Smoothing Algorithms for State-Space Models

Mark Briers, Arnaud Doucet, and Simon Maskell

Abstract

A prevalent problem in statistical signal processing, applied statistics, and time series analysis is the calculation of the smoothed posterior distribution, which describes the uncertainty associated with a state, or a sequence of states, conditional on data from the past, the present, and the future. The aim of this paper is to provide a rigorous foundation for the calculation, or approximation, of such smoothed distributions, to facilitate a *robust* and *efficient* implementation. Through a cohesive and generic exposition of the scientific literature we offer several novel extensions such that one can perform smoothing in the most general case. Experimental results for: a Jump Markov Linear System; a comparison of particle smoothing methods; and parameter estimation using a particle implementation of the EM algorithm, are provided.

Index Terms

State space smoothing, Hidden Markov Model, Kalman filter, Kalman smoother, Jump Markov Linear System, Particle filter, Particle smoother, Parameter estimation.

I. Introduction

One is often interested in quantifying the uncertainty associated with an unobserved variable, X, given some information pertaining to that variable, Y. In a sequential context, this relates to the calculation of the posterior density,[1] $p(x_{1:T}|y_{1:T})$, where $x_{1:T} = \{x_1, \ldots, x_T\}$, $y_{1:T} = \{y_1, \ldots, y_T\}$, are generic points in path space of the signal and observation processes, and the discrete time index $t \in \mathbb{N}^+$. To facilitate sequential estimation, a first order Markovian state space model is often assumed[46]. Furthermore, the observations are assumed to be conditionally independent given the state $X_t = x_t$. One thus has mathematical expressions for the state transition and observation densities, as follows:

$$X_t|X_{t-1} = x_{t-1} \sim f(\cdot|x_{t-1}); \tag{1}$$

$$Y_t|X_t = x_t \sim g(\cdot|x_t), \tag{2}$$

with an appropriately defined prior density, $X_1 \sim \mu(\cdot)$.

For many state-space applications (e.g. tracking[4]) it is more convenient to compute the marginal posterior distribution of the state at a particular time instance conditional on a sequence of data, $p(x_t|y_{1:l})$. If $l < t$ then this process is known as *prediction*; if $l = t$ then it is commonly referred to as *filtering*; and if $l > t$ then one is conducting the process of *smoothing*. The main objective of this article is to present a generic exposition of the vast array of literature available in the scientific research community that attempt to solve the 'smoothing' problem. Moreover, we are able to offer several novel improvement strategies for the techniques available by identifying and solving problems which were previously ignored. We illustrate these novel algorithms on several examples.

The smoothing problem is commonly segmented into three problems: fixed-interval smoothing, where one is interested in calculating $p(x_t|y_{1:T})$ for all time indices $t = 1, \ldots, T$; fixed lag smoothing, where one calculates $p(x_t|y_{1:t+L})$ where $L > 0$ is some fixed value (this density is calculated on-line); and fixed point smoothing, where $p(x_t|y_{1:D})$ is calculated for a fixed value t with $D > t$ increasing.

- The most common application of smoothing is the (off-line) fixed-interval smoothing algorithm. The authors assert that it is possible to employ a single smoothing scheme, based on fixed-interval smoothing, to solve all three problems. Details of this proposition are now provided.
- Several schemes have been suggested to solve the fixed-lag smoothing problem, with a favoured technique being the augmentation of the states over the lag L. In general, however, one then has a computational cost which is exponential in the dimension of the state space, n_x. A more suitable technique would be to store the latest filtering distribution $p(x_{T-1}|y_{1:T-1})$, and calculate the new filtering distribution $p(x_T|y_{1:T})$ on the receipt of a new measurement. One is then able to employ a (traditionally considered) fixed-interval smoothing algorithm to calculate $p(x_{T-L}|y_{1:T})$. This gives rise to a computational cost which is linear in the length of the lag.
- A similar situation occurs with fixed point smoothing, in that it is possible to calculate a distribution $p(x_t|y_{1:D})$ on the receipt of a new measurement by calculating a new filtering distribution at the latest time instance and stepping

Manuscript received XX; revised XX.

M. Briers is with the Advanced Signal and Information Processing group, QinetiQ Ltd, Malvern, UK, and Cambridge University Engineering Department, Cambridge, UK, Email: m.briers@signal.qinetiq.com, or mb511@eng.cam.ac.uk

A. Doucet is with Cambridge University Engineering Department, Cambridge, UK, Email: ad2@eng.cam.ac.uk

S. Maskell is with the Advanced Signal and Information Processing group, QinetiQ Ltd, Malvern, UK, Email: s.maskell@signal.qinetiq.com

[1] Appropriate nomenclature and notation can be substituted for the discrete analogue.

back through the lag. However, one can calculate a revised smoothed distribution without the need for recursing through the intermediate history of the states, using the same fixed-interval type smoothing algorithm repository by calculating a distribution over the joint state (X_t, X_D). Note that while this method is computationally attractive and allows the methodology to be presented in a generic framework, particle based methods for fixed point smoothing would exhibit degeneracy as the number of data increased[11]. However, in most cases, the fixed-lag smoothing distribution converges exponentially fast to a fixed distribution so it is possible to stop including new data.

It is noted that one is also often interested in calculating the joint smoothing distribution, particularly in the fixed-interval $(p(x_{1:T}|y_{1:T}))$ and fixed-lag $(p(x_{T-L:T}|y_{1:T}))$ scenarios, and we are able to perform such calculations by exploiting the sequential nature of the problem, which is discussed in the sequel.

The format of the paper is as follows: the following section identifies the filtering and smoothing recursions for general state space models and suggests a rigorous framework on which one can base smoothing algorithms. The subsequent five sections then discuss the algorithmic implementation of such recursions under differing scenarios: finite state space hidden Markov models, linear Gaussian state space models, jump Markov linear systems, analytic approximations for general state models, and sequential Monte Carlo approximations for general state space models, respectively. Simulation results are presented in Section VIII, with conclusions drawn in Section IX.

II. Filtering and smoothing for general state space models

The derivations of the recursions used in the filtering, forward-backward smoothing, and two-filter smoothing procedures for general state space models are now described. Note that by replacing the integrals with summations, and the notation used to denote a probability density function with an appropriate term to denote a probability mass function, one arrives at expressions for both the continuous and discrete state space case.

A. Filtering

Filtering is the term used to describe the process of recursively calculating the marginal posterior distribution, and is a pre-requisite in all of the algorithms discussed herein. We therefore summarise the filtering procedure in each of the subsequent sections. The filtering recursion can be derived as follows:

$$\begin{aligned} p(x_t|y_{1:t}) &= \frac{p(y_t|x_t, y_{1:t-1})p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \\ &\propto g(y_t|x_t)p(x_t|y_{1:t-1}), \end{aligned} \tag{3}$$

where:

$$p(x_t|y_{1:t-1}) = \int f(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1}.$$

One can see that this algorithm is recursive in nature; equation (3) relies solely upon the state transition and observation densities, and the posterior density from the previous time instance.

B. Forward-backward smoothing

It is possible to deduce the 'standard' forward-backward recursive expression for the marginal smoothed posterior distribution, $p(x_t|y_{1:T})$, as follows[1]:

$$\begin{aligned} p(x_t|y_{1:T}) &= \int p(x_t, x_{t+1}|y_{1:T})dx_{t+1} \\ &= \int p(x_{t+1}|y_{1:T})p(x_t|x_{t+1}, y_{1:T})dx_{t+1} \\ &= \int p(x_{t+1}|y_{1:T})p(x_t|x_{t+1}, y_{1:t})dx_{t+1} \\ &= p(x_t|y_{1:t}) \int \frac{p(x_{t+1}|y_{1:T})f(x_{t+1}|x_t)}{p(x_{t+1}|y_{1:t})}dx_{t+1}. \end{aligned} \tag{4}$$

It is thus possible to compute the filtered and predicted distributions in a forward (filtering) recursion of the algorithm (by calculating $p(x_t|y_{1:t})$), and then execute a backward recursion with each smoothed distribution $(p(x_t|y_{1:T}))$ relying upon the quantities calculated and the previous (in reverse time) smoothed distribution $(p(x_{t+1}|y_{1:T}))$.

C. Two-filter smoothing

One can compute the marginal smoothed posterior distribution by combining the output of two (independent) filters, one that recurses in the forwards time direction and calculates $p(x_t|y_{1:t-1})$, the other in the backward time direction calculating $p(y_{t:T}|x_t)$, to give the required distribution $p(x_t|y_{1:T})$[6], [31], [32]. This is constructed as follows:

$$\begin{aligned} p(x_t|y_{1:T}) &= p(x_t|y_{1:t-1}, y_{t:T}) \\ &= \frac{p(x_t|y_{1:t-1})p(y_{t:T}|y_{1:t-1}, x_t)}{p(y_{t:T}|y_{1:t-1})} \\ &\propto p(x_t|y_{1:t-1})p(y_{t:T}|x_t). \\ &\propto p(x_t|y_{1:t})p(y_{t+1:T}|x_t) \end{aligned} \tag{5}$$

This form of smoothing reduces the complexity of the smoothing procedure for certain modelling assumptions.

A *Backward Information Filter*[37] can be used to calculate $p(y_{t:T}|x_t)$ sequentially from $p(y_{t+1:T}|x_{t+1})$:

$$\begin{aligned} p(y_{t:T}|x_t) &= \int p(y_t, y_{t+1:T}, x_{t+1}|x_t)dx_{t+1} \\ &= \int p(y_{t+1:T}|x_{t+1})f(x_{t+1}|x_t)g(y_t|x_t)dx_{t+1}. \end{aligned} \tag{6}$$

Note that $p(y_{t:T}|x_t)$ is not a probability density function (pdf) in argument x_t and thus its integral over x_t might not be finite. However, it is often more convenient in practice to propagate a pdf. Moreover, particle based methods can only be used to approximate finite measures. To cater for these cases, and thus ensure that $p(y_{t:T}|x_t)$ is integrable over x_t, we introduce an *artificial prior distribution* over x_t denoted $\gamma_t(x_t)$. This concept is now described.

Given the initial condition:

$$\widetilde{p}(x_T|y_T) = \frac{g(y_T|x_T)\gamma_T(x_T)}{p(y_T)}$$

and defining the sequence of probability distributions:

$$\widetilde{p}(x_{t:T}|y_{t:T}) \propto \gamma_t(x_t) \prod_{i=t+1}^{T} f(x_i|x_{i-1}) \prod_{i=t}^{T} g(y_i|x_i),$$

where $t < T$, one can see that:

$$\begin{aligned} p(y_{t:T}|x_t) &= \int \cdots \int p(y_{t:T}, x_{t+1:T}|x_t)dx_{t+1:T} \\ &= \int \cdots \int p(x_{t+1:T}|x_t)p(y_{t:T}|x_{t:T})dx_{t+1:T} \\ &= \int \cdots \int \prod_{i=t+1}^{T} f(x_i|x_{i-1}) \prod_{i=t}^{T} g(y_i|x_i)dx_{t+1:T} \\ &= \int \cdots \int \frac{\gamma_t(x_t)}{\gamma_t(x_t)} \prod_{i=t+1}^{T} f(x_i|x_{i-1}) \prod_{i=t}^{T} g(y_i|x_i)dx_{t+1:T} \\ &\propto \int \cdots \int \frac{\widetilde{p}(x_{t:T}|y_{t:T})}{\gamma_t(x_t)} dx_{t+1:T} \\ &= \frac{\widetilde{p}(x_t|y_{t:T})}{\gamma_t(x_t)}. \end{aligned} \tag{7}$$

This derivation allows one to construct a filtering-like recursion to determine the information quantity, whilst ensuring that one propagates finite measures:

$$\begin{aligned} p(y_{t:T}|x_t) &= \int p(y_{t+1:T}|x_{t+1})f(x_{t+1}|x_t)g(y_t|x_t)dx_{t+1} \\ &\propto \int \frac{\widetilde{p}(x_{t+1}|y_{t+1:T})}{\gamma_{t+1}(x_{t+1})} f(x_{t+1}|x_t)g(y_t|x_t)dx_{t+1} \\ &= \frac{g(y_t|x_t)}{\gamma_t(x_t)} \int \widetilde{p}(x_{t+1}|y_{t+1:T}) \frac{f(x_{t+1}|x_t)\gamma_t(x_t)}{\gamma_{t+1}(x_{t+1})} dx_{t+1} \end{aligned}$$

1) Prediction: The "prediction" stage can be defined as follows:

$$\widetilde{p}(x_t|y_{t+1:T}) \triangleq \int \widetilde{p}(x_{t+1}|y_{t+1:T}) \frac{f(x_{t+1}|x_t)\gamma_t(x_t)}{\gamma_{t+1}(x_{t+1})} dx_{t+1}. \tag{8}$$

Note that (8) is only a probability measure if one defines the artificial prior as follows:

$$\gamma_t(x_t) \triangleq \begin{cases} \gamma_1(x_1) & \text{for } t = 1; \\ \int \cdots \int \gamma_1(x_1) \prod_{t'=1}^{t-1} f(x_{t'+1}|x_{t'}) dx_{1:t-1} = p(x_t) & \text{for } t \geq 2; \end{cases} \tag{9}$$

which gives the (generally time inhomogeneous) backward Markov kernel:

$$\gamma(x_t|x_{t+1}) = \frac{f(x_{t+1}|x_t)\gamma_t(x_t)}{\gamma_{t+1}(x_{t+1})}$$

where on substitution:

$$\widetilde{p}(x_t|y_{t+1:T}) = \int \widetilde{p}(x_{t+1}|y_{t+1:T})\gamma(x_t|x_{t+1})dx_{t+1}.$$

Note that the prior distribution $\gamma_1(x_1)$ appearing in (9) is an arbitrary prior distribution and not necessarily that used in the filtering recursion. There are certain circumstances where employing this formulation $\gamma_1(x_1) \neq \mu(x_1)$ is useful. For example, if the Markov kernel $f(x_{t+1}|x_t)$ admits an invariant distribution $\pi(x)$ it is convenient to define $\gamma_t(x_t) = \pi(x)$, even if $\mu(x_1) \neq \pi(x)$, as then $\gamma_t(x_t) = \pi(x)$ for all $t \geq 2$. Moreover, $\gamma(x_t|x_{t+1}) = f(x_t|x_{t+1})$ if f is π-reversible. If one takes the actual (filtering) prior distribution $\mu(x_1)$ then one uses the abusive notation $p(x_t|x_{t+1}) = \gamma(x_t|x_{t+1})$. The use of such a prior distribution was (implicitly) used in references [6], [22], [35].

The calculation of (9) is not generally analytically tractable when one uses the prior $\mu(\cdot)$. That is, in the general case the backward Markov kernel $p(x_t|x_{t+1})$ may not be known or even analytic and one necessarily has to propagate a finite measure, and so the additional degree of freedom provided by (8) is important in applications; see experimental results.

2) Update: The "update" stage is simply taken to be:

$$\widetilde{p}(x_t|y_{t:T}) = \frac{g(y_t|x_t)\widetilde{p}(x_t|y_{t+1:T})}{\int g(y_t|x'_t)\widetilde{p}(x'_t|y_{t+1:T})dx'_t}, \tag{10}$$

which has been renormalised to be a probability measure.

One can therefore see that:

$$\begin{aligned} p(x_t|y_{1:T}) \propto & p(x_t|y_{1:t-1})p(y_{t:T}|x_t) \\ \propto & \frac{p(x_t|y_{1:t-1})\widetilde{p}(x_t|y_{t:T})}{\gamma_t(x_t)}. \end{aligned} \tag{11}$$

Since the artificial prior distribution is introduced in (8), it needs to be removed when calculating the smoothing distribution, hence the factor $\gamma_t^{-1}(x_t)$.

For clarity, we will refer to (5) as an independent two-filter smoother, and to (11) as an artificial two-filter smoother.

3) Propagation of a probability measure: It is often algorithmically appealing to be able to propagate a probability measure, for example in the unscented information filter (as will be discussed later in the paper), rather than a finite measure as one would by the (general) use of equation (8). As stated, the calculation of (9) is not analytically tractable, could be computationally infeasible, or an approximation could be insufficient. Thus, it is convenient to define the joint distribution:

$$\widetilde{p}(x_t, x_{t+1}|y_{t:T}) \propto \widetilde{p}(x_t, x_{t+1}|y_{t+1:T})\widetilde{q}(x_t|x_{t+1}, y_t)\frac{g(y_t|x_t)f(x_{t+1}|x_t)\gamma_t(x_t)}{\gamma_{t+1}(x_{t+1})\widetilde{q}(x_t|x_{t+1}, y_t)}, \tag{12}$$

such that one is able to directly compute $p(y_{t:T}|x_t, x_{t+1})$ in an analogous manner to that done in (10). Further details will be provided on the use of (12) when deemed appropriate later in the paper.

D. Joint Distribution

As mentioned, it is possible to capitalise on the sequential nature of the problem when one constructs the joint distribution, by considering the following factorisation[15]:

$$p(x_{1:T}|y_{1:T}) = p(x_T|y_{1:T}) \prod_{t=1}^{T-1} p(x_t|x_{t+1:T}, y_{1:T}). \tag{13}$$

It is possible to exploit the modelling assumptions (1)-(2) to yield the following simple result:

$$\begin{aligned} p(x_t|x_{t+1:T}, y_{1:T}) &= p(x_t|x_{t+1}, y_{1:t}) \\ &= \frac{p(x_t|y_{1:t})f(x_{t+1}|x_t)}{p(x_{t+1}|y_{1:t})} \\ &\propto p(x_t|y_{1:t})f(x_{t+1}|x_t). \end{aligned}$$

It is thus possible to calculate $p(x_{1:T}|y_{1:T})$ based solely on $p(x_t|y_{1:t})$ and the state transition density $f(x_{t+1}|x_t)$, which are required for all values of t.

The authors note that alternative factorisations exist[5], which can be constructed as the symmetrical relationship of (13) when one uses an artificial prior distribution defined by (9), as follows:

$$p(x_{1:T}|y_{1:T}) = p(x_1|y_{1:T}) \prod_{t=2}^{T} p(x_t|x_{t-1}, y_{1:T}),$$

where

$$\begin{aligned} p(x_t|x_{t-1}, y_{1:T}) &= p(x_t|x_{t-1}, y_{t:T}) \\ &= \frac{p(y_{t:T}|x_t)f(x_{t+1}|x_t)}{p(y_{t:T}|x_{t-1})} \\ &\propto \frac{\widetilde{p}(x_t|y_{t:T})\gamma(x_t|x_{t+1})}{\widetilde{p}(x_{t-1}|y_{t:T})}, \end{aligned}$$

where $\gamma(x_t|x_{t+1})$ is the backward Markov kernel if $\gamma_t(x_t)$ is defined through (9). These alternative factorisations, however, do not seem to bring any advantage over (13).

III. Finite state space hidden Markov model

Define X_t as a discrete-time, time-homogenous, n_x-state, first-order Markov chain. By using a discrete analogue to the previous section, one is able to construct the following algorithmic arguments.

A. Filtering

Let $\alpha_{t-1}(x_{t-1}) \triangleq p(X_{t-1} = x_{t-1}|y_{1:t-1})$. The filtering recursion based on the algorithm described by Rabiner[42] is given as follows:

$$\begin{aligned} p(X_t = x_t|y_{1:t-1}) &= \sum_{x_{t-1}} f(X_t = x_t|X_{t-1} = x_{t-1})\alpha_{t-1}(x_{t-1}) \\ \alpha_t(x_t) &= \frac{p(X_t = x_t|y_{1:t-1})g(y_t|X_t = x_t)}{\sum_{x'_t} p(X_t = x'_t|y_{1:t-1})g(y_t|X_t = x'_t)} \end{aligned}$$

with suitably defined initial conditions $\mu(X_1 = x_1),\ \forall x_1 \in S = \{1, \ldots, n_x\}$.

B. Forward-backward smoothing

A solution that enables one to conduct two passes through the data, the first (forward) pass calculating $\alpha_t(j)$ and the second (backward) pass calculating the required quantity, $p(X_t = x_t|y_{1:T})$, is given as:

$$p(X_t = x_t|y_{1:T}) = \alpha_t(x_t) \sum_{x_{t+1}} \frac{p(X_{t+1} = x_{t+1}|y_{1:T})f(X_{t+1} = x_{t+1}|X_t = x_t)}{\sum_{x'_t} \alpha_t(x'_t)f(X_{t+1} = x_{t+1}|X_t = x'_t)}.$$

Surprisingly, this formula is not used very often, not least because it avoids the need to rescale the results to prevent numerical instabilities of the independent smoother which is used routinely in this context.

C. Two-filter smoothing

1) Independent smoother: Defining $\beta_t(x_t) \triangleq p(y_{t:T}|X_t = x_t)$, one recurses in the backward direction:

$$\beta_t(x_t) = g(y_t|X_t = x_t) \sum_{x_{t+1}} \beta_{t+1}(x_{t+1}) f(X_{t+1} = x_{t+1}|X_t = x_t),$$

with the combination step given as:

$$p(X_t = x_t|y_{1:T}) = \frac{\sum_{x_{t-1}} \alpha_{t-1}(x_{t-1}) f(X_t = x_t|X_{t-1} = x_{t-1}) \beta_t(x_t)}{\sum_{x'_t} \left[\sum_{x_{t-1}} \alpha_{t-1}(x_{t-1}) f(X_t = x'_t|X_{t-1} = x_{t-1}) \beta_t(x'_t) \right]}.$$

As previously stated, in practice one has to rescale $\beta_t(\cdot)$ otherwise the technique is numerically unstable.

The Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm[3] can be used to calculate the maximum *a posteriori* (MAP) estimate of the marginal state X_t based on the independent two-filter smoothing algorithm. A more recent article describes on online version of the BCJR algorithm[41].

2) Artificial smoother: It is possible to derive a recursion for the artificial smoother as follows:

$$\widetilde{p}(X_t = x_t|y_{t+1:T}) = \frac{\sum_{x_{t+1}} \widetilde{p}(X_{t+1} = x_{t+1}|y_{t+1:T}) f(X_{t+1} = x_{t+1}|X_t = x_t) \gamma_t(X_t = x_t)}{\gamma_{t+1}(X_{t+1} = x_{t+1})}$$

$$\widetilde{p}(X_t = x_t|y_{t:T}) = \frac{1}{c_t} g(y_t|X_t = x_t) \widetilde{p}(X_t = x_t|y_{t+1:T}),$$

and where $c_t = \sum_{x_t} g(y_t|X_t = x_t)\widetilde{p}(X_t = x_t|y_{t+1:T})$. We reiterate that $\widetilde{p}(X_t = x_t|y_{t+1:T})$ is not a probability measure when one does not choose $\gamma_t(\cdot)$ as in equation (9). The combination step is thus:

$$p(X_t = x_t|y_{1:T}) \propto \frac{p(X_t = x_t|y_{1:t-1})\widetilde{p}(X_t = x_t|y_{t:T})}{\gamma_t(X_t = x_t)},$$

which is subsequently renormalised.

D. Joint distribution

Using the factorisation (13) it is easy to compute the joint distribution. However, one is often interested in maximising it. The Viterbi algorithm[45] is a dynamic programming tool that enables one to circumvent the need to exhaustively search all combinations of state trajectories when calculating the MAP estimate of the distribution $p(x_{1:T}|y_{1:T})$:

$$x_{1:T}^{MAP} = \arg\max_{x_{1:T}} \; p(x_{1:T}|y_{1:T}).$$

Reference [20] introduces a technique to reduce the computational complexity of the forward-backward algorithm, and Viterbi algorithm.

IV. Linear Gaussian state space model

The remainder of this article assumes that the state X_t and measurement Y_t are respectively $\mathbb{R}^{n_x}$ and $\mathbb{R}^{n_y}$-valued random variables.

A. Filtering

The optimal Bayesian solution that allows exact calculation of the filtering distribution is available through the Kalman filter[30] when one assumes linear Gaussian models. That is:

$$f(x_t|x_{t-1}) = N(x_t; B_t x_{t-1}, Q_t)$$
$$g(y_t|x_t) = N(y_t; H_t x_t, R_t),$$

or equivalently:

$$X_t = B_t X_{t-1} + \epsilon_t \tag{14}$$
$$Y_t = H_t X_t + \nu_t, \tag{15}$$

where $p(\epsilon_t) = N(\epsilon_t; 0, Q_t)$ and $p(\nu_t) = N(\nu_t; 0, R_t)$ are mutually independent, uncorrelated noise processes[2]; $N(x; \mu, P)$ denotes a Gaussian density of argument x with mean μ and covariance P. The Kalman filter recursion can be summarised as follows:

$$
\begin{aligned}
p(x_{t-1}|y_{1:t-1}) &= N(x_{t-1}; \mu^x_{t-1|1:t-1}, P^x_{t-1|1:t-1}) \\
p(x_t, y_t|y_{1:t-1}) &= N\left(\begin{bmatrix} x_t \\ y_t \end{bmatrix}; \begin{bmatrix} \mu^x_{t|1:t-1} \\ \mu^y_{t|1:t-1} \end{bmatrix}, \begin{bmatrix} P^x_{t|1:t-1} & P^{xy}_{t|1:t-1} \\ P^{xy\ T}_{t|1:t-1} & P^y_{t|1:t-1} \end{bmatrix} \right) \\
p(x_t|y_{1:t}) &= N(x_t;\ \mu^x_{t|1:t}, P^x_{t|1:t}),
\end{aligned}
$$

where:

$$\mu^x_{t|1:t} = \mu^x_{t|1:t-1} + P^{xy}_{t|1:t-1} P^{y\ -1}_{t|1:t-1} (y_t - \mu^y_{t|1:t-1}) \tag{16}$$

$$P^x_{t|1:t} = P^x_{t|1:t-1} - P^{xy}_{t|1:t-1} P^{y\ -1}_{t|1:t-1} P^{xy\ T}_{t|1:t-1}, \tag{17}$$

and with:

$$
\begin{aligned}
\mu^x_{t|1:t-1} &= B_t \mu^x_{t-1|1:t-1} \\
P^x_{t|1:t-1} &= B_t P^x_{t-1|1:t-1} B_t^T + Q_t \\
\mu^y_{t|1:t-1} &= H_t \mu^x_{t|1:t-1} \\
P^y_{t|1:t-1} &= H_t P^x_{t|1:t-1} H_t^T + R_t \\
P^{xy}_{t|1:t-1} &= P^x_{t|1:t-1} H_t^T.
\end{aligned}
$$

Note that the conditioning $\mu^x_{l|1:l}$ is used to denote the conditioning on the data $y_{1:l}$, and by definition $\mathbb{E}_{p(x_l|y_{1:l})}[X_l] = \mu^x_{l|1:t}$ (with an appropriate definition for the covariance). In the above equations, the transpose of a matrix M is denoted by M^T, and the inverse is denoted M^{-1}.

B. Forward-backward smoothing

Assuming the linear Gaussian system described by equations (14) and (15), the marginal smoothed posterior distribution at time t also follows a Gaussian distribution, with mean $\mu^x_{t|1:T}$ and covariance $P^x_{t|1:T}$ and its parameters can be determined through (4). Simple algebraic manipulations yield the following equations[1]:

$$
\begin{aligned}
\mu^x_{t|1:T} &= \mu^x_{t|1:t} + \Gamma_t(\mu^x_{t+1|1:T} - \mu^x_{t+1|1:t}) \\
P^x_{t|1:T} &= P^x_{t|1:t} + \Gamma_t(P^x_{t+1|1:T} - P^x_{t+1|1:t})\Gamma_t^T \\
\Gamma_t &= P^x_{t|1:t} B_t^T P^{x\ -1}_{t+1|1:t}.
\end{aligned}
$$

Several researchers [10], [33] have independently derived an alternative to the standard forward-backward smoothing algorithm, to allow a more efficient and computationally stable algorithm to be produced. The algorithm is outlined below, but the reader is referred to the aforementioned references for a thorough discussion and derivation.

$$x^x_{t|1:T} = x^x_{t|1:t-1} + P^x_{t|1:t-1} r_{t-1} \tag{18}$$

$$P^x_{t|1:T} = P^x_{t|1:t-1} - P^x_{t|1:t-1} W_{t-1} P^x_{t|1:t-1}, \tag{19}$$

where:

$$
\begin{aligned}
r_{t-1} &= H_t^T P^{y\ -1}_{t|1:t-1} e_t + L_t^T \\
W_{t-1} &= H_t^T P^{y\ -1}_{t|1:t-1} H_t + L_t^T W_t L_t \\
L_t &= B_t - (B_t P^x_{t|1:t-1} H_t^T P^{y\ -1}_{t|1:t-1}) H_t \\
e_t &= y_t - \mu^y_{t|1:t-1}.
\end{aligned}
$$

A special case of the smoothing algorithm (18)-(19), known as the disturbance smoother[34], [38], has been developed, that allows the calculation of the smoothed disturbance vector. Typically, this algorithm has the appealing property of a reduced computational complexity, whilst being practically useful in many application domains.

[2]Dependent noise processes also yield solutions to the general filtering/smoothing problem, a topic that is not discussed in this article.

C. Two-filter smoothing

The following section discusses how one can avoid integrability issues through the use of a canonical representation of a Gaussian distribution[37]. The inclusion of an artificial smoother is therefore deemed redundant since the most general case, under the assumptions made, has been catered for.

1) Independent smoother: If $p(y_{t:T}|x_t)$ (with argument $y_{t:T}$) follows a Gaussian distribution and is integrable in x_t, then $p(y_{t:T}|x_t)/\int p(y_{t:T}|x_t)dx_t$ also follows a Gaussian distribution (with argument x_t). In the general case, the integrability of such functions need not be true, but it is possible to use a *canonical* representation of the Gaussian distribution[3][37]. This recursion is now summarised. Initialise at time T:

$$\begin{aligned}\tau_{T|T:T} &= {P^x_{T|T:T}}^{-1} = H_T^T R_T^{-1} H_T \\ \theta_{T|T:T} &= {P^x_{T|T:T}}^{-1}\mu^x_{T|T:T} = H_T^T R_T^{-1} y_T.\end{aligned}$$

One can then recurse in the reverse-time direction ($t = T-1,\ldots,1$) using the following:

$$\begin{aligned}\Delta_{t+1} &= \left[I + \Theta_{t+1}^T \tau_{t+1|t+1:T}\Theta_{t+1}\right]^{-1} \\ \tau_{t|t+1:T} &= B_{t+1}^T \tau_{t+1|t+1:T}\left(I - \Theta_{t+1}\Delta_{t+1}\Theta_{t+1}^T\tau_{t+1|t+1:T}\right)B_{t+1} \\ \theta_{t|t+1:T} &= B_{t+1}^T\left(I - \tau_{t+1|t+1:T}\Theta_{t+1}\Delta_{t+1}\Theta_{t+1}^T\right)\theta_{t+1|t+1:T} \\ \tau_{t|t:T} &= \tau_{t|t+1:T} + H_{t+1}^T R_{t+1}^{-1} H_{t+1} \\ \theta_{t|t:T} &= \theta_{t|t+1:T} + H_{t+1}^T R_{t+1}^{-1} y_t,\end{aligned}$$

where the *information vector* ($\theta_{t|t:T}$) and *information matrix* ($\tau_{t|t:T}$) completely parameterise $p(y_{t:T}|x_t)$ (with no need for integrability assumptions) and the notation $\Theta_{t+1} \triangleq Q_{t+1}^{\frac{1}{2}}$. This alternative representation also evades the need to compute any state transition matrix inversions. This recursion is the information form of the Kalman filter being used to parameterise $p(y_{t:T}|x_t)$; the inability to integrate $p(y_{t:T}|x_t)$ with respect to x_t (under the Gaussian assumptions stated above) is equivalent to having a diffuse prior in a reverse-time Kalman filter.

One is able to take the product of the two Gaussian distributed random variables (composed of the predicted density $p(x_t|y_{1:t-1})$ and $p(y_{t:T}|x_t)$) as in equation (5) to obtain the first and second moments of the (Gaussian distributed) smoothed marginal posterior distribution:

$$\begin{aligned}\mu^x_{t|1:T} &= P^x_{t|1:T}({P^x_{t|1:t-1}}^{-1}\mu^x_{t|1:t-1} + \theta_{t|t:T}) \\ P^x_{t|1:T} &= ({P^x_{t|1:t-1}}^{-1} + \tau_{t|t:T})^{-1}.\end{aligned}$$

D. Joint distribution

Using the factorisation of the joint distribution (13), one only has to store $\mu^x_{t|1:t}$, $P^x_{t|1:t}$, $\mu^x_{t|1:t-1}$ and $P^x_{t|1:t-1}$ (for $t = 1,\ldots,T$). The construction of the joint is then a simple operation.

V. Jump Markov linear system

Jump Markov linear systems are widely used in several fields of signal processing including seismic signal processing, digital communications such as interference suppression in CDMA spread spectrum systems, target tracking, and de-interleaving of pulse trains[16]. A derivation of a novel smoothing strategy which is mathematically rigorous is included in this section.

Following the convention in section III, we let A_t denote the discrete-time, time-homogenous, n_a-state, first order Markov chain with transition probability $p(A_{t+1} = a_{t+1}|A_t = a_t)$ for any $a_{t+1}, a_t \in S$, where $S = \{1, 2, \ldots, n_a\}$.

Throughout this section we consider the following Jump Markov Linear System (JMLS):

$$\begin{aligned}X_t &= B_t(A_t)X_{t-1} + \epsilon_t(A_t) \\ Y_t &= H_t(A_t)X_t + \nu_t(A_t),\end{aligned}$$

with $p(\epsilon_t(A_t)) = N(\epsilon_t(A_t); 0, Q_t(A_t))$ and $p(\nu_t(A_t)) = N(\nu_t(A_t); 0, R_t(A_t))$, and with the matrices $B_t(\cdot)$ and $H_t(\cdot)$ being functions of the Markov chain state a_t.

[3]The canonical representation of a Gaussian is parameterised by an information matrix (τ), with zero entries corresponding to infinite entries in the corresponding covariance matrix, and an information vector (θ). Note that we distinguish this from a (backwards) information filter, which we define to be $p(y_{t:T}|x_t)$ (as a function of x_t for fixed values of $y_{t:T}$). We explicitly avoid using the term "information filter" to describe this canonical representation of a Gaussian. The lack of a distinction in the literature is a cause for confusion and the authors hope making the distinction is useful to the reader.

A. Filtering

The filtering process is interested in calculating:

$$\begin{aligned} p(x_t, a_t|y_{1:t}) &= \sum_{a_{1:t-1}} p(x_t, a_{1:t}|y_{1:t}) \\ &= \sum_{a_{1:t-1}} p(a_{1:t}|y_{1:t}) p(x_t|a_{1:t}, y_{1:t}) \end{aligned} \tag{20}$$

which represents the (joint) uncertainty over the continuous state and the discrete state for the mode of the system at time t.

One is able to calculate the factor $p(a_{1:t}|y_{1:t})$ for a particular component by a simple filtering recursion:

$$\begin{aligned} p(a_{1:t}|y_{1:t}) &= p(a_{1:t}|y_{1:t-1}, y_t) \\ &\propto p(a_{1:t}|y_{1:t-1}) p(y_t|a_{1:t}, y_{1:t-1}), \end{aligned}$$

where:

$$p(a_{1:t}|y_{1:t-1}) = p(a_{1:t-1}|y_{1:t-1}) p(a_t|a_{t-1}),$$

and, in this case:

$$p(y_t|a_{1:t}, y_{1:t-1}) = N(y_t; \mu^y_{t|1:t-1,a_{1:t}}, P^y_{t|1:1-t,a_{1:t}}).$$

The notation $\mu^y_{t|1:t-1,a_{1:t}}$ and $P^y_{t|1:1-t,a_{1:t}}$ denote the mean and covariance of $p(y_t|a_{1:t}, y_{1:t-1})$, respectively, which are both calculated in the recursion for $p(x_t|a_{1:t}, y_{1:t})$ (which can be derived using techniques from section IV, as this entity is simply conditionally linear Gaussian).

In many practical applications, one is only interested in the distribution of the state x_t and so one can integrate over the random variable $A_{1:t}$ as follows:

$$p(x_t|y_{1:t}) = \sum_{a_{1:t}} p(a_{1:t}|y_{1:t}) p(x_t|a_{1:t}, y_{1:t}). \tag{21}$$

Similarly, one can estimate

$$p(x_t, a_{1:t}|y_{1:t}) = p(a_{1:t}|y_{1:t}) p(x_t|a_{1:t}, y_{1:t}).$$

As time increases, the number of mixture components in the joint filtering distribution (20) grows exponentially. The use of approximation methods is therefore paramount to retain a computationally feasible calculation. A popular technique in the tracking literature to perform such an approximation is to use mixture reduction (potentially reducing components with the same lagged-transition history)[4], resulting in the following:

$$p(x_t, a_t|y_{1:t}) \approx \sum_{a_{\{t-1\}}} p(a_{\{t-1\}}|y_{1:t}) p(x_t|a_{\{t-1\}}, y_{1:t}).$$

The notation $a_{\{t-1\}}$ represents the joint index of the states of all the transitions of the time instances at $t-1$ that have *not* been approximated (through mixture reduction or another approximation method). For example, within a tracking context, the use of the standard Generalised Pseudo-Bayes (GPB(1)) algorithm would give rise to a single component at time t (as all of the components up to this time instance would have been approximated by moment matching). Clearly, a similar phenomenon occurs for (21) and such approximation techniques can be further utilised.

B. Two-filter smoothing

A generic presentation of the methodology behind inference in a JMLS is now presented. It is the authors' belief that the material contained herein allows *any* JMLS approximation scheme to be implemented, without the complication of idiosyncrasies of such techniques. The use of an artificial smoothing algorithm is the *only* method with which one can ensure integrability of $p(y_{t:T}|x_t, a_t)$ over x_t. We have therefore omitted details of any attempt at an independent two-filter smoothing algorithm.

1) Artificial smoother: One can use the results of section II to give rise to the following:

$$p(x_t|y_{1:T}) \propto \sum_{a_t} p(x_t, a_t|y_{1:t-1}) p(y_{t:T}|x_t, a_t). \tag{22}$$

One can obtain the required quantities for the first term on the right hand side of equation (22) based on the discussion given above. All that is left to specify is the backward information quantity $p(y_{t:T}|x_t, a_t)$. This can be constructed through the use of equation (12) since the calculation of a joint prior distribution over x_t and a_t is computationally infeasible. This substitution (for the joint distribution of x_t, a_t and x_{t+1}, a_{t+1}) is left as an exercise for the reader, to conserve space. Note that one is

required to introduce an artificial prior distribution over x_t such that one can perform approximation (be it pruning, merging,...). This is because the information filtering quantity can be written as follows:

$$p(y_{t:T}|x_t, a_t) = \sum_{a_{t+1:T}} p(a_{t+1:T}|x_t, a_t, y_{t:T}) p(y_{t:T}|x_t, a_t), \tag{23}$$

where the term $p(a_{t+1:T}|x_t, a_t, y_{t:T})$ is dependent on x_t which prevents one from performing such an approximation. The use of such a prior distribution would eliminate this problem.

C. Joint distribution

The construction of the joint (state) distribution can be done in a similar manner to that outlined in sections II-D and IV-D:

$$p(x_{1:T}|y_{1:T}) = \sum_{a_{1:T}} p(a_{1:T}|y_{1:T}) p(x_T|a_{1:T}, y_{1:T}) \prod_{t=1}^{T-1} p(x_t|x_{t+1:T}, a_{1:T}, y_{1:T})$$

where

$$\begin{aligned} p(x_t|x_{t+1:T}, a_{1:T}, y_{1:T}) =& p(x_t|x_{t+1}, a_{1:t+1}, y_{1:t}) \\ \propto& p(x_t|a_{1:t}, y_{1:t}) p(x_{t+1}|x_t, a_{t+1}). \end{aligned}$$

This computation is computationally infeasible for any practical value of T, and approximation techniques will have to be utilised.

VI. Analytic approximation for general state space models

The assumptions made in the previous sections restrict the analysis of many 'real-life' statistical signal processing problems, and so one often has to resort to approximation methods. One popular technique is to approximate the possibly non-linear (and/or) non-Gaussian densities to both be (locally) linear Gaussian and so obtain the Extended Kalman Filter (EKF)[1]. A more recent development is to use a deterministic sample based approximation to ensure that the models (1)-(2) are never approximated, only the distributions under examination[28], [29], the so-called "unscented transform". We will describe both of these in turn.

An exemplar system which requires such an approximation is described as:

$$X_t = \varphi(X_{t-1}, \epsilon_t) \tag{24}$$

$$Y_t = \phi(X_t, \nu_t) \tag{25}$$

with ϵ_t and ν_t are independent and identically distributed (i.i.d.) mutually independent noise processes.

A. Filtering

1) The Extended Kalman filter: The Extended Kalman filter uses a Taylor series expansion to produce a (local) linear-Gaussian approximation of the state-transition and observation densities. A first order Taylor series expansion about the points $\hat{x}_{t-1}$ and $\hat{\epsilon}_t$ for the state and noise variables, respectively, leads to an approximation of (24) as follows:

$$X_t \approx \varphi(\hat{x}_{t-1}, \hat{\epsilon}_t) + F_{t-1}(X_{t-1} - \hat{x}_{t-1}) + G_{t-1}(\epsilon_t - \hat{\epsilon}_t)$$

where:

$$\begin{aligned} F_{t-1} =& \left. \frac{\partial \varphi(X_{t-1}, \epsilon_t)}{\partial X_{t-1}} \right|_{X_{t-1}=\hat{x}_{t-1}, \epsilon_t=\hat{\epsilon}_t} \\ G_{t-1} =& \left. \frac{\partial \varphi(X_{t-1}, \epsilon_t)}{\partial \epsilon_t} \right|_{X_{t-1}=\hat{x}_{t-1}, \epsilon_t=\hat{\epsilon}_t} \end{aligned}$$

and a similar expansion for the measurement equation about the points $\hat{x}_t$ and $\hat{\nu}_t$ provides:

$$Y_t \approx \phi(\hat{x}_t, \hat{\nu}_t) + M_t(X_t - \hat{x}_t) + N_t(\nu_t - \hat{\nu}_t),$$

where again:

$$\begin{aligned} M_t =& \left. \frac{\partial \phi(X_t, \nu_t)}{\partial X_t} \right|_{X_t=\hat{x}_t, \nu_t=\hat{\nu}_t} \\ N_t =& \left. \frac{\partial \phi(X_t, \nu_t)}{\partial \nu_t} \right|_{X_t=\hat{x}_t, \nu_t=\hat{\nu}_t}. \end{aligned}$$

Based on these approximations and the assumption that the noise processes are Gaussian, one can calculate the required moments for use in the Kalman filter equations (16) and (17):

$$\begin{aligned}
\mu^x_{t|1:t-1} &= \varphi(\hat{x}_{t-1}, \hat{\epsilon}_t) + F_{t-1}(\mu^x_{t-1|1:t-1} - \hat{x}_{t-1}) - G_{t-1}\hat{\epsilon}_t \\
P^x_{t|1:t-1} &= F_{t-1}P^x_{t-1|1:t-1}F^T_{t-1} + G_{t-1}Q_tG^T_{t-1} \\
\mu^y_{t|1:t-1} &= \phi(\hat{x}_t, \hat{\nu}_t) + M_t(\mu^x_{t|1:t-1} - \hat{x}_t) - N_t\hat{\nu}_t \\
P^y_{t|1:t-1} &= M_tP^x_{t|1:t-1}M^T_t + N_tR_tN^T_t \\
P^{xy}_{t|1:t-1} &= P^x_{t|1:t-1}M^T_t
\end{aligned}$$

By performing the state expansion about the posterior mean $\hat{x}_{t-1} = \mu^x_{t-1|1:t-1}$, and the measurement expansion about $\hat{x}_t = \mu^x_{t|1:t-1}$ (and their respective noises about zero), one obtains the standard EKF prediction/update equations which are conceptually equivalent to the Kalman filter/smoother equations discussed in section IV.

There are several strategies available to improve the posterior mode estimation by attempting to reduce the linearisation errors: the *iterated* EKF[19], where one re-linearises about the one-step smoothed estimate $\mu^x_{t-1|1:t}$ until convergence is obtained, and the use of higher order terms in the Taylor series expansion[43]. Both of these strategies, however, result in an increased computational burden. A comparison of the EKF, iterated EKF, and EKF with second order terms is provided in reference [47].

2) The Unscented transformation: The Unscented Transform (UT) has been used to conduct an approximation of the state transition and observation densities, (1) and (2) respectively, though quasi-Monte Carlo sampling[28]. The resulting filter, the *Unscented Kalman Filter* (UKF), considers a set of (*sigma*) points that are deterministically selected from the Gaussian approximation to $p(x_{t-1}|y_{1:t-1})$. These points are all propagated through the true models and the parameters of $p(x_t, y_t|y_{1:t-1})$ are then estimated from the transformed samples. The UKF algorithm selects deterministically N_s points such that:

$$N\left(\begin{bmatrix} x_{t-1} \\ \epsilon_t \\ \nu_t \end{bmatrix};\ V,\ Z\right) \approx \sum_{i=1}^{N_s} w^{(i)}_{t-1|1:t-1}\delta\left(\begin{bmatrix} x_{t-1} \\ \epsilon_t \\ \nu_t \end{bmatrix} - \begin{bmatrix} x^{(i)}_{t-1} \\ \epsilon^{(i)}_t \\ \nu^{(i)}_t \end{bmatrix}\right),$$

where

$$V = \begin{bmatrix} \mu^x_{t-1|1:t-1} \\ 0 \\ 0 \end{bmatrix} \qquad Z = \begin{bmatrix} P^x_{t-1|1:t-1} & 0 & 0 \\ 0 & Q_t & 0 \\ 0 & 0 & R_t \end{bmatrix},$$

where $\delta\left(x_t - x^{(i)}_t\right)$ denotes the delta-Dirac mass located in $x^{(i)}_t$. One chooses the location of such points according to an appropriate procedure[40]. The update stage is the same as the Kalman filter equations (16) and (17), with:

$$\begin{aligned}
\mu^x_{t|1:t-1} &= \sum_{i=1}^{N_s} w^{(i)}_{t-1|1:t-1}\varphi(x^{(i)}_{t-1}, \epsilon^{(i)}_t) \\
P^x_{t|1:t-1} &= \sum_{i=1}^{N_s} w^{(i)}_{t-1|1:t-1}\left[\varphi(x^{(i)}_{t-1}, \epsilon^{(i)}_t) - \mu^x_{t|1:t-1}\right]\left[\varphi(x^{(i)}_{t-1}, \epsilon^{(i)}_t) - \mu^x_{t|1:t-1}\right]^T \\
\mu^y_{t|1:t-1} &= \sum_{i=1}^{N_s} w^{(i)}_{t-1|1:t-1}\phi\left(\varphi(x^{(i)}_{t-1}, \nu^{(i)}_t), v^{(i)}_t\right) \\
P^y_{t|1:t-1} &= \sum_{i=1}^{N_s} w^{(i)}_{t-1|1:t-1}\left[\phi\left(\varphi(x^{(i)}_{t-1}, \epsilon^{(i)}_t), \nu^{(i)}_t\right) - \mu^y_{t|1:t-1}\right]\left[\phi\left(\varphi(x^{(i)}_{t-1}, \epsilon^{(i)}_t), \nu^{(i)}_t\right) - \mu^y_{t|1:t-1}\right]^T \\
P^{xy}_{t|1:t-1} &= \sum_{i=1}^{N_s} w^{(i)}_{t-1|1:t-1}\left[\varphi(x^{(i)}_{t-1}, \epsilon^{(i)}_t) - \mu^x_{t|1:t-1}\right]\left[\phi\left(\varphi(x^{(i)}_{t-1}, \epsilon^{(i)}_t), v^{(i)}_t\right) - \mu^y_{t|1:t-1}\right]^T,
\end{aligned}$$

Note that, under certain conditions, one is able to perform some calculations analytically and thus reduce the quasi-Monte Carlo variance within the deterministic sampling scheme[7].

B. Two-filter smoother

The use of the UT in forward-backward smoothing is not straightforward or intuitive and relies on some restrictive assumptions. Consequently, we only present the two-filter smoothing algorithm.

1) Artificial smoother: We present a novel scheme for performing unscented smoothing. Note that we limit our discussion to the use of (9) as the artificial prior distribution to appeal to the readers intuition, as the use of a general artificial prior distribution requires one to approximate a finite measure, rather than a probability measure as is usually done when one employs an unscented approximation.

One can approximate the artificial prior distribution stated in equation (9) by using an unscented approximation as follows:

$$N\left(\left[\begin{array}{c} x_t \\ \epsilon_t \end{array}\right];\ V,\ Z\right) \approx \sum_{i=1}^{N_s} w_t^{(i)} \delta\left(\left[\begin{array}{c} x_t \\ \epsilon_{t+1} \end{array}\right] - \left[\begin{array}{c} x_t^{(i)} \\ \epsilon_{t+1}^{(i)} \end{array}\right]\right),$$

where

$$V = \left[\begin{array}{c} \mu_t^m \\ 0 \end{array}\right] \qquad Z = \left[\begin{array}{cc} P_t^m & 0 \\ 0 & Q_{t+1} \end{array}\right]$$

and with μ_t^m, P_t^m denoting the mean and covariance of the (m)arginal distribution, respectively. One is then able to calculate the parameters of (the Gaussian approximation to) equation (9):

$$\begin{aligned}
\mu_{t+1}^m &= \sum_{i=1}^{N_s} w_t^{(i)} \varphi(x_t^{(i)}, \epsilon_{t+1}^{(i)}) \\
P_{t+1}^m &= \sum_{i=1}^{N_s} w_t^{(i)} \left[\varphi(x_t^{(i)}, \epsilon_{t+1}^{(i)}) - \mu_{t+1}^m\right] \left[\varphi(x_t^{(i)}, \epsilon_{t+1}^{(i)}) - \mu_{t+1}^m\right]^T \\
P_{t,t+1} &= \sum_{i=1}^{N_s} w_t^{(i)} \left[x_t^{(i)} - \mu_t^m\right] \left[\varphi(x_t^{(i)}, \epsilon_{t+1}^{(i)}) - \mu_{t+1}^m\right]^T.
\end{aligned}$$

One would expect this approximation to converge to the invariant distribution in many practical applications. One can now construct a Gaussian approximation to the (b)ackward Markov kernel, $\gamma(x_t|x_{t+1})$, using elementary statistical results:

$$\begin{aligned}
\mu_t^b &= \mu_t^m + P_{t,t+1}^m (P_{t+1}^m)^{-1} (x_{t+1} - \mu_{t+1}) \\
P_t^b &= P_t^m - P_{t,t+1}^m (P_{t+1}^m)^{-1} (P_{t,t+1}^m)^T.
\end{aligned}$$

Based on this Gaussian approximation and the Gaussian approximation to $\widetilde{p}(x_{t+1}|y_{t+1:T})$ one can construct a Gaussian approximation to the joint distribution $p(x_t, y_t|y_{t+1:T})$ by sampling from:

$$N\left(\left[\begin{array}{c} x_t \\ \nu_t \end{array}\right];\ V,\ Z\right) \approx \sum_{i=1}^{N_s} w_{t|t+1:T}^{(i)} \delta\left(\left[\begin{array}{c} x_t \\ \nu_t \end{array}\right] - \left[\begin{array}{c} x_t^{(i)} \\ \nu_t^{(i)} \end{array}\right]\right),$$

where

$$V = \left[\begin{array}{c} \mu_t^b \\ 0 \end{array}\right] \qquad Z = \left[\begin{array}{cc} P_t^b & 0 \\ 0 & R_t \end{array}\right].$$

One thus calculates the following moments:

$$\begin{aligned}
\mu_{t|t+1:T}^x &= \mu_t^b \\
\mu_{t|t+1:T}^y &= \sum_{i=1}^{N_s} w_{t|t+1:T}^{(i)} \phi\left(x_t^{(i)}, \nu_t^{(i)}\right) \\
P_{t|t+1:T}^x &= P_t^b \\
P_{t|t+1:T}^y &= \sum_{i=1}^{N_s} w_{t|t+1:T}^{(i)} \left[\phi\left(x_t^{(i)}, \nu_t^{(i)}\right) - \mu_{t|t+1:T}^y\right] \left[\phi\left(x_t^{(i)}, \nu_t^{(i)}\right) - \mu_{t|t+1:T}^y\right]^T \\
P_{t|t+1:T}^{xy} &= \sum_{i=1}^{N_s} w_{t|t+1:T}^{(i)} \left[x_t^{(i)} - \mu_{t|t+1:T}^x\right] \left[\phi\left(x_t^{(i)}, \nu_t^{(i)}\right) - \mu_{t|t+1:T}^y\right]^T,
\end{aligned}$$

such that one can then use the update equations:

$$\begin{aligned}
\mu_{t|t:T}^x &= \mu_{t|t+1:T}^x + P_{t|t+1:T}^{xy} P_{t|t+1:T}^{y\ -1} (y_t - \mu_{t|t+1:T}^y) \\
P_{t|t:T}^x &= P_{t|t+1:T}^x - P_{t|t+1:T}^{xy} P_{t|t+1:T}^{y\ -1} P_{t|t+1:T}^{xy\ T}.
\end{aligned}$$

The first and second moments of the smoothed distribution can be calculated through the intermediate step:

$$\widetilde{x}_{t|1:T} = \widetilde{P}^{x\ -1}_{t|1:T} \left(P^{x\ -1}_{t|1:t-1} \mu^x_{t|1:t-1} + P^{x\ -1}_{t|t:T} \mu^x_{t|t:T} \right)$$
$$\widetilde{P}^{x\ -1}_{t|1:T} = P^{x\ -1}_{t|1:t-1} + P^{x\ -1}_{t|t:T},$$

where one removes the artificial prior as follows:

$$\mu^x_{t|1:T} = P^{x\ -1}_{t|1:T} \left(\widetilde{P}^{-1}_{t|1:T} \widetilde{x}_{t|1:T} - P^{m\ -1}_{t-1} x^m_{t-1} \right)$$
$$P^{x\ -1}_{t|1:T} = \widetilde{P}^{-1}_{t|1:T} - P^{m\ -1}_{t-1}.$$

We remark that it is possible to propagate a probability measure on the joint space x_t, x_{t+1}. This would result in a (Gaussian) approximation of $p(x_t, x_{t+1}|y_{t:T})$, as stated in equation (12) and would eliminate the need to approximate $p(x_t)$ as is presented in this section, or to use an unscented approximation to a finite measure rather than a probability measure as previously discussed.

VII. Monte Carlo approximation for general state space models

The use of Monte Carlo (MC) methods for stochastic simulation of an analytically intractable distribution has received a great amount of attention in the past 15 years. It is the intention of this section to review a subset of the MC methodology, known as Sequential Monte Carlo (SMC) methods, commonly referred to as particle filters (and smoothers)[13].

A. Filtering

1) Importance Sampling: If one was able to obtain N i.i.d. random samples $\{X^{(i)}_{1:t};\ i = 1, \ldots, N\}$ drawn from $p(x_{1:t}|y_{1:t})$, then an estimate of the (joint) posterior density would be given by:

$$\hat{p}(x_{1:t}|y_{1:t}) = \frac{1}{N} \sum_{i=1}^{N} \delta(x_{1:t} - X^{(i)}_{1:t}).$$

It is typically impossible to sample from $p(x_{1:t}|y_{1:t})$ so instead one uses instead the importance sampling identity

$$p(x_{1:t}|y_{1:t}) = \frac{w\left(x_{1:t}\right) q(x_{1:t}|y_{1:t})}{\int w\left(x_{1:t}\right) q(x_{1:t}|y_{1:t}) dx_{1:t}}$$

where $q(x_{1:t}|y_{1:t})$ is a so-called importance distribution whose support includes the one of $p(x_{1:t}|y_{1:t})$ and

$$w\left(x_{1:t}\right) = \frac{p(x_{1:t}|y_{1:t})}{q(x_{1:t}|y_{1:t})}.$$

Given N i.i.d random samples $\{X^{(i)}_{1:t};\ i = 1, \ldots, N\}$,

$$\begin{aligned}
\hat{p}(x_{1:t}|y_{1:t}) &= \frac{w\left(x_{1:t}\right) \frac{1}{N} \sum_{i=1}^{N} \delta(x_{1:t} - X^{(i)}_{1:t})}{\int w\left(x_{1:t}\right) \frac{1}{N} \sum_{i=1}^{N} \delta(x_{1:t} - X^{(i)}_{1:t}) dx_{1:t}} \\
&= \frac{\sum_{i=1}^{N} w\left(X^{(i)}_{1:t}\right) \delta(x_{1:t} - X^{(i)}_{1:t})}{\sum_{i=1}^{N} w\left(X^{(i)}_{1:t}\right)} \\
&= \sum_{i=1}^{N} w^{(i)}_t \delta(x_{1:t} - X^{(i)}_{1:t}).
\end{aligned}$$

It follows that

$$\begin{aligned}
\mathbb{E}_{p(x_{1:t}|y_{1:t})}\left[\psi\left(x_{1:t}\right)\right] &\approx \mathbb{E}_{\hat{p}(x_{1:t}|y_{1:t})}\left[\psi\left(x_{1:t}\right)\right] \\
&= \sum_{i=1}^{N} w^{(i)}_t \psi\left(X^{(i)}_{1:t}\right).
\end{aligned}$$

Note that since this is a ratio of estimates the resulting estimate is biased. However this negligible bias is of order $\frac{1}{N}$ and this estimate is still asymptotically consistent[11].

2) Sequential Importance Sampling and Resampling: One is able to achieve a sequential update of the importance weights by (implicitly) assuming the state at the current time is independent of future measurements, thus making the algorithm recursive:

$$q(x_{1:t}|y_{1:t}) \triangleq q(x_{1:t-1}|y_{1:t-1})q(x_t|x_{t-1}, y_t). \tag{26}$$

Based on this importance function and the Markovian assumptions stated in Section I, it is possible to obtain a sequential weight update equation. Unfortunately such a factorisation (26) of the importance function leads to a phenomenon known as the degeneracy of the algorithm. That is, the unconditional variance of the importance weights increases over time; all but one of the normalised importance weights are very close to zero. To avoid this problem, the concept of 'resampling' was introduced[24] to rejuvenate the particle approximation of the posterior distribution under interest. Various schemes can be adopted to perform resampling, with the interested reader being referred to [13] and references therein. Resampling is deemed necessary when the number of effective samples, $N_{eff} = \left(\sum_{i=1}^{N}\left(w_t^{(i)}\right)^2\right)^{-1}$, falls below a pre-defined threshold. After resampling, the particles are no longer statistically independent. It has been shown, however, that under mild assumptions, the estimates one obtains are still asymptotically consistent[9], [14].

One would like, however, to minimise the number of times that resampling has to be conducted to avoid unnecessary errors being introduced into the algorithm. The choice of importance function $q(x_t|x_{t-1}, y_t)$ is therefore critical, as the closer to the posterior that this function is, the smaller is the variance of the importance weights, and the less resampling is required. The optimal choice of importance function (in terms of minimising the variance of the importance weights, conditional on $X_{1:t-1}^{(i)}$ and $y_{1:t}$) is $p(x_t|X_{t-1}^{(i)}, y_t)$. Details of the implementation of such a proposal are given in references [2], [14].

GENERIC PARTICLE FILTER

1) Initialise at time $t = 1$.
   - For $i = 1, \ldots, N$, sample $X_1^{(i)} \sim q(\cdot|y_1)$.
   - For $i = 1, \ldots, N$, compute the importance weights:

$$w_1^{(i)} \propto \frac{\mu(X_1^{(i)})g(y_1|X_1^{(i)})}{q(X_1^{(i)}|y_1)}, \sum_{i=1}^{N} w_1^{(i)} = 1.$$

2) For $t = 2, \ldots, T$.
   - For $i = 1, \ldots, N$, sample $X_t^{(i)} \sim q(\cdot|X_{t-1}^{(i)}, y_t)$
   - For $i = 1, \ldots, N$ compute the importance weights

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{g(y_t|X_t^{(i)})f(X_t^{(i)}|X_{t-1}^{(i)})}{q(X_t^{(i)}|X_{t-1}^{(i)}, y_t)}, \quad \sum_{i=1}^{N} w_t^{(i)} = 1.$$

3) Resampling
   - Evaluate the number of effective samples, N_{eff}.
   - If N_{eff} is greater than a pre-specified value then resample using an appropriate scheme[13].

B. Forward-backward smoothing

An SMC forward-backward smoothing algorithm can be derived as follows, based on the approximations to equation (4)[14]:

$$\begin{aligned}\widehat{p}(x_t|y_{1:T}) &= \sum_{i=1}^{N} w_t^{(i)} \left[\sum_{j=1}^{N} w_{t+1|T}^{(j)} \frac{f\left(X_{t+1}^{(j)}|X_t^{(i)}\right)}{\left[\sum_{l=1}^{N} w_t^{(l)} f\left(X_{t+1}^{(i)}|X_t^{(l)}\right)\right]}\right] \delta(x_t - X_t^{(i)}) \\ &\triangleq \sum_{i=1}^{N} w_{t|T}^{(i)} \delta(x_t - X_t^{(i)}),\end{aligned}$$

where $w_{t|T}^{(i)}$ denotes the (smoothed) weight of the ith particle at time t conditioned on the data $y_{1:T}$.

This particle implementation iterates recursively through the filtered posterior estimates and, without changing the support of the distribution, modifies the particle weights.

Note that the memory requirement is $O(TN)$, with the complexity being quadratic in the number of particles, $O(TN^2)$. Reference [26] introduces a rejection sampling method when constructing such an approximation using the forward-backward algorithm.

The algorithm suffers from one major problem: its reliance on the support of the filtering distribution. If an insufficient proposal distribution was used in the filtering operation, in so much as it did not represent the support of the filtering distribution accurately, then one cannot necessarily expect the representation of the smoothed distribution to be accurately modelled by these samples. This point is exemplified in the smoothing example of a non-linear time series problem examined in Section VIII.

C. Two-filter smoothing

References [27], [32] implicitly assume that $\int p(y_{t:T}|x_t)dx_t < \infty$ and develop particle methods in this context. However, if this assumption is violated, particle methods do not apply. We advocate here the use of the artificial two-filter smoother as by construction $\widetilde{p}(x_t|y_{t:T})$ defined through (7) is always a finite measure. To approximate (7) within an SMC context, one factorises a proposal distribution $\widetilde{q}(x_{t:T}|y_{t:T})$ in a similar manner to Section VII-A.2 to allow the backward information filtering algorithm to be formed, see equation (12).

INFORMATION PARTICLE FILTER

1) Initialise at time $t = T$.
   - For $i = 1, \ldots, N$, sample $\widetilde{X}_T^{(i)} \sim \widetilde{q}(\cdot|y_T)$.
   - For $i = 1, \ldots, N$, compute the importance weights:

$$\widetilde{w}_T^{(i)} \propto \frac{\gamma_T(\widetilde{X}_T^{(i)})g(y_T|\widetilde{X}_T^{(i)})}{\widetilde{q}(\widetilde{X}_T^{(i)}|y_T)}, \quad \sum_{i=1}^{N} \widetilde{w}_T^{(i)} = 1.$$

2) For times $t = (T-1), \ldots, 1$,
   - For $i = 1, \ldots, N$, sample $\widetilde{X}_t^{(i)} \sim \widetilde{q}(\cdot|\widetilde{X}_{t+1}^{(i)}, y_t)$.
   - For $i = 1, \ldots, N$, compute the importance weights:

$$\widetilde{w}_t^{(i)} \propto \widetilde{w}_{t+1}^{(i)} \frac{g(y_t|\widetilde{X}_t^{(i)})\gamma_t(\widetilde{X}_t^{(i)})f(\widetilde{X}_{t+1}^{(i)}|\widetilde{X}_t^{(i)})}{\gamma_{t+1}(\widetilde{X}_{t+1}^{(i)})\widetilde{q}(\widetilde{X}_t^{(i)}|\widetilde{X}_{t+1}^{(i)}, y_t)}, \quad \sum_{i=1}^{N} \widetilde{w}_t^{(i)} = 1.$$

3) Resampling
   - Evaluate the number of effective samples, N_{eff}.
   - If N_{eff} is greater than a pre-specified value then resample using an appropriate scheme[13].

Based on the particle estimates of the objects required to estimate the smoothed marginal posterior distribution, one can easily see that:

$$\begin{aligned}\widehat{p}(x_t|y_{1:T}) &\approx \int \left(\sum_{i=1}^{N} w_{t-1}^{(i)} \delta(x_{t-1} - X_{t-1}^{(i)}) \right) f(x_t|x_{t-1}) \frac{\left(\sum_{j=1}^{N} \widetilde{w}_t^{(j)} \delta(x_t - \widetilde{X}_t^{(j)}) \right)}{\gamma_t(x_t)} dx_{t-1} \\ &\propto \sum_{j=1}^{N} \widetilde{w}_t^{(j)} \sum_{i=1}^{N} w_{t-1}^{(i)} \frac{f\left(\widetilde{X}_t^{(j)}|X_{t-1}^{(i)}\right)}{\gamma_t\left(\widetilde{X}_t^{(j)}\right)} \delta(x_t - \widetilde{X}_t^{(j)}).\end{aligned}$$

The computational complexity of this algorithm is again quadratic in the number of particles, $O(TN^2)$, but one would expect the smoothed distribution to be better approximated, since it does not necessarily rely on the support of the approximation to the filtered density to be an accurate representation of the smoothed density. Approximation schemes such as rejection sampling can be used to reduce the computational complexity of the particle smoothing algorithm.

D. Joint Distribution

Based on the factorisation in section II, one has a particle implementation as described below[15]. An equation for the weights is simply given as:

$$w_{t|t+1}^{(i)} = \frac{w_t^{(i)} f\left(\widetilde{X}_{t+1}|X_t^{(i)}\right)}{\sum_{j=1}^{N} w_t^{(i)} f\left(\widetilde{X}_{t+1}|X_t^{(j)}\right)},$$

where $\tilde{X}_{t+1}$ is a randomly selected particle chosen from the support of the filtering distribution with weight $w_{t|t+1}^{(i)}$. The random variable $X_{1:T}$ is approximately distributed according to $p(x_{1:T}|y_{1:T})$. This procedure can be repeated to produce further (independent) realisations, as required. Note that schemes to reduce the Monte Carlo variance such as Rao-Blackwellisation have also been employed in reference [21].

It is possible to approximate the MAP sequence estimate within a particle filtering context through the use of the Viterbi algorithm. This has been applied in the filtering case in reference [23].

E. Parameter estimation using the Expectation Maximisation (EM) algorithm

In many cases of interest, the state-space model depends on unknown parameters $\theta \in \Theta$. That is,

$$X_t|X_{t-1} = x_{t-1} \sim f_\theta(\cdot|x_{t-1});$$
$$Y_t|X_t = x_t \sim g_\theta(\cdot|x_t),$$

where for the sake of simplicity we assume here that the initial distribution μ is independent of θ. To estimate this parameter given $y_{1:T}$, we propose to maximize the log-likelihood

$$\log\left(p_\theta\left(y_{1:T}\right)\right) = \log\left(p_\theta\left(y_1\right)\right) + \sum_{t=2}^{T} \log\left(p_\theta\left(y_t|\, y_{1:t-1}\right)\right).$$

A direct maximization of the likelihood is difficult and we instead use the standard EM algorithm[12]. This iterative algorithm proceeds as follows: given a current estimate $\theta^{(i-1)}$ of θ then

$$\theta^{(i)} = \arg\max_{\theta\in\Theta} Q\left(\theta^{(i-1)}, \theta\right)$$

where

$$\begin{aligned} Q\left(\theta^{(i-1)}, \theta\right) &= \mathbb{E}_{\theta^{(i-1)}}\left[\log\left(p_\theta\left(x_{1:T}, y_{1:T}\right)\right)|\, y_{1:T}\right] \\ &= \int \log\left(p_\theta\left(x_{1:T}, y_{1:T}\right)\right) p_{\theta^{(i-1)}}\left(x_{1:T}|\, y_{1:T}\right) dx_{1:T}. \end{aligned} \tag{27}$$

The EM algorithm guarantees that $p_{\theta^{(i)}}\left(y_{1:T}\right) \geq p_{\theta^{(i-1)}}\left(y_{1:T}\right)$[4] making it a popular and intuitively appealing inference technique.

When the complete data distribution is from the exponential family, then computing $Q\left(\theta^{(i-1)}, \theta\right)$ only requires evaluating expectations of the form

$$\mathbb{E}_{\theta^{(i-1)}}\left[\varphi_1\left(x_t\right)|\, y_{1:T}\right],\ \mathbb{E}_{\theta^{(i-1)}}\left[\varphi_2\left(x_{t-1}, x_t\right)|\, y_{1:T}\right] \text{ and } \mathbb{E}_{\theta^{(i-1)}}\left[\varphi_3\left(x_t, y_t\right)|\, y_{1:T}\right].$$

This can be done using any smoothing technique approximating the marginal distributions $p\left(x_t|\, y_{1:T}\right)$ and $p\left(x_{t-1:t}|\, y_{1:T}\right)$. In the particular case of general state-space models, we recommend using the two-filter formula. The maximization of $Q\left(\theta^{(i-1)}, \theta\right)$ with respect to θ can be typically performed analytically. Experimental results from the application of this technique can be found in Section VIII.

VIII. Experimental Results

A. Jump Markov linear system

In several problems related to seismic signal processing and nuclear science [8], [39], the signal of interest can be modelled as the output of a linear filter excited by a Bernoulli-Gaussian (BG) process and observed in white Gaussian noise. We therefore provide an example of the detection of a Bernoulli-Gauss process related to the aforementioned application domain. The input sequence is $\nu_t \sim \lambda N(0, \sigma_\nu^2) + (1 - \lambda_0),\ 0 < \lambda < 1$, and the observation noise is $\epsilon_t' \sim N(0, \sigma_w^2)$. ν_t' and ϵ_t' are mutually independent sequences. The linear filter is modelled by and AR(2) model. Thus, we have $S = \{1, 2\}$, and the signal admits the following state-space model:

$$B = \begin{pmatrix} a_1 & a_2 \\ 1 & 0 \end{pmatrix},\ H = (10)$$

[4]If $Q\left(\theta^{(i-1)}, \theta\right)$ is evaluated numerically using particle methods, then we cannot ensure this property.

$$Q(1) = (\sigma_\nu^2 \;\; 0)^T,\; Q(2) = (0 \;\; 0)^T,\; R = \sigma_w^2.$$

In the following simulations, we set the parameters to $a_1 = 1.51$, $a_2 = -0.55$, and $\sigma_\nu = 0.50$, $\sigma_w = 0.25$. $T = 250$ observations are generated and the exemplar data set is shown in Figure 1.

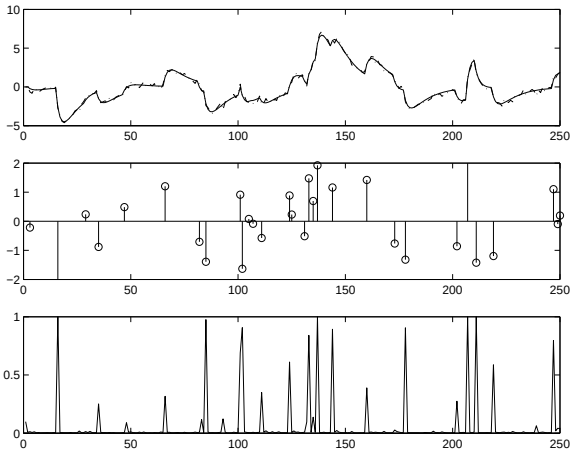


Fig. 1. Top: simulated signal X_t (solid line) and observations Y_t (dotted line). Middle: simulated sequence ν_t'.. Bottom: smoothed posterior estimates $p(A_t = 1|y_{1:T})$.

The results show that it is possible to employ any mixture component reduction approximation technique such that one has a reasonable computational expense without compromising the fidelity of the estimate. The scheme employed here was to perform mixture reduction at each time instance using the GBP methodology, which is similar in nature to that performed in reference [25], although one can utilise any approximation scheme such as retaining the N-top weighted components or sampling based approaches, for example, using the material presented.

B. Non-linear time series

Consider the time series problem[24], [32]:

$$X_{t+1} = \frac{1}{2}X_t + 25\frac{X_t}{1+X_t^2} + 8\cos(1.2t) + v_{t+1} \tag{28}$$

$$Y_t = \frac{X_t^2}{20} + \omega_t, \tag{29}$$

where ν_{t+1} and ω_{t+1} are independent zero mean Gaussian noise sequences respectively. As discussed by in [24], the likelihood is bimodal when the measurements are (strictly) positive, making the inference problem difficult for Kalman filter based algorithms. We now compare the results produced using differing proposal distributions in an SMC context, since Kalman based methods are deemed inadequate for this example. An exemplar set of data are displayed in Figure 2.

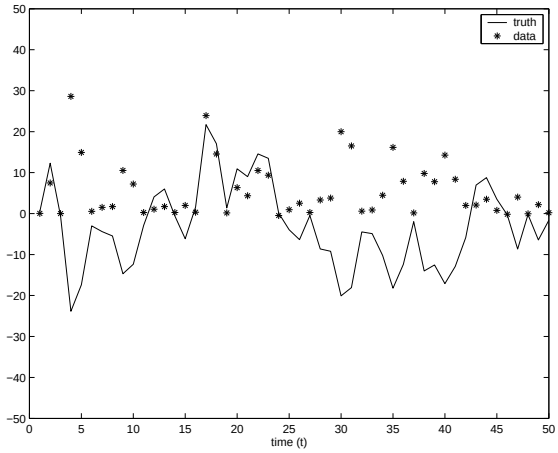


Fig. 2. Exemplar data from the model defined by equations (28)-(29). Simulated signal X_t (solid line) and observations Y_t (star)

Method	Average RMS(a)	Average RMS(b)
Filter (prior)	98.87	164.10
Forward-backward (prior)	67.0	138.57
Filter (optimal)	96.9	147.42
Forward-backward (optimal)	65.59	101.07
Information filter (Gaussian mixture)	30.40	92.40
Two-filter (optimal/Gaussian mixture)	22.53	42.98

TABLE I
AVERAGE RMS VALUES FOR 100 MONTE CARLO RUNS

To enable a comparison between the two (forward-backward and two-filter) smoothing methodologies, 100 Monte Carlo simulations were performed. Following reference [24] we used 500 particles over 50 time epochs. Two sets of experiments were conducted with different noise variances: the process noise variance and measurement noise variance were taken to be 5.0, 0.1 and 15.0, 0.001 for experiment sets (a) and (b), respectively. The measurement noise variance was taken as 0.1 for both experiments. Furthermore, we compared two proposal distributions (prior and 'optimal' - implemented using an Unscented approximation to the proposal[44]) to test the hypothesis that the support of the filtering distribution dictates the accuracy (in terms of root mean square (RMS) error) in forward-backward smoothing. Note that a (relatively) flat Gaussian distribution was used as the artificial prior in this example, which is needed since the information filtering quantity is not integrable in x_t.

The average RMS error values for the minimum mean square estimate (MMSE) for these simulations are displayed in Table I. The proposal distribution used for each of the simulations appears as a bracketed term next to each method. One can see that for experiment set (a) in the forward-backward smoothing procedure there does not appear to be any advantage of using the optimal proposal over the prior. This suggests that there is no reliance on the support of the filtering distribution for this particular example. Experiment set (b) was therefore conducted in conditions that one would recommend using the (approximate) optimal distribution. The results produced allow the conclusions sought (that is, there are cases when the support of the filtering distribution dictates the accuracy of the forward-backward smoothed estimate) to be reached. The authors suggest that the practitioner should ensure that the most suitable proposal is used as an application specific compromise; either in terms of computational reduction (clearly the prior distribution should be used in experiment set (a)) or state estimation accuracy (the optimal distribution would be the authors' choice for experiment (b)). In both sets of experiments, the two-filter smoother outperformed the forward-backward filtering algorithm, which one would expect, since the smoothed distribution is being approximated from two sets of realisations of the respective two-filter sweeps.

We now comment on the choice of importance function in the backward information filter: we choose to use a bimodal importance function that incorporates information from the measurements but is independent of the previous set of particles. Note that this is not sampling from the likelihood[36], which would involve the application of the change of variable theorem (and would require sampling from a truncated Gamma distribution after algebraic manipulations). The authors note that if one chooses the process noise distribution from which to sample for this example, then the normalisation constant for the weight update cannot be calculated (since the integral is intractable).

C. Parameter estimation

Stochastic volatility models are used extensively in the analysis of financial time series data. A natural inference problem is to estimate the parameters in such a complex model, with the EM algorithm being a popular gradient based method to perform such inference. The application of particle smoothing methods to circumvent the non-linear estimation problem when calculating expectations for the EM algorithm is now considered.

The stochastic volatility model can be written as follows:

$$\begin{aligned} X_{t+1} &= \theta_1 X_t + \theta_2 \nu_{t+1}, \;\; X_1 \sim \left(0, \frac{\theta_2^2}{1-\theta_1^2}\right) \\ Y_t &= \theta_3 \exp\left(X_t/2\right) \omega_t, \end{aligned}$$

where ν_{t+1} and ω_t are two independent standard Gaussian noise processes. The initial state, X_1, which is independent of the states at all other times, is distributed according to the invariant distribution of process evolution. Moreover, this invariant distribution is used as the pseudo-prior in the backward information particle filter. As we use the invariant distribution for $\mu(\cdot)$,

we cannot maximise Q in closed form. However, we can use a gradient algorithm initialised at the value given by:

$$\theta_1^{(i)} = \frac{\sum_{t=2}^{T} \mathbb{E}_{\theta^{(i-1)}}\left[X_{t-1}X_t \middle| y_{1:T}\right]}{\sum_{t=1}^{T-1} \mathbb{E}_{\theta^{(i-1)}}\left[X_t^2 \middle| y_{1:T}\right]}$$

$$\theta_2^{(i)} = \left((T-1)^{-1}\left(\sum_{t=2}^{T} \mathbb{E}_{\theta^{(i-1)}}\left[X_t^2 \middle| y_{1:T}\right] + \theta_1^{(i)2}\sum_{t=2}^{T} \mathbb{E}_{\theta^{(i-1)}}\left[X_{t-1}^2 \middle| y_{1:T}\right] - 2\theta_1^{(i)}\sum_{t=2}^{T} \mathbb{E}_{\theta^{(i-1)}}\left[X_{t-1}X_t \middle| y_{1:T}\right]\right)\right)^{1/2},$$

$$\theta_3^{(i)} = \left(T^{-1}\sum_{t=2}^{T} y_t^2 \mathbb{E}_{\theta^{(i-1)}}\left[\exp\left(-X_t\right) \middle| y_{1:T}\right]\right)^{1/2}.$$

This initialisation corresponds to the maximum of a modified Q function where the initial state is discarded. It should be noted that the sample-based approximations, $\hat{p}(x_{t-1}, x_t|y_{1:T}) \approx p(x_{t-1}, x_t|y_{1:T})$ and $\hat{p}(x_t, x_{t+1}|y_{1:T}) \approx p(x_t, x_{t+1}|y_{1:T})$, have the peculiar property that $\int \hat{p}(x_{t-1}, x_t|y_{1:T})dx_{t-1} \neq \int \hat{p}(x_t, x_{t+1}|y_{1:T})dx_{t+1}$. So the expectations, $\mathbb{E}_{\theta^{(i-1)}}\left[X_{t-1}X_t \middle| y_{1:T}\right]$, $\mathbb{E}_{\theta^{(i-1)}}\left[X_{t-1}^2 \middle| y_{1:T}\right]$ and $\mathbb{E}_{\theta^{(i-1)}}\left[X_t^2 \middle| y_{1:T}\right]$ must all be calculated using $\hat{p}(x_{t-1}, x_t|y_{1:T})$ alone. In the experiences of the authors, this is shown to have a significant impact on convergence.

This algorithm can be applied to the pound/dollar daily exchange rates; see [17], [18]. Using $N = 500$ particles, the results obtained were $\hat{\theta}_1 = 0.877$, $\hat{\theta}_2 = 0.1055$, and $\hat{\theta}_3 = 0.6834$, which compare favourably with those in reference [17], [18].

IX. Conclusions

This article has presented a review of the common methods used in smoothing for state space models. Consequently, the generic exposition and creation of a rigorous mathematical framework has resulted in the production of several important modifications and novel extensions which improve the robustness and efficiency of such algorithms.

Acknowledgements

The authors would like to thank the Royal Commission for the Exhibition of 1851 (www.royalcommission1851.org.uk). This research was supported by the UK MOD Corporate Research Programme through work funded by the Defence Technology Centre for Data and Information Fusion, and the Engineering and Physical Sciences Research Council (EPSRC), UK.

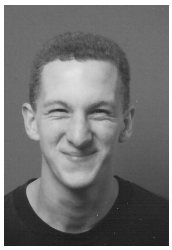
References

[1] B D Anderson and J B Moore. *Optimal Filtering*. Prentice-Hall, New Jersey, 1979.
[2] S Arulampalam, S R Maskell, N J Gordon, and T Clapp. A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2001.
[3] L R Bahl, J Cocke, F Jelinek, and J Raviv. Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Transactions on Information Theory*, IT-20:284–287, 1974.
[4] Y Bar-Shalom and X R Li. *Multitarget-Multisensor Tracking: Principles and Techniques*. Academic Press, 1995.
[5] F Bartolucci and J Besag. A recursive algorithm for Markov random fields. *Biometrika*, 89(3):724–730, 2002.
[6] Y Bresler. Two-filter formula for discrete-time non-linear Bayesian smoothing. *International Journal of Control*, 43(2):629–641, 1986.
[7] M Briers, S R Maskell, and R Wright. A Rao-Blackwellised unscented Kalman filter. In *The 6th International Conference on Information Fusion*, pages 55–61, 2003.
[8] O Cappe, A Doucet, E Moulines, and M Lavielle. Simulation-based methods for blind maximum-likelihood filter identification. *Signal Processing*, 73(1):3–25, 1999.
[9] D Crisan, P Del Moral, and T Lyons. Discrete filtering using branching and interacting particle systems. *Markov Processes Related Fields*, 5(3):293–318, 1999.
[10] P de Jong. Smoothing and interpolation with the state-space model. *Journal of the American Statistical Association*, 84(408):1085–1088, 1991.
[11] P Del Moral. *Feynman-Kac Formulae Genealogical and Interacting Particle Systems with Applications*. Springer-Verlag: New York, 2004.
[12] A P Dempster, N M Laird, and D B Rubin. Maximum-likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B*, 39, 1977.
[13] A Doucet, J F G de Freitas, and N J Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag: New York, 2001.
[14] A Doucet, S J Godsill, and C Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.
[15] A Doucet, S J Godsill, and M. West. Monte Carlo filtering and smoothing with application to time-varying spectral estimation. In *IEEE International Conference on Acoustics*, volume II, pages 701–704, 2000.
[16] A Doucet, N J Gordon, and V Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):613–624, 2001.
[17] A Doucet and V B Tadic. Parameter estimation in general state-space models using particle methods. *Annals of the Institute of Statistical Mathematical*, 55(2):409–422, 2003.
[18] J Durbin and S J Koopman. Time series analysis of non-Gaussian observations based on state space models from both classical and Bayesian perspectives (with discussion). *Journal of the Royal Statistical Society*, 62(B):3–56, 2000.
[19] L Fahrmeir. Posterior mode estimation by extended Kalman filtering for multivariate dynamic generalized linear model. *Journal of the American Statistical Association*, 87(418):501–509, 1992.
[20] P F Felzenszwalb, D P Huttenlocher, and J M Kleinberg. Fast algorithms for large-state-space HMMs with applications to web usage analysis. In *Advances in Neural Information Processing Systems*, 2003.
[21] W Fong, S J Godsill, A Doucet, and M West. Monte Carlo smoothing with application to audio signal enhancement. *IEEE Transactions on Signal Processing*, 50(2):438–448, 2002.
[22] D C Fraser and J E Potter. The optimum linear smoother as a combination of two optimum linear filters. *IEEE Transactions on Automatic Control*, pages 387–390, 1969.

[23] S J Godsill, A Doucet, and M West. Maximum a posteriori sequence estimation using monte carlo particle filters. *Annals of the Institute of Statistical Mathematics*, 51(1):82–96, 2001.
[24] N J Gordon, D J Salmond, and A F M Smith. Novel approach to nonlinear/non-gaussian Bayesian state estimation. *IEE Proceedings - F*, 140(2):107–113, 1993.
[25] R E Helmick, W D Blair, and S A Hoffman. Fixed-interval smoothing for Markovian switching systems. *IEEE Transactions on Information Theory*, 41(6):1845–1855, 1995.
[26] M Hürzler and H R Künsch. Monte Carlo approximations for general state space models. *Journal of Computational and Graphical Statistics*, 7:175–193, 1998.
[27] M Isard and A Blake. A smoothing filter for condensation. *Proceedings of 5th European Conference on Computer Vision*, 1:767–781, 1998.
[28] S Julier and J K Uhlmann. A new extension of the Kalman filter to nonlinear systems. *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls*, 1997.
[29] S J Julier and J K Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004.
[30] R E Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, pages 35–46, 1960.
[31] G Kitagawa. The two-filter formula for smoothing and an implementation of the gaussian-sum smoother. *Annals of the Institute of Statistical Mathematics*, 46(4):605–623, 1994.
[32] G Kitagawa. Monte Carlo filter and smoother for non-Gaussian non-linear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.
[33] R Kohn and C F Ansley. Filtering and smoothing algorithms for state space models. *Computers Mathematical Applications*, 18(6/7):515–528, 1989.
[34] S J Koopman. Disturbance smoother for state space models. *Biometrika*, 80(1):117–126, 1993.
[35] S R Maskell. *Sequentially Structured Bayesian Solutions*. PhD thesis, University of Cambridge, 2003.
[36] S R Maskell, M Briers, and R Wright. Tracking using a radar and a problem specific proposal distribution in a particle filter. In *IEE Symposium on Target Tracking: Algorithms and Applications*, 2004.
[37] D Q Mayne. A solution of the smoothing problem for linear dynamic systems. *Automatica*, 4:73–92, 1966.
[38] J M Mendel. White noise estimators for seismic data processing in oil exploration. *IEEE Transactions on Automatic Control*, AC-22(5):694–706, 1977.
[39] J M Mendel, editor. *Maximum-Likelihood Deconvolution: A Journey into Model-Based Signal Processing*. Springer-Verlag: New York, 1990.
[40] H Niederreiter and P J Shiue, editors. *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*. Springer-Verlag: New York, 1995.
[41] M Park and D J Miller. Low-delay optimal MAP state estimation in HMM's with application to symbol decoding. *IEEE Signal Processing Letters*, 4(10):289–292, 1997.
[42] L R Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.
[43] H W Sorenson and A R Stubberud. Recursive filtering for systems with small but non-negligible nonlinearities. *International Journal of Control*, 7:271–280, 1968.
[44] R van der Merwe, A Doucet, J F G de Freitas, and E Wan. The unscented particle filter. In T K Leen, T G Dietterich, and V Tresp, editors, *Advances in Neural Information Processing Systems 13*. MIT Press, 2000.
[45] A J Viterbi. Error bounds for convolution code and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, IT-13:260–269, 1967.
[46] M West and J Harrison. *Bayesian Forecasting and Dynamic Models*. Springer-Verlag: New York, 1997.
[47] R P Wishner, M Athans, and J A Tabaczynski. A comparison of three filters. *Automatica*, 5:487–496, 1969.

Mark Briers joined QinetiQ after receiving a First Class (Hons) degree in Mathematics in July 2001. As part of his undergraduate studies, Mark was awarded the Dr. David Whiteman prize for "outstanding work" and was also presented with the Institute of Mathematics and its Applications (IMA) prize. In September 2003 he was awarded one of six prestigious Royal Commission for the Exhibition of 1851 Industrial fellowships, which funds his work at QinetiQ towards his PhD from Cambridge University Engineering Department. He has authored several international conference papers and several technical reports.

Arnaud Doucet was born in Melle, France, on November 2, 1970. He received the M.S. degree from the Institut National des Telecommunications in 1993 and the Ph.D. degree from University Paris XI Orsay in December 1997. From 1998 to 2000, he was a research associate in the Signal Processing group of Cambridge University. In 2001-2002 he was a senior lecturer in the department of Electrical and Electronic Engineering of Melbourne University, Australia. Since September 2002, he is University Lecturer in Information Engineering in Cambridge University. He is a member of the IEEE Signal Processing Theory and Methods committee and an associate editor for The Annals of The Institute of Statistical Mathematics and Test. His research interests include simulation-based methods for estimation and control. He has co-edited with J.F.G. de Freitas and N.J. Gordon "Sequential Monte Carlo Methods in Practice", New York: Springer-Verlag, 2001.



Simon Maskell is a PhD graduand and holds a first class degree with Distinction from Cambridge University Engineering Department. His PhD was funded by one of six prestigious Royal Commission for the Exhibition of 1851 Industrial fellowships awarded on the basis of outstanding excellence to researchers working in British industry. At QinetiQ, he is a lead researcher for tracking in the Advanced Signal and Information Processing group, ASIP. As such, he leads on several projects and co-ordinates ASIP's tracking research while also supervising a number of other researchers. Simon has authored a number of papers, as well as several technical reports and a patent. He has also been the leading force behind the development of a QinetiQ product to provide a generic solution to all of QinetiQ's tracking problems.