Adaptive sequential Monte Carlo by means of mixture of experts

Julien Cornebise · Eric Moulines · Jimmy Olsson

Abstract Appropriately designing the proposal kernel of particle filters is an issue of significant importance, since a bad choice may lead to deterioration of the particle sample and, consequently, waste of computational power. In this paper we introduce a novel algorithm adaptively approximating the so-called optimal proposal kernel by a mixture of integrated curved exponential distributions with logistic weights. This family of distributions, referred to as *mixtures of experts*, is broad enough to be used in the presence of multi-modality or strongly skewed distributions. The mixtures are fitted, via online-EM methods, to the optimal kernel through minimisation of the Kullback-Leibler divergence between the auxiliary target and instrumental distributions of the particle filter. At each iteration of the particle filter, the algorithm is required to solve only a single optimisation problem for the whole particle sample, yielding an algorithm with only linear complexity. In addition, we illustrate in a simulation study how the method can be successfully applied to optimal filtering in nonlinear state-space models.

J. Cornebise (✉)
Department of Statistical Science, University College London, London, UK
e-mail: julien@stats.ucl.ac.uk

E. Moulines
Département Traitement du Signal et de l'Image, Telecom ParisTech, Paris, France

J. Olsson
Centre for Mathematical Science, Lund University, Lund, Sweden

1 Introduction

During the last decade, sequential Monte Carlo (SMC) methods have developed from being a tool for solving specific problems, such as the optimal filtering problem in general state-space models or simulation of rare events (these topics are abundantly covered in Liu 2001; Doucet et al. 2001; Ristic et al. 2004; Cappé et al. 2005; Del Moral et al. 2006, and the references therein). The high flexibility of these methods makes it possible to use them for efficiently executing key steps of composite algorithms tackling very complex problems, e.g., as in particle Markov chain Monte Carlo (MCMC) methods for joint inference on static and dynamic variables (see Andrieu et al. 2010).

However, while there has been abundant work on adaptation of MCMC methods, from the early algorithms of Haario et al. (2001) to the mathematical study of diminishing adaptation established in Andrieu and Moulines (2006) and Roberts and Rosenthal (2009), adaptive methods for SMC are still in their infancy. This article, which sets out from a theoretical framework for adaptive SMC presented in Cornebise et al. (2008), is part of the effort to fill this gap—which is required to achieve maximum efficiency and user-friendliness. A mathematically less detailed pre-version of the present paper appeared as part of the Ph.D. dissertation by Cornebise (2009).

Such adaptive methods are especially suited to cope with the rising complexity of *state-space models* (SSMs) in recent applications. While simple unidimensional filtering problems will rarely justify even a slight computational overhead of adaptive methods over plain SMC methods

(such as the *bootstrap particle filter*; see Sect. 3.1), the situation is reversed for many state-of-the-art SSMs, where forward simulation may require costly physical emulators or the evaluation of computationally expensive transition densities. This occurs, e.g., as soon as the model involves differential equations and numeric solutions of the same. Recent examples range from health sciences (see e.g. Korotsil et al. 2012, for a study of the epidemiology of human papillomavirus infections) to environmental sciences (see e.g. Robins et al. 2009, who studied data assimilation for atmospheric dispersion of hazardous airborne biological material).

In the SMC framework, the objective is to approximate a *sequence* of target densities, defined on a sequence of state spaces as follows. Given a density μ on some state space $\boldsymbol{\Xi}$, the next density $\tilde{\mu}$ in the sequence is defined by

$$\tilde{\mu}(\tilde{\mathbf{x}}) := \frac{\int \mu(\mathbf{x}) l(\mathbf{x}, \tilde{\mathbf{x}}) \, d\mathbf{x}}{\iint \mu(\mathbf{x}) l(\mathbf{x}, \tilde{\mathbf{x}}) \, d\mathbf{x} \, d\tilde{\mathbf{x}}}, \quad \tilde{\mathbf{x}} \in \tilde{\boldsymbol{\Xi}}, \tag{1.1}$$

where $\tilde{\boldsymbol{\Xi}}$ is some other state space and $l : \boldsymbol{\Xi} \times \tilde{\boldsymbol{\Xi}} \to \mathbb{R}_+$ is an un-normalised transition density. Here the dimensions of $\tilde{\boldsymbol{\Xi}}$ and $\boldsymbol{\Xi}$ typically differ. This framework encompasses, inter alia, the classical problem of *optimal filtering* in SSMs recalled in Sect. 6 (see in particular Eq. (6.4)).

SMC methods approximate online the densities generated by (1.1) by a recursively updated set of random *particles* $\{\mathbf{X}_i\}_{i=1}^N$ and associated *importance weights* $\{\omega_i\}_{i=1}^N$. At each iteration of the algorithm, the weighted particle sample $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^N$ approximates the corresponding target density μ in the sense that $\sum_{i=1}^N \omega_i f(\mathbf{X}_i)/\Omega$, with $\Omega := \sum_{j=1}^N \omega_j$, approximates $\mathbb{E}_\mu[f(\mathbf{X})] := \int f(\mathbf{x}) \mu(\mathbf{x}) \, d\mathbf{x}$ (this will be our generic notation for expectations) for all bounded measurable functions f on the corresponding state space $\boldsymbol{\Xi}$. All SMC methods have two main operations in common: a *mutation step* and a *selection step*. In the former, the particles are randomly scattered in the state space by means of a *proposal* transition density $r : \boldsymbol{\Xi} \times \tilde{\boldsymbol{\Xi}} \to \mathbb{R}_+$; more specifically, the mutation operation generates a set of new, $\tilde{\boldsymbol{\Xi}}$-valued particles by moving the particles according to the transition density r and assigning each mutated particle an updated importance weight proportional to l/r; consequently, each weight reflects the relevance of the corresponding particle as measured by the likelihood ratio. In general, the mutated particles are conditionally independent given the ancestor particles. In the selection step, the particles are duplicated or eliminated with probabilities depending on the importance weights. This is typically done by multinomially resampling the particles with probabilities proportional to the corresponding weights. In the *auxiliary particle filter* (proposed by Pitt and Shephard 1999) the particles are resampled according to weights that are modified by a factor determined by an *adjustment multiplier function* $a : \boldsymbol{\Xi} \to \mathbb{R}_+$. Introducing adjustment multipliers makes it possible to, by using

some convenient lookahead strategy, increase the weight of those particles that will most likely contribute significantly to the approximation of the subsequent target distribution after having been selected. In this way computational power is directed toward zones of the state space where the mass of the subsequent target density is located.

The choice of the adjustment weight function a and the proposal density r affects significantly the quality of the generated sample, and in this paper we focus on adaptive design of the latter. Letting $r(\mathbf{x}, \tilde{\mathbf{x}}) = l(\mathbf{x}, \tilde{\mathbf{x}}) / \int l(\mathbf{x}, \tilde{\mathbf{x}}) \, d\tilde{\mathbf{x}}$ is appealing as the particles in this case are propagated according to a dynamics that is closely related to that of the recursion (1.1). Indeed, by combining this choice of proposal kernel with the adjustment multiplier function $a(\mathbf{x}) = \int l(\mathbf{x}, \tilde{\mathbf{x}}) \, d\tilde{\mathbf{x}}$ yields perfectly uniform importance weights. In this case the SMC algorithm is referred to as *fully adapted*. However, this so-called *optimal* proposal kernel and adjustment multiplier weight function can be expressed on closed-form only in a few cases and one is in general referred to approximations of these quantities.

There is a large literature dealing with the problem of approximating the optimal proposal kernel. When using SMC for optimal filtering in SSMs, Doucet et al. (2000) suggest approximating each function $l(\mathbf{x}, \cdot)$ by a Gaussian density whose mean and covariance are obtained using the *extended Kalman filter*. A similar approach based on the *unscented Kalman filter* was proposed by Van der Merwe et al. (2000) (see also Van der Merwe and Wan 2003). This technique presupposes that the model under consideration can be well approximated by a nonlinear Gaussian SSM with linear measurement equation; however, this far from always the case, and careless linearisation (e.g. in cases where the local likelihood is multimodal) may boomerang and cause severe deterioration of the particle sample. Cappé et al. (2005, Sect. 7.2.2.4) make a form of Laplace approximation of $l(\mathbf{x}, \cdot)$ by simply locating the mode of the function in question and centering a Gaussian density or the density of a Student's t-distribution around the same. This technique, which goes back to Pitt and Shephard (1999), is appropriate when the function is log-concave (or strongly unimodal); nevertheless, as the mode of each $l(\mathbf{x}, \cdot)$ is a function of $\mathbf{x}$, this approach involves solving one optimisation problem per particle. Thus, in spite of this intense recent activity in the field, the state-of-the-art algorithms have met only mitigated success as they implicitly assume that the functions $l(\mathbf{x}, \cdot)$ have a single mode.

A common practice (see e.g. Oh and Berger 1993) for designing proposal distributions in standard (non-sequential) importance sampling is to consider a parametric family of proposal distributions and then identify a parameter that minimizes some measure of discrepancy between the target and the proposal distributions. Common choices are the widely used squared *coefficient of variation* of the importance weights or the negated *Shannon entropy* (see e.g.

Cappé et al. 2005, p. 235, for definitions of these quantities) of the same. Both are maximal when the sample is completely degenerated, i.e. when one single particle carries all the weight, and minimal when all the importance weights are equal. The coefficient of variation often appears in a transformed form as the *effective sample size* (we refer again to Cappé et al. 2005, for a definition). The same quantities have been widely used also within the framework of SMC, but until recently only for triggering selection and not as a tool for adaptive design of the instrumental distribution. The key result of Cornebise et al. (2008, Theorems 1 and 2) was to relate the Shannon entropy to the *Kullback-Leibler divergence* (KLD; see Eq. (3.6)) between the instrumental and proposal distributions of the particle filter. More specifically, when selection is carried through using multinomial resampling, each particle cloud update (selection followed by resampling) can be viewed as standard importance sampling where a weighted mixture of N stratas—the ith stratum being proportional to $\omega_i l(\mathbf{X}_i, \cdot)$—approximating $\tilde{\mu}$ is targeted using a mixture proposal π comprising N stratas proportional to $\{\omega_i r(\mathbf{X}_i, \cdot)\}_{i=1}^N$. As the number of particles tends to infinity, one may prove that the KLD between these two mixtures tends to the KLD between two distributions having densities $\mu^*(\mathbf{x}, \tilde{\mathbf{x}}) \propto l(\mathbf{x}, \tilde{\mathbf{x}})\mu(\mathbf{x})$ and $\pi^*(\mathbf{x}, \tilde{\mathbf{x}}) \propto a(\mathbf{x})r(\mathbf{x}, \tilde{\mathbf{x}})\mu(\mathbf{x})$, which can be viewed as "asymptotic" target and proposal distributions on the product space $\boldsymbol{\Xi} \times \tilde{\boldsymbol{\Xi}}$. In addition, one may prove that the Shannon entropy has the same limit. (Cornebise et al. 2008 established a similar relation between the coefficient of variation of the particle weights and the *chi-square divergence* between the same distributions.) This gives a sound theoretical support for using the Shannon entropy of the importance weights for measuring the quality of the particle sample. In addition, it suggests that the KLD between the target mixture and π could be used in lieu of the Shannon entropy for all purposes, especially adaptation. As a matter of fact, in the context of adaptive design of SMC methods, the KLD is (as pointed out by Cornebise et al. 2008, Sect. 2.3) highly practical since it decouples the adaptation of the adjustment weight function a and that of the proposal kernel r; see the next section.

Henceforth, in the present article we extend and implement fully the methodology indicated and sketched by Cornebise et al. (2008) and select the auxiliary proposal distribution π from a family $\{\pi_{\boldsymbol{\theta}}; \boldsymbol{\theta} \in \Theta\}$ of candidate distributions by picking $\boldsymbol{\theta}^*$ such that the KLD between the target mixture and $\pi_{\boldsymbol{\theta}^*}$ is minimal and letting $\pi = \pi_{\boldsymbol{\theta}^*}$. On the one hand, the chosen parametric family should be flexible enough to approximate complex transition kernels; on the other hand, sampling from $\pi_{\boldsymbol{\theta}}$ should be easy. Finally, the parameterisation should be done in such a way that the problem of estimating the parameters is as simple as possible. In

this article, we suggest modeling the proposal $\pi_{\boldsymbol{\theta}}$ as a mixture of integrated curved exponential distributions. By letting the mixture weights of the chosen proposal depend on the ancestor particles $\{\mathbf{X}_i\}_{i=1}^N$, we allow for partitioning of the input space into regions corresponding to a specialised kernel. Each component of the mixture belongs to a family of integrated curved exponential distributions, whose two most known members are the multivariate Gaussian distribution and the Student's t-distribution. Also the parameters of the mixture depend on the ancestors $\{\mathbf{X}_i\}_{i=1}^N$. This parameterisation of the proposal distribution is closely related to the (hierarchical) *mixture of experts* appearing in the machine learning community and described in Jordan and Jacobs (1994), Jordan and Xu (1995). The flexibility of our approach allows for efficient approximation of the optimal kernel for a large class of intricate (nonlinear non-Gaussian) models. Unlike typical EM-type parameter estimation procedures, which are in general run iteratively until stabilisation, our adaptive SMC algorithm only requires a decrease of the KLD, not an exact minimum. As illustrated in our examples in Sect. 6, such a gain typically occurs already at early iterations of the algorithm (a characteristic of EM-type algorithms), implying the need of only very few EM iterations and therefore a minimal computational overhead.

The paper is organized as follows. In the next section we introduce some matrix notation and list the most common notation used throughout the paper. Section 3 describes, in a nutshell, auxiliary SMC methods as well as a KLD-based approach to adaptation of these methods that is used in the paper. Section 4 recalls mixtures of experts and Sect. 5 treats optimisation of the mixture parameters by means of stochastic approximation methods; the latter section is the core of the paper and contains our main results. In Sect. 6 we illustrate the efficiency of the method on several simulation-based examples and Appendix contains some proofs.

2 Notation

2.1 Matrix notation

All vectors and matrices are typeset in boldface. Vectors are *column vectors* unless precised differently. The ith column vector of a matrix $\mathbf{A}$ is denoted by $\mathbf{A}_{|i}$. We denote by $\mathrm{Tr}(\mathbf{A})$ the trace of a matrix $\mathbf{A}$ and for any matrices $\mathbf{A}$ and $\mathbf{B}$ of dimensions $m \times n$ and $p \times q$, respectively, we denote their *direct sum* by

$$\mathbf{A} \oplus \mathbf{B} := \begin{pmatrix} \mathbf{A} & \mathbf{0}_{p \times q} \\ \mathbf{0}_{m \times n} & \mathbf{B} \end{pmatrix}, \tag{2.1}$$

which is such that $(\mathbf{A} \oplus \mathbf{B})^{\mathsf{T}} = \mathbf{A}^{\mathsf{T}} \oplus \mathbf{B}^{\mathsf{T}}$ and, for any two matrices $\mathbf{C}$ and $\mathbf{D}$ of compatible dimensions, $(\mathbf{A} \oplus \mathbf{B})(\mathbf{C} \oplus \mathbf{D})^{\mathsf{T}} = (\mathbf{A}\mathbf{C}^{\mathsf{T}}) \oplus (\mathbf{B}\mathbf{D}^{\mathsf{T}})$, where T denotes the transpose.

2.2 List of notation

The following quantities are defined in the corresponding equations.

Quantity	Equation
a^*	(3.3)
α_j	(4.5)
d_{KL}	(3.6)
$\boldsymbol{\eta}$	(4.2)
ϕ_k	(6.3)
G	(6.2)
g	(6.2)
$\mathbf{h}$	(5.15)
κ	(5.23)
l	(5.8)
l^*	(3.2)
$\tilde{\mu}$	(1.1)
$\bar{\mathbf{p}}$	(5.7)
$\tilde{\mathbf{p}}$	(5.21)
$\pi_{\boldsymbol{\theta}}$	(4.3)
$\bar{\pi}_{\boldsymbol{\theta}}()$	(5.1)
$\bar{\pi}_{\boldsymbol{\theta}}(\,|\,)$	(5.5)
$\oplus$	(2.1)
Q	(6.1)
q	(6.1)
$r_{\boldsymbol{\theta}}$	(4.1)
ρ	(4.1)
$\bar{\rho}()$	(4.6)
$\bar{\rho}(\,|\,)$	(5.2)
$\bar{\mathbf{s}}$	(5.6)
$\tilde{\mathbf{s}}$	(5.20)
$\bar{\mathbf{t}}$	(5.27)
$\tilde{\mathbf{t}}$	(5.28)
$\boldsymbol{\theta}$	(4.2)
$\bar{\boldsymbol{\theta}}$	(5.9)
$\bar{\mathbf{v}}$	(5.27)
$\tilde{\mathbf{v}}$	(5.28)
$w_{\boldsymbol{\theta}}$	(4.4)

3 Preliminaries

3.1 Auxiliary SMC methods

In order to precisely describe one iteration of the SMC algorithm, let μ be a target probability density function over a space Ξ and suppose that we have at hand a weighted sample $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^N$ such that $\sum_{i=1}^N \omega_i f(\mathbf{X}_i)/\Omega$, with $\Omega = \sum_{i=1}^N \omega_i$, estimates $\int f(\mathbf{x})\mu(\mathbf{x})\,d\mathbf{x}$ for any μ-integrable function f. As a notational convention, we use capitals to denote random variables and lower case for function arguments.

We now wish to transform (by moving the particles and adjusting accordingly the importance weights) the sample $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^N$ into a new weighted particle sample approximating the probability density $\tilde{\mu}$ defined on $\tilde{\boldsymbol{\Xi}}$ through (1.1). Having access to $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^N$, an approximation of $\tilde{\mu}$ is naturally obtained by plugging the weighted empirical measure associated with this weighted sample into (1.1), yielding the mixture density

$$
\begin{aligned}
\tilde{\mu}(\tilde{\mathbf{x}}) &\approx \frac{\sum_{i=1}^N \omega_i l(\mathbf{X}_i, \tilde{\mathbf{x}})}{\sum_{j=1}^N \omega_j \int l(\mathbf{X}_j, \tilde{\mathbf{x}})\,d\tilde{\mathbf{x}}} \\
&= \sum_{i=1}^N \left(\frac{\omega_i a^*(\mathbf{X}_i)}{\sum_{j=1}^N \omega_j a^*(\mathbf{X}_j)} l^*(\mathbf{X}_i, \tilde{\mathbf{x}}) \right),
\end{aligned}
\tag{3.1}
$$

where we have introduced the normalised transition density

$$
l^*(\mathbf{x}, \tilde{\mathbf{x}}) := \frac{l(\mathbf{x}, \tilde{\mathbf{x}})}{\int l(\mathbf{x}, \tilde{\mathbf{x}})\,d\tilde{\mathbf{x}}}
\tag{3.2}
$$

and the partition function

$$
a^*(\mathbf{x}) := \int l(\mathbf{x}, \tilde{\mathbf{x}})\,d\tilde{\mathbf{x}}
\tag{3.3}
$$

of $l(\mathbf{x}, \cdot)$, and ideally an updated particle sample would be obtained by drawing new particles $\{\tilde{\mathbf{X}}_i\}_{i=1}^N$ independently from (3.1). There are however two problems with this approach: firstly, in general, the integral $\int l(\cdot, \tilde{\mathbf{x}})\,d\tilde{\mathbf{x}}$ lacks closed-form expression, ruling out direct computation of the mixture weights; secondly, even if the integral was known in closed form, the resulting algorithm would have an $\mathcal{O}(N^2)$ computational complexity due to the normalisation of the mixture weights. To cope with these issues, we will, as proposed by Pitt and Shephard (1999), aim instead at sampling from an *auxiliary target distribution* having density

$$
\bar{\mu}(i, \tilde{\mathbf{x}}) := \frac{\omega_i a^*(\mathbf{X}_i)}{\sum_{j=1}^N \omega_j a^*(\mathbf{X}_j)} l^*(\mathbf{X}_i, \tilde{\mathbf{x}}),
\tag{3.4}
$$

over the product space $\{1, \ldots, N\} \times \tilde{\boldsymbol{\Xi}}$ of indices and particle positions. To sample $\bar{\mu}$ we take an importance sampling approach consisting in drawing independent pairs $\{(I_i, \tilde{\mathbf{X}}_i)\}_{i=1}^N$ of indices and positions from the proposal distribution

$$
\pi(i, \tilde{\mathbf{x}}) := \frac{\omega_i a(\mathbf{X}_i)}{\sum_{j=1}^N \omega_j a(\mathbf{X}_j)} r(\mathbf{X}_i, \tilde{\mathbf{x}})
\tag{3.5}
$$

over the same extended space and assigning each draw $(I_i, \tilde{\mathbf{X}}_i)$ the importance weight $\tilde{\omega}_i := w(I_i, \tilde{\mathbf{X}}_i)$, where

$$
w(i, \tilde{\mathbf{x}}) := \frac{l(\mathbf{X}_i, \tilde{\mathbf{x}})}{a(\mathbf{X}_i) r(\mathbf{X}_i, \tilde{\mathbf{x}})} \propto \frac{\bar{\mu}(i, \tilde{\mathbf{x}})}{\pi(i, \tilde{\mathbf{x}})}.
$$

Here r and a are the proposal transition density resp. adjustment multiplier weight function mentioned in the introduction. As for all importance sampling, it is crucial that the target distribution $\bar{\mu}$ is absolutely continuous with respect to the proposal distribution π. We thus require that the adjustment multiplier weight function is positive and that the proposal transition density is such that for all $\mathbf{x} \in \boldsymbol{\Xi}$,

$$r(\mathbf{x}, \tilde{\mathbf{x}}) = 0 \quad \Rightarrow \quad l(\mathbf{x}, \tilde{\mathbf{x}}) = 0,$$

i.e. the support of $l(\mathbf{x}, \tilde{\mathbf{x}})$ in included in that of $r(\mathbf{x}, \tilde{\mathbf{x}})$. Finally, we discard the indices I_i and keep $\{(\tilde{\mathbf{X}}_i, \tilde{\omega}_i)\}_{i=1}^N$ as an approximation of $\tilde{\mu}$.

This scheme, which is traditionally referred to as the auxiliary particle filter (Pitt and Shephard 1999), encompasses the simpler framework of the bootstrap particle filter proposed originally by Gordon et al. (1993), which simply amounts to setting $a = \mathbb{1}_{\boldsymbol{\Xi}}$ (implying a gain of simplicity at the price of flexibility). Moreover, SMC schemes where resampling is performed only at random times can similarly be cast into the setting of the auxiliary particle filter by composing the kernels involved in several consecutive steps of the of recursion (1.1) (see e.g. Cornebise 2009, Chap. 4, for details). A theoretical analysis of bootstrap-type SMC methods with random resampling schedule is also given by Del Moral et al. (2012). Therefore, any methodology built for the auxiliary particle filter also applies to these simpler frameworks without modification.

3.2 A foundation of SMC adaptation

We now describe a foundation of SMC adaptation underpinning the coming developments. Recall that the KLD between two distributions μ and ν, defined on the same state space $\boldsymbol{\Xi}$, is defined as

$$d_{\mathrm{KL}}(\mu \| \nu) := \mathbb{E}_\mu \left[\log \frac{d\mu}{d\nu}(\mathbf{X}) \right], \tag{3.6}$$

provided that μ is absolutely continuous with respect to ν. Here $\mathbb{E}_\mu$ denotes the expectation under μ. Casting the KLD into the framework of SMC methods and replacing μ and ν by the auxiliary target distribution $\bar{\mu}$ and the proposal π, respectively, yields

$$d_{\mathrm{KL}}(\bar{\mu} \| \pi) = \mathbb{E}_{\bar{\mu}} \left[\log \frac{\bar{\mu}(I, \tilde{\mathbf{X}})}{\pi(I, \tilde{\mathbf{X}})} \right], \tag{3.7}$$

which decouples as

$$d_{\mathrm{KL}}(\bar{\mu} \| \pi) = \underbrace{\mathbb{E}_{\bar{\mu}} \left[\log \frac{l^*(\mathbf{X}_I, \tilde{\mathbf{X}})}{r(\mathbf{X}_I, \tilde{\mathbf{X}})} \right]}_{\text{Depends only on } r}$$

$$+ \underbrace{\mathbb{E}_{\bar{\mu}} \left[\log \frac{\omega_I a^*(\mathbf{X}_I) / \sum_{j=1}^N \omega_j a^*(\mathbf{X}_j)}{\omega_I a(\mathbf{X}_I) / \sum_{j=1}^N \omega_j a(\mathbf{X}_j)} \right]}_{\text{Depends only on } a}.$$

(3.8)

As clear from the definition (3.4), the measure $\bar{\mu}$ is random as it depends on the ancestor particles $\{\mathbf{X}_i\}_{i=1}^N$; the locations of the latter should thus be viewed as fixed within the brackets of the expectations in (3.7) and (3.8). The first term in (3.8) corresponds to the discrepancy induced by mutating the particles $\{\tilde{\mathbf{X}}_i\}_{i=1}^N$ using a suboptimal proposal kernel, and the second term corresponds to the discrepancy induced by sampling the ancestor indices $\{I_i\}_{i=1}^N$ according to suboptimal adjustment weights. Moreover,

$$d_{\mathrm{KL}}(\bar{\mu} \| \pi) = -\mathbb{E}_{\bar{\mu}} \left[\log r(\mathbf{X}_I, \tilde{\mathbf{X}}) \right]$$
$$- \mathbb{E}_{\bar{\mu}} \left[\log \frac{a(\mathbf{X}_I)}{\sum_{j=1}^N \omega_j a(\mathbf{X}_j)} \right] + c,$$

where the additive term c involves the optimal quantities a^* and l^* only and is irrelevant for the optimisation problem. Equality up to a constant will in the following be denoted by $\equiv$. *Restricting ourselves to adaptation of the proposal kernel*, we obtain the simple expression

$$d_{\mathrm{KL}}(\bar{\mu} \| \pi) \equiv -\mathbb{E}_{\bar{\mu}} \left[\log r(\mathbf{X}_I, \tilde{\mathbf{X}}) \right]. \tag{3.9}$$

In the next section we present a method for minimising the KLD (3.9) over a family $\{\pi_{\boldsymbol{\theta}}; \boldsymbol{\theta} \in \Theta\}$ of instrumental distributions where the proposal transition density of each member belongs to parameterised family of mixtures of experts. Formally, we will solve

$$\boldsymbol{\theta}^* := \underset{\boldsymbol{\theta} \in \Theta}{\arg\min} \, d_{\mathrm{KL}}(\bar{\mu} \| \pi_{\boldsymbol{\theta}}) \tag{3.10}$$

and take $\pi = \pi_{\boldsymbol{\theta}^*}$. Although the right hand side of (3.9) is most often not directly computable, such a quantity can be approximated *on-the-fly* using the weighted sample already generated; the resulting algorithms (Algorithms 1 and 2; see Sect. 5) closely resembles the *cross-entropy method* (see Rubinstein and Kroese 2004).

4 Mixture of experts

In this contribution, we consider $\boldsymbol{\Xi} = \mathbb{R}^p$ and $\tilde{\boldsymbol{\Xi}} = \mathbb{R}^{\tilde{p}}$ and let the proposal kernel have density (with respect to Lebesgue measure)

$$r_{\boldsymbol{\theta}}(\mathbf{x}, \tilde{\mathbf{x}}) := \sum_{j=1}^d \alpha_j(\mathbf{x}; \underline{\boldsymbol{\beta}}) \rho(\mathbf{x}, \tilde{\mathbf{x}}; \boldsymbol{\eta}_j), \tag{4.1}$$

where $\{\alpha_j\}_{j=1}^d$ are nonnegative *weighting functions* summing into unity and ρ is a Markov transition kernel from Ξ to $\tilde{\Xi}$. The weighting functions and the Markov transition density are parameterised by parameters $\underline{\beta}$ and η, respectively, and the jth stratum of the mixture is characterised by a certain choice η_j of the latter. The integer d is fixed. Denote

$$\underline{\eta} := \{\eta_1, \ldots, \eta_d\},$$
$$\theta := \{\underline{\beta}, \underline{\eta}\}. \tag{4.2}$$

Putting (4.1) into the auxiliary particle filter framework in Sect. 3.1, the associated proposal distribution is given by

$$\pi_\theta(i, \tilde{x}) = \frac{\omega_i a(X_i)}{\sum_{\ell=1}^N \omega_\ell a(X_\ell)} \sum_{j=1}^d \alpha_j(X_i, \underline{\beta}) \rho(X_i, \tilde{x}; \eta_j). \tag{4.3}$$

We then assign the importance weight $\tilde{\omega}_i = w_\theta(I_i, \tilde{X}_i)$ to each draw $(I_i, \tilde{X}_i)$ from π_θ, where

$$w_\theta(i, \tilde{x}) = \frac{l(X_i, \tilde{x})}{a(X_i) \sum_{j=1}^d \alpha_j(X_i, \underline{\beta}) \rho(X_i, \tilde{x}; \eta_j)}$$
$$\propto \frac{\bar{\mu}(i, \tilde{x})}{\pi_\theta(i, \tilde{x})}. \tag{4.4}$$

The set $\{\alpha_j\}_{j=1}^d$ of weight functions typically partitions the input space Ξ into sub-regions with smooth transitions by assigning a vector of mixture weights to each point of the input space. In each sub-region, one proposal kernel will eventually dominate the mixture, specialising the whole proposal r_θ on a per-region basis. As in Jordan and Xu (1995), we consider *logistic* weight functions, i.e. $\underline{\beta} = \{\beta_1, \ldots, \beta_{d-1}\}$ and

$$\alpha_j(x, \underline{\beta}) := \begin{cases} \dfrac{\exp(\beta_j^\mathsf{T} \bar{x})}{1 + \sum_{\ell=1}^{d-1} \exp(\beta_\ell^\mathsf{T} \bar{x})}, & j \neq d, \\ \dfrac{1}{1 + \sum_{\ell=1}^{d-1} \exp(\beta_\ell^\mathsf{T} \bar{x})}, & j = d, \end{cases} \tag{4.5}$$

where $\bar{x} := (x^\mathsf{T}\ 1)^\mathsf{T}$ and the β_j's are vectors in $\mathbb{R}^{p+1}$. It is sometimes of interest to resort to simpler mixtures whose weights do not depend on x by letting, for a set $\underline{\beta} = (\beta_j)_{j=1}^d$ nonnegative scalars summing into unity, $\alpha_j(x, \underline{\beta}) := \beta_j$, $1 \leq j \leq d$, independently of x and hence without partitioning the input space. This model for the transition kernel is then similar to the *switching regression model* (Quandt and Ramsey 1972).

The original mixture of experts proposed by Jordan and Jacobs (1994) was based on Gaussian strata ρ, which is indeed is the most straightforward choice. However the same algorithm can, at no additional cost, be extended to the broader class of densities of *integrated curved exponential form*, described in Assumption 1 below, which still allows

for the same trade-off between flexibility, ease of sampling, and convenient estimation.

Assumption 1 There exist a state space $\Upsilon \subseteq \mathbb{R}^u$ and a Markov transition kernel from Ξ to $\tilde{\Xi} \times \Upsilon$ having a density of form

$$\bar{\rho}(x, (\tilde{x}, u); \eta) := \gamma(u) h(x, \tilde{x}, u) \exp\big(-a(\eta) + \mathrm{Tr}(b(\eta)^\mathsf{T} s(x, \tilde{x}, u))\big), \tag{4.6}$$

where the functions γ, h, a and s, b are real-valued resp. $\mathbb{R}^{s \times s'}$-valued, such that

$$\rho(x, \tilde{x}; \eta) = \int_\Upsilon \bar{\rho}(x, (\tilde{x}, u); \eta) \, du. \tag{4.7}$$

In other words, $\rho(x, \tilde{x}; \eta)$ is the density of the marginal of a curved exponential distribution. In (4.6), s is a vector of sufficient statistics.

In the next section we present a method fitting efficiently the proposal π_θ to the target $\bar{\mu}$ by solving (3.10). We will illustrate the details for the specific case of multidimensional Gaussian distributions and multidimensional Student's t-distributions. It should however be kept in mind that the algorithm is valid for any member of the family of integrated curved exponential distributions as long as Assumption 2 below is fulfilled.

5 Main results and algorithms

In the following we discuss how the intricate optimisation problem (3.10) can be cast into the framework of *latent data problems*. Parameter estimation for mixtures of experts is most often carried through using the EM algorithm (see McLachlan and Krishnan 2008, Sect. 8.4.6), and also in this paper a recursive EM-type algorithm will be used for finding close to optimal mixture parameters. More precisely, we develop an algorithm that is closely related to the *online EM algorithm* for latent data models proposed by Cappé and Moulines (2009). From a practical point of view, a key feature of the adaptive SMC approach described in Sect. 3.2 is that it aims at decreasing the KLD rather than obtaining a fully converged parameter estimate: very few EM iterations are therefore needed for obtaining a significant computational gain. This will be illustrated further in the simulation study of Sect. 6.

5.1 Main assumptions and definitions

In order to describe clearly the algorithm, *we limit ourselves initially to the case of constant mixture weights*, i.e. we let $\alpha_j(x, \underline{\beta}) := \beta_j$, $1 \leq j \leq d$, for a set $\underline{\beta} = \{\beta_j\}_{j=1}^d$ of nonnegative scalars summing into unity. The more general choice (4.5) will be treated in Sect. 5.3.

We now augment the pair $(I, \tilde{\mathbf{X}})$ with the index J of the mixture component as well as the auxiliary, $\boldsymbol{\Upsilon}$-valued variable $\mathbf{U}$ of the curved exponential family. More specifically, we introduce an extended random variable $(I, J, \tilde{\mathbf{X}}, \mathbf{U})$ having mixed-type distribution with probability function

$$\bar{\pi}_{\boldsymbol{\theta}}(i, j, \tilde{\mathbf{x}}, \mathbf{u}) := \frac{\omega_i a(\mathbf{X}_i)}{\sum_{\ell=1}^{N} \omega_\ell a(\mathbf{X}_\ell)} \alpha_j(\mathbf{X}_i, \underline{\boldsymbol{\beta}})$$
$$\times \bar{\rho}\big(\mathbf{X}_i, (\tilde{\mathbf{x}}, \mathbf{u}); \boldsymbol{\eta}_j\big) \tag{5.1}$$

on the product space $\{1, \ldots, N\} \times \{1, \ldots, d\} \times \tilde{\boldsymbol{\Xi}} \times \boldsymbol{\Upsilon}$. It is easily checked that $\pi_{\boldsymbol{\theta}}$ is the marginal of $\bar{\pi}_{\boldsymbol{\theta}}$ in I and $\tilde{\mathbf{X}}$. The following assumption is critical for convenient estimation. Although surprising at first glance, it simply requires that the state space extension used for handling integrated curved exponential distributions can be easily reverted for the sufficient statistics: the expectation of the sufficient statistics under the conditional distribution of $\mathbf{U}$ given $\mathbf{X}_I = \mathbf{x}$ and $\tilde{\mathbf{X}} = \tilde{\mathbf{x}}$, having density

$$\bar{\rho}(\mathbf{u}|\mathbf{x}, \tilde{\mathbf{x}}; \boldsymbol{\eta}) := \frac{\bar{\rho}(\mathbf{x}, (\tilde{\mathbf{x}}, \mathbf{u}); \boldsymbol{\eta})}{\rho(\mathbf{x}, \tilde{\mathbf{x}}; \boldsymbol{\eta})}, \tag{5.2}$$

should be available in closed form. Since the sufficient statistics of exponential families often have a simple form, this assumption is most often satisfied; e.g. it is satisfied for the Student's t-distribution in Example 2.

Assumption 2 For all $\boldsymbol{\eta}$ and $(\mathbf{x}, \tilde{\mathbf{x}}) \in \boldsymbol{\Xi} \times \tilde{\boldsymbol{\Xi}}$, the expectation

$$\int \mathbf{s}(\mathbf{x}, \tilde{\mathbf{x}}, \mathbf{u})\bar{\rho}(\mathbf{u}|\mathbf{x}, \tilde{\mathbf{x}}; \boldsymbol{\eta})\, d\mathbf{u} \tag{5.3}$$

is available in closed form.

Under Assumption 2, let for $j \in \{1, \ldots, d\}$,

$$\bar{p}_j(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}})\big],$$
$$\bar{\mathbf{s}}_j(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}}\bigg[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}}) \int \mathbf{s}(\mathbf{X}_I, \tilde{\mathbf{X}}, \mathbf{u})\bar{\rho}(\mathbf{u}|\mathbf{X}_I, \tilde{\mathbf{X}}; \boldsymbol{\eta}_j)\, d\mathbf{u}\bigg], \tag{5.4}$$

where we have defined the *responsibilities*

$$\bar{\pi}_{\boldsymbol{\theta}}(j|i, \tilde{\mathbf{x}}) := \frac{\int \bar{\pi}_{\boldsymbol{\theta}}(i, j, \tilde{\mathbf{x}}, \mathbf{u})\, d\mathbf{u}}{\sum_{\ell=1}^{d} \int \bar{\pi}_{\boldsymbol{\theta}}(i, \ell, \tilde{\mathbf{x}}, \mathbf{u})\, d\mathbf{u}}$$
$$= \frac{\beta_j \rho(\mathbf{X}_i, \tilde{\mathbf{x}}; \boldsymbol{\eta}_j)}{\sum_{\ell=1}^{d} \beta_\ell \rho(\mathbf{X}_i, \tilde{\mathbf{x}}; \boldsymbol{\eta}_\ell)}. \tag{5.5}$$

Note that the quantities defined in (5.4) depend implicitly on the ancestor sample. In addition, we collect these statistics in the matrix

$$\bar{\mathbf{s}}(\boldsymbol{\theta}) := \bar{\mathbf{s}}_1(\boldsymbol{\theta}) \oplus \bar{\mathbf{s}}_2(\boldsymbol{\theta}) \oplus \cdots \oplus \bar{\mathbf{s}}_d(\boldsymbol{\theta}) \tag{5.6}$$

and the vector

$$\bar{\mathbf{p}}(\boldsymbol{\theta}) := \big(\bar{p}_1(\boldsymbol{\theta}), \bar{p}_2(\boldsymbol{\theta}), \ldots, \bar{p}_d(\boldsymbol{\theta})\big)^\mathsf{T}. \tag{5.7}$$

We impose the following additional assumption.

Assumption 3 There are subsets $\boldsymbol{\Pi} \subseteq \mathbb{R}_+^d$ and $\boldsymbol{\Sigma} \subseteq \mathbb{R}^{s \times s'}$ such that for all $\{\mathbf{s}_j\}_{j=1}^d \in \boldsymbol{\Sigma}^d$ and $\mathbf{p} \in \boldsymbol{\Pi}$, the mapping

$$\boldsymbol{\theta} \mapsto l(\mathbf{s}, \mathbf{p}; \boldsymbol{\theta})$$
$$:= -\big(a(\boldsymbol{\eta}_1), \ldots, a(\boldsymbol{\eta}_d)\big)\mathbf{p}$$
$$+ (\log \beta_1, \ldots, \log \beta_d)\mathbf{p} \tag{5.8}$$
$$+ \sum_{j=1}^{d} \mathrm{Tr}\big(\mathbf{b}(\boldsymbol{\eta}_j)^\mathsf{T}\mathbf{s}_j\big)$$

has a *unique* global maximum denoted by $\bar{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{p})$. In particular,

$$\bar{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{p}) := \arg\max_{\boldsymbol{\theta} \in \Theta} l(\mathbf{s}, \mathbf{p}; \boldsymbol{\theta}). \tag{5.9}$$

Remark 1 (Decoupling, optimal strata weights) It should be noted that due to the additive form of l, the optimisation problem (5.9) can be split into the two separate subproblems

1.

$$\begin{cases} \arg\max_{\underline{\boldsymbol{\beta}}} (\log \beta_1, \ldots, \log \beta_d)\mathbf{p} \\ \text{subject to } \sum_{j=1}^{d} \beta_j = 1 \end{cases}$$

2.

$$\arg\max_{\underline{\boldsymbol{\eta}}} \Bigg(-\big(a(\boldsymbol{\eta}_1), \ldots, a(\boldsymbol{\eta}_d)\big)\mathbf{p} + \sum_{j=1}^{d} \mathrm{Tr}\big(\mathbf{b}(\boldsymbol{\eta}_j)^\mathsf{T}\mathbf{s}_j\big)\Bigg),$$

corresponding to maximisation over mixture weights and strata parameters, respectively. In the framework considered so far, where the mixture weights are assumed to be constant, the first of these problems (1) has, for all $\mathbf{p} \in \mathbb{R}_+^d$ the solution

$$\bar{\beta}_j = \frac{p_j}{\sum_{\ell=1}^{d} p_\ell}$$

for all j.

In the following two examples we specify the solution to the second problem (2) in Remark 1 for the two—fundamental—special cases of *multivariate Gaussian distributions* and *multivariate Student's t-distributions*; it should however be kept in mind that the method is valid for any integrated curved exponential distribution satisfying Assumption 2.

Example 1 (Updating formulas for Gaussian strata) In a first example we let each stratum be the density of a linear Gaussian regression parameterised by $\boldsymbol{\eta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The original hierarchical mixture of experts was based on plain Gaussian distributions and a possibly deep hierarchy of experts, incurring a computational overhead possibly larger than affordable for an algorithm used for adaptation. Dropping the hierarchical approach is a possible way to reduce this overhead at the price of reduced flexibility. Here we compensate somewhat for this loss by using a linear regression within each expert. Thus, given $J = j$, the new particle location $\tilde{\mathbf{X}}$ has p'-dimensional Gaussian distribution with mean $\boldsymbol{\mu}_j \overline{\mathbf{X}}_I$, where the *regression matrix* $\boldsymbol{\mu}_j$ is of size $p' \times (p + 1)$, and symmetric, positive definite *covariance matrix* $\boldsymbol{\Sigma}_j$ of size $p' \times p'$. In this case, the sufficient statistics are $\mathbf{s}(\mathbf{x}, \tilde{\mathbf{x}}, \mathbf{u}) = (\tilde{\mathbf{x}}\tilde{\mathbf{x}}^\mathsf{T}) \oplus (\overline{\mathbf{x}}\overline{\mathbf{x}}^\mathsf{T}) \oplus (\tilde{\mathbf{x}}\overline{\mathbf{x}}^\mathsf{T})$, leading to the expected responsibilities

$$\bar{p}_j(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}})\big]$$

and the expected sufficient statistics $\bar{\mathbf{s}}_j(\boldsymbol{\theta}) := \bar{\mathbf{s}}_{j,1}(\boldsymbol{\theta}) \oplus \bar{\mathbf{s}}_{j,2}(\boldsymbol{\theta}) \oplus \bar{\mathbf{s}}_{j,3}(\boldsymbol{\theta})$ with

$$\bar{\mathbf{s}}_{j,1}(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}}) \tilde{\mathbf{X}}\tilde{\mathbf{X}}^\mathsf{T}\big],$$
$$\bar{\mathbf{s}}_{j,2}(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}}) \overline{\mathbf{X}}_I\overline{\mathbf{X}}_I^\mathsf{T}\big],$$
$$\bar{\mathbf{s}}_{j,3}(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}}) \tilde{\mathbf{X}}\overline{\mathbf{X}}_I^\mathsf{T}\big].$$

In addition, the functions a and $\mathbf{b}$ in (4.6) are in this case given by

$$a(\boldsymbol{\eta}) = \frac{1}{2} \log |\boldsymbol{\Sigma}|,$$
$$\mathbf{b}(\boldsymbol{\eta}) = \left(-\frac{1}{2}\boldsymbol{\Sigma}^{-1}\right) \oplus \left(-\frac{1}{2}\boldsymbol{\mu}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}^\mathsf{T}\right) \quad (5.10)$$
$$\oplus \left(-\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\right).$$

Consequently, the argument $\boldsymbol{\eta}_j$ that maximizes $l(\mathbf{s}, \mathbf{p}; \boldsymbol{\theta})$ for a given sufficient statistics $\mathbf{s}_j$ is given by

$$\bar{\boldsymbol{\mu}}_j(\mathbf{s}_j) := \mathbf{s}_{j,3}\mathbf{s}_{j,2}^{-1},$$
$$\bar{\boldsymbol{\Sigma}}_j(p_j, \mathbf{s}_j) := p_j^{-1}\big(\mathbf{s}_{j,1} - \mathbf{s}_{j,3}\mathbf{s}_{j,2}^{-1}\mathbf{s}_{j,3}^\mathsf{T}\big).$$

Remark 2 (Pooling the covariances) In practice, a robustified version of the algorithm above can be obtained by *pooling* the covariances (see e.g. Rayens and Greene 1991). This means that a common covariance matrix is used for all the components of the mixture, i.e. $\boldsymbol{\Sigma}_j = \boldsymbol{\Sigma}$ for all j. By doing this, the well-known problem of mixture models with strata components degenerating to Dirac masses is avoided. It is straightforward to enforce such a restriction in the optimisa-

tion procedure above, leading to

$$\bar{\boldsymbol{\Sigma}}_j(p_j, \mathbf{s}_j) := d^{-1} \sum_{\ell=1}^{d} (\mathbf{s}_{\ell,1} - \mathbf{s}_{\ell,3}\mathbf{s}_{\ell,2}^{-1}\mathbf{s}_{\ell,3}^\mathsf{T}) \quad (5.11)$$

for all j.

Example 2 (Updating formulas for Student's t-distribution-based strata) A common grip in importance sampling (see, for instance, Oh and Berger 1993) is to replace, in order to allow for more efficient exploration of the state space, Gaussian distributions by Student's t-distributions. Therefore, as a more robust alternative to the approach taken in Example 1, one may use instead a p'-dimensional Student's t-distribution with ν degrees of freedom, i.e.

$$\rho(\mathbf{x}, \tilde{\mathbf{x}}; \boldsymbol{\eta}_j) = t_{p'}(\tilde{\mathbf{x}}; \boldsymbol{\mu}_j\overline{\mathbf{x}}, \boldsymbol{\Sigma}_j, \nu). \quad (5.12)$$

Remark 3 (Fixing the degrees of freedom) The number ν of degrees of freedom of the Student's t-distributions is fixed, typically to $\nu \in \{3, 4\}$, beforehand and is common to all strata. A similar choice has been made by, among others, Peel and McLachlan (2000, Sect. 7) and Cappé et al. (2008) as it allows for closed form optimisation in the M-step.

The choice (5.12) can be cast into the framework of Sect. 4, with $\Upsilon = \mathbb{R}$, thanks to the *Gaussian-Gamma decomposition* of multivariate Student's t-distributions used by Liu and Rubin (1995, Sect. 2) and Peel and McLachlan (2000, Sect. 3):

$$t(\tilde{\mathbf{x}}; \boldsymbol{\mu}\overline{\mathbf{x}}, \boldsymbol{\Sigma}, \nu) = \int_0^\infty \mathcal{N}(\tilde{\mathbf{x}}; \boldsymbol{\mu}\overline{\mathbf{x}}, \boldsymbol{\Sigma}u^{-1})\gamma\left(u; \frac{\nu}{2}, \frac{\nu}{2}\right) du,$$

where $\gamma(u; a, b) := b^a u^{a-1} \exp(-bu)/\Gamma(a)$ is the density of the Gamma distribution with shape parameter a and scale parameter b. Hence, the multivariate Student's t-distribution is an integrated curved exponential distribution (4.6) with $\gamma(u) = \gamma(u; \nu/2, \nu/2)$, $h(\mathbf{x}, \tilde{\mathbf{x}}, u) = (2\pi)^{-p/2}$, sufficient statistics $\mathbf{s}(\mathbf{x}, \tilde{\mathbf{x}}, u) = (u\tilde{\mathbf{x}}\tilde{\mathbf{x}}^\mathsf{T}) \oplus (u\overline{\mathbf{x}}\overline{\mathbf{x}}^\mathsf{T}) \oplus (u\tilde{\mathbf{x}}\overline{\mathbf{x}}^\mathsf{T})$, and the same a and $\mathbf{b}$ as in the Gaussian case (5.10). Since the Gamma distribution is conjugate prior with respect to precision for the Gaussian distribution with known mean, the expectation η of U under $\bar{\rho}(u|\mathbf{x}, \tilde{\mathbf{x}}; \boldsymbol{\eta})$ can in this case be expressed as

$$\eta(\tilde{\mathbf{x}}, \boldsymbol{\mu}\overline{\mathbf{x}}, \boldsymbol{\Sigma}) := \int_0^\infty u\bar{\rho}(u|\mathbf{x}, \tilde{\mathbf{x}}; \boldsymbol{\eta}) du$$
$$= \frac{\nu + p'}{\nu + \delta(\tilde{\mathbf{x}}, \boldsymbol{\mu}\overline{\mathbf{x}}, \boldsymbol{\Sigma})}, \quad (5.13)$$

where $\delta(\tilde{\mathbf{x}}, \boldsymbol{\mu}\overline{\mathbf{x}}, \boldsymbol{\Sigma}) = (\tilde{\mathbf{x}} - \boldsymbol{\mu}\overline{\mathbf{x}})^\mathsf{T}\boldsymbol{\Sigma}^{-1}(\tilde{\mathbf{x}} - \boldsymbol{\mu}\overline{\mathbf{x}})$ is the *Mahalanobis distance* with covariance matrix $\boldsymbol{\Sigma}$. This leads to the expected responsibilities

$$\bar{p}_j(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}})\big]$$

and the expected sufficient statistics $\bar{\mathbf{s}}_j(\boldsymbol{\theta}) = \bar{\mathbf{s}}_{j,1}(\boldsymbol{\theta}) \oplus \bar{\mathbf{s}}_{j,2}(\boldsymbol{\theta}) \oplus \bar{\mathbf{s}}_{j,3}(\boldsymbol{\theta})$ with

$$
\begin{aligned}
\bar{\mathbf{s}}_{j,1}(\boldsymbol{\theta}) &:= \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}})\eta(\tilde{\mathbf{X}}, \boldsymbol{\mu}\overline{\mathbf{X}}, \boldsymbol{\Sigma})\tilde{\mathbf{X}}\tilde{\mathbf{X}}^{\mathsf{T}}\big], \\
\bar{\mathbf{s}}_{j,2}(\boldsymbol{\theta}) &:= \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}})\eta(\tilde{\mathbf{X}}, \boldsymbol{\mu}\overline{\mathbf{X}}, \boldsymbol{\Sigma})\overline{\mathbf{X}}_I\overline{\mathbf{X}}_I^{\mathsf{T}}\big], \\
\bar{\mathbf{s}}_{j,3}(\boldsymbol{\theta}) &:= \mathbb{E}_{\bar{\mu}}\big[\bar{\pi}_{\boldsymbol{\theta}}(j|I, \tilde{\mathbf{X}})\eta(\tilde{\mathbf{X}}, \boldsymbol{\mu}\overline{\mathbf{X}}, \boldsymbol{\Sigma})\tilde{\mathbf{X}}\overline{\mathbf{X}}_I^{\mathsf{T}}\big].
\end{aligned}
\tag{5.14}
$$

Since the functions a and $\mathbf{b}$ are unchanged from the Gaussian case, the same equations (5.11) can be applied for updating the parameter $\boldsymbol{\eta}$ based on the expected sufficient statistics (5.14). Moreover, covariance pooling can be achieved in exactly same way as in Remark 2.

5.2 Main algorithm

Under the assumptions above we define the *mean field*

$$
\mathbf{h}(\mathbf{s}, \mathbf{p}) := \bar{\mathbf{s}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{p})\big) \oplus \bar{\mathbf{p}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{p})\big) - \mathbf{s} \oplus \mathbf{p}.
\tag{5.15}
$$

We now have the following result (proved in Appendix), which serves as a basis for the method proposed in the following.

Proposition 1 *Under Assumptions* 1–3, *the following holds true. If* $(\mathbf{s}^*, \mathbf{p}^*)$ *is a root of the mean field* $\mathbf{h}$, *defined in* (5.15), *then* $\bar{\boldsymbol{\theta}}(\mathbf{s}^*, \mathbf{p}^*)$ *is a stationary point of* $\boldsymbol{\theta} \mapsto \nabla_{\boldsymbol{\theta}} d_{\mathrm{KL}}(\bar{\mu}\|\pi_{\boldsymbol{\theta}})$. *Conversely, if* $\boldsymbol{\theta}^*$ *is a stationary point of the latter mapping, then* $(\mathbf{s}^*, \mathbf{p}^*)$, *with*

$$
\begin{aligned}
\mathbf{s}^* &= \bar{\mathbf{s}}\big(\boldsymbol{\theta}^*\big), \\
\mathbf{p}^* &= \bar{\mathbf{p}}\big(\boldsymbol{\theta}^*\big),
\end{aligned}
\tag{5.16}
$$

is a root of $\mathbf{h}$.

From Proposition 1 it is clear that the optimal parameter $\boldsymbol{\theta}^*$ solving the optimisation problem (3.10) over the parameter space Θ can, under the assumptions stated above, be obtained by solving the equation $\mathbf{h}(\mathbf{s}, \mathbf{p}) = \mathbf{0}$ in the space $\boldsymbol{\Sigma}^d \times \boldsymbol{\Pi}$ of sufficient statistics. Nevertheless, solving the latter equation is obstructed by the fact that the expected sufficient statistics (5.4), and consequently the mean field itself, cannot be expressed on closed form (as we do not know the target distribution $\bar{\mu}$). Thus, following the online EM approach taken by Cappé and Moulines (2009), we aim at finding this root using stochastic approximation. More specifically, we apply the classical *Robbins-Monro procedure*

$$
\hat{\mathbf{s}}_{\ell+1} \oplus \hat{\mathbf{p}}_{\ell+1} = \hat{\mathbf{s}}_\ell \oplus \hat{\mathbf{p}}_\ell + \lambda_{\ell+1}\tilde{\mathbf{h}}(\hat{\mathbf{s}}_\ell, \hat{\mathbf{p}}_\ell),
\tag{5.17}
$$

where $\{\lambda_\ell\}_{\ell \geq 1}$ is a decreasing sequence of positive step sizes such that

$$
\sum_{\ell=1}^{\infty} \lambda_\ell = \infty, \qquad \sum_{\ell=1}^{\infty} \lambda_\ell^2 < \infty
\tag{5.18}
$$

and $\tilde{\mathbf{h}}(\hat{\mathbf{s}}_\ell, \hat{\mathbf{p}}_\ell)$ is a noisy observation of $\mathbf{h}(\hat{\mathbf{s}}_\ell, \hat{\mathbf{p}}_\ell)$. We form these noisy observations by means of importance sampling in the following way: at iteration $\ell + 1$ (i.e. when computing $\hat{\mathbf{s}}_{\ell+1}$ and $\hat{\mathbf{p}}_{\ell+1}$ given $\hat{\mathbf{s}}_\ell$ and $\hat{\mathbf{p}}_\ell$) we estimate

$$
\mathbf{h}(\hat{\mathbf{s}}_\ell, \hat{\mathbf{p}}_\ell) = \bar{\mathbf{s}}\big(\bar{\boldsymbol{\theta}}(\hat{\mathbf{s}}_\ell, \hat{\mathbf{p}}_\ell)\big) \oplus \bar{\mathbf{p}}\big(\bar{\boldsymbol{\theta}}(\hat{\mathbf{s}}_\ell, \hat{\mathbf{p}}_\ell)\big) - \hat{\mathbf{s}}_\ell \oplus \hat{\mathbf{p}}_\ell
$$

by, first, setting $\boldsymbol{\theta}_\ell := \bar{\boldsymbol{\theta}}(\hat{\mathbf{s}}_\ell, \hat{\mathbf{p}}_\ell)$, second, estimating $\bar{\mathbf{s}}(\boldsymbol{\theta}_\ell)$ and $\bar{\mathbf{p}}(\boldsymbol{\theta}_\ell)$ by drawing particles and indices $\{(I_i^\ell, \tilde{\mathbf{X}}_i^\ell)\}_{i=1}^{N_\ell}$ from $\pi_{\boldsymbol{\theta}_\ell}$ (the currently best fitted proposal distribution), computing the associated weights $\{\tilde{\omega}_i^\ell\}_{i=1}^{N_\ell}$, and forming the importance sampling estimates

$$
\begin{aligned}
\tilde{p}_j(\boldsymbol{\theta}_\ell) &:= \sum_{i=1}^{N_\ell} \tilde{\omega}_i^\ell \bar{\pi}_{\boldsymbol{\theta}_\ell}\big(j|I_i^\ell, \tilde{\mathbf{X}}_i^\ell\big), \\
\tilde{\mathbf{s}}_j(\boldsymbol{\theta}_\ell) &:= \sum_{i=1}^{N_\ell} \tilde{\omega}_i^\ell \bar{\pi}_{\boldsymbol{\theta}_\ell}\big(j|I_i^\ell, \tilde{\mathbf{X}}_i^\ell\big) \\
&\quad \times \int \mathbf{s}\big(\mathbf{X}_{I_i^\ell}, \tilde{\mathbf{X}}_i^\ell, \mathbf{u}\big)\bar{\rho}\big(\mathbf{u}|\mathbf{X}_{I_i^\ell}, \tilde{\mathbf{X}}_i^\ell; \boldsymbol{\eta}_j^\ell\big)\,d\mathbf{u}.
\end{aligned}
\tag{5.19}
$$

Note that we let the Monte Carlo sample size N_ℓ, which can be relatively small compared to the sample size N of the particle filter, increase with the iteration index ℓ. As before, we collect the statistics defined in (5.19) in a matrix

$$
\tilde{\mathbf{s}}(\boldsymbol{\theta}_\ell) := \big(\tilde{\mathbf{s}}_1(\boldsymbol{\theta}_\ell) \quad \tilde{\mathbf{s}}_2(\boldsymbol{\theta}_\ell) \quad \cdots \quad \tilde{\mathbf{s}}_d(\boldsymbol{\theta}_\ell)\big)
\tag{5.20}
$$

and a vector

$$
\tilde{\mathbf{p}}(\boldsymbol{\theta}_\ell) := \big(\tilde{p}_1(\boldsymbol{\theta}_\ell), \tilde{p}_2(\boldsymbol{\theta}_\ell), \ldots, \tilde{p}_d(\boldsymbol{\theta}_\ell)\big)^{\mathsf{T}}.
\tag{5.21}
$$

Note that the importance sampling estimates defined in (5.19) lack normalisation $\sum_{i=1}^{N_\ell} \tilde{\omega}_i^\ell$. The role of the latter is to estimate the normalising constant $\iint \mu(\mathbf{x})l(\mathbf{x}, \tilde{\mathbf{x}})\,d\mathbf{x}\,d\tilde{\mathbf{x}}$ of the target distribution defined in (1.1), as it holds that

$$
\frac{1}{N}\sum_{i=1}^{N} \tilde{\omega}_i \xrightarrow{\mathbb{P}} \iint \mu(\mathbf{x})l(\mathbf{x}, \tilde{\mathbf{x}})\,d\mathbf{x}\,d\tilde{\mathbf{x}}
$$

as $N \to \infty$ (see e.g. Douc et al. 2008). Here $\xrightarrow{\mathbb{P}}$ denotes convergence in probability. However, as N_ℓ is supposed to be considerably smaller that N, which is necessary for obtaining a computationally efficient algorithm, this estimator suffers from large variance in our case. Thus, in order to robustify the estimator we combine the Robbins-Monro procedure (5.17) with a similar procedure

$$
\hat{c}_{\ell+1} = (1 - \lambda_{\ell+1})\hat{c}_\ell + \frac{\lambda_{\ell+1}}{N_\ell}\sum_{i=1}^{N_\ell} \tilde{\omega}_i^\ell
\tag{5.22}
$$

for the normalising constant. The recursion (5.22) is typically initialised with

$$\hat{c}_0 = \frac{1}{N_0} \sum_{i=1}^{N_0} \tilde{\omega}_i^0.$$

Obviously, (5.22) does not guarantee that the normalised weights $\{\tilde{\omega}_i^\ell / \hat{c}_{\ell+1} N_\ell\}_{i=1}^{N_\ell}$ sum to unity, and our approach thus leads to approximation of a probability measure with a finite measure. However, this does not impede the convergence of the stochastic approximation algorithm. Needless to say, this does not affect the properties of the particle approximation as the weights of the final particle sample obtained using the adapted kernel are renormalised in the usual manner and thus sum to unity. The final weighted empirical measure associated with the particle cloud is therefore still a probability distribution.

So, the Robbins-Monro recursion (5.17) is executed for, say, L iterations *before every updating step of the SMC algorithm* or only at time steps decided by the user, and the final parameter fit $\boldsymbol{\theta}_L$ defines the instrumental distribution $\pi_{\boldsymbol{\theta}*}$ used for propagating the particles. The following assumption is needed to formulate a practically useful algorithm.

Assumption 4 There is a set $\boldsymbol{\Gamma} \subseteq \mathbb{R}_+$ such that for all $N \geq 1$, $\{\tilde{\mathbf{X}}_i\}_{i=1}^N \in \tilde{\boldsymbol{\Xi}}^N$, $\{I_i\}_{i=1}^N \in \{1, \ldots N\}^N$, $\boldsymbol{\theta} \in \Theta$, $c \in \boldsymbol{\Gamma}$, $\mathbf{s} \in \boldsymbol{\Sigma}^d$, $\mathbf{p} \in \boldsymbol{\Pi}$, and $\lambda \in (0, 1)$, it holds that

$$(1-\lambda)c + \frac{\lambda}{N} \sum_{i=1}^N \tilde{\omega}_i^\ell \in \boldsymbol{\Gamma},$$

$$(1-\lambda)\mathbf{s} + \frac{\lambda}{cN} \tilde{\mathbf{s}}(\boldsymbol{\theta}) \in \boldsymbol{\Sigma}^d,$$

$$(1-\lambda)\mathbf{p} + \frac{\lambda}{cN} \tilde{\mathbf{p}}(\boldsymbol{\theta}) \in \boldsymbol{\Pi},$$

where the quantities $\tilde{\mathbf{s}}(\boldsymbol{\theta})$ and $\tilde{\mathbf{p}}(\boldsymbol{\theta})$ are defined through (5.20) and (5.21), respectively.

Under Assumption 4 our main algorithm, combining the two procedures (5.17) and (5.22), is well defined and goes as follows.

5.3 Algorithm formulation for logistic weights

We now extend the case of constant mixture weights to the considerably more complicated case of logistic weights (4.5). In this case the gradient $\nabla_{\boldsymbol{\theta}} d_{\mathrm{KL}}(\bar{\mu} \| \pi_{\boldsymbol{\theta}})$, the crucial quantity in the proof of Proposition 1 (given in Appendix), is given by

---

Algorithm 1

Require: $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^N$, $\boldsymbol{\theta}_0, \hat{c}_0, \hat{\mathbf{s}}_0, \hat{\mathbf{p}}_0$
1: **for** $\ell = 0 \to L - 1$ **do**
2:     **for** $i = 1 \to N_\ell$ **do**
3:         draw $(I_i^\ell, \tilde{\mathbf{X}}_i^\ell) \sim \pi_{\boldsymbol{\theta}_\ell}$
4:         set $\tilde{\omega}_i^\ell \leftarrow w_{\boldsymbol{\theta}_\ell}(I_i^\ell, \tilde{\mathbf{X}}_i^\ell)$
5:     **end for**
6:     compute $\tilde{\mathbf{s}}(\boldsymbol{\theta}_\ell)$ and $\tilde{\mathbf{p}}(\boldsymbol{\theta}_\ell)$ as in (5.19)
7:     set

$$\hat{c}_{\ell+1} \leftarrow (1 - \lambda_{\ell+1})\hat{c}_\ell + \frac{\lambda_{\ell+1}}{N_\ell} \sum_{i=1}^{N_\ell} \tilde{\omega}_i^\ell$$

8:     set

$$\hat{\mathbf{s}}_{\ell+1} \oplus \hat{\mathbf{p}}_{\ell+1} \leftarrow (1 - \lambda_{\ell+1})\hat{\mathbf{s}}_\ell \oplus \hat{\mathbf{p}}_\ell$$
$$+ \frac{\lambda_{\ell+1}}{\hat{c}_{\ell+1} N_\ell} \tilde{\mathbf{s}}(\boldsymbol{\theta}_\ell) \oplus \tilde{\mathbf{p}}(\boldsymbol{\theta}_\ell)$$

9:     set $\boldsymbol{\theta}_{\ell+1} \leftarrow \bar{\boldsymbol{\theta}}(\hat{\mathbf{s}}_{\ell+1}, \hat{\mathbf{p}}_{\ell+1})$
10: **end for**

$$\nabla_{\boldsymbol{\theta}} d_{\mathrm{KL}}(\bar{\mu} \| \pi_{\boldsymbol{\theta}})$$
$$= \mathbb{E}_{\bar{\mu}} \left[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(I, \tilde{\mathbf{X}}) \right]$$
$$= -\left(\nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_1) \cdots \nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_d) \right) \bar{\mathbf{p}}(\boldsymbol{\theta})$$
$$+ \sum_{j=1}^d \sum_{m=1}^{s'} \nabla_{\boldsymbol{\theta}} \mathbf{b}_{|m}(\boldsymbol{\eta}_j)^\mathsf{T} \bar{\mathbf{s}}_{j|m}(\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} \kappa(\boldsymbol{\theta}),$$

where

$$\kappa(\boldsymbol{\theta}) := \mathbb{E}_{\bar{\mu}} \left[\sum_{j=1}^d \log \alpha_j(\mathbf{X}_I, \underline{\boldsymbol{\beta}}) \bar{\pi}_{\boldsymbol{\theta}}(j | I, \tilde{\mathbf{X}}) \right], \quad (5.23)$$

with responsibilities $\bar{\pi}_{\boldsymbol{\theta}}(j | i, \tilde{\mathbf{x}})$ given by, in this case,

$$\bar{\pi}_{\boldsymbol{\theta}}(j | i, \tilde{\mathbf{x}}) := \frac{\alpha_j(\mathbf{X}_i, \underline{\boldsymbol{\beta}}) \rho(\mathbf{X}_i, \tilde{\mathbf{x}}; \boldsymbol{\eta}_j)}{\sum_{\ell=1}^d \alpha_\ell(\mathbf{X}_i, \underline{\boldsymbol{\beta}}) \rho(\mathbf{X}_i, \tilde{\mathbf{x}}; \boldsymbol{\eta}_\ell)}$$

and the vectors $\bar{\mathbf{p}}(\boldsymbol{\theta})$ and $\bar{\mathbf{s}}(\boldsymbol{\theta})$ being defined as in (5.7). However, since we are no longer within the framework of exponential families, we will not be able to find a closed-form zero of $\nabla_{\boldsymbol{\theta}} \kappa(\boldsymbol{\theta})$ (e.g. a closed-form global maximum of $\kappa(\boldsymbol{\theta})$) when $\bar{\mathbf{p}}(\boldsymbol{\theta})$ and $\bar{\mathbf{s}}(\boldsymbol{\theta})$ are replaced by stochastic approximation iterates $\hat{\mathbf{p}}_\ell$ and $\hat{\mathbf{s}}_\ell$, respectively. On the other hand, for all $\boldsymbol{\theta}_\ell \in \Theta$, the mapping

$$\underline{\boldsymbol{\beta}} \mapsto \bar{\kappa}(\underline{\boldsymbol{\beta}}; \boldsymbol{\theta}_\ell)$$
$$:= \mathbb{E}_{\bar{\mu}} \left[\sum_{j=1}^d \log \alpha_j(\mathbf{X}_I, \underline{\boldsymbol{\beta}}) \bar{\pi}_{\boldsymbol{\theta}_\ell}(j | I, \tilde{\mathbf{X}}) \right]$$

is *concave* (this stems directly from the multinomial logistic regression by computing conditional expectations) and can thus be approximated well by a second degree polynomial (in $\underline{\beta}$). More specifically, consider the gradient $\nabla_{\underline{\beta}}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell)$, whose jth block ($j \in \{1,\ldots,d-1\}$) is given by

$$
\nabla_{\boldsymbol{\beta}_j}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell)
$$
$$
= \mathbb{E}_{\bar{\mu}}\big[\big(\bar{\pi}_{\boldsymbol{\theta}_\ell}(j|I,\tilde{\mathbf{X}}) - \alpha_j(\mathbf{X}_I,\underline{\beta})\big)\overline{\mathbf{X}}_I\big], \tag{5.24}
$$

and the Hessian $\nabla_{\underline{\beta}}\nabla_{\underline{\beta}}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell)$, having the matrix

$$
\nabla_{\boldsymbol{\beta}_j}\nabla_{\boldsymbol{\beta}_{j'}}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell)
$$
$$
= \mathbb{E}_{\bar{\mu}}\bigg[\alpha_j(\mathbf{X}_I,\underline{\beta})\bigg(\alpha_{j'}(\mathbf{X}_I,\underline{\beta})
$$
$$
-\frac{\mathbb{1}_{j=j'}}{1+\sum_{m=1}^{d-1}\exp(\boldsymbol{\beta}_m^\intercal\overline{\mathbf{X}}_I)}\bigg)\overline{\mathbf{X}}_I\overline{\mathbf{X}}_I^\intercal\bigg] \tag{5.25}
$$

as block $(j,j') \in \{1,\ldots,d-1\}^2$. Note that the Hessian above does not depend on $\boldsymbol{\theta}_\ell$.

Using these definitions, the function $\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell)$ can be approximated by means of a second order Taylor expansion around the previous parameter estimate $\underline{\beta}^\ell$ according to

$$
\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell) \approx \bar{\kappa}(\underline{\beta}^\ell;\boldsymbol{\theta}_\ell) + (\underline{\beta}-\underline{\beta}^\ell)\cdot\nabla_{\underline{\beta}}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell)\big|_{\underline{\beta}=\underline{\beta}^\ell}
$$
$$
+\frac{1}{2}(\underline{\beta}-\underline{\beta}^\ell)\cdot\big[(\underline{\beta}-\underline{\beta}^\ell)
$$
$$
\cdot\nabla_{\underline{\beta}}\nabla_{\underline{\beta}}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}_\ell)\big|_{\underline{\beta}=\underline{\beta}^\ell}\big]. \tag{5.26}
$$

In the light of (5.26), we approximate the gradient and Hessian quantities (which again lack closed-form expressions due to the expectation over $\bar{\mu}$) using stochastic approximation. For this purpose, we set

$$
\bar{\mathbf{t}}_j(\boldsymbol{\theta}) := \nabla_{\boldsymbol{\beta}_j}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}),
$$
$$
\bar{\mathbf{v}}_{j,j'}(\boldsymbol{\theta}) := \nabla_{\boldsymbol{\beta}_j}\nabla_{\boldsymbol{\beta}_{j'}}\bar{\kappa}(\underline{\beta};\boldsymbol{\theta}),
$$

define

$$
\bar{\mathbf{t}}(\boldsymbol{\theta}) := \big(\bar{\mathbf{t}}_1(\boldsymbol{\theta}) \quad \cdots \quad \bar{\mathbf{t}}_{d-1}(\boldsymbol{\theta})\big),
$$
$$
\bar{\mathbf{v}}(\boldsymbol{\theta}) := \begin{pmatrix} \bar{\mathbf{v}}_{1,1}(\boldsymbol{\theta}) & \cdots & \bar{\mathbf{v}}_{1,d-1}(\boldsymbol{\theta}) \\ \vdots & \ddots & \vdots \\ \bar{\mathbf{v}}_{d-1,1}(\boldsymbol{\theta}) & \cdots & \bar{\mathbf{v}}_{d-1,d-1}(\boldsymbol{\theta}) \end{pmatrix}, \tag{5.27}
$$

and find, using again the Robbins-Monro procedure, a root to the extended mean field

$$
\mathbf{h}(\mathbf{s},\mathbf{p},\mathbf{v},\mathbf{t}) := \bar{\mathbf{s}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s},\mathbf{p},\mathbf{v},\mathbf{t})\big) \oplus \bar{\mathbf{p}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s},\mathbf{p},\mathbf{v},\mathbf{t})\big)
$$
$$
\oplus \bar{\mathbf{t}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s},\mathbf{p},\mathbf{v},\mathbf{t})\big) \oplus \bar{\mathbf{v}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s},\mathbf{p},\mathbf{v},\mathbf{t})\big)
$$
$$
-\mathbf{s}\oplus\mathbf{p}\oplus\mathbf{v}\oplus\mathbf{t}.
$$

In this case, the mapping $\bar{\boldsymbol{\theta}}$ updates the strata parameters in analogy with the case of constant mixture weights (Sect. 5.1) while the $\underline{\beta}$ parameter gets updated according to the Newton-Raphson-type formula

$$
\underline{\beta}^{\ell+1} := \underline{\beta}^\ell - \mathbf{v}^{-1}\mathbf{t}.
$$

The full procedure is summarised below, $\tilde{\mathbf{p}}(\boldsymbol{\theta}_\ell)$ as in (5.21) and

$$
\tilde{\mathbf{t}}_j(\boldsymbol{\theta}_\ell) := \sum_{i=1}^{N_\ell}\tilde{\omega}_i^\ell\bar{\pi}_{\boldsymbol{\theta}_\ell}\big(j|I_i^\ell,\tilde{\mathbf{X}}_i^\ell\big)
$$
$$
\times \int \bar{\mathbf{t}}\big(\mathbf{X}_{I_i^\ell},\tilde{\mathbf{X}}_i^\ell,\mathbf{u}\big)\bar{\rho}\big(\mathbf{u}|\mathbf{X}_{I_i^\ell},\tilde{\mathbf{X}}_i^\ell;\boldsymbol{\eta}_j^\ell\big)\,d\mathbf{u},
$$
$$
\tilde{\mathbf{v}}_j(\boldsymbol{\theta}_\ell) := \sum_{i=1}^{N_\ell}\tilde{\omega}_i^\ell\bar{\pi}_{\boldsymbol{\theta}_\ell}\big(j|I_i^\ell,\tilde{\mathbf{X}}_i^\ell\big) \tag{5.28}
$$
$$
\times \int \bar{\mathbf{v}}\big(\mathbf{X}_{I_i^\ell},\tilde{\mathbf{X}}_i^\ell,\mathbf{u}\big)\bar{\rho}\big(\mathbf{u}|\mathbf{X}_{I_i^\ell},\tilde{\mathbf{X}}_i^\ell;\boldsymbol{\eta}_j^\ell\big)\,d\mathbf{u}.
$$

Algorithm 2

Require: $\{(\mathbf{X}_i,\omega_i)\}_{i=1}^N, \boldsymbol{\theta}_0, \hat{c}_0, \hat{\mathbf{s}}_0, \hat{\mathbf{p}}_0, \hat{\mathbf{t}}_0, \hat{\mathbf{v}}_0$

1: **for** $\ell = 0 \to L-1$ **do**
2:      **for** $i = 1 \to N_\ell$ **do**
3:          draw $(I_i^\ell,\tilde{\mathbf{X}}_i^\ell) \sim \pi_{\boldsymbol{\theta}_\ell}$
4:          set $\tilde{\omega}_i^\ell \leftarrow w_{\boldsymbol{\theta}_\ell}(I_i^\ell,\tilde{\mathbf{X}}_i^\ell)$
5:      **end for**
6:      compute $\tilde{\mathbf{s}}(\boldsymbol{\theta}_\ell)$ and $\tilde{\mathbf{p}}(\boldsymbol{\theta}_\ell)$ through (5.19)
7:      compute $\tilde{\mathbf{t}}(\boldsymbol{\theta}_\ell)$ and $\tilde{\mathbf{v}}(\boldsymbol{\theta}_\ell)$ through (5.28)
8:      set

$$
\hat{c}_{\ell+1} \leftarrow (1-\lambda_{\ell+1})\hat{c}_\ell + \frac{\lambda_{\ell+1}}{N_\ell}\sum_{i=1}^{N_\ell}\tilde{\omega}_i^\ell
$$

9:      set

$$
\hat{\mathbf{s}}_{\ell+1}\oplus\hat{\mathbf{p}}_{\ell+1}\oplus\hat{\mathbf{t}}_{\ell+1}\oplus\hat{\mathbf{v}}_{\ell+1}
$$
$$
\leftarrow (1-\lambda_{\ell+1})\hat{\mathbf{s}}_\ell\oplus\hat{\mathbf{p}}_\ell\oplus\hat{\mathbf{t}}_\ell\oplus\hat{\mathbf{v}}_\ell
$$
$$
+\frac{\lambda_{\ell+1}}{\hat{c}_{\ell+1}N_\ell}\tilde{\mathbf{s}}(\boldsymbol{\theta}_\ell)\oplus\tilde{\mathbf{p}}(\boldsymbol{\theta}_\ell)\oplus\tilde{\mathbf{t}}(\boldsymbol{\theta}_\ell)\oplus\tilde{\mathbf{v}}(\boldsymbol{\theta}_\ell)
$$

10:     set $\boldsymbol{\theta}_{\ell+1} \leftarrow \bar{\boldsymbol{\theta}}(\hat{\mathbf{s}}_{\ell+1},\hat{\mathbf{p}}_{\ell+1})$ ▷ Here $\underline{\beta}$ is updated according to $\underline{\beta}^{\ell+1} := \underline{\beta}^\ell - \hat{\mathbf{v}}_{\ell+1}^{-1}\hat{\mathbf{t}}_{\ell+1}$.
11: **end for**

5.4 Some practical guidelines to choose the algorithm's parameters

The algorithm outlined above involves parameters, such as the number d of mixture components and the Monte Carlo

sample sizes $\{N_\ell\}_{\ell=0}^{L-1}$, that need to be set. Fortunately, these parameters are easy to tune. Moreover, we may expect a significant improvement of the SMC proposal distribution also with a suboptimal choice of the same. Without constituting a definite answer on how to tune the algorithmic parameters, the following guidelines emerge from the simulations in Sect. 6.

– Concerning the number d of experts, a too small number of strata averts good approximation of strongly nonlinear kernels while a too large d leads to computational overhead. For instance, in the simulations in Sect. 6 we use up to $d = 8$ strata to approximate well a kernel with a ring-shaped, 2-dimensional support.

– The initial values of the mixture parameters could be chosen based on prior knowledge of the model by, e.g., fitting, in the case of optimal filtering with informative or non-informative observations, the local likelihood or the prior kernel, respectively. In our simulations, we initialise the algorithm with uniform logistic weights (i.e. with $\underline{\beta}$ being a null matrix), regressions being null except for the intercept (implying independence of the ancestors), and covariances making the strata widely overspread on the support of the target.

– For simplicity, the importance sampling sizes can be taken constant $N_0 = N_1 = \cdots = N_{L-1}$ over all iterations. This is well in line with the theory of stochastic approximation (see e.g. Duflo 1997; Benveniste et al. 1990), which in general guarantees, if the step size sequence is chosen correctly, convergence of the algorithm as long as the variance of the observation noise is not increasing with the iteration index. Moreover, the importance sampling size can be chosen relatively to the computational budget, by using, say, a proportion $\alpha = 20\ \%$ or $\alpha = 50\ \%$ of the total number N of particles that would be used for a plain SMC algorithm for the adaptation step. More specifically, this is achieved by using $N_\ell = \alpha N / L$ samples per iteration in the adaptation step, and propagating $(1-\alpha)N$ particles through the SMC updating step (based on the adapted proposal). In order assure reasonably fast convergence, the proportion α should be large enough to assure a decent sampling size N_ℓ, typically not less than a hundred particles. It can be helpful to let the sample size N_0 be twice or three times larger than the size used at a typical iteration, to recover from a potentially bad initial choice.

– The number L of iterations can typically be very small compared to typical stochastic approximation EM runs: as mentioned at the beginning of Sect. 5, the main gain in KLD typically occurs in the first dozen iterations.

– The least precise guideline concerns the step sizes $\{\lambda_\ell\}_{\ell=0}^{L-1}$. Since we not search an exact optimum but rather aim at fast convergence towards the region of interest, we use a slowly decreasing step size. A rate matching this could be, say, letting $\lambda_\ell = \ell^{-0.6}$ (which was used by e.g. Cappé et al. 2005, Sect. 11.1.5, within the framework of stochastic approximation EM methods), which satisfies (5.18). However, in our simulations we use, for simplicity, a constant step size. This is well motivated by the fact that the adaptation algorithm is run for only a few iterations.

6 Applications to nonlinear SSMs

As mentioned, SMC methods can be successfully applied to optimal filtering in SSMs. An SSM is a bivariate process $\{(\mathbf{X}_k, \mathbf{Y}_k)\}_{k \geq 0}$, where $\mathbf{X} := \{\mathbf{X}_k\}_{k \geq 0}$ is an unobserved Markov chain on some state space $\mathsf{X} \subseteq \mathbb{R}^x$ and $\{\mathbf{Y}_k\}_{k \geq 0}$ is an observation process taking values in some space $\mathsf{Y} \subseteq \mathbb{R}^y$. We denote by Q and π_0 the transition density (in the following sometimes referred to as the *prior kernel*) and initial distribution of $\mathbf{X}$, respectively. Conditionally on $\mathbf{X}$, the observations are assumed to be conditionally independent with the conditional distribution G of a particular $\mathbf{Y}_k$ depending on the corresponding $\mathbf{X}_k$ only. We will assume that Q and G have densities q and g, respectively, with respect to Lebesgue measure, i.e.

$$\mathbb{P}(\mathbf{X}_{k+1} \in \mathsf{A}|\mathbf{X}_k) = Q(\mathbf{X}_k, \mathsf{A}) = \int_\mathsf{A} q(\mathbf{X}_k, \mathbf{x})\, d\mathbf{x} \qquad (6.1)$$

and

$$\mathbb{P}(\mathbf{Y}_k \in \mathsf{B}|\mathbf{X}_k) = G(\mathbf{X}_k, \mathsf{B}) = \int_\mathsf{B} g(\mathbf{X}_k, \mathbf{y})\, d\mathbf{y}. \qquad (6.2)$$

For simplicity, we assume that the Markov chain $\mathbf{X}$ is time-homogeneous and that the distribution G does not depend on k; however, all developments that follow may be extended straightforwardly to the time-inhomogeneous case. The optimal filtering problem consists of computing, given a fixed record $\mathbf{Y}_{0:n} := (\mathbf{Y}_0, \ldots, \mathbf{Y}_n)$ of observations, the *filter distributions*

$$\phi_k(\mathsf{A}) := \mathbb{P}(\mathbf{X}_k \in \mathsf{A}|\mathbf{Y}_{0:k}) \qquad (6.3)$$

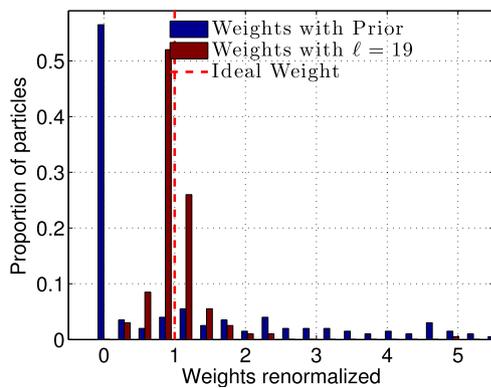for $k = 0, 1, \ldots, n$. As the sequence of filter distributions satisfies the nonlinear recursion

$$\phi_k(\mathbf{x}_{k+1})$$
$$= \frac{\int g(\mathbf{x}_{k+1}, \mathbf{Y}_{k+1}) q(\mathbf{x}_k, \mathbf{x}_{k+1}) \phi_k(\mathbf{x}_k)\, d\mathbf{x}_k}{\iint g(\mathbf{x}_{k+1}, \mathbf{Y}_{k+1}) q(\mathbf{x}_k, \mathbf{x}_{k+1}) \phi_k(\mathbf{x}_k)\, d\mathbf{x}_k\, d\mathbf{x}_{k+1}}, \quad (6.4)$$

the optimal filtering problem can be perfectly cast into the sequential importance sampling framework (1.1) with $\boldsymbol{\Xi} = \tilde{\boldsymbol{\Xi}} \equiv \mathsf{X}$, $\mu \equiv \phi_k$, $\tilde{\mu} \equiv \phi_{k+1}$, and $l(\mathbf{x}, \tilde{\mathbf{x}}) \equiv g(\tilde{\mathbf{x}}, \mathbf{Y}_{k+1}) q(\mathbf{x}, \tilde{\mathbf{x}})$. Consequently, a particle sample $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^N$ (here and in

the following we have omitted, for brevity, the time index from the particles and the associated weights) approximating ϕ_k can be transformed into a sample $\{(\tilde{\mathbf{X}}_i, \tilde{\omega}_i)\}_{i=1}^N$ approximating the filter ϕ_{k+1} at the subsequent time step by executing Algorithm 3. Here $\{a_i\}_{i=1}^N$ is a set of nonnegative adjustment multipliers and the proposal r is a Markov transition density. In the following examples we illustrate how the proposal r can be designed adaptively using Algorithm 2 described in the previous section. In Sects. 6.1 and 6.2 we adapt mixtures of Gaussian strata (following Example 1) only, while Sect. 6.3 provides also a comparison with Student's t-distributed strata.

6.1 Multivariate linear Gaussian model

We start by considering a simple multivariate linear Gaussian model. In this toy example, the optimal kernel l^* and the

---

Algorithm 3

Require: $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^N$
1: **for** $i = 1 \to N$ **do**
2:      draw $I_i \sim \{a_\ell \omega_\ell\}_{\ell=1}^N$
3:      draw $\tilde{\mathbf{X}}_i \sim r(\mathbf{X}_{I_i}, \cdot)$
4:      set $\tilde{\omega}_i \leftarrow \dfrac{g(\tilde{\mathbf{X}}_i, \mathbf{y}_{k+1}) q(\mathbf{X}_{I_i}, \tilde{\mathbf{X}}_i)}{a_{I_i} r(\mathbf{X}_{I_i}, \tilde{\mathbf{X}}_i)}$
5: **end for**
6: **return** $\{(\tilde{\mathbf{X}}_i, \tilde{\omega}_i)\}_{i=1}^N$

---

optimal adjustment weight function a^* (defined in (3.2) and (3.3), respectively) are available in closed form; here we use the term "optimal"—which is standard in the literature—as a proposal distribution π based on these quantities minimises the KLD (3.7) (which of course then vanishes as $\pi = \bar{\mu}$ in that case). This makes it possible to compare our algorithm to an exact reference. The optimal kernel does not belong to our family of mixture of experts. In the model under consideration:

(a) Histogram of the importance weights obtained using the prior kernel and the final adapted kernel in the linear Gaussian model. Weights are re-normalised and multiplied by the size of the sample. In the fully adapted case, all weights would be equal to unity
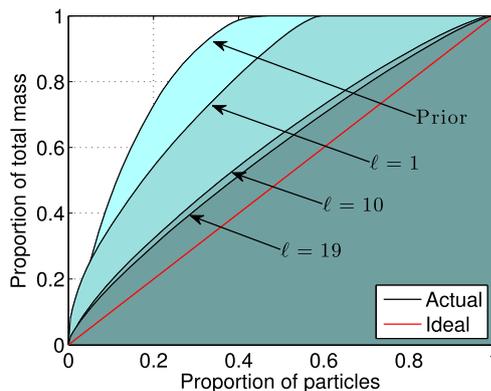
(b) *Curve of proportions*: proportion of particles, sorted by decreasing order of importance weight, against proportion of the total mass, for the prior kernel and for different numbers of iterations of the adaptation algorithm

Fig. 1 Evolution of the distribution of the importance weights before and after adaptation
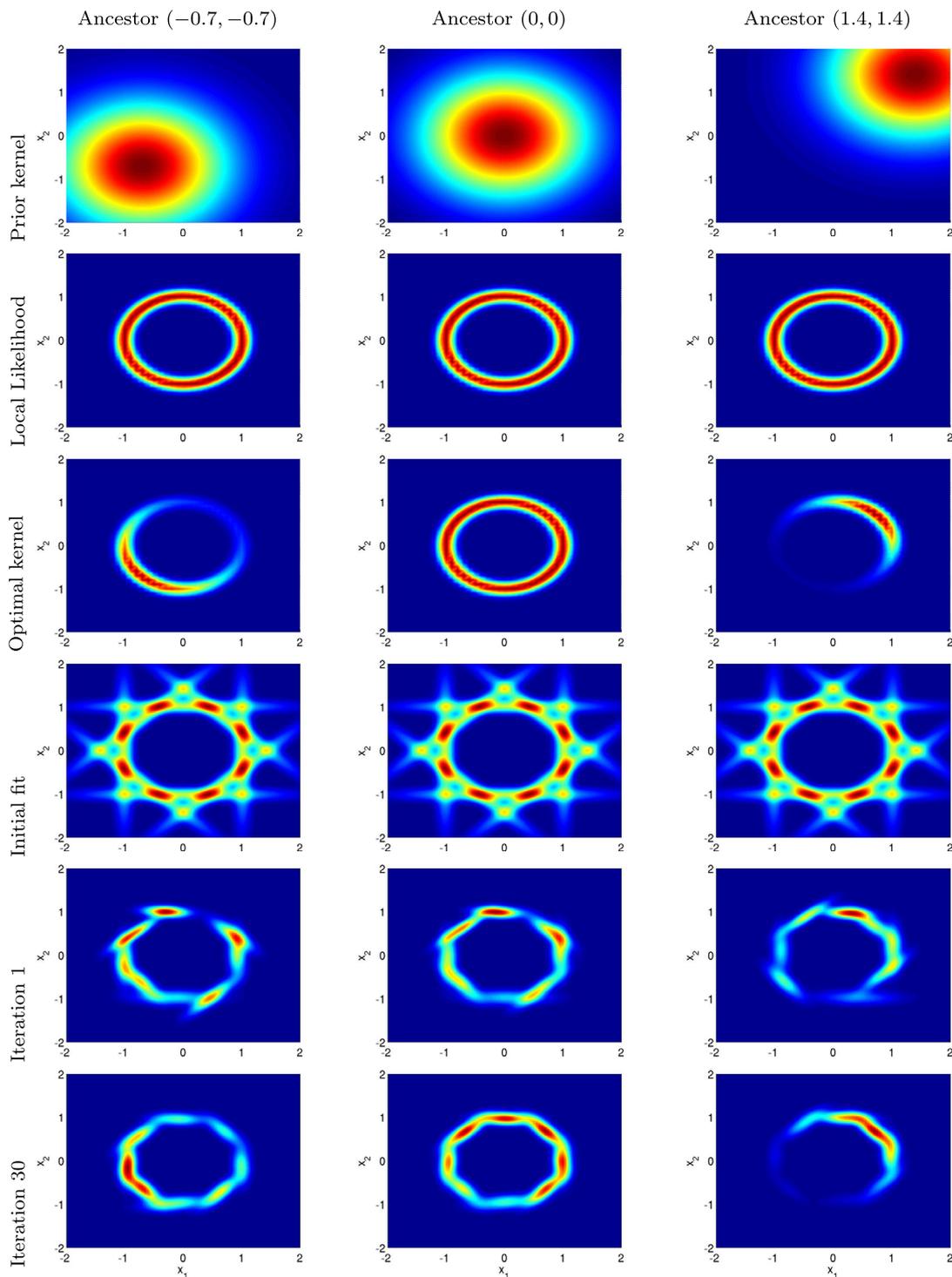

(a) Evolution of the KLD for the step size $0.1/\sqrt{L}$
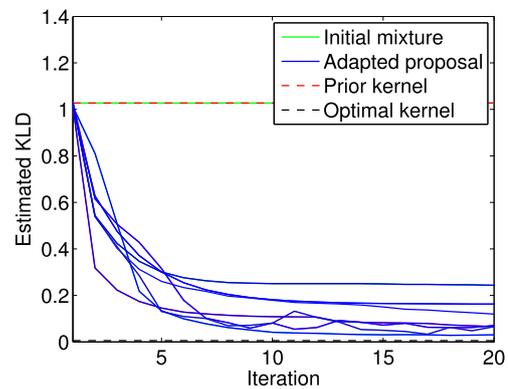
(b) Comparison of evolution of KLDs for step sizes $\{0.001; 0.01; 0.1; 1; 10; 15\}/\sqrt{L}$

**Fig. 2** Evolution of the KLD over $L = 20$ iterations of adaptation for the linear Gaussian model

Fig. 3 Evolution of the optimal and adapted kernels for the Bessel model for 3 distinct ancestors in different regions and for the observation $Y_{k+1} = 1.0$. The *last three rows* show the evolution of the adapted proposal kernel after the initial fit, one iteration, and 30 iterations, respectively. The adaptation algorithm used a constant sample size $N_\ell = 200$ for all $\ell \in \{0, 2, \ldots, 30\}$. After 30 iterations (the *last row*), the adapted kernel is visually very close to the optimal kernel

– Each $\mathbf{Y}_k$, taking values in $\mathsf{Y} = \mathbb{R}^2$, is a noisy observation of the corresponding hidden state $\mathbf{X}_k$ with local likelihood $g(\tilde{\mathbf{x}}, \mathbf{y}) = \mathcal{N}_2(\mathbf{y}; \tilde{\mathbf{x}}, \mathbf{\Sigma_Y})$, where $\mathbf{\Sigma_Y} = 0.1 \times \mathbf{I}_2$.

– The prior kernel density is a mixture of multivariate Gaussian distributions $q(\mathbf{x}, \tilde{\mathbf{x}}) = (\mathcal{N}_2(\tilde{\mathbf{x}}; \mathbf{\Lambda}_1\bar{\mathbf{x}}, \mathbf{\Sigma}) + \mathcal{N}_2(\tilde{\mathbf{x}}; \mathbf{\Lambda}_2\bar{\mathbf{x}}, \mathbf{\Sigma}))/2$ with regression matrices
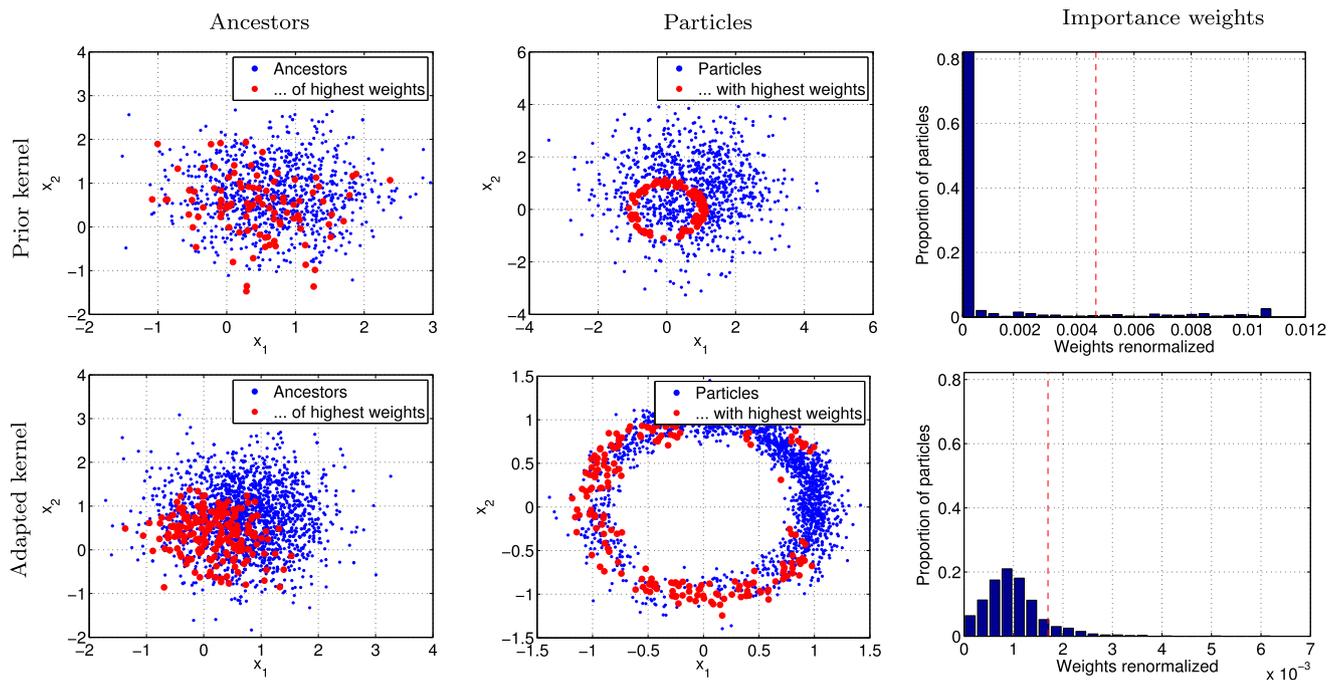
Fig. 4 Comparison of 1,000 and 200 particles from the prior and final adapted kernels, respectively, with the 10 % largest weights plotted in *red*, along with the corresponding ancestors and the distribution of the corresponding importance weights. For the prior kernel, the support of the proposal distribution is over-spread and as much as 80 % of particles have null weight. On the other hand, the adaptation algorithm concentrates the particles on the support of the target distribution, thus narrowing the span of the importance weights and balancing the contributions of all particles (Color figure online)

$$\boldsymbol{\Lambda}_1 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}, \qquad \boldsymbol{\Lambda}_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}.$$

– The filter ϕ_k at time k is assumed to be a bimodal distribution $\mathsf{X} = \mathbb{R}^2$ with density $(\mathcal{N}_2(\mathbf{x}; (0, 1)^\mathsf{T}, \boldsymbol{\Sigma}) + \mathcal{N}_2(\mathbf{x}; (0, -1)^\mathsf{T}, \boldsymbol{\Sigma}))/2$, where $\boldsymbol{\Sigma} = 0.1 \times \mathbf{I}_2$. The two modes are hence well separated.

In this first example, we consider *one single updating step of the particle filter*, with initial particles $\{\mathbf{X}_i\}_{i=1}^N$ sampled exactly from the filter ϕ_k. For this single step we study the performance of Algorithm 2. The observation $\mathbf{Y}_{k+1}$ is chosen to be $\mathbf{Y}_{k+1} = (1, 0)^\mathsf{T} = \boldsymbol{\Lambda}_2(0, 1)^\mathsf{T} = \boldsymbol{\Lambda}_1(0, -1)^\mathsf{T}$. The likelihood of $\mathbf{X}_k$ given $\mathbf{Y}_{k+1}$ is thus symmetric around the x-axis, giving equal weight to each of the two components of ϕ_k. Even though the optimal kernel l^* does not belong to our family of experts, it is still a mixture whose weights are highly dependent of location of the ancestor, and we can expect our algorithm to adapt accordingly.

The histogram in Fig. 1a of the importance weights of the particle swarm produced using the prior kernel shows that 50 % of these weights are nearly equal to zero, and the remaining ones are spread over a large range of values. This is confirmed by looking at the corresponding curve of proportions in Fig. 1b, showing that 80 % of the total mass is carried by only 25 % of the particles and 99 % of the total mass by 40 % of the particles.

One single iteration of the adaptation scheme described in Sect. 5.3 is sufficient to improve those proportions to 80 % of the mass for 40 % of the particles and 99 % of the mass for 55 % of the particles; see the corresponding curve of proportions in Fig. 1b. After 10 such iterations, the corresponding curve of proportions (again in Fig. 1b) shows that close to maximum improvement has been reached: the first few steps of the optimisation are enough to bring significant improvement. The histogram in Fig. 1a of the weights obtained at the final iteration shows how the weights are concentrated around unity, the value that corresponds to sampling with the optimal kernel.

We display in Fig. 2a the KLD (3.8) between the fit and the target (estimated by means of a Monte Carlo approximation with a large sample size), and the same KLD for the proposal based on the prior kernel as well as the proposal based on the optimal kernel with uniform adjustment weights—i.e. all ancestor particles have practically the same optimal adjustment weight: choosing $a \equiv \mathbb{1}_\mathsf{X}$ makes the second term in (3.8) negligible. As mentioned earlier, the KLD decreases very fast, and most of the improvement is obtained after only a few iterations. Figure 2b compares the evolution of the KLD for several step sizes covering four orders of magnitude. As the step size is increased, the algorithm converges faster, although less smoothly: this has no impact from a practical point of

(a) Components of the logistic weight parameters $\underline{\boldsymbol{\beta}}^{\ell}$
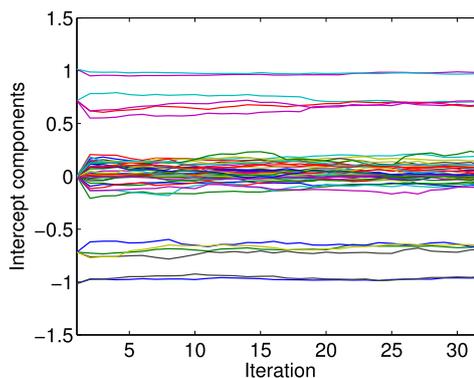
(b) Components of the intercept matrices $\boldsymbol{\mu}_{\ell}$

Fig. 5 Improvement of the proposal vai adaptation over 30 iterations. The plot of proportions (*top*) and the KLD (*bottom*) improve dramatically over the very first iterations. While the prior kernel puts 90 % of the mass on 15 % of the particles, adaptation increases it to 70 % of the particles in one step and stabilises to 80 % after a very few iterations

view as we are only looking for a good proposal kernel in an importance sampling setting, not for the exact optimal one.

6.2 Bessel process observed in noise

As a more challenging example we consider optimal filtering of the Brownian motion underlying a Bessel process observed in noise, also known as *range-only filtering* of a Gaussian random walk. We let the state process evolve in the plane for visualisation purposes. More specifically, the SSM is given by

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{V}_k,$$
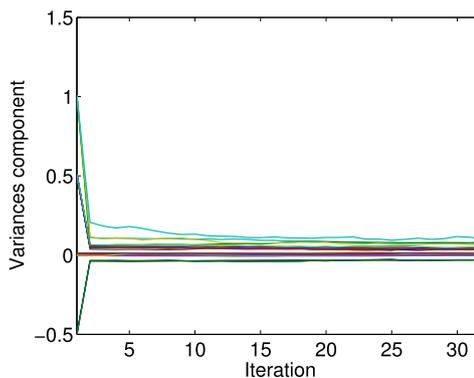
$$Y_k = \|\mathbf{X}_k\|_2 + W_k,$$

where $\{\mathbf{V}_k\}_{k\geq 0}$ and $\{W_k\}_{k\geq 0}$ are sequences of mutually independent $\mathcal{N}_2(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{X}})$-distributed and $\mathcal{N}_2(0, \sigma_Y^2)$-distributed, respectively, noise variables and $\|\mathbf{X}\|_2$ is the Euclidean $\mathcal{L}_2$ norm on the state space $\mathsf{X} := \mathbb{R}^2$. Note that the observations $\{Y_k\}_{k\geq 0}$ are real-valued in this case. With the notation in (6.1) and (6.2), it holds that $q(\mathbf{x}, \tilde{\mathbf{x}}) := \mathcal{N}_2(\tilde{\mathbf{x}}; \mathbf{x}, \boldsymbol{\Sigma}_{\mathbf{X}})$ the

(c) Components of the covariance matrices $\boldsymbol{\Sigma}_{\ell}$

Fig. 6 Parameters $\boldsymbol{\theta}_{\ell}$ of the adapted kernel over 30 iterations of the algorithm: practically, convergence is achieved after a few steps only

density of the prior kernel and by $g(\tilde{\mathbf{x}}, y) := \mathcal{N}(y, \|\tilde{\mathbf{x}}\|_2, \sigma_Y^2)$ the local likelihood. We ensure a diffuse prior kernel and informative observations by setting $\boldsymbol{\Sigma}_{\mathbf{X}} = \mathbf{I}_2$ and $\sigma_Y^2 = 0.01$. As the hidden state is observed range-only, the state equation provides most of the information concerning the bearing while the local likelihood is invariant under rotation of the state around the origin. This induces a variety of nonlinear shapes of the optimal kernel depending on the location of the ancestor—see the three top rows of Fig. 3.

333

(a) Negated Shannon entropy

(b) Relative effective sample size
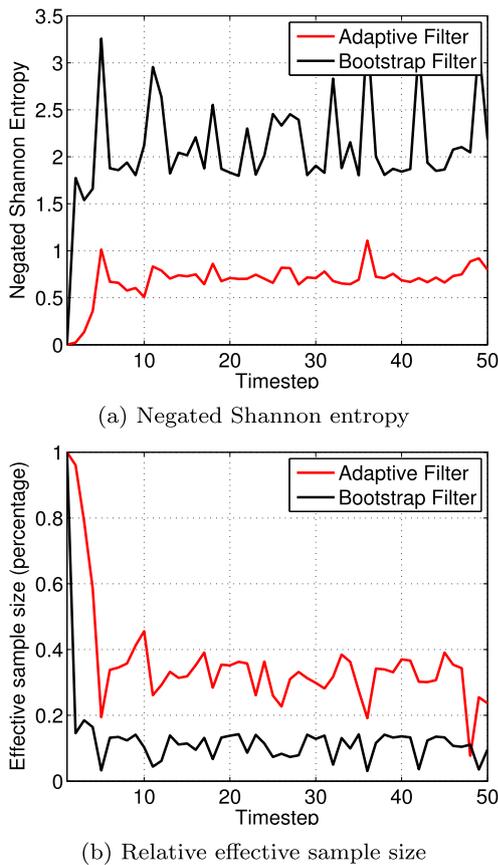
Fig. 7 Evolution of the effective sample size and the entropy between target and proposal for the bootstrap filter and the adaptive filter. As intended, the adaptive filter leads to higher effective sample size and lower negated Shannon entropy

To start with, we consider again *a single step* of the particle filter, updating a weighted particle sample $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^{N}$ approximating the filter ϕ_k at some time index k to a weighted sample $\{(\tilde{\mathbf{X}}_i, \tilde{\omega}_i)\}_{i=1}^{N}$ approximating the filter ϕ_{k+1} at the next time step. We assume that $\phi_k = \mathcal{N}_2((0.7, 0.7)^\top, 0.5 \times \mathbf{I}_2)$. In our experiment, the original weighted sample $\{(\mathbf{X}_i, \omega_i)\}_{i=1}^{N}$ consists of a sample of $N = 20,000$ i.i.d. particles drawn exactly from the filter distribution ϕ_k (hence, the particles have uniform weights). The observation at time $k + 1$ is $Y_{k+1} = 1.0$. We initialise the algorithm with a sample $\{(\tilde{\mathbf{X}}_i, \tilde{\omega}_i)\}_{i=1}^{N_0}$ of size $N_0 = 1,000$ using the prior kernel. The resulting cloud, along with the 100 particles with largest weights and the corresponding ancestors, is plotted in Fig. 4 (top row). The support of the proposal distribution is overspread: most of the particles have negligible weights, and only a few particles have comparatively large weights; indeed, 80 % of particles have null weight. This is confirmed by the curve of proportions in Fig. 5 (top): only 20 % of the proposed particles carry the total mass. Adaptation of the proposal kernel is thus highly relevant. Here again, adaptation of the adjustment weights is not required, as the ances-

tors of the particles with highest importance weights are not located in any specific region of the state space.

Based on these 1,000 particles, the first iteration of the adaptation is carried through using conditional probabilities from the initial fit displayed in Fig. 3 (fourth row), whose components are chosen to be independent of the ancestor, i.e. only the constant term is non-zero. The resulting kernel is plotted in Fig. 3 (fifth row). We use it to propose 200 new particles, serving as an importance sampling approximation of $\bar{\mu}$ at the second iteration. After 30 such iterations, the adapted kernel visible in Fig. 3 (last row) is visually very close to the optimal kernel. Note the impact of the location of the ancestor on the un-normalised transition kernel, whose mass shifts on a circle, and how the adapted kernel shifts accordingly. Figure 4 (bottom row) shows that the adaptation algorithm concentrates the particles on the support of the target distribution, thus narrowing the span of the importance weights and balancing the contributions of all particles.

Most importantly, a very small number of iterations suffices to achieve significantly more uniformly distributed importance weights, i.e. to significantly lower the KLD: Fig. 5 (bottom) shows how the KLD drops after the first 2 iterations and then stabilises near null, while the curve of proportions in Fig. 5 (top) shows that the distribution of the weights is essentially unchanged past the first few iterations.

As a final look at convergence, Fig. 6 displays the evolution of all the estimated parameters over the 30 iterations, confirming that the fit stabilises after a few steps: the result of the very first couple of iterations could serve as a more efficient proposal than the prior kernel. The parameters of the logistic weights $\underline{\beta}^\ell$ are the slowest to stabilise due to the stochastic gradient used to palliate for the lack of closed-form update as mentioned in Sect. 5.3.

We now turn focus away from a single time step and run the adaptive particle filter in parallel with a plain bootstrap filter for 50 times steps with a simulated observation record $\mathbf{Y}_{0:50}$ as input. At each iteration we let the adaptive filter optimise the proposal kernel using Algorithm 2. The forgetting of the initial distribution of the random walk entails that the filter distribution flow converges to a distribution symmetric under rotation and supported on a ring centered at $\mathbf{0}$. To connect with widespread measures of efficiency discussed in Cornebise et al. (2008), we compare in Fig. 7 the negated Shannon entropy (lower is better) of the importance weights and relative effective sample size (in percentage of the total sample; higher is better) of the two filters. The negated Shannon entropy shows improvement by our adaptive filter, which is not surprising in the light of the convergence results of Cornebise et al. (2008, Theorems 1–2): it is a consistent estimate of the same KLD when the number of particles tends to infinity. In this example also the relative effective sample size also exhibits an improvement, even
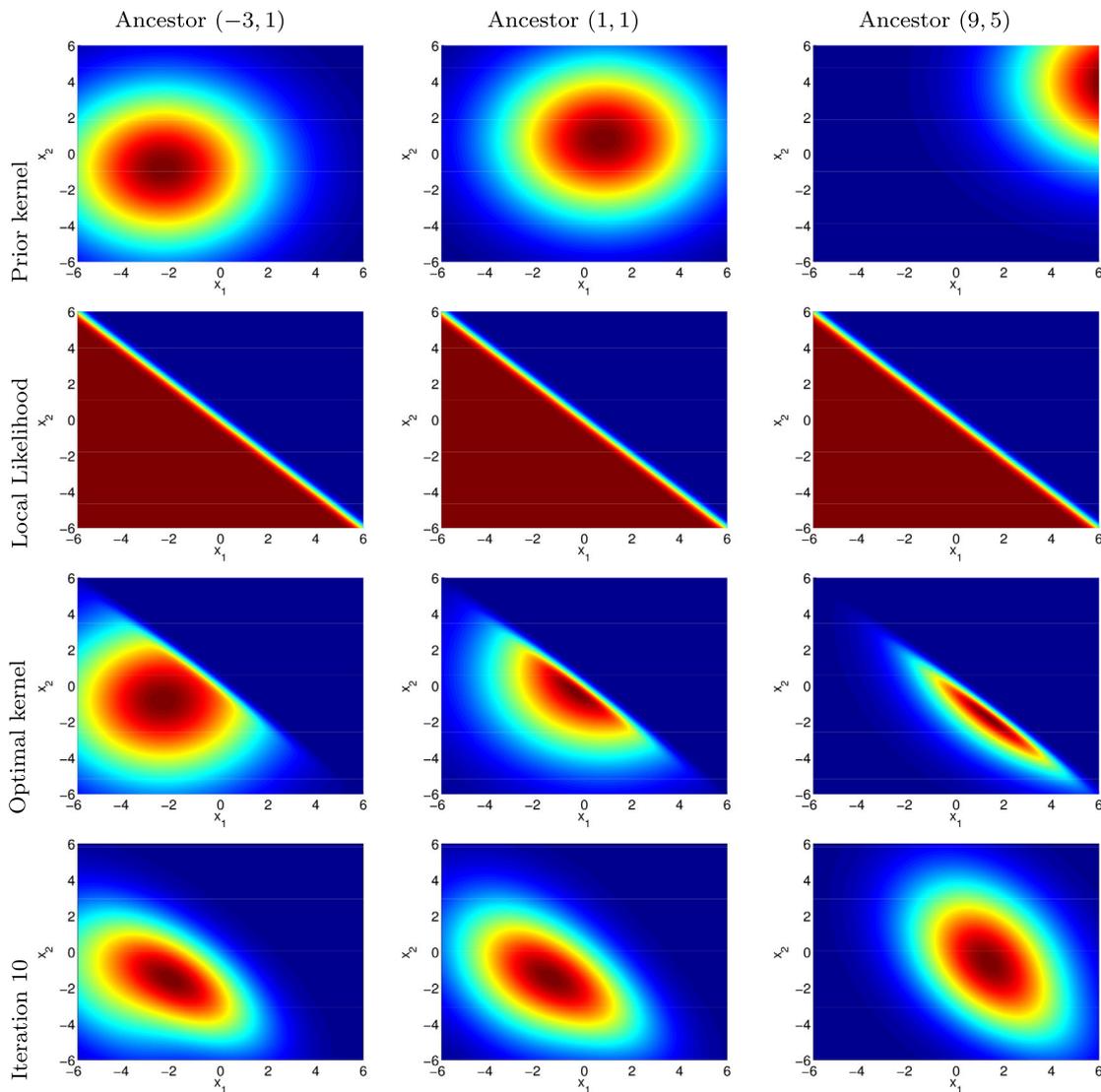
Fig. 8 Evolution of the adapted kernel, for three ancestors in different regions for the tobit model. In this example the observation brings, thanks to the censorship, a lot of information concerning the location of the ancestor. Thus, adapting only the proposal kernel will not lead to perfect adaptation, as can be seen on the fit after 10 iterations. In this case it is thus highly relevant to consider adaptation also of the adjustment multiplier weights

though it is a consistent estimate of the chi-square divergence between the same distributions rather than the KLD (see again Cornebise et al. 2008, Theorems 1–2, for an exact formulation of this result). Even though this makes sense to some extent, one should not expect this to hold systematically.

6.3 Multivariate tobit model

We now briefly illustrate how far adaptation of the proposal kernel leads—and where it stops. Consider a partially observed multivariate dynamic *tobit* model

$$\mathbf{X}_{k+1} = \mathbf{A}\mathbf{X}_k + \mathbf{U}_{k+1},$$

$$Y_k = \max\big(\mathbf{B}^\mathsf{T}\mathbf{X}_k + V_k, 0\big),$$

where $\mathsf{X} = \mathbb{R}^2$, $\mathbf{A} = 0.8 \times \mathbf{I_2}$, and $\mathbf{B} = (1,1)^\mathsf{T} \in \mathbb{R}^2$, so that Y_k takes values in $\mathbb{R}$. Here $\{\mathbf{U}_k\}_{k \geq 1}$ and $\{V_k\}_{k \geq 0}$ are independent sequences of mutually independent and identically distributed Gaussian variables with variances $\mathbf{\Sigma_U} = 2 \times \mathbf{I_2}$ and $\sigma_v^2 = 0.1$, respectively. The observation process consists of noisy observations of the sum of its components of the hidden states. In addition, the observations are left-censored. We consider again a single update of the particle swarm for a given time step k, where we have $N = 20,000$ ancestor particles distributed according to $\mathcal{N}_2((1,1)^\mathsf{T}, 10 \times \mathbf{I_2})$ and set $Y_{k+1} = 0$. The local likelihood is hence null above the line $\Delta = \{\mathbf{x} \in \mathbb{R}^2 : \mathbf{B}^\mathsf{T}\mathbf{x} = 0\}$ and constant below with a nar-

row transition region, as displayed in Fig. 8 (second row). The prior kernel displayed in Fig. 8 (first row) can have most of its mass out of the high-likelihood regions, depending on the ancestor. Figure 8 (third row) illustrates the unnormalised optimal transition kernel for three reference ancestors, showing how the match between the supports of the prior kernel and the local likelihood varies depending on the position of the ancestor particle relatively to the line $\mathbf{A}^{-1}\Delta$. Hence, half of the original particles have very small adjustment multiplier weights.

Adapting the proposal kernel will hence require at least two components, and will not lead to perfect adaptation, as can be seen on the fit after 10 iterations displayed in Fig. 8 (last row). The ancestor $(1,1)^{\mathsf{T}}$ is the center of the ancestor sample, and the un-normalised optimal kernel is the prior kernel truncated in its middle; $(-3,-1)^{\mathsf{T}}$ is the bottom left of the ancestor sample, and the un-normalised optimal kernel almost matches the prior, save for a truncation in the upper right tail; $(9,5)^{\mathsf{T}}$ is the top right of the ancestor sample, and the un-normalised optimal kernel differs widely from the prior kernel, as only the very far tails of the latter match non-null local likelihood.

Finally, without getting into details, we make our point through Fig. 9 showing again how the KLD drops in the first iteration for families of both Gaussian distributions and Student's t-distributions. We nevertheless purposefully keep the algorithm running for a large number of iterations, to illustrate the difference between the Gaussian and the Student's t parameterisations. The Gaussian parameterisation stabilises close to the minimum achieved by the optimal kernel with uniform adjustment weights, which is not negligible. In addition, the Student's t-distribution allows for heavier tails at the price of a higher attainable lower bound on the KLD.

Finally, Fig. 9 illustrates a limit case for kernel-only adaptation: replacing the prior kernel by the optimal one reduces only the KLD, which is still far from zero, by half. In this example, chosen for that purpose, the observation brings, thanks to the censorship, a lot of information concerning the location of the ancestor. In this case it is thus highly relevant to consider adaptation also of the adjustment multiplier weights. Naturally, this could, as a logical next step in development of adaptive SMC algorithms, be done by designing a minimisation algorithm for the second term of (3.8), which we leave as an open problem.

7 Future work and conclusion

Relying on the results of Cornebise et al. (2008), we have built new algorithms approximating the so-called optimal proposal kernel at a given time step of an auxiliary particle filter by means of minimisation of the KLD between the auxiliary target and instrumental distributions of the particle
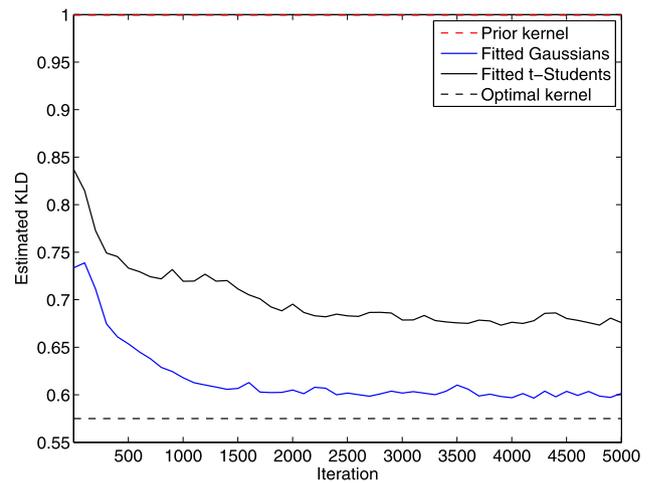
Fig. 9 Comparison of the evolution of the KLD for the Gaussian experts and the Student's t-distributed expert over $L = 5{,}000$ iterations of the algorithm

filter. More specifically, the algorithm fits a weighted mixture of integrated curved exponential distributions with logistic weights to the auxiliary target distribution by minimising the KLD between the two using a Monte Carlo version of the online EM method proposed by Cappé and Moulines (2009).

In addition, we have applied successfully this relatively simple algorithm to optimal filtering in SSMs; indeed, running the stochastic approximation-based adaptation procedure for only a few iterations at every time step as the particles evolve leveled off significantly the distribution of the total weight mass among the particles also for models exhibiting very strong nonlinearity. Thus, adding the adaptation step to an existing particle filter implementation implies only limited computational demands.

A proof of convergence, combining the theory developed in Sect. 5 with existing theory of stochastic approximation with Markovian perturbations (treated by, e.g., Duflo 1997; Benveniste et al. 1990) of the algorithm is currently in progress. In addition, we investigate at present the possibility of extending the approach to comprise adaptation also of the adjustment multipliers.

Appendix: Proof of Proposition 1

The proof follows the lines the proof of Cappé and Moulines (2009, Proposition 1). For vector-valued differentiable functions $h = (h_1, \ldots, h_m)^{\mathsf{T}}$ from Θ to $\mathbb{R}^m$ we denote by $\nabla_{\boldsymbol{\theta}} h^{\mathsf{T}}$ the $|\Theta| \times m$ matrix having the gradient $\nabla_{\boldsymbol{\theta}} h_j$ as jth column, that is, the inverse of the Jacobian of the same mapping.

Let $(\mathbf{s}^*, \mathbf{p}^*)$ be a zero of $\mathbf{h}$ and set $\boldsymbol{\theta}^* := \bar{\boldsymbol{\theta}}(\mathbf{s}^*, \mathbf{p}^*)$. Since for all $\boldsymbol{\eta}$ and $\mathbf{s} \in \mathbb{R}^{s \times s'}$,

$$\nabla_{\boldsymbol{\theta}} \mathrm{Tr}\big(\mathbf{b}(\boldsymbol{\eta})^{\mathsf{T}} \mathbf{s}\big) = \sum_{\ell=1}^{s'} \nabla_{\boldsymbol{\theta}} \mathbf{b}_{|\ell}(\boldsymbol{\eta})^{\mathsf{T}} \mathbf{s}_{|\ell}, \tag{A.1}$$

it holds, by definition,

$$
\begin{aligned}
\nabla_{\boldsymbol{\theta}} &\, l\big(\mathbf{s}^*, \mathbf{p}^*; \boldsymbol{\theta}\big)\big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} \\
&= -\big(\nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_1) \cdots \nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_d)\big)\big|_{\underline{\boldsymbol{\eta}}=\underline{\boldsymbol{\eta}}^*} \mathbf{p}^* \\
&\quad + (\nabla_{\boldsymbol{\theta}} \log \beta_1 \cdots \nabla_{\boldsymbol{\theta}} \log \beta_d)|_{\underline{\boldsymbol{\beta}}=\underline{\boldsymbol{\beta}}^*} \mathbf{p}^* \\
&\quad + \sum_{j=1}^{d} \sum_{\ell=1}^{s'} \nabla_{\boldsymbol{\theta}} \mathbf{b}_{|\ell}(\boldsymbol{\eta}_j)^{\mathsf{T}}\big|_{\underline{\boldsymbol{\eta}}=\underline{\boldsymbol{\eta}}^*} \mathbf{s}^*_{j|\ell} \\
&= \mathbf{0}.
\end{aligned}
\tag{A.2}
$$

On the other hand, by Fisher's identity,

$$
\begin{aligned}
\nabla_{\boldsymbol{\theta}} &\, \log \pi_{\boldsymbol{\theta}}(i, \tilde{\mathbf{x}}) \\
&= \mathbb{E}_{\bar{\pi}_{\boldsymbol{\theta}}}\big[\nabla_{\boldsymbol{\theta}} \log \bar{\pi}_{\boldsymbol{\theta}}(I, J, \tilde{\mathbf{X}}, \mathbf{U}) | \tilde{\mathbf{X}} = \tilde{\mathbf{x}}, I = i\big] \\
&= -\sum_{j=1}^{d} \bar{\pi}_{\boldsymbol{\theta}}(j|i, \tilde{\mathbf{x}}) \nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_j) \\
&\quad + \sum_{j=1}^{d} \bar{\pi}_{\boldsymbol{\theta}}(j|i, \tilde{\mathbf{x}}) \nabla_{\boldsymbol{\theta}} \log \beta_j \\
&\quad + \sum_{j=1}^{d} \bar{\pi}_{\boldsymbol{\theta}}(j|i, \tilde{\mathbf{x}}) \int \nabla_{\boldsymbol{\theta}} \mathrm{Tr}\big(\mathbf{b}(\boldsymbol{\eta}_j)^{\mathsf{T}} \mathbf{s}(\mathbf{X}_i, \tilde{\mathbf{x}}, \mathbf{u})\big) \\
&\qquad \times \bar{\rho}(\mathbf{u}|\mathbf{X}_i, \tilde{\mathbf{x}}; \boldsymbol{\eta}_j) \, d\mathbf{u},
\end{aligned}
$$

and, by (A.1),

$$
\begin{aligned}
\nabla_{\boldsymbol{\theta}} &\, \log \pi_{\boldsymbol{\theta}}(i, \tilde{\mathbf{x}}) \\
&= -\sum_{j=1}^{d} \bar{\pi}_{\boldsymbol{\theta}}(j|i, \tilde{\mathbf{x}}) \nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_j) \\
&\quad + \sum_{j=1}^{d} \bar{\pi}_{\boldsymbol{\theta}}(j|i, \tilde{\mathbf{x}}) \nabla_{\boldsymbol{\theta}} \log \beta_j \\
&\quad + \sum_{j=1}^{d} \sum_{\ell=1}^{s'} \bar{\pi}_{\boldsymbol{\theta}}(j|i, \tilde{\mathbf{x}}) \nabla_{\boldsymbol{\theta}} \mathbf{b}_{|\ell}(\boldsymbol{\eta}_j)^{\mathsf{T}} \\
&\qquad \times \int \mathbf{s}_{|\ell}(\mathbf{X}_i, \tilde{\mathbf{x}}, \mathbf{u}) \bar{\rho}(\mathbf{u}|\mathbf{X}_i, \tilde{\mathbf{x}}; \boldsymbol{\eta}_j) \, d\mathbf{u}.
\end{aligned}
$$

Consequently,

$$\nabla_{\boldsymbol{\theta}} d_{\mathrm{KL}}(\bar{\mu} \| \pi_{\boldsymbol{\theta}})$$

$$
\begin{aligned}
&= \mathbb{E}_{\bar{\mu}}\big[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(I, \tilde{\mathbf{X}})\big] \\
&= -\big(\nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_1) \cdots \nabla_{\boldsymbol{\theta}} a(\boldsymbol{\eta}_d)\big) \bar{\mathbf{p}}(\boldsymbol{\theta}) \\
&\quad + (\nabla_{\boldsymbol{\theta}} \log \beta_1 \cdots \nabla_{\boldsymbol{\theta}} \log \beta_d) \bar{\mathbf{p}}(\boldsymbol{\theta}) \\
&\quad + \sum_{j=1}^{d} \sum_{\ell=1}^{s'} \nabla_{\boldsymbol{\theta}} \mathbf{b}_{|\ell}(\boldsymbol{\eta}_j)^{\mathsf{T}} \bar{\mathbf{s}}_{j|\ell}(\boldsymbol{\theta}).
\end{aligned}
\tag{A.3}
$$

Thus, as

$$\mathbf{h}(\mathbf{s}^*, \mathbf{p}^*) = \mathbf{0}$$
$$\Rightarrow \quad \bar{\mathbf{s}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s}^*, \mathbf{p}^*)\big) \oplus \bar{\mathbf{p}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s}^*, \mathbf{p}^*)\big) = \mathbf{s}^* \oplus \mathbf{p},$$

we obtain

$$\nabla_{\boldsymbol{\theta}} d_{\mathrm{KL}}(\bar{\mu} \| \pi_{\boldsymbol{\theta}})|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} = \mathbf{0}. \tag{A.4}$$

Conversely, let $\boldsymbol{\theta}^*$ be a stationary point of $\boldsymbol{\theta} \mapsto d_{\mathrm{KL}}(\bar{\mu} \| \pi_{\boldsymbol{\theta}})$ (i.e. (A.4) holds) and let $(\mathbf{s}^*, \mathbf{p})$ be given by (5.16). Then we conclude, via (A.2) and (A.3), that $\boldsymbol{\theta}^*$ is a stationary point of $\boldsymbol{\theta} \mapsto l(\mathbf{s}^*, \mathbf{p}^*; \boldsymbol{\theta})$ as well. However, by Assumption 3, this point is unique and equal to $\bar{\boldsymbol{\theta}}(\mathbf{s}^*, \mathbf{p}^*)$; thus,

$$\mathbf{s}^* \oplus \mathbf{p}^* = \bar{\mathbf{s}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s}^*, \mathbf{p}^*)\big) \oplus \bar{\mathbf{p}}\big(\bar{\boldsymbol{\theta}}(\mathbf{s}^*, \mathbf{p}^*)\big)$$
$$\Rightarrow \quad \mathbf{h}(\mathbf{s}^*, \mathbf{p}^*) = \mathbf{0},$$

which completes the proof.

References

Andrieu, C., Moulines, E.: On the ergodicity properties of some adaptive MCMC algorithms. Ann. Appl. Probab. **16**, 1462–1505 (2006)

Andrieu, C., Doucet, A., Holenstein, R.: Particle Markov chain Monte Carlo methods. J. R. Stat. Soc., Ser. B, Stat. Methodol. **72**(3), 269–342 (2010)

Benveniste, A., Métivier, M., Priouret, P.: Adaptive Algorithms and Stochastic Approximations, vol. 22. Springer, Berlin (1990). Translated from the French by Stephen S.S. Wilson

Cappé, O., Moulines, E.: On-line expectation–maximization algorithm for latent data models. J. R. Stat. Soc., Ser. B **71**(3), 593–613 (2009)

Cappé, O., Moulines, E., Rydén, T.: Inference in Hidden Markov Models. Springer, Berlin (2005)

Cappé, O., Douc, R., Guillin, A., Marin, J.M., Robert, C.P.: Adaptive importance sampling in general mixture classes. Stat. Comput. **18**(4), 447–459 (2008)

Cornebise, J.: Adaptive sequential Monte Carlo methods. Ph.D. thesis, Université Pierre et Marie Curie–Paris 6 (2009)

Cornebise, J., Moulines, E., Olsson, J.: Adaptive methods for sequential importance sampling with application to state space models. Stat. Comput. **18**(4), 461–480 (2008)

Del Moral, P., Doucet, A., Jasra, A.: Sequential Monte Carlo samplers. J. R. Stat. Soc., Ser. B **68**(3), 411 (2006)

Del Moral, P., Doucet, A., Jasra, A.: On adaptive resampling strategies for sequential Monte Carlo methods. Bernoulli **18**(1), 252–278 (2012)

Douc, R., Moulines, E., Olsson, J.: Optimality of the auxiliary particle filter. Probab. Math. Stat. **29**(1), 1–28 (2008)

Doucet, A., Godsill, S., Andrieu, C.: On sequential Monte Carlo sampling methods for Bayesian filtering. Stat. Comput. **10**(3), 197–208 (2000)

Doucet, A., De Freitas, N., Gordon, N. (eds.): Sequential Monte Carlo Methods in Practice. Springer, New York (2001)

Duflo, M.: Random Iterative Models, vol. 34. Springer, Berlin (1997). Translated from the 1990 French original by S.S. Wilson and revised by the author

Gordon, N., Salmond, D., Smith, A.F.: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proc., F, Radar Signal Process. **140**, 107–113 (1993)

Haario, H., Saksman, E., Tamminen, J.: An adaptive metropolis algorithm. Bernoulli **7**(2), 223–242 (2001)

Jordan, M., Jacobs, R.: Hierarchical mixtures of experts and the EM algorithm. Neural Comput. **6**, 181–214 (1994)

Jordan, M., Xu, L.: Convergence results for the EM approach to mixtures of experts architectures. Neural Netw. **8**(9), 1409–1431 (1995)

Korotsil, I., Peters, G., Cornebise, J., Regan, D.: Adaptive Markov chain Monte Carlo forward simulation for statistical analysis in epidemic modeling of human papilloma virus. To appear Stat. Med. arxiv:1108.3137 (2012)

Liu, J.: Monte Carlo Strategies in Scientific Computing. Springer, New York (2001)

Liu, C., Rubin, D.: ML estimation of the t distribution using EM and its extensions, ECM and ECME. Stat. Sin. **5**(1), 19–39 (1995)

McLachlan, G., Krishnan, T.: The EM Algorithm and Extensions, 2nd edn. Wiley, New York (2008)

Oh, M.S., Berger, J.O.: Integration of multimodal functions by Monte Carlo importance sampling. J. Am. Stat. Assoc. **88**(422), 450–456 (1993)

Peel, D., McLachlan, G.: Robust mixture modelling using the t distribution. Stat. Comput. **10**(4), 339–348 (2000)

Pitt, M.K., Shephard, N.: Filtering via simulation: auxiliary particle filters. J. Am. Stat. Assoc. **94**(446), 590–599 (1999)

Quandt, R., Ramsey, J.: A new approach to estimating switching regressions. J. Am. Stat. Assoc. **67**, 306–310 (1972)

Rayens, W., Greene, T.: Covariance pooling and stabilization for classification. Comput. Stat. Data Anal. **11**(1), 17–42 (1991)

Ristic, B., Arulampalam, M., Gordon, A.: Beyond Kalman Filters: Particle Filters for Target Tracking. Artech House, Norwood (2004)

Roberts, G.O., Rosenthal, J.S.: Examples of adaptive MCMC. J. Comput. Graph. Stat. **18**(2), 349–367 (2009)

Robins, P., Rapley, V.E., Green, N.: Realtime sequential inference of static parameters with expensive likelihood calculations. J. R. Stat. Soc., Ser. B **58**, 641–662 (2009)

Rubinstein, R.Y., Kroese, D.P.: The Cross-Entropy Method. Springer, Berlin (2004)

Van der Merwe, R., Wan, E.: Sigma-point Kalman filters for probabilistic inference in dynamic state-space models. In: Proceedings of the Workshop on Advances in Machine Learning, Montreal, Canada (2003)

Van der Merwe, R., Doucet, A., De Freitas, N., Wan, E.: The unscented particle filter. In: Leen, T.K., Dietterich, T.G., Tresp, V. (eds.) Adv. Neural Inf. Process. Syst., vol. 13. MIT Press, Cambridge (2000)