phi = \pi_{t-1}$ , with equation (13)

$$\sum_{m=1}^{N_a} \omega_{t-1}^m \sum_{n=1}^{N_i} w_{t-1}^{n,m} \nabla \log l_{1:t} \left( A a_{t-1}^m + I i_{t-1}^{n,m} \right) \nabla \log l_{1:t} \left( A a_{t-1}^m + I i_{t-1}^{n,m} \right)^T. \tag{14}$$

The eigendecomposition, together with our chosen approach to choose the active and inactive variables, yields the AS that we represent using  $A_t$  and  $I_t$ .

The next step is that we need to find a representation of the current particle set in the new AS. To do this, for each particle we select a single inactive variable, then transform the active variable and the selected inactive variable from iteration t to the original space of  $\theta$ , then project the resultant  $\theta$  into the new AS. We then require an additional  $N_i - 1$  inactive variables for estimating the marginal likelihood. This move needs to be carefully constructed: we use a conditional IS update (Andrieu et al., 2010; Chopin et al., 2013).

We allow for the possibility for the dimension  $d_{t,i}$  of the inactive subspace to reduce to zero, but we require that the dimension  $d_{t,a}$  of the AS is always greater than zero. When  $d_{t,i} = 0$ , we no longer require an estimated likelihood and the target is simplified significantly.

#### 2.3.2 Target distributions

This algorithm uses the target

$$\pi_t \left( a_t, \{i_t^n\}_{n=1}^{N_i} \right) := \frac{1}{Z_t} p_{t,a} \left( a_t \right) \prod_{j=1}^{N_i} q_{t,i} \left( i_t^j \mid a_t \right) \frac{1}{N_i} \sum_{n=1}^{N_i} \frac{p_{t,i} \left( i_t^n \mid a_t \right) l_{1:t} \left( A_t a_t + i_t i_t^n \right)}{q_{t,i} \left( i_t^n \mid a_t \right)}$$

$$(15)$$

$$= \frac{\pi_{t,a}(a_t)}{N_i} \sum_{n=1}^{N_i} \pi_{t,i}(i_t^n \mid a_t) \left( \prod_{\substack{j=1\\j \neq n}}^{N_i} q_{t,i} \left( i_t^j \mid a_t \right) \right)$$
(16)

at iteration t, where  $p_{t,a}$  and  $\pi_{t,a}$  denote the marginal prior and posterior of a under the AS determined at iteration t and  $p_{t,i}$  and  $\pi_{t,i}$  denote the corresponding conditional prior and posterior on  $i \mid a$ . Since the AS has changed between iterations, to easily describe the algorithm we additionally need to define a target that uses likelihood  $l_{1:t-1}$ , but which uses the AS at iteration t. This is given by

$$\lambda_{t-1}\left(a_{t},\left\{i_{t}^{n}\right\}_{n=1}^{N_{i}}\right) := \frac{1}{Z_{t-1}}p_{t,a}\left(a_{t}\right)\prod_{j=1}^{N_{i}}\kappa_{t-1,i}\left(i_{t}^{j}\mid a_{t}\right)\frac{1}{N_{i}}\sum_{n=1}^{N_{i}}\frac{p_{t,i}\left(i_{t}^{n}\mid a_{t}\right)l_{1:t-1}\left(A_{t}a_{t}+I_{t}i_{t}^{n}\right)}{\kappa_{t-1,i}\left(i_{t}^{n}\mid a_{t}\right)}$$
(17)

$$= \frac{\lambda_{t-1,a}(a_t)}{N_i} \sum_{n=1}^{N_i} \lambda_{t-1,i} (i_t^n \mid a_t) \left( \prod_{\substack{j=1\\j \neq n}}^{N_i} \kappa_{t-1,i} \left( i_t^j \mid a_t \right) \right), \tag{18}$$

where  $\lambda_{t-1,a}$  and  $\lambda_{t-1,i}$  denote the posterior using  $l_{1:t-1}$ , projected onto the AS from iteration t. For simplicity we take  $\kappa_{t-1,i}$  to be equal to  $q_{t,i}$  (in practice both will be taken to be  $p_{t,i}$ ), although in full generality these two proposals could be chosen independently. The SMC then uses two steps at iteration t: the first a conditional IS step

to move from  $\pi_{t-1}$  to  $\lambda_{t-1}$  which we call 'reprojection'; the second to move from  $\pi_{t-1\to t}$  to  $\pi_t$  which is a standard weight update. The following two sections detail these steps. Following these steps, we resample, then for each particle use an MCMC move with invariant distribution  $\pi_t\left(a_t, \{i_t^n\}_{n=1}^{N_i}\right)$ , using an AS-MH move with likelihood  $l_{1:t}$ .

#### 2.3.3 Reprojection

At the beginning of each iteration, an eigendecomposition of equation (14) is used to determine  $A_t$  and  $I_t$ . We then perform the following conditional IS step on each particle to obtain a particle in the new active and inactive subspaces, moving particle  $\left(a_{t-1}^m, \left\{i_{t-1}^{n,m}\right\}_{n=1}^{N_i}\right)$  to  $\left(a_{t-1\to t}^m, \left\{i_{t-1\to t}^{n,m}\right\}_{n=1}^{N_i}\right)$ . We cannot easily transform the first collection of weighted points to the second, since we have multiple inactive points per each active point each with its own (internal) weight. Instead we need to select one of the inactive points to pair with the active, transform back to the original  $\theta$ -space, then transform to the new parameterisation. This involves discarding all but one of the inactive points, then regenerating additional inactive points after the reparameterisation. To construct this move we use the observation Chopin et al. (2013) that the target in equation (16) can be seen as a marginalisation of an extended distribution  $\pi_t^*$  over a uniformly distributed particle index variable  $u_t$ 

$$\pi_t^* \left( u_t, a_t, \{i_t^n\}_{n=1}^{N_i} \right) = \frac{\pi_{t,a} \left( a_t \right)}{N_i} \pi_{t,i} \left( i_t^{u_t} \mid a_t \right) \left( \prod_{\substack{j=1\\j \neq u_t}}^{N_i} q_{t,i} \left( i_t^j \mid a_t \right) \right). \tag{19}$$

Chopin et al. (2013) notes that it is simple to extend a set of weighted particles from  $\pi_{t-1}$  so that they are from  $\pi_{t-1}^*$  for each particle we simulate from the conditional distribution of  $u_{t-1}$ , which is given by  $u_{t-1} \mid a_{t-1}, \left\{i_{t-1}^n\right\}_{n=1}^{N_i} \sim \mathcal{M}\left(\left(w_{t-1}^{1,m},...,w_{t-1}^{N_i,m}\right)\right)$ , where  $w_{t-1}^{n,m}$  is the normalised version of  $\tilde{w}_{t-1}^{n,m}\left(a_{t-1}^m,i_{t-1}^{n,m}\right)$ . At the beginning of iteration t, our method performs this simulation of  $u_{t-1}$  for each particle, then makes use of the following transformation of the extended state

$$a_{t-1\to t} := G_{t-1\to t,a} \left( u_{t-1}, a_{t-1}, \left\{ i_{t-1}^n \right\}_{n=1}^{N_i} \right) := A_t^T \left( A_{t-1} a_{t-1} + I_{t-1} i_{t-1}^{u_{t-1}} \right)$$
$$i_{t-1\to t}^n := G_{t-1\to t,i} \left( u_{t-1}, a_{t-1}, \left\{ i_{t-1}^n \right\}_{n=1}^{N_i} \right) := I_t^T \left( A_{t-1} a_{t-1} + I_{t-1} i_{t-1}^n \right)$$

The conditional IS move makes use of this transformation, along with a proposal from Chopin *et al.* (2013). Our desired target distribution for the new point is  $\lambda_{t-1}^* \left( u_{t-1}, a_{t-1 \to t}, \left\{ i_{t-1 \to t}^n \right\}_{n=1}^{N_i} \right)$ , the extension of the target in equation (17), just as equation (19) is an extension of (15). The

Our proposal uses the current particle with values  $(u_{t-1}, a_{t-1}, \{i_{t-1}^n\}_{n=1}^{N_i})$  passed through the transformation above to give the point  $(u_{t-1}, a_{t-1 \to t}, i_{t-1 \to t}^{u_{t-1}})$ , plus the variables  $\{i_{t-1 \to t}^n\}_{n=1, n \neq u_{t-1}}^{N_i}$  which will be discarded. We then let  $a_t = a_{t-1 \to t}$  and  $i_t^{u_{t-1}} = i_{t-1 \to t}^{u_{t-1}}$ . In place of the discarded variables we additionally require the variables  $\{i_t^n\}_{n=1, n \neq u_{t-1}}^{N_i}$ , proposing them from the conditional distribution on the inactive variables resulting from  $\lambda_{t-1}^*$ . Following the structure of equation (19) and using equation (18), this conditional distribution is given by

$$\frac{\lambda_{t-1}^* \left( u, a_t, \{i_t^n\}_{n=1}^{N_i} \right)}{\frac{\lambda_{t-1, a}(a_t)}{N_i} \lambda_{t-1, i} \left( i_t^u \mid a_t \right)} = \prod_{\substack{j=1 \ j \neq u}}^{N_i} \kappa_{t-1, i} \left( i_t^j \mid a_t \right),$$

For the IS to be valid, we need to artificially extend the target distribution using a backwards kernel L (as in Del Moral *et al.* (2006)) to form a joint distribution over all of the variables involved in the proposal, such that the desired target  $\pi_t^*$  is a marginal distribution. The IS target is then

$$\lambda_{t-1}^* \left( u, a_t, \left\{ i_t^n \right\}_{n=1}^{N_i} \right) L \left( \left\{ i_{t-1 \to t}^n \right\}_{n=1, n \neq u}^{N_i} \mid u, a_t, \left\{ i_t^n \right\}_{n=1}^{N_i} \right)$$

and the proposal is

$$\pi_{t-1 \to t}^* \left( u, a_t, i_t^n, \left\{ i_{t-1 \to t}^n \right\}_{n=1, n \neq u}^{N_i} \right) \prod_{\substack{j=1 \ j \neq u}}^{N_i} \kappa_{t-1, i} \left( i_t^j \mid a_t \right),$$

where  $\pi_{t-1\to t}^*$  pushforward distribution of  $\pi_{t-1}^*$  under  $G_{t-1\to t}$ . We choose the backwards kernel to be

$$L\left(\left\{i_{t-1\to t}^{n}\right\}_{n=1, n\neq u}^{N_{i}} \mid u, a_{t}, \left\{i_{t}^{n}\right\}_{n=1}^{N_{i}}\right) = \frac{N_{i}\pi_{t-1\to t}^{*}\left(u, a_{t}, i_{t}^{n}, \left\{i_{t-1\to t}^{n}\right\}_{n=1, n\neq u}^{N_{i}}\right)}{\lambda_{t-1, a}\left(a_{t}\right)\lambda_{t-1, i}\left(i_{t}^{u} \mid a_{t}\right)},$$

which gives an importance weight of 1 (as in Chopin *et al.* (2013)). To proceed with the next iteration of the SMC, we then discard the value  $u_{t-1}$ , so that our particle is from the marginal distribution  $\pi_t$ , rather than the extended target  $\pi_t^*$ .

In summary:

- this additional step is run after determining the AS for the next iteration, for each of the  $N_a$  particles;
- we sample one of the inactive particles, using the weights of the particles in the inactive space;
- we project the active variable and sampled inactive variable into the new active and inactive subspaces;
- we sample  $N_i 1$  additional inactive variables from the proposal  $\kappa_{t-1,i}$ .

#### 2.3.4 Weight update

The weight update at iteration t is similar to that in section 2.2

$$\begin{split} \tilde{\omega}_{t}^{m} &= \omega_{t-1}^{m} \frac{\pi_{t} \left( a_{t-1 \to t}^{m}, \left\{ i_{t-1 \to t}^{n,m} \right\}_{n=1}^{N_{i}} \right)}{\pi_{t-1 \to t} \left( a_{t-1 \to t}^{m}, \left\{ i_{t-1 \to t}^{n,m} \right\}_{n=1}^{N_{i}} \right)} \\ &= \omega_{t-1}^{m} \frac{\prod_{j=1}^{N_{i}} q_{t,i} \left( \theta_{i_{t}}^{j} \mid \theta_{a_{t}} \right) \sum_{n=1}^{N_{i}} \tilde{w}_{t}^{n,m} \left( a_{t-1 \to t}^{m}, i_{t-1 \to t}^{n,m} \right)}{\prod_{j=1}^{N_{i}} q_{t-1 \to t,i} \left( \theta_{i_{t}}^{j} \mid \theta_{a_{t}} \right) \sum_{n=1}^{N_{i}} \tilde{w}_{t-1}^{n,m} \left( a_{t-1 \to t}^{m}, i_{t-1 \to t}^{n,m} \right)}, \end{split}$$

where

$$\tilde{w}_{t}^{n,m}\left(a_{t-1\to t}^{m},i_{t-1\to t}^{n,m}\right) = \frac{p_{i}\left(i_{t-1\to t}^{n,m}\mid a_{t-1\to t}^{m}\right)l_{1:t}\left(A_{t}a_{t-1\to t}^{m} + I_{t}i_{t-1\to t}^{n}\right)}{q_{t,i}\left(i_{t-1\to t}^{n,m}\mid a_{t-1\to t}^{m}\right)},$$

similarly to section 2.2 and

$$\tilde{w}_{t-1}^{n,m}\left(a_{t-1\to t}^{m}, i_{t-1\to t}^{n,m}\right) = \frac{p_{i_{t}}\left(i_{t-1\to t}^{n,m} \mid a_{t-1\to t}^{m}\right) l_{1:t-1}\left(A_{t}a_{t-1\to t}^{m} + I_{t}i_{t-1\to t}^{n}\right)}{q_{t-1\to t, i}\left(i_{t-1\to t}^{n,m} \mid a_{t-1\to t}^{m}\right)}.$$

The full algorithm, adaptive AS-SMC is given in algorithm 3.

#### **2.3.5** MH on *a*

The use of an AS means that we expect our moves on a to be more 'difficult' than those on i. To ensure the MCMC is efficient, we may require moves on a that use the gradient of the posterior  $\pi_{t,a}(a)$ , such as the Metropolis-adjusted Langevin algorithm (MALA). In the pseudo-marginal case this gradient is not available analytically, but it may be estimated as in, e.g. Dahlin  $et\ al.\ (2015)$ . For simplicity we do not consider this approach in the paper; instead we use a Gaussian random walk proposal. At iteration t we take  $q_{t,a}$  to be a multivariate Gaussian with mean the current point and covariance  $\Sigma_{t,q_a}$ . Based on Sherlock  $et\ al.\ (2010)$ , at iteration t we take  $\Sigma_{t,q_a}$  to be the scaled empirical covariance of the reprojected particles from iteration t-1

$$\Sigma_{q_a} = \hat{\Sigma}_{t-1 \to t,a} = \frac{2.38^2}{d_a} \sum_{m=1}^{N_a} \omega_t^m \left( a_{t-1 \to t}^m - \hat{\mu}_{t-1 \to t,a} \right) \left( a_{t-1 \to t}^m - \hat{\mu}_{t-1 \to t,a} \right)^T$$

where

$$\hat{\mu}_{t-1 \to t, a} = \sum_{m=1}^{N_a} \omega_t^m a_{t-1 \to t}^m.$$

This adaptation takes place directly following line 20 of algorithm 2, or after line 28 of algorithm 3.

#### **2.3.6** Choice of $q_{t,i}$

We propose to adapt the parameters  $\mu_{t,q_i}$ ,  $\Sigma_{t,q_i}$  following line 10 of algorithm 2, or after line 17 of algorithm 3. The optimal choice of  $q_{t,i}$  ( $i \mid a$ ) would be proportional to  $p_i$  ( $i \mid a$ )  $l_{1:t}$  ( $A_t a + I_t i$ ). From the previous iteration, for each  $a^m$ , we have a population of  $N_i - 1$  weighted points distributed according to  $q_{t-1,i}$  ( $i \mid a^m$ ). We use these populations to approximate the mean and covariance of the optimal choice using importance sampling. We first calculate the weight

$$\tilde{\kappa}_{t}^{n,m} = w_{t-1}^{n,m} l_{t} \left( A_{t} a_{t-1 \to t}^{m} + I_{t} i_{t-1 \to t}^{n,m} \right)$$

 $\text{for } m=1: N_a \text{ and } n=1: N_i, n \neq u^m_{t-1 \to t}, \text{ then for } m=1: N_a \text{ normalise } \left(\tilde{\kappa}^{1,m}_t, ..., \tilde{\kappa}^{u^m_{t-1}-1,m}_t, \tilde{\kappa}^{u^m_{t-1}+1,m}_t, ..., \tilde{\kappa}^{N_i,m}_t\right)$  to yield normalised weights  $\left(\kappa^{1,m}_t, ..., \kappa^{u^m_{t-1 \to t}-1,m}_t, \kappa^{u^m_{t-1 \to t}+1,m}_t, ..., \kappa^{N_i,m}_t\right).$  For  $m=1: N_a$  we then calculate

$$\hat{\mu}_{t-1\to t,i}^{m} = \sum_{\substack{n=1\\n\neq u_{t-1}^{m} \\ t \neq i}}^{N_{i}} \kappa_{t}^{n,m} i_{t-1\to t}^{n,m}$$

$$\hat{\Sigma}_{t-1 \to t, i}^{m} = \sum_{\substack{n=1 \\ n \neq u_{t-1 \to t}^{m}}}^{N_{i}} \kappa_{t}^{n, m} \left( i_{t-1 \to t}^{n, m} - \hat{\mu}_{t-1 \to t}^{m} \right) \left( i_{t-1 \to t}^{n, m} - \hat{\mu}_{t-1 \to t, i}^{m} \right)^{T}$$

and use  $\hat{\mu}_{t-1\to t,i}^m$  and  $\hat{\Sigma}_{t-1\to t,i}^m$  as the parameters of the proposal  $q_{t,i}$  ( $\cdot \mid \theta_a^m$ ). For algorithm 3, we take  $q_{t-1\to t,i} = p_{t,i}$ .

# 2.4 Active subspace SMC<sup>2</sup>

A weakness of the AS-SMC, is the same as AS-MH: that the variance of the IS estimator of the likelihood in equation 6 will grow exponentially as the dimension of the inactive variables increases, even if the likelihood is only a little informative about these variables. A way of circumventing this problem is to change the marginal likelihood estimator from the one in equation 6 to an SMC estimator, which is likely to have a lower variance in high dimensions.

We propose to estimate 5 using an SMC sampler with the sequence of distributions  $\pi_{0,i}(i \mid a) = p_i(i \mid a)$  and  $\pi_{t,i}(i \mid a) \propto p_i(i \mid a) l_{1:t}(Aa + Ii)$  for t = 1:T, assuming once again that the AS is fixed across the SMC iterations. For the move step in this SMC, at iteration t we propose to use an MH move with independent proposal  $q_{t,i}$ . This 'internal' SMC sampler may be embedded within an 'external' SMC sampler on the space of a using the approach of Chopin et al. (2013). The algorithm follows exactly the SMC<sup>2</sup> framework of Chopin et al. (2013). Full details are found in algorithms 4 and 5 in the appendix, where algorithm 4 is called by algorithm 5. The primary difference between AS-SMC and AS-SMC<sup>2</sup> are that the inactive variables are not drawn independently at each SMC iteration. Instead the inactive variables from the previous step are stored and reused: SMC<sup>2</sup> guides a set of particles to find 'good' values for the inactive variables. The justification for this algorithm can be found in appendix C.

To choose  $q_{s,i}$  in algorithm 4, we use a similar scheme to that in section 2.3.6, where the adaptation is performed at line 10, and  $\hat{\mu}_{s-1,i}$  and  $\hat{\Sigma}_{s-1,i}$  are used as the parameters of the proposal  $q_{s,i}(\cdot)$ , with

$$\hat{\mu}_{s-1,i} = \sum_{m=1}^{N_a} \omega_t^m \sum_{n=1}^{N_i} w_s^n i_{s-1}^n$$

$$\hat{\Sigma}_{s-1,i} = \sum_{m=1}^{N_a} \omega_t^m \sum_{n=1}^{N_i} w_s^n \left( i_{s-1}^n - \hat{\mu}_{s-1,i} \right) \left( i_{s-1}^n - \hat{\mu}_{s-1,i} \right)^T.$$

## 2.4.1 SMC<sup>2</sup> variant of adaptive AS-SMC

We might also consider the SMC<sup>2</sup> extension in adaptive AS-SMC, although it is significantly more complex and with less potential gain. The reason for this is that, since the AS is changed throughout the algorithm, the reprojection step introduced in section 2.3.2 is required to simulate the inactive particles at each new SMC iteration, and this involves resampling  $N_i - 1$  particles from the proposal distribution  $q_{t,i}$ . The main advantage of AS-SMC<sup>2</sup> is that the set of  $N_i$  inactive particles are maintained and refined throughout the algorithm; an advantage which is mostly lost through use of the reprojection step, where only one of the existing inactive particles would be propagated.

As a substitute for this SMC<sup>2</sup> approach, we propose the following alternative: that algorithm 3 is followed for the initial SMC iterations then, the after some criterion is met, AS-SMC<sup>2</sup> is followed for the remainder of the target distributions. This switching criterion should be based on the estimates of the AS stabilising, which we might expect as t increases and the likelihood becomes increasingly influential. After switching, the particles representing the inactive variables may be replaced with particles drawn from the SMC sampler on  $\theta_i$ -space in AS-SMC<sup>2</sup>, with the  $\theta_a$  particles reweighted according to the 'exchange importance sampling step' from Chopin *et al.* (2013). We do not pursue this approach further in this paper.

# 3 Empirical results

# 3.1 Comparing SMC, AS-SMC and AS-SMC<sup>2</sup>

#### 3.1.1 Models

As an example of where active subspaces are applicable, we use a toy Gaussian model. We perform inference for parameter  $\theta = (\theta_1, ..., \theta_d)^T \in \mathbb{R}^d$ , given observations of variable  $y \in \mathbb{R}$ . We use prior  $\theta_i \sim \mathcal{N}\left(0, \sigma = \sqrt{5000}\right)$  and model  $y \sim \mathcal{N}\left(\sum_{i=1}^d \theta_i, 1\right)$ . We fit this model to 100 observations of y drawn from  $\mathcal{N}(0, 1)$ . This model is useful for studying the case of an 'ideal' AS. The  $\theta_i$  can only be identified as lying on a hyperplane of dimension k-1 with equation  $\sum_{i=1}^k \theta_i = 0$ . Figure 2a gives an illustration of this model for d=3. We call this model the 'plane' model. This model is very unrealistic. The artificial construction of a number of variables that are not involved in the likelihood is tailor-made for AS algorithms. However, this construction results in a ridge in the posterior, which is possible in some real models. However, it would be very unusual to see observe a ridge in the posterior that is linear; for this reason we consider a variation on the model where the ridge is not linear. Specifically, we consider instead the model  $y \sim \mathcal{N}\left(\sum_{i=1}^d \theta_i + b \sum_{j=1}^k \theta_i^2, 1\right)$ , for  $1 \le k \le d$ . Figure 3a gives an illustration of this model for d=3, k=2 and b=0.001. We call this model the 'banana' model, after the shape of the posterior it produces in two dimensions. Whilst this model is again unrealistic, it provides a useful test of the effectiveness of AS approaches outside of the ideal case. The same models are studied in Ripoli and Everitt (2024).

#### 3.1.2 Results

We take d=25 for the first model, and performed inference using a standard SMC sampler and AS-SMC with a sequence of targets given by annealing the likelihood. The eigenvalue approach led to the choice of AS of dimension 1 (figure 2b). We used a fixed sequence of 25 targets for both algorithms, with the sequence being determined by a preliminary run of an adaptive SMC sampler. The standard SMC used  $10^4$  particles; AS-SMC used the same number of likelihood evaluations, taking  $N_a=10^3$  and  $N_i=10$ . Both algorithms used a Gaussian random walk proposal, tuned using the adaptive approach from section 2.3.5. The proposal on inactive variables was taken to be the projected prior  $p_i$ . We estimated the root mean squared error (RMSE) of the posterior mean of each of the 25 parameters (approximating the truth as 0 for each) from 50 runs of the algorithms. Figure 2c compares these estimated RMSEs, where the distribution is across the different parameters. We see that AS-SMC outperforms standard SMC in this idealised case.

We now compared standard SMC, AS-SMC and AS-SMC<sup>2</sup> on the banana model, for d = 25, k = 3 and b = 0.001. The eigenvalue approach led to the choice of AS of dimension 4 (figure 2b). This indicates a significant flaw in the use of active subspaces in this example: the small curve in the 'plane' introduced by the coefficient b = 0.001 results in four variables being grouped together in the AS: it is only the variables that are absent from the likelihood that are treated as inactive.

We again used a fixed sequence of 25 targets determined by a preliminary run of an adaptive SMC. The standard SMC used  $10^4$  particles and AS-SMC and AS-SMC<sup>2</sup> used  $N_a = 10^3$  and  $N_i = 10$ . This gives AS-SMC the same number of likelihood evaluations as the standard SMC, but due to the costlier MCMC moves the computational cost

of AS-SMC<sup>2</sup> is higher with  $3 \times 10^5$  likelihood evaluations. All algorithms used a Gaussian random walk proposal, tuned using the adaptive approach from section 2.3.5, and the prior was always used as the proposal for the inactive variables. Figure 2c compares the estimated RMSE of the estimated posterior expectation over all parameters from 50 runs of the three algorithms. In this example we observe poor performance from both the AS-SMC and AS-SMC<sup>2</sup> algorithms. In both cases this is due to the smaller number of (active space) particles used for these algorithms. We use this smaller number of particles due to the additional computational effort expended in the IS/SMC on the inactive space. This highlights a weakness of AS approaches that build on the method of Constantine *et al.* (2016): that we spend additional computational effort on simulating points for the inactive variables, which are the variables that should be easiest to simulate.

## 3.2 Example of adaptive AS-SMC

To illustrate adaptive AS-SMC, we applied it to the toy example from section 1.3. Our implementation used an annealed sequence of distributions determined by a pilot run of an adaptive SMC sampler. We performed 10 runs with  $N_a = 10^3$  active particles and  $N_i = 10$  inactive IS points, with a Gaussian random walk proposal on the active variables whose variance was adapted using the method in section 2.3.5. Figure 5 illustrates how the AS changes across the iterations of the adaptive AS-SMC algorithm through showing, in the PCA on equation (3), the proportion of variance explained by the active direction in the posterior AS. However, although we see the expected evolution of the AS in this example, we find that the adaptive AS-SMC algorithm has poor performance compared to standard SMC. This is since, as can be seen readily from the construction in section 1.3, there is no evidence that there are any inactive directions in this example.

## 4 Discussion

In this paper we have introduced three SMC samplers that make use of an AS. We have seen how the first of these, AS-SMC, can exhibit improved performance over a standard SMC sampler in a model with variables that are unidentifiable from the likelihood. The second method, adaptive AS-SMC, can be used to estimate an appropriate AS at each iteration, moving from the prior to the posterior. The third, AS-SMC<sup>2</sup>, addresses a different weakness of AS-SMC, substituting the IS step in AS-SMC for an 'internal' SMC move on the inactive variables in order to avoid the exponential scaling of the variance of IS as the dimension of the inactive variables increases.

SMC approaches can offer advantages over MCMC, since they use a population of particles and are easier to tune. In this sense, AS-SMC has some advantages over the current state-of-the-art AS approach, AS-MH. Adaptive AS-SMC might improve on AS-SMC in the case where the posterior AS is far from the prior AS, and where an 'ideal' AS exists. Parente (2020); Zahm et al. (2022) study such models: these papers also consider adaptive AS approaches, although without theoretical guarantees. AS-SMC<sup>2</sup> has the potential to provide lower variance estimates than AS-SMC, although it has a higher computational cost.

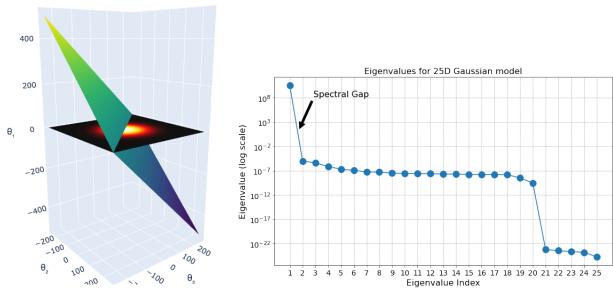
All approaches considered in this paper are limited by the assumptions underpinning active subspaces: that there is a subspace under a linear transformation for which the likelihood is not informative. Outside of this setting, it is not clear that AS-MH, AS-SMC, adaptive AS-SMC or AS-SMC<sup>2</sup> will outperform a standard approach. Despite some evidence in previous studies (e.g. Constantine et al. (2016); Parente (2020); Zahm et al. (2022)) that using an AS can result in more efficient inference we are skeptical that approaches that build on the marginal likelihood estimation approach of Constantine et al. (2016) will find wide applicability, due to the poor performance of the methods when the linear transformation is not suitable. This is studied further, in the context of MCMC, in Ripoli and Everitt (2024).

# Acknowledgements

Leonardo Ripoli's work was supported by EPSRC grant EP/L016613/1 (the Centre for Doctoral Training in the Mathematics of Planet Earth). Richard G. Everitt was supported by EPSRC grant EP/W006790/1 and NERC grant NE/T00973X/1.

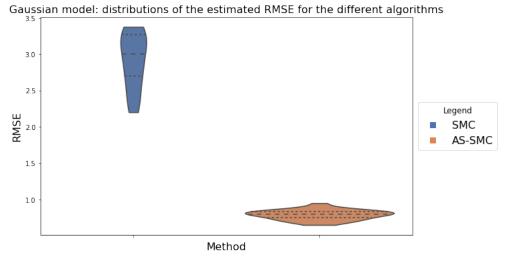
# A Full algorithms

This section contains step-by-step descriptions of the algorithms devised in the paper. Algorithm 3 contains the adaptive AS-SMC algorithm. Algorithm 4 contains the internal SMC algorithm used on inactive variables, which



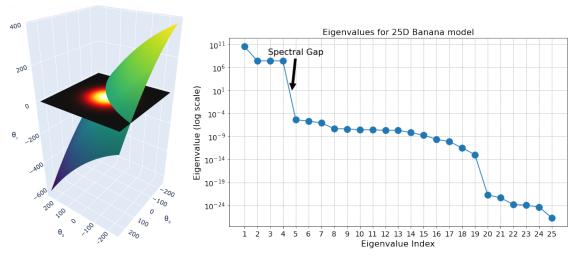
(a) A 2d slice of the Gaussian prior on the horizontal plane  $\theta_1=0$  together with the level surface of the likelihood in the particular case  $\theta_1+\theta_2+\theta_3=0$  (green plane).

(b) The eigenvalues used in determining the AS.



(c) Estimated RMSE of the posterior expectations over 50 runs.

Figure 2: The plane model: illustration and results.



(a) A 2d slice of the Gaussian prior on the horizontal plane  $\theta_1=0$  together with the level surface of the likelihood in the particular case  $\theta_1+\theta_2+\theta_3+b\theta_2^2+b\theta_3^2=0$  for b=0.001 (green curved plane).

(b) The eigenvalues used in determining the AS.



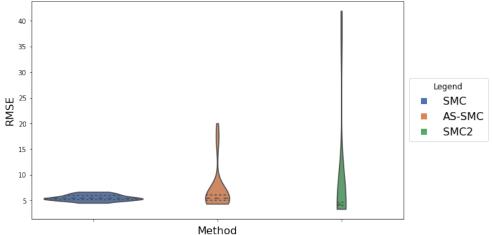


Figure 3: Estimated RMSE of the posterior expectations over 50 runs.

Figure 4: The banana model: illustration and results.

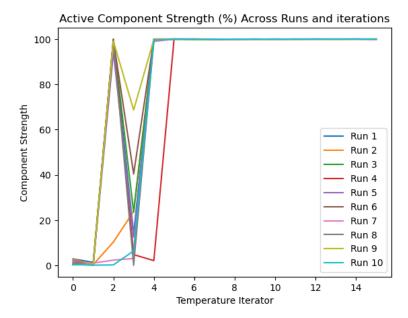


Figure 5: The evolution of the estimated AS across the iterations of the adaptive SMC.

is called by algorithm 5.

# B Special cases for adaptive AS-SMC

This section describes the special cases of adaptive AS-SMC when no inactive variables are found in either iteration t-1 or iteration t.

## B.1 Moving to no inactive variables

When there are no inactive variables at iteration t,  $a = \theta$ , and the target distribution is chosen to be

$$\pi_{t}(\theta) = \frac{1}{Z_{t}} p(\theta) l_{1:t}(\theta).$$

We first consider the reprojection step at the beginning of the SMC iteration, when moving from inactive variables existing at iteration t-1, but not existing at iteration t. This step is more straightforward than previously. The desired target distribution for the move is

$$\pi_{t-1}^{*}(u,\theta) = \frac{1}{N_{i}} \frac{1}{Z_{t-1}} p(\theta) l_{1:t-1}(\theta),$$

and the extended target then

$$\pi_{t-1}^{*}\left(u,\theta\right)L\left(\left\{ i_{t-1}^{n}\right\} _{n=1,n\neq u}^{N_{i}}\mid u,\theta\right).$$

The proposal in the reprojection is  $\pi_{t-1}^* \left( u, a_{t-1}, \left\{ i_{t-1}^n \right\}_{n=1}^{N_i} \right)$ , i.e. the target distribution from the previous step. The proposal takes a point  $\left( u_{t-1}, a_{t-1}, \left\{ i_{t-1}^n \right\}_{n=1}^{N_i} \right)$  from this target and transforms it using

$$\theta_{t-1\to t} = G_{t-1\to t,a} \left( u_{t-1}, a_{t-1}, \left\{ i_{t-1}^n \right\}_{n=1}^{N_i} \right) = A_{t-1} a_{t-1} + I_{t-1} i_{t-1}^{u_{t-1}}$$

$$u_{t-1\to t} = u_{t-1}. \tag{20}$$

The L kernel is again chosen so that the importance weights are 1, and the indexing variable is again discarded. The result is that this step simplifies to applying the transform in equation (20) to each particle.

# Algorithm 3: Adaptive AS-SMC

```
1 Simulate N_a points, \left\{\theta_0^m\right\}_{m=1}^{N_a} \sim p and set each weight \omega_0^m = 1/N_a;
2 Find AS using points \left\{\theta_0^m\right\}_{m=1}^{N_a} in equation (3), yielding A_0 and I_0;
                                                     7
                                                       Set w_0^{n,m} = 1/N_i for n = 1:N_i
       8
10 for t = 1 : T do
                                                     Find AS using points \left\{a_{t-1}^m \left\{i_{t-1}^{n,m}\right\}_{n=1}^{N_i}\right\}_{m=1}^{N_a} in equation (14), yielding A_t and I_t;
 11
 12
                                                                                       \begin{array}{l} \text{Set } a_{t-1 \to t}^T = A_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right), \ i_{t-1 \to t}^{u_{t-1}, m} = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1 \to t}^m = u_{t-1}^m \right) \\ \text{Set } a_{t-1}^m = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1 \to t}^m = u_{t-1}^m \right) \\ \text{Set } a_{t-1}^m = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1 \to t}^m = u_{t-1}^m \right) \\ \text{Set } a_{t-1}^m = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1 \to t}^m = u_{t-1}^m \right) \\ \text{Set } a_{t-1}^m = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1 \to t}^m = u_{t-1}^m \right) \\ \text{Set } a_{t-1}^m = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1 \to t}^m = u_{t-1}^m \right) \\ \text{Set } a_{t-1}^m = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1 \to t}^m = u_{t-1}^m \right) \\ \text{Set } a_{t-1}^m = I_t^T \left( A_{t-1} a_{t-1}^m + I_{t-1} i_{t-1}^{u_{t-1}, m} \right) \ \text{and} \ u_{t-1}^m = u_{t-1}^m + u_{t-1
 13
                                                                                   \begin{array}{l} \text{for } n=1:N_i, \ n\neq u_{t-1}^m \ \text{do} \\ & \begin{vmatrix} i^{n,m}_{t-1\rightarrow t} \sim \kappa_{t-1,i} \left(\cdot \mid a_{t-1\rightarrow t}^m \right) \\ \end{vmatrix} \end{array} end
 14
 15
 16
 17
 18
19
                                                                                                           \begin{split} & \cdot n = 1 : N_i \text{ do} \\ & \quad \tilde{w}_t^{n,m} \left( a_{t-1 \to t}^m, i_{t-1 \to t}^{n,m} \right) = \frac{p_{t,i} \left( i_{t-1 \to t}^{n,m} | a_{t-1 \to t}^m \right) l_{1:t} \left( A_t a_{t-1 \to t}^m + I_t i_{t-1 \to t}^n \right)}{q_{t,i} \left( i_{t-1 \to t}^{n,m} | a_{t-1 \to t}^m \right) \left( i_{t-1 \to t}^{n,m} | a_{t-1 \to t}^m \right)}; \\ & \quad \tilde{w}_{t-1}^{n,m} \left( a_{t-1 \to t}^m, i_{t-1 \to t}^{n,m} \right) = \frac{p_{t,i} \left( i_{t-1 \to t}^{n,m} | a_{t-1 \to t}^m \right) l_{1:t} - \left( A_t a_{t-1 \to t}^m + I_t i_{t-1 \to t}^n \right)}{\kappa_{t-1,i} \left( i_{t-1 \to t}^{n,m} | a_{t-1 \to t}^m \right)}. \end{split}
 21
                                                                                                                                                                                                                                                                                                                          \tilde{\boldsymbol{\omega}}_{t}^{m} = \boldsymbol{\omega}_{t-1}^{m} \frac{\prod_{j=1}^{N_{i}} q_{t,i} \left(\boldsymbol{\theta}_{i_{t}}^{j} + \boldsymbol{\theta}_{a_{t}}\right) \sum_{n=1}^{N_{i}} \tilde{\boldsymbol{w}}_{t}^{n,m} \left(\boldsymbol{a}_{t-1 \to t}^{m}, i_{t-1 \to t}^{n,m}\right)}{\prod_{i=1}^{N_{i}} \sum_{n=1}^{N_{i}} \left(\boldsymbol{\theta}_{i_{t}}^{j} + \boldsymbol{\theta}_{a_{t}}\right) \sum_{n=1}^{N_{i}} \tilde{\boldsymbol{w}}_{t}^{n,m} \left(\boldsymbol{a}_{t-1 \to t}^{m}, i_{t-1 \to t}^{n,m}\right)};
                                                       \begin{aligned} &\{\boldsymbol{\omega}_{t}^{m}\}_{m=1}^{N_{a}} \leftarrow \text{ normalise}\left(\{\tilde{\boldsymbol{\omega}}_{t}^{m}\}_{m=1}^{N_{a}}\right); \\ &\text{for } m=1:N_{a} \text{ do} \\ & \left\{\boldsymbol{w}_{t}^{n,m}\right\}_{n=1}^{N_{i}} \leftarrow \text{ normalise}\left(\left\{\tilde{\boldsymbol{w}}_{t}^{n,m}\right\}_{n=1}^{N_{i}}\right); \end{aligned}
 25
 26
 27
                                                         \begin{array}{l} \vdots \\ \text{end} \\ \text{for } m = 1: N_a \text{ do} \\ \\ u_t^m \sim \mathcal{M}\left(\left(w_t^{1,m}, ..., w_t^{N_i,m}\right)\right); \end{array}
28
29
 30
                                                       31
 32
                                                                                                                     Simulate \left(\theta_{t,a_t}^m, \theta_{t,i_t}^{1:N_i,m}\right) from the mixture distribution
                                                                                                                                                                                                                                                                                                                                                                                                                                                 \sum_{i=1}^{N_a} \omega_t^j K_{t,a} \left\{ \cdot \mid \left( a_{t-1 \to t}^j, i_{t-1 \to t}^{1:N_i, j} \right) \right\},
                                                                                                                                 where K_{t,a} is an AS-MH move, i.e.:
                                                                                                                     j^* \sim \mathcal{M}\left(\left\{\omega_t^j\right\}_{j=1}^{N_a}\right);
 35
 36
 37
    38
                                                                                                                                                                                                                                                                                                                                                            \bar{w}_{t}^{*n,m}\left(a_{t}^{*m},i_{t}^{*n,m}\right) = \frac{p_{t,i}\left(i_{t}^{*n,m} \mid a_{t}^{*m}\right)l_{1:t}\left(A_{t}a_{t}^{*m} + I_{t}i_{t}^{*n,m}\right)}{q_{t,i}\left(i_{t}^{*n,m} \mid a_{t}^{*m}\right)};
 40
                                                                                                                      u_t^{*m} \sim \mathcal{M}\left(\left(w_t^{*1,m}, ..., w_t^{*N_i,m}\right)\right) , where for n=1:N_i
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            w_t^{*n,m} = \frac{\tilde{w}_t^{*n,m}}{\sum_{n=1}^{N} \tilde{w}_t^{*p,m}}
                                                                                                                     \text{Set}\left(a_t^m, \left\{i_t^{n,m}, \tilde{w}_t^{n,m}\right\}_{n=1}^{N_i}, u_t^m\right) = \left(a_t^{*m}, \left\{i_t^{*n,m}, \tilde{w}_t^{*n,m}\right\}_{n=1}^{N_i}, u_t^{*m}\right) \text{ with probability}
                                                                                                                                                                                                                                                                                                         \alpha_{t,a_{t}}^{m} = 1 \wedge \frac{p_{t,a}\left(a_{t}^{*m}\right)\sum_{n=1}^{N_{i}} \bar{w}_{t}^{*n,m}\left(a_{t}^{*m}, i_{t}^{*n,m}\right)}{p_{t,a}\left(a_{t-1 \to t}^{j*}\right)\sum_{n=1}^{N_{i}} \bar{w}_{t}^{*n,j*}\left(a_{t-1 \to t}^{j*}, i_{t-1 \to t}^{*n,j*}\right)} \frac{q_{t,a}\left(a_{t}^{j*} \mid a_{t}^{*m}\right)}{q_{t,a}\left(a_{t,a}^{*m} \mid a_{t}^{j*}\right)};
                                                                                                                     \text{Else let } \left(a_t^m, \left\{i_t^{n,m}, \bar{w}_t^{n,m}\right\}_{n=1}^{N_i}, u_t^m\right) = \left(a_t^{j^*}, \left\{i_t^{n,j^*}, \bar{w}_t^{n,j^*}\right\}_{n=1}^{N_i}, u_t^{j^*}\right) = \left(a_t^{j^*}, \left\{i_t^{n,j^*}, \bar{w}_t^{n,j^*}\right\}_{n=1}^{N_i}, u_t^{j^*}\right)
 43
                                                                                      \begin{array}{l} \mathbf{end} \\ \boldsymbol{\omega}_t^m = 1/N_a \text{ for } m = 1:N_a; \end{array}
```

## **Algorithm 4:** SMC on inactive variables for a given a and t.

```
1 Simulate N_i points, \{i_0^n\}_{n=1}^{N_i} \sim p_i \left(\cdot \mid a\right) and set each weight w_0^n = 1/N_i;
 2 for s = 1 : t \ do
 3
            for n = 1 : N_i \text{ do // reweight}
                  if s = 1 then
  4
  5
                                                                                           \tilde{w}_{s}^{n} = w_{s-1}^{n} l_{1:s} (B_{a}a + B_{i}i^{n});
                   else
 6
                                                                                      \tilde{w}_{s}^{n} = w_{s-1}^{n} \frac{l_{1:s} \left( B_{a} a + B_{i} i_{s-1}^{n} \right)}{l_{1:s-1} \left( B_{a} a + B_{i} i_{s-1}^{n} \right)};
                   end
  8
 9
            \left\{w_s^n\right\}_{n=1}^{N_i} \leftarrow \text{ normalise}\left(\left\{\tilde{w}_s^n\right\}_{n=1}^{N_i}\right);
10
            If s = t, go to line 32;
11
            for n = 1 : N_i \ do
12
                  Simulate the index v_{s-1}^n \sim \mathcal{M}\left(\left(w_s^1,...,w_s^{N_i}\right)\right) of the ancestor of particle n;
13
14
            if some degeneracy condition is met then // resample
15
                   for n = 1 : N_i do
16
                       Set i_s^n = i_{s-1}^{v_{s-1}^n};
17
18
                  w_s^n = 1/N_i \text{ for } n = 1:N_i;
19
            end
20
            else
21
                   for n = 1 : N_i do
\mathbf{22}
                    Set i_s^n = i_{s-1}^n;
23
                   end
\mathbf{24}
            \mathbf{end}
25
            \mathbf{for}\ n=1:N_i\ \mathbf{do}\ //\ \mathtt{move}
26
                  Simulate i_s^{n*} \sim q_{t,i} (\cdot \mid i_s^n, a);
Set i_s^n = i_s^{n*} with probability
27
28
29
                                                                         1 \wedge \frac{l_{1:s} (B_a a + B_i i_s^{n*}) p_i (i_s^{n*} \mid a) q_{t,i} (i_s^{n} \mid a)}{l_{1:s} (B_a a + B_i i_s^{n}) p_i (i_s^{n} \mid a) q_{t,i} (i_s^{n*} \mid a)},
            end
30
31 end
32 Estimate l_{t,a}\left(a\right) using
                                                                                        \bar{l}_{t,a}\left(a\right) = \prod_{s}^{t} \sum_{s}^{N_{i}} \tilde{w}_{s}^{n}.
```

```
Algorithm 5: Active subspace SMC<sup>2</sup>
```

```
1 Simulate N_a points, \{\theta_0^m\}_{m=1}^{N_a} \sim p and set each weight \omega_0^m = 1/N_a;
 2 Find AS using points \{\theta_0^m\}_{m=1}^{N_a} in equation (3), yielding B_a and B_i;
 3 for m = 1 : N_a do
             Set a_0^m = B_a^T(\theta_0^m), i_0^{1,m} = B_i^T(\theta_0^m) \text{ and } u_0^m = 1;
             for n=2:N_i do \mid i_0^{n,m} \sim q_{0,i}\left(\cdot \mid a_0^m\right):=p_i\left(\cdot \mid a_0^m\right);
  5
  6
  7
             Set w_0^{n,m} = 1/N_i for n = 1 : N_i;
  8
 9 end
10 for t = 1 : T do
              \mathbf{for}\ m = 1: N_a\ \mathbf{do}\ //\ \mathtt{reweight}
11
                     if t = 1 then
12
                            Simulate (i_t^{1:N_i,m}, v_{t-1}^{1:N_i,m}) using lines 3-30 (ignoring line 11) of algorithm 4 taking s in these lines equal
13
                               to t, then compute
                                                                                                              \widehat{l_{t,a}\left(a_{t-1}^{m}\right)} = \sum_{t=1}^{N_i} \widetilde{w}_t^{n,m};
                                                                                                              \widetilde{\omega}_{t}^{m} = \widehat{\omega_{t-1}^{m}} l_{t,a} \left( \widehat{a_{t-1}^{m}} \right);
14
                            Simulate (i_t^{1:N_i,m}, v_{t-1}^{1:N_i,m}) using lines 3-30 (ignoring line 11) of algorithm 4 taking s in these lines equal
15
                               to t, then compute:
                                                                                                           \frac{l_{t,a}\left(a_{t-1}^{m}\right)}{l_{t-1,a}\left(a_{t-1}^{m}\right)} = \sum_{i=1}^{N_{i}} \tilde{w}_{t}^{n,m};
                                                                                                           \tilde{\omega}_t^m = \omega_{t-1}^m \frac{l_{t,a}\left(a_{t-1}^m\right)}{l_{t-1,a}\left(a_{t-1}^m\right)};
                     end
16
17
              end
              \{\omega_t^m\}_{m=1}^{N_a} \leftarrow \text{ normalise } \left(\{\tilde{\omega}_t^m\}_{m=1}^{N_a}\right);
18
              if some degeneracy condition is met then // resample and move
19
                     for m = 1 : N_a do
20
                            Simulate \left(a_t^m, i_{1:t}^{1:N_i,m}, v_{1:t-1}^{1:N_i,m}\right) from the mixture distribution
21
                                                                                             \sum_{t=1}^{N_a} \omega_t^j K_{t,a} \left\{ \cdot \mid \left( a_{t-1}^j, i_t^{1:N_u,j}, v_{t-1}^{1:N_t,j} \right) \right\},\,
                            where K_{t,a} is an AS-PMMH move, i.e.: j^* \sim \mathcal{M}\left(\left\{\omega_t^j\right\}_{j=1}^{N_{\theta}}\right), then a^* \sim q_{t,a}\left(\cdot \mid a_{t-1}^{j^*}\right), then run algorithm 4 up to target t conditional on a^*.
22
                            Set a_t^m = a^* and i_{1:t}^{n,m}, v_{1:t-1}^{n,m} and \tilde{w}_{1:t}^{n,m} to be the variables and unnormalised weights generated when
23
                               running algorithm 4 with probability
24
                                                                                    1 \wedge \frac{p_{a}\left(a^{*}\right)}{p_{a}\left(a_{t-1}^{j^{*}}\right)} \frac{q_{t,a}\left(a_{t-1}^{j^{*}} \mid a^{*}\right)}{q_{t,a}\left(a^{*} \mid a_{t-1}^{j^{*}}\right)} \frac{\overline{l}_{t,a}\left(a^{*}\right)}{\prod_{t=1}^{T} \sum_{n=1}^{N_{t}} \tilde{w}_{t}^{n,j^{*}}},
                            where \bar{l}_{t,a}\left(a^{*}\right) given by algorithm 4 run conditional on \theta_{a}^{*};
Else set a_{t}^{m}=a_{t-1}^{j^{*}},\; \tilde{w}_{1:t}^{n,m}=\tilde{w}_{1:t}^{n,j^{*}},\; i_{1:t}^{n,m}=i_{1:t}^{n,j^{*}} and v_{1:t-1}^{n,m}=v_{1:t-1}^{n,j^{*}}.
25
26
                     \omega_t^m = 1/N_a for m = 1: N_a;
27
              end
28
     end
\mathbf{29}
```

The weight update then follows as

$$\tilde{\omega}_{t}^{m} = \frac{p\left(\theta_{t-1\to t}^{m}\right) l_{1:t}\left(\theta_{t-1\to t}^{m}\right)}{p\left(\theta_{t-1\to t}^{m}\right) l_{1:t-1}\left(\theta_{t-1\to t}^{m}\right)}$$
$$= l_{t}\left(\theta_{t-1\to t}^{m}\right),$$

and the MCMC move at iteration t is then a standard MH step.

## B.2 Moving from no inactive variables

When there are no inactive variables at iteration t-1 and there are again no inactive variables at iteration t, we simply follow section 20, but do not need to the transform in equation (20), i.e. the algorithm is a standard SMC sampler. When there no inactive variables at iteration t-1, but there are inactive variables at iteration t we mostly follow section 2.3.2, except that the conditional IS step at the beginning of the SMC iteration is slightly simpler.

As the target distribution for this IS step, we may use  $\pi_{t-1}\left(a_t, \{i_t^n\}_{n=1}^{N_i}\right)$ : the indexing variable u is not required. We use the transformation

$$a_{t-1 \to t} := G_{t-1 \to t,a} (\theta_{t-1}) := A_t^T (\theta_{t-1})$$
$$i_{t-1 \to t} := G_{t-1 \to t,i} (\theta_{t-1}) := I_t^T (\theta_{t-1})$$

on the point  $\theta_{t-1}$ , then, if  $N_i > 1$ , need to generate the remaining variables  $i_{t-1 \to t}^n$  for  $n = 2 : N_i$ . We simulate each from  $q_{t-1 \to t}$  and see straightforwardly that this results in an importance weight of 1.

# C Justification for AS-SMC<sup>2</sup>

This section follows exactly the corresponding argument in Chopin *et al.* (2013): here we sketch the main ideas, with the full argument being found in that paper.

At iteration t we require our algorithm to have as one of its marginals the target distribution

$$\pi_{t}\left(a,i\right) = \frac{p_{a}\left(a\right)p_{i}\left(i\mid a\right)l_{1:t}\left(B_{a}a + B_{i}i\right)}{Z_{t}},$$

with marginal distribution

$$\pi_{t,a}\left(a\right) = \frac{p\left(a\right)l_{t,a}\left(a\right)}{Z_{t}},$$

where

$$l_{t,a}\left(a\right) = \int_{i} p_{i}\left(i \mid a\right) l_{1:t}\left(B_{a}a + B_{i}i\right) di.$$

At t = 0,  $\{a_0^m\}_{m=1}^{N_a} \sim p_a$  and  $\{i_0^{n,m}\}_{n=1}^{N_i} \sim p_i \left( \cdot \mid a_0^m \right)$  for  $m = 1: N_u$ . The target at iteration 0 is

$$p\left(a\right)\prod_{n=1}^{N_{i}}p_{i}\left(i_{0}^{n}\mid a\right),$$

from which we simulate  $N_a$  a-points and  $N_i$  i-points for every a. At t=1 each particle is assigned the weight

$$\hat{l}_{1,a}(a) = \frac{1}{N_i} \sum_{n=1}^{N_i} l_1 \left( B_a a + B_i i_0^n \right). \tag{21}$$

We introduce notation for the distribution of the *i*-variables generated at iteration 0 used to estimate  $l_{1,a}$ :  $\psi_0\left(\left\{i_0^n\right\}_{n=1}^{N_i}\mid a\right) = \prod_{n=1}^{N_i} p_i\left(i_0^n\mid a\right)$ . The target distribution at t=1 being

$$\pi_1\left(a, \{i_0^n\}_{n=1}^{N_i}\right) = p_a\left(a\right)\psi_0\left(\{i_0^n\}_{n=1}^{N_i} \mid a\right) \frac{\hat{l}_{1,a}\left(a\right)}{Z_1}.$$
(22)

This results in the weight update in equation (21). We can rewrite the target as

$$\pi_{1}\left(a,\left\{i_{0}^{n}\right\}_{n=1}^{N_{i}}\right) = \frac{p_{a}\left(a\right)}{Z_{1}} \prod_{n=1}^{N_{i}} p_{i}\left(i_{0}^{n} \mid a\right) \left(\frac{1}{N_{i}} \sum_{n=1}^{N_{i}} l_{1}\left(B_{a}a + B_{i}i_{0}^{n}\right)\right)$$

$$= \frac{1}{N_{i}} \sum_{n=1}^{N_{i}} \frac{p_{a}\left(a\right)}{Z_{1}} p_{i}\left(i_{0}^{n} \mid a\right) l_{1}\left(B_{a}a + B_{i}i_{0}^{n}\right) \left(\prod_{\substack{j=1\\j \neq n}}^{N_{i}} p_{i}\left(i_{0}^{j} \mid a\right)\right)$$

$$= \frac{\pi_{1,a}\left(a\right)}{N_{i}} \sum_{n=1}^{N_{i}} \pi_{t,i}\left(i_{0}^{n} \mid a\right) \left(\prod_{\substack{j=1\\j \neq n}}^{N_{i}} p_{i}\left(i_{0}^{j} \mid a\right)\right)$$

$$(23)$$

where we used the result

$$p_a(a) p_i(i_0^n \mid a) l_1(B_a a + B_i i_0^n) = Z_1 \pi_{1,a}(a) \pi_{t,i}(i_0^n \mid a).$$

In the marginals of equation (23) we have the target at iteration 1 of  $\pi_1(a, i \mid y) = \pi_{1,a}(a) \pi_{1,i}(i_0^n \mid a)$ .

For  $t \geq 2$ , similarly to equation 22, we again have that our target distribution is defined to be the prior, multiplied by the likelihood estimate, multiplied by the distribution of the variables used in the likelihood estimator, multiplied by the normalising constant (which again follows from the unbiasedness of the likelihood estimator).

Let  $\psi_{t-1}$  be the distribution of all of the random variables generated by the internal SMC up to time t.

$$\pi_t \left( a, \left\{ i_{0:t-1}^n, v_{0:t-1}^n \right\}_{n=1}^{N_i} \right) = p_a \left( a \right) \psi_{t-1} \left( \left\{ i_{0:t-1}^n, v_{0:t-1}^n \right\}_{n=1}^{N_i} \mid a \right) \frac{\hat{l}_{t,a} \left( a \right)}{Z_t}, \tag{24}$$

Similarly to the t=1 case, we may rearrange equation (24) to see that  $\pi_t(a,i)$  is included in its marginals:

$$\begin{split} \pi_t \left( a, \left\{ i_{0:t-1}^n, v_{0:t-1}^n \right\}_{n=1}^{N_i} \right) &= \frac{\pi_{t,a} \left( a \right)}{N_i} \times \sum_{n=1}^{N_i} \frac{\pi_{t,i} \left( \mathbf{i}_{1:t-1}^n \mid a \right)}{N_u^{t-1}} \\ & \left( \prod_{\substack{j=1 \\ j \neq \mathbf{h}_t^n \left( 0 \right)}}^{N_i} p_i \left( i_0^j \mid a \right) \right) \left( \prod_{s=2}^t \prod_{\substack{j=1 \\ j \neq \mathbf{h}_t^n \left( s-1 \right)}}^{N_i} w_{s-1}^{v_{s-1}^j} K_{s-1,i} \left( i_{s-1}^j \mid i_{s-2}^{v_{s-2}^j}, a \right) \right), \end{split}$$

where  $\mathbf{i}_{1:t-1}^n$  and  $\mathbf{h}_t^n$  are deterministic functions of  $\left\{i_{0:t-1}^n\right\}_{n=1}^{N_i}$  and  $\left\{v_{0:t-1}^n\right\}_{n=1}^{N_i}$ :  $\mathbf{h}_t^n = (\mathbf{h}_t^n\left(0\right),...,\mathbf{h}_t^n\left(t-1\right))$  denotes the index history of  $v_{t-1}^n$ , i.e.  $\mathbf{h}_t^n\left(t-1\right) = n$  and  $\mathbf{h}_t^n\left(s\right) = v^{\mathbf{h}_t^n\left(s+1\right)}$ , recursively for s=t-2,...,0, and  $\mathbf{i}_{1:t-1}^n = \left(\mathbf{i}_{1:t-1}^n\left(0\right),...,\mathbf{i}_{1:t-1}^n\left(t-1\right)\right)$  denotes the state trajectory of particle  $i_{t-1}^n$ , i.e.  $\mathbf{i}_{1:t-1}^n\left(s\right) = i_s^{h_t^n\left(s\right)}$ , for s=0,...,t-1. For the remainder of the proof we follow Chopin  $et\ al.\ (2013)$ , with the one difference in the notation from that paper that here the index of the i variable is one fewer: i.e. equation (24) uses  $\psi_{t-1}\left(\left\{i_{0:t-1}^n,v_{0:t-1}^n\right\}_{n=1}^{N_i}\mid a\right)$ , whereas the equivalent in Chopin  $et\ al.\ (2013)$  would be  $\psi_t\left(\left\{i_{1:t}^n,v_{1:t-1}^n\right\}_{n=1}^{N_i}\mid a\right)$ . The reason is that here the weight update in the internal SMC only involves the values of the particles from the previous iteration.

# References

Agapiou, S., Papaspiliopoulos, O., Sanz-Alonso, D., and Stuart, A. M. (2017). Importance Sampling: Intrinsic Dimension and Computational Cost. *Statistical Science* 32(3), 405–431.

Alquier, P., Friel, N., Everitt, R. G., and Boland, A. (2016). Noisy Monte Carlo: Convergence of Markov chains with approximate transition kernels. *Statistics and Computing* 26(1), 29–47.

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. Journal of the Royal Statistical Society: Series B 72(3), 269–342.

Andrieu, C. and Roberts, G. O. (2009). The pseudo-marginal approach for efficient Monte Carlo computations. The Annals of Statistics 37(2), 697–725.

- Beaumont, M. A. (2003). Estimation of population growth or decline in genetically monitored populations. *Genetics* 164(3), 1139–1160.
- Chopin, N. (2002). A sequential particle filter method for static models. Biometrika 89(3), 539-552.
- Chopin, N., Jacob, P. E., and Papaspiliopoulos, O. (2013). SMC2 an efficient algorithm for sequential analysis of state space models. *Journal of the Royal Statistical Society: Series B* 75(3), 397–426.
- Constantine, P. G., Kent, C., and Bui-Thanh, T. (2016). Accelerating Markov Chain Monte Carlo with Active Subspaces. SIAM Journal on Scientific Computing 38(5), A2779–A2805.
- Dahlin, J., Lindsten, F., and Schön, T. B. (2015). Particle Metropolis-Hastings using gradient and Hessian information. Statistics and Computing 25(1), 81–92.
- Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B* 68(3), 411–436.
- Del Moral, P., Doucet, A., and Jasra, A. (2012). An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing* 22(5), 1009–1020.
- Douc, R., Cappé, O., and Moulines, E. (2005). Comparison of resampling schemes for particle filtering. In *ISPA* 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005., Zagreb, Croatia, pp. 64–69. IEEE.
- Everitt, R. G., Culliford, R., Medina-Aguayo, F., and Wilson, D. J. (2020). Sequential Monte Carlo with transformations. *Statistics and Computing* 30(3), 663–676.
- Kerama, I., Thorne, T., and Everitt, R. G. (2022). Rare event ABC-SMC2.
- Kong, A., Liu, J. S., and Wong, W. H. (1994). Sequential Imputations and Bayesian Missing Data Problems. Journal of the American Statistical Association 89 (425), 278–288.
- Mengersen, K. L. and Tweedie, R. L. (1996). Rates of convergence of the Hastings and Metropolis algorithms. *The Annals of Statistics* 24(1), 101–121.
- Parente, M. M. T. (2020). Active Subspaces in Bayesian Inverse Problems. Ph. D. thesis, Technische Universität München.
- Ripoli, L. and Everitt, R. G. (2024). Improved mcmc with active subspaces. arXiv.
- Schuster, I., Constantine, P. G., and Sullivan, T. J. (2017). Exact active subspace Metropolis-Hastings, with applications to the Lorenz-96 system.
- Sherlock, C., Fearnhead, P., and Roberts, G. O. (2010). The Random Walk Metropolis: Linking Theory and Practice Through a Case Study. *Statistical Science* 25(2), 172–190.
- South, L. F., Pettitt, A. N., and Drovandi, C. C. (2016). Sequential Monte Carlo for Static Bayesian Models with Independent MCMC Proposals. Technical report, Queensland University of Technology.
- Zahm, O., Cui, T., Law, K., Spantini, A., and Marzouk, Y. (2022). Certified dimension reduction in nonlinear Bayesian inverse problems.