Particle Smoothing Algorithms for Variable Rate Models

Pete Bunch*, *Member, IEEE,* Simon Godsill, *Member, IEEE,*

***Abstract*—Standard state-space methods assume that the latent state evolves uniformly over time, and can be modelled with a discrete-time process synchronous with the observations. This may be a poor representation of some systems in which the state evolution displays discontinuities in its behaviour. For such cases, a variable rate model may be more appropriate; the system dynamics are conditioned on a set of random changepoints which constitute a marked point process. In this paper, new particle smoothing algorithms are presented for use with conditionally linear-Gaussian and conditionally deterministic dynamics. These are demonstrated on problems in financial modelling and target tracking. Results indicate that the smoothing approximations provide more accurate and more diverse representations of the state posterior distributions.**

***Index Terms*—Bayesian inference, state-space model, variable rate, particle filter, filtering, smoothing**

I. Introduction

THE objective of sequential Bayesian inference is to estimate an imperfectly observed quantity as it varies over time. This is accomplished through the use of probabilistic models for the state evolution and measurement processes. Often, the latent state is a continuously varying quantity, whereas the observations are made at a discrete set of times. In these circumstances, it is simplest to discretise the state onto the same set of times as the observations. When the system is also Markovian, this leads to the standard "fixed rate" hidden Markov model (HMM). The standard HMM is poorly suited to systems where the state evolution contains discontinuities; for example, the price of a financial asset which may display large jumps at random times between periods of diffusion-like behaviour, or the kinematic state of a manoeuvring vehicle which may have sudden changes in the acceleration when turns begin or end. Such problems can be handled more naturally using a "variable rate" model, in which the state dynamics are conditioned upon a set of changepoints which characterise transitions in behaviour.

In a variable rate setting, changepoints and associated parameters are modelled as a marked point process (MPP), the mathematical properties of which are explored comprehensively in [1]. Conditional upon this MPP the state evolves according to some benign dynamics. In [2]–[4], the conditional state evolution is treated as deterministic, while in [5], [6] a conditionally linear-Gaussian state model is considered.

The posterior distribution for the changepoint MPP is inherently nonlinear, and cannot be calculated analytically. Instead, inference may be conducted using numerical approximations. The particle filter (introduced by [7]) and smoother (see [8], [9]) are schemes which approximate a posterior distribution using a set of samples, or "particles", drawn sequentially from it. A thorough introduction to particle filtering and smoothing methods can be found in [10], [11]. In [2], [3], [5], the particle filter was adapted for use with variable rate models, resulting in the variable rate particle filter (VRPF).

The VRPF allows the changepoint sequence — and hence the current state — to be estimated sequentially as observations are received. However, estimates can often be improved later once further observations have been made. In this paper, we address the problem of fixed-interval smoothing in variable rate models, i.e. the estimation of the changepoint sequence and latent state given all the observations. We achieve this with an efficient backward sweep through the observations, in a new version of the method for standard HMMs described in [9]. Two new schemes are introduced: one for conditionally linear-Gaussian models which exploits the method of Rao-Blackwellisation; and a second for use with conditionally-deterministic models which uses an augmented target distribution in the style of an SMC sampler [12].

We introduce the general structure of variable rate models in section II. The VRPF is reviewed in section III, and new variable rate smoothing algorithms are described in section IV. In section V, algorithm performance is demonstrated through simulations.

II. Variable Rate Models

We consider a general model from time 0 to T, between which observations, $\{y_1 \dots y_N\}$, are made at times $\{t_1 \dots t_N = T\}$. During this period, an unknown number of changepoints, K, occur at times $\{\tau_0 = 0, \tau_1 \dots \tau_K\}$, each with associated changepoint parameters, $\{u_0, u_1 \dots u_K\}$. The pairs $\{\tau_k, u_k\}$ are the elements of a marked point process (MPP). The latent state is a continuous-time process denoted $x(t)$. Discrete sets containing multiple values over time will be written as, e.g. $y_{1:n} = \{y_1 \dots y_n\}$.

The objective for inference will be to estimate the changepoint sequence. This will be denoted as $\theta = \{\tau_{0:K}, u_{0:K}\}$. At a particular time t_n, the sequence will be divided into past $\theta_n^- = \{\tau_j, u_j \forall j : 0 \leq \tau_j < t_n\}$, and future $\theta_n^+ = \{\tau_j, u_j \forall j : t_n \leq \tau_j < T\}$. It will also be useful to define a variable for the changepoints which occur in the interval $[t_{n-1}, t_n)$, $\theta_{n \backslash n-1} = \{\tau_j, u_j \forall j : t_{n-1} \leq \tau_j < t_n\}$.

For notational simplicity, the following counting variables are introduced to keep track of the most recent changepoint to have occurred,

$$K(t) = \max(k : \tau_k < t) \tag{1}$$

$$K_n = K(t_n). \tag{2}$$

P. Bunch and S. Godsill are with the Department of Engineering, Cambridge University, UK. email: {pb404,sjg30}@cam.ac.uk

The changepoint sequence is assumed to be a Markov process,

$$\{\tau_k, u_k\} \sim p(\tau_k, u_k|\tau_{k-1}, u_{k-1}). \tag{3}$$

This density will be constructed such that $P(\tau_k < \tau_{k-1}) = 0$. As in [3], a survivor function is defined as the probability that no new changepoint occurs before a given time,

$$\begin{aligned} S(\tau_k, u_k, t) &= P(\tau_{k+1} > t|\tau_k, u_k) \\ &= 1 - \int_{\tau_k}^{t} p(\xi|\tau_k, u_k)d\xi. \end{aligned} \tag{4}$$

It is now possible to write down a prior for the changepoint sequence. This comprises a density term for each changepoint in the sequence and a survivor function term accounting for the probability that no additional changepoints occur within the time interval. We use the convention that $\tau_0 = 0$.

$$p(\theta_n^-) = S(\tau_{K_n}, u_{K_n}, t_n)p(u_0)\prod_{k=1}^{K_n} p(\tau_k, u_k|\tau_{k-1}, u_{k-1}) \tag{5}$$

The existence of such a density for a MPP is addressed in [1]. The number of changepoints, K_n, is not fixed. If u_k varies over $\mathcal{U}$, and τ_k varies over the real line $\mathbb{R}$, then the space spanned by θ_n^- is the union,

$$\Theta = \bigcup_k \Theta_k \tag{6}$$

$$\Theta_k = \{k\} \times \mathbb{R}^k \times \mathcal{U}^k. \tag{7}$$

This closely resembles the variable-dimension spaces employed by reversible jump Markov chain Monte Carlo (MCMC) algorithms [13].

A. Conditionally Linear-Gaussian Models

We first consider the class of variable rate models whose state dynamics are linear-Gaussian conditional on the changepoint sequence. Such a model may be discretised onto the set of observation times in exactly the same manner as a standard HMM,

$$x_n = A_n(\theta_n^-)x_{n-1} + w_n \tag{8}$$

$$y_n = C_n(\theta_n^-)x_n + v_n, \tag{9}$$

where w_n and v_n are independent Gaussian variables (although the independence condition may be relaxed [14]),

$$w_n \sim \mathcal{N}(w_n|0, Q_n(\theta_n^-)) \tag{10}$$

$$v_n \sim \mathcal{N}(v_n|0, R_n(\theta_n^-)). \tag{11}$$

In addition, the prior state distribution should be Gaussian, with known mean and covariance,

$$x_0 \sim \mathcal{N}(x_0|m_0, P_0). \tag{12}$$

If the changepoint sequence is known, or has been estimated, then the state posterior distributions may be inferred conditionally using optimal Kalman filtering and smoothing recursions. As well as the basic Kalman filter [15], the Rauch-Tung-Striebel (RTS) smoother [16] and two-filter smoother [17] will prove useful for this step later in the paper.

A Jump-Diffusion model for Financial Assets: Conditionally linear-Gaussian models for use with variable rate particle filters were introduced in [5], [6] for financial applications. A version of these models is used for testing the new smoothing algorithm. Prices of an asset are treated as noisy observations of a latent state, which evolves according to a drift-diffusion with occasional jumps.

The state is a vector with two elements, the underlying value of the asset, and the trend followed by this value,

$$x(t) = [z(t), \dot{z}(t)]^T. \tag{13}$$

This evolves continuously according to a jump-diffusion model,

$$dx(t) = \begin{bmatrix} 0 & 1 \\ 0 & -\lambda \end{bmatrix} x(t)dt + \begin{bmatrix} 0 \\ \sigma \end{bmatrix} d\beta(t) + dJ(t) \tag{14}$$

where λ introduces a mean reversion effect on the trend and $\beta(t)$ is standard Brownian motion (with unit diffusion constant). The jump term, $dJ(t)$, is zero everywhere except when jumps occur,

$$dJ(t) = \begin{cases} J_k & t \in \{\tau_k\} \\ 0 & \text{elsewhere} \end{cases} \tag{15}$$

$$J_k \sim \mathcal{N}(J_k|0, Q_{J,u_k}). \tag{16}$$

$\{\tau_k\}$ is the set of random jump times, each of which is one of two types: a value jump, indicated by $u_k = 1$, or a trend jump, indicated by $u_k = 2$. The jump covariance matrices are,

$$Q_{J,u_k} = \begin{cases} \begin{bmatrix} \sigma_{J1}^2 & 0 \\ 0 & 0 \end{bmatrix} & u_k = 1 \\ \begin{bmatrix} 0 & 0 \\ 0 & \sigma_{J2}^2 \end{bmatrix} & u_k = 2 \end{cases}. \tag{17}$$

Only the value is observed, and that corrupted by additive Gaussian noise with standard deviation σ_y^2.

This model may be discretised at the observation times using standard methods, detailed in appendix A, leading to,

$$x_n = Ax_{n-1} + w_n \tag{18}$$

$$y_n = Cx_n + v_n \tag{19}$$

where the w_n and v_n are Gaussian random variables with covariance matrixes Q_n and R respectively. Q_n is a function of the sequence of jump times and their types, giving rise to the desired conditionally linear-Gaussian structure.

B. Conditionally Deterministic Models

Next we examine the class of variable rate models in which the state is completely specified by the changepoint sequence, with no additional random components. Such a process is commonly referred to as "piecewise-deterministic", as the latent state follows a deterministic path between changepoints. In this case, it is not necessary to discretise the state — it may be kept as a continuous variable.

In general, the state dynamics will be governed by a differential equation which may depend on the entire changepoint sequence. Here we assume that only the most recent changepoint determines the future state trajectory,

$$dx(t) = \varphi(x(t), \tau_{K(t)}, u_{K(t)})dt. \tag{20}$$

By introducing a new sequence, $\{x_0, x_1 \dots x_K\}$, which denotes the value of the state at each changepoint (i.e. $x_k = x(\tau_k)$), and assuming that an analytic solution exists, a piecewise function describing the path may be found,

$$x(t) = f(x_{K_n}, u_{K_n}, \tau_{K_n}, t), \tau_{K_n} < t \leq \tau_{K_n+1}. \tag{21}$$

By choosing $t = \tau_{K_n+1}$, this equation specifies the state at the next changepoint time. Similarly, by choosing $t = t_n$, the state at the observation times may be evaluated — these latter points will be denoted $\hat{x}_n$. To complete the framework, a probabilistic measurement model must be devised for the observation process, $p(y_n|\hat{x}_n)$.

For convenience, we assume that x_0 is known in the following sections. This means that $x(t)$ may be calculated deterministically for all t given θ. This condition is easily relaxed by including x_0 as a random variable in the posterior distribution.

An Intrinsic Coordinate Model for Manoeuvring Vehicles: Target tracking algorithms are commonly based upon fixed rate models (see, e.g. [18] for a thorough survey), in which the target kinematics (position, velocity, etc.) are estimated at a set of fixed times at which observations (e.g. radar measurements) are made. In [2]–[5], [19], variable rate models were introduced for tracking. The state trajectory is divided up by a set of changepoints, between which the motion follows a deterministic path governed by motion parameters (accelerations, etc.) which are fixed for that division. In this paper we use the 2-dimensional intrinsic coordinate model of [19] to test the new smoothing algorithm.

The target is modelled as a particle subject to perpendicular forces tangential and normal to the direction of motion. These forces are held constant between changepoint times, $\tau_{1:K}$, which represent the beginnings and ends of manoeuvres. The state vector consists of Cartesian position, $x_1(t)$ and $x_2(t)$, plus heading angle, $\psi(t)$, and speed, $\dot{s}(t)$,

$$x(t) = [x_1(t), x_2(t), \psi(t), \dot{s}(t)]^T. \tag{22}$$

In addition to the tangential and normal accelerations, $a_{T,k}$ and $a_{N,k}$, we introduce two linear velocity terms, $d_{X,k}$ and $d_{Y,k}$. As will become apparent later, these are necessary for the operation of an efficient smoothing algorithm. They provide us with enough degrees of freedom to match the state trajectory between an arbitrary past changepoint ($\tau_{K(t)}$, $\mathbf{x}_{K(t)}$) and an arbitrary future changepoint ($\tau_{K(t)+1}$, $\mathbf{x}_{K(t)+1}$). These linear velocity parameters may be considered to be merely relaxation terms, or they may represent real physical effects, such as wind (for aircraft) or currents (for boats). Together, these four variables make up the vector of motion parameters,

$$u_k = [a_{T,k}, a_{N,k}, d_{X,k}, d_{Y,k}]^T. \tag{23}$$

The target dynamics are described by four differential equations,

$$\ddot{s}(t) = a_{T,K(t)} \tag{24}$$

$$\dot{s}(t)\dot{\psi}(t) = a_{N,K(t)} \tag{25}$$

$$\dot{x}_1(t) = \dot{s}(t)\cos(\psi(t)) + d_{X,K(t)} \tag{26}$$

$$\dot{x}_2(t) = \dot{s}(t)\sin(\psi(t)) + d_{Y,K(t)}. \tag{27}$$

Solving these yields the following state equations (where $\Delta\ t = t - \tau_{K(t)}$),

$$\dot{s}(t) = \dot{s}_{K(t)} + a_{T,k}\Delta t \tag{28}$$

$$\psi(t) = \psi_{K(t)} + \frac{a_{N,k}}{a_{T,k}} \log\left(\frac{\dot{s}(t)}{\dot{s}_{K(t)}}\right) \tag{29}$$

$$\begin{aligned} x_1(t) &= x_{1,K(t)} + d_{X,K(t)}\Delta t \\ &+ \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2}\left[a_{N,k}\sin(\psi(t)) + 2a_{T,k}\cos(\psi(t))\right] \\ &- \frac{\dot{s}_{K(t)}^2}{4a_{T,k}^2 + a_{N,k}^2}\left[a_{N,k}\sin(\psi_{K(t)}) + 2a_{T,k}\cos(\psi_{K(t)})\right] \end{aligned} \tag{30}$$

$$\begin{aligned} x_2(t) &= x_{2,K(t)} + d_{Y,K(t)}\Delta t \\ &+ \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2}\left[-a_{N,k}\cos(\psi(t)) + 2a_{T,k}\sin(\psi(t))\right] \\ &- \frac{\dot{s}_{K(t)}^2}{4a_{T,k}^2 + a_{N,k}^2}\left[-a_{N,k}\cos(\psi_{K(t)}) + 2a_{T,k}\sin(\psi_{K(t)})\right]\ . \end{aligned} \tag{31}$$

When formulating the smoothing algorithm, it will be necessary that u_k can be calculated for any pair of adjacent changepoints, (τ_k, x_k), and (τ_{k+1}, x_{k+1}). This condition is met by the preceding model — the equations are solved in the order specified to find $a_{T,k}$ from (28), $a_{N,k}$ from (29), and finally $d_{X,K(t)}$ and $d_{Y,K(t)}$ from (30) and (31).

III. The Variable Rate Particle Filter

The variable rate particle filter (VRPF) is described in [2], [3], [5]. The objective of the algorithm is to sequentially estimate the posterior distribution of the changepoint sequence, $p(\theta_n^-|y_{1:n})$, at each time t_n, which is defined on the same space as the prior density (6). This distribution may be expanded using Bayes' rule,

$$\begin{aligned} p(\theta_n^-|y_{1:n}) &\propto p(y_n|\theta_n^-, y_{1:n-1}) \\ &\times p(\theta_{n\backslash n-1}|\theta_{n-1}^-)p(\theta_{n-1}^-|y_{1:n-1}). \end{aligned} \tag{32}$$

The transition term, $p(\theta_{n\backslash n-1}|\theta_{n-1}^-)$, has a similar form to the changepoint prior of (5) [1], but an additional survivor function is included to account for the condition that changepoints cannot occur before t_{n-1},

$$\begin{aligned} &p(\theta_{n\backslash n-1}|\theta_{n-1}^-) \\ &= S(\tau_{K_n}, u_{K_n}, t_n)/S(\tau_{K_{n-1}}, u_{K_{n-1}}, t_{n-1}) \\ &\quad \times \prod_{j:t_{n-1}\leq\tau_j<t_n} p(\tau_j, u_j|\tau_{j-1}, u_{j-1}). \end{aligned} \tag{33}$$

Practically, because changepoints will be relatively rare events in most models, it is not likely that more than one new changepoint will occur between t_{n-1} and t_n.

The target distribution of (32) cannot be calculated analytically, but may be approximated numerically, for example, by using a particle filter.

A particle filter is an algorithm for approximating a probability distribution using a set of weighted samples (or "particles") drawn from that distribution using importance sampling (IS). In this case, each particle will be a set of changepoint times and parameters.

$$\hat{p}(\theta_n^-|y_{1:n}) = \sum_j w_n^{(j)} \delta_{\theta_n^{-(j)}}(\theta_n^-) \tag{34}$$

where $\delta_x(X)$ is a unit probability mass at $X = x$. The particle filter works recursively. At the nth step, a particle, $\theta_{n-1}^{-(i)}$, is first resampled from those approximating the filtering distribution at the $(n-1)$th step, using an appropriately chosen set of proposal weights, $\{v_{n-1}^{(j)}\}$ (where $\sum_j v_{n-1}^{(j)} = 1$),

$$q(\theta_{n-1}^-) = \sum_j v_{n-1}^{(j)} \delta_{\theta_{n-1}^{-(j)}}(\theta_{n-1}^-). \tag{35}$$

The choice of weights determines the type of resampling used. The simplest choice, $v_{n-1}^{(j)} = 1/\aleph_F$ (where $\aleph_F$ is the number of filter particles) may be achieved by simply omitting this step all together and using the particles of $\hat{p}(\theta_{n-1}^-|y_{1:n-1})$. This, however, leads to degeneracy of the particle weights over time. Conventional resampling is achieved by using $v_{n-1}^{(j)} = w_{n-1}^{(j)}$. Any other choice results in an auxiliary particle filter [20]. For further discussion of resampling, see [10], [11].

Next, an extension to the changepoint sequence, $\theta_{n\backslash n-1}^{(i)}$, is sampled from an importance distribution, $q(\theta_{n\backslash n-1}|\theta_{n-1}^{-(i)}, y_n)$, and concatenated with $\theta_{n-1}^{-(i)}$ to create a proposal for $\theta_n^{-(i)}$. Finally, the particle is weighted according to the ratio of the target and proposal densities,

$$\begin{aligned} w_n^{(i)} &= \frac{p(\theta_n^{-(i)}|y_{1:n})}{q(\theta_n^{-(i)})} \\ &\propto \frac{p(y_n|\theta_n^{-(i)}, y_{1:n-1}) p(\theta_{n\backslash n-1}^{(i)}|\theta_{n-1}^{-(i)}) p(\theta_{n-1}^{-(i)}|y_{1:n-1})}{q(\theta_{n-1}^{-(i)}) q(\theta_{n\backslash n-1}^{(i)}|\theta_{n-1}^{-(i)}, y_n)} \\ &= \frac{w_{n-1}^{(i)}}{v_{n-1}^{(i)}} \times \frac{p(y_n|\theta_n^{-(i)}, y_{1:n-1}) p(\theta_{n\backslash n-1}^{(i)}|\theta_{n-1}^{-(i)})}{q(\theta_{n\backslash n-1}^{(i)}|\theta_{n-1}^{-(i)}, y_n)}. \end{aligned} \tag{36}$$

The normalisation may be enforced by scaling the weights so that they sum to 1.

For the most basic "bootstrap" [7] form of the VRPF, $\theta_{n\backslash n-1}$ may be proposed from the prior transition density (33). This can be achieved by sampling new changepoints sequentially from the transition model (3) until one falls after the current time, t_n. This final future changepoint is discarded. (This process can be thought of as sampling an entire future changepoint sequence from t_{n-1} onwards, and then marginalising those which fall after t_n.) The bootstrap proposal leads to the usual simplification of the weight formula,

$$w_n^{(i)} = \frac{w_{n-1}^{(i)}}{v_{n-1}^{(i)}} \times p(y_n|\hat{x}_n). \tag{37}$$

In [2], the following choice of proposal weights was found to work well, as it preserves low probability particles which might turn out to be good estimates at later time steps,

$$v_{n-1}^{(i)} \propto \max(1, \aleph_F w_{n-1}^{(i)}). \tag{38}$$

It only remains to consider the likelihood term required for evaluation of the importance weights, $p(y_n|\theta_n^-, y_{1:n-1})$. The form of this term depends on the particular model under consideration. In the following sections, the likelihood expressions for the conditionally linear-Gaussian and deterministic cases are considered.

A. Conditionally Linear-Gaussian Likelihoods

For the conditionally linear-Gaussian models of section II-A, the required likelihood term, $p(y_n|\theta_n^-, y_{1:n-1})$ is the predictive distribution estimated by the Kalman filter. Conveniently, the Kalman filter also provides us with an estimate of the current state given the changepoint sequence and all the preceding observations. It follows from the Gaussian dynamics and prior that these distributions are all Gaussian as well, [14],

$$p(x_n|\theta_n^-, y_{1:n-1}) = \mathcal{N}(x_n|m_n^-, P_n^-) \tag{39}$$
$$p(x_n|\theta_n^-, y_{1:n}) = \mathcal{N}(x_n|m_n, P_n) \tag{40}$$
$$p(y_n|\theta_n^-, y_{1:n-1}) = \mathcal{N}(y_n|\mu_n, S_n), \tag{41}$$

with means and variances given by the following standard recursions (dependence on θ_n^- suppressed for clarity),

$$\begin{aligned} m_n^- &= A_n m_{n-1} \\ P_n^- &= A_n P_{n-1} A_n^T + Q_n \\ \mu_n &= C_n m_n^- \\ S_n &= C_n P_n^- C_n^T + R_n \end{aligned} \tag{42}$$

$$\begin{aligned} G_n &= P_n^- C_n^T S_n^{-1} \\ m_n &= m_n^- + G_n(y_n - \mu_n) \\ P_n &= P_n^- - G_n S_n G_n^T. \end{aligned} \tag{43}$$

This completes the requirements for the particle filter, resulting in the final algorithm shown in Fig. 1.

The VRPF applied to a conditionally linear-Gaussian model estimates a posterior distribution over the changepoints only. The contribution of the linear state is calculated analytically using the Kalman recursions. This is an example of Rao-Blackwellisation (see, e.g. [21], [22]).

B. Conditionally Deterministic Likelihoods

When the conditionally deterministic models of section II-B are used, the state at observation time t_n is specified by the changepoint sequence θ_n^- (plus the initial state, x_0), using (21). Thus, the required likelihood term is simply given by,

$$p(y_n|\theta_n^-, y_{1:n-1}) = p(y_n|\hat{x}_n). \tag{44}$$

This leads to a piecewise-deterministic variable rate particle filter, as summarised in Fig. 2.

In [4], an improvement to this algorithm for conditionally-deterministic models was introduced in which the linear parts

For each i, initialise particle changepoint sequence with $\{\theta_0^{-(i)}\} \leftarrow \{\tau_0^{(i)}, u_0^{(i)}\}$, where $\tau_0^{(i)} = 0$ and $u_0^{(i)} \sim p(u_0)$.
For each i, initialise particle sufficient statistics, $m_0^{(i)}$ and $P_0^{(i)}$ with prior values.
for $n = 1 \dots N$ **do**
  for $i = 1 \dots \aleph_F$ **do**
    Sample a history $\theta_{n-1}^{-(i)} \sim \sum_j v_{n-1}^{(j)} \delta_{\theta_{n-1}^{-(j)}}(\theta_{n-1}^{-})$.
    Propose an extension $\theta_{n\backslash n-1}^{(i)} \sim q(\theta_{n\backslash n-1}|\theta_{n-1}^{-(i)})$.
    Add extension to sequence $\theta_n^{-(i)} \leftarrow \theta_{n-1}^{-(i)} \cup \theta_{n\backslash n-1}^{(i)}$.
    Predict observation mean and covariance $\mu_n^{(i)}$ and $S_n^{(i)}$ using (42).
    Calculate weight $w_n^{(i)}$ using (36).
    Update state mean and covariance $m_n^{(i)}$ and $P_n^{(i)}$ using (43).
  end for
  Scale weights such that $\sum_i w_n^{(i)} = 1$.
end for

Fig. 1. Rao-Blackwellised variable rate particle filter

For each i, initialise particle changepoint sequence with $\{\theta_0^{(i)}\} \leftarrow \{\tau_0^{(i)}, u_0^{(i)}\}$, where $\tau_0^{(i)} = 0$ and $u_0^{(i)} \sim p(u_0)$.
for $n = 1 \dots N$ **do**
  for $i = 1 \dots \aleph_F$ **do**
    Sample a history $\theta_{n-1}^{-(i)} \sim \sum_j v_{n-1}^{(j)} \delta_{\theta_{n-1}^{-(j)}}(\theta_{n-1}^{-})$.
    Propose an extension $\theta_{n\backslash n-1}^{(i)} \sim q(\theta_{n\backslash n-1}|\theta_{n-1}^{-(i)}, y_n)$.
    Add extension to sequence $\theta_n^{-(i)} \leftarrow \theta_{n-1}^{-(i)} \cup \theta_{n\backslash n-1}^{(i)}$.
    Calculate state $\hat{x}_n$ using (21).
    Calculate weight $w_n^{(i)}$ using (36).
  end for
  Scale weights such that $\sum_i w_n^{(i)} = 1$.
end for

Fig. 2. Piecewise deterministic variable rate particle filter

of the motion parameters are Rao-Blackwellised. Note that this is different to the scheme used in the previous section in which the state variables are Rao-Blackwellised.

C. Improving the Variable Rate Particle Filter

The bootstrap versions of the VRPF may perform poorly if changepoints are not obvious until significantly after they occur. For example, in a tracking example, if a jump occurs in the acceleration, but only the position is observed, then this change may not be obvious until a number of additional observations have arrived. In this case, the estimation may be improved by the introduction of resample-move (RM) steps [23]. In an RM scheme, optional Metropolis-Hastings (MH) moves are conducted to alter the particle states after the importance sampling has taken place. For variable rate models, any one of the previous changepoints, τ_k, or associated parameters, u_k, could be adjusted. Because more observations are available than when the changepoint was first proposed, it may be possible to construct more informed proposals and so move the changepoints towards regions with higher posterior probability. It is even possible to retrospectively add or remove changepoints, using reversible jump MH moves [13]. Variable rate particle filters using RM with piecewise deterministic models are described in [3], [24].

Rather than conducting the IS and MH steps separately, it is possible to combine them using the framework of SMC samplers [12]. This was suggested in [3], again for piecewise deterministic dynamics, but the extension to conditionally linear-Gaussian models is straightforward.

IV. Variable Rate Particle Smoothing

A filter conducts inference sequentially as new observations are introduced. The purpose of a smoother is to produce a revised estimate once all the observations have been made, using future values to improve upon the filter performance. Estimating changepoints online is a challenging task because the presence of a change may not be obvious until after it has happened, as discussed above. Thus, a smoothing algorithm is expected to achieve improved performance in changepoint and state estimation.

The target distribution for a variable rate smoothing algorithm is the posterior distribution over the entire changepoint sequence, $p(\theta|y_{1:N})$. This distribution is approximated by the final step of the VRPF. However, in the same manner as the fixed rate filter-smoother of [25], this approximation is likely to lack path-space diversity; because of the necessary resampling step in the filtering algorithm, the particles all share the same set of changepoints before a particular time, with variation only appearing for changepoints closer to T. For a better characterisation of the smoothing distribution, it is necessary to rejuvenate the set of particles. This is achieved here with a backward pass through the observations in an extension of the forward-filtering-backward-sampling (FFBS) method described in [9]. Since these algorithms are based on the FFBS algorithm, the convergence proofs of [9], [26] are believed to apply.

For standard HMMs, it is common to distinguish between joint smoothing distributions, $p(x_{1:N}|y_{1:N})$, and marginal smoothing distributions, $p(x_k|y_{1:N})$. Note that for variable rate models, the latter is ill defined — we need the entire sequence of changepoints — and thus algorithms for estimating marginal smoothing distributions, such as reweighting smoothers [22] and two-filter smoothers [27], [28] will not be considered here.

A. Conditionally Linear-Gaussian Smoothing

For the conditionally linear-Gaussian models of section II-A, the target distribution may be factorised as follows at time t_n,

$$p(\theta|y_{1:N}) = p(\theta_n^+|y_{1:N})p(\theta_n^-|\theta_n^+, y_{1:N}). \tag{45}$$

This suggests a sequential strategy for particle smoothing. Starting with a low diversity set of particles from $p(\theta|y_{1:N})$, we first marginalise θ_n^- by simply discarding the changepoints before t_n. New values of θ_n^- are then sampled from the conditional distribution, $p(\theta_n^-|\theta_n^+, y_{1:N})$, which is approximated using the filter particles, and the past and future sequences are concatenated. By iterating backwards over $n = N \dots 1$, a diverse set of particles approximating $p(\theta|y_{1:N})$ will be generated.

It remains to devise a scheme for sampling the conditional distribution, $p(\theta_n^-|\theta_n^+, y_{1:N})$. The simplifications used by the fixed rate particle smoother which exploit the Markovian nature of the state sequence [9] are of no help, because past changepoints are not independent of future observations in variable rate models (i.e. $p(\theta_n^-|\theta_n^+, y_{1:N}) \neq p(\theta_n^-|\theta_n^+, y_{1:n})$). However, when the state dynamics are linear-Gaussian conditional on the changepoint sequence, it is possible split the dependence on the observations into past and future by introducing the current state as an additional variable. This device was used in [29] in the derivation of the Rao-Blackwellised particle smoother for fixed rate conditionally linear-Gaussian models,

$$\begin{aligned}
&p(\theta_n^-|\theta_n^+, y_{1:N}) \\
&\quad \propto p(\theta_n^-, \theta_n^+|y_{1:N}) \\
&\quad = \int p(x_n, \theta_n^-, \theta_n^+|y_{1:N})dx_n \\
&\quad \propto \int p(y_{n+1:N}|x_n, \theta_n^-, \theta_n^+, y_{1:n})p(x_n, \theta_n^-, \theta_n^+|y_{1:n})dx_n \\
&\quad = \int p(y_{n+1:N}|x_n, \theta_n^+)p(x_n|\theta_n^-, y_{1:n})dx_n \\
&\qquad \times p(\theta_n^+|\theta_n^-)p(\theta_n^-|y_{1:n}).
\end{aligned} \tag{46}$$

The particle approximation of (34) is substituted for the filtering distribution, $p(\theta_n^-|y_{1:n})$, yielding,

$$\hat{p}(\theta_n^-|\theta_n^+, y_{1:N}) = \sum_i \tilde{w}_n^{(i)} \delta_{\theta_n^{-(i)}}(\theta_n^-), \tag{47}$$

where the backwards conditional weights are given by,

$$\begin{aligned}
\tilde{w}_n \propto w_n \int &p(y_{n+1:N}|x_n, \theta_n^+) \\
&\times p(x_n|\theta_n^{-(i)}, y_{1:n})dx_n p(\theta_n^+|\theta_n^{-(i)}).
\end{aligned} \tag{48}$$

As before, normalisation is enforced by scaling the weights so that they sum to 1.

When calculating the changepoint transition density of the future sequence, only the terms depending on the past sequence need to be considered, because the future sequence is fixed,

$$\begin{aligned}
p(\theta_n^+|\theta_n^-) &= p(\tau_{K_n+1:K}, u_{K_n+1:K}|\tau_{K_n}, u_{K_n}, \tau_{K_n+1} > t_n) \\
&\propto p(\tau_{K_n+1}, u_{K_n+1}|\tau_{K_n}, u_{K_n}, \tau_{K_n+1} > t_n).
\end{aligned} \tag{49}$$

The integral in (48) contains two terms involving the current state, x_n. The first is the familiar state posterior generated by the Kalman filter using (42) and (43),

$$p(x_n|\theta_n^{-(i)}, y_{1:n}) = \mathcal{N}(x_n|m_n^{(i)}, P_n^{(i)}). \tag{50}$$

The mean and covariance of this distribution will have been calculated during the filtering stage, and can be stored for use now in the smoother.

The second state-dependent term in (48) is the likelihood, $p(y_{n+1:N}|x_n, \theta_n^+)$. This may be regarded as an improper density over x_n, and may be calculated analytically using a backwards Kalman filter, in a similar manner to that used in the two-filter smoother [17], [29], [29]–[31]. Such a backwards Kalman filter uses the following recursions,

$$p(y_{n+1:N}|x_n, \theta_n^+) \propto \mathcal{N}(x_n|\tilde{m}_n^-, \tilde{P}_n^-) \tag{51}$$

$$p(y_{n:N}|x_n, \theta_n^+) \propto \mathcal{N}(x_n|\tilde{m}_n, \tilde{P}_n) \tag{52}$$

$$\begin{aligned}
\tilde{m}_n^- &= A_{n+1}^{-1}\tilde{m}_{n+1} \\
\tilde{P}_n^- &= A_{n+1}^{-1}(\tilde{P}_{n+1} + Q_{n+1})A_{n+1}^{-T}
\end{aligned} \tag{53}$$

$$\begin{aligned}
\tilde{\mu}_n &= C_n\tilde{m}_n^- \\
\tilde{S}_n &= C_n\tilde{P}_n^- C_n^T + R_n \\
\tilde{G}_n &= \tilde{P}_n^- C_n^T \tilde{S}_n^{-1} \\
\tilde{m}_n &= \tilde{m}_n^- + \tilde{G}_n(y_n - \tilde{\mu}_n) \\
\tilde{P}_n &= \tilde{P}_n^- - \tilde{G}_n\tilde{S}_n\tilde{G}_n^T.
\end{aligned} \tag{54}$$

Exact methods for initialising this recursion are discussed in [29], [30]. However, it is often sufficient to use an approximation, for example $\tilde{m}_N^{(i)} \leftarrow m_N^{(i)}$ and $\tilde{P}_N^{(i)} \leftarrow P_N^{(i)}$.

Substituting the Gaussian densities into (48) and integrating yields the following backwards conditional weights,

$$\tilde{w}_n \propto w_n p(\theta_n^+|\theta_n^{-(i)})\mathcal{N}(\tilde{m}_n^-|m_n, \tilde{P}_n^- + P_n). \tag{55}$$

Using these weights, a sample of θ_n^- may be drawn from the particle distribution of (47), completing one step of the smoothing algorithm. Once sampling has progressed backwards over $n = N \dots 1$, a particle from the smoothing distribution will have been generated. This procedure may then be repeated until sufficient particles have been obtained. The algorithm is summarised in Fig. 3.

The algorithmic complexity of this variable rate particle smoother for conditionally linear-Gaussian models is $O(\aleph_F \times \aleph_S \times N)$. It is possible to reduce this by using a rejection sampling [26] or MCMC [32] sampling scheme, which avoids the necessity of calculating the sampling weights for every filter particle.

Run Rao-Blackwellised variable rate particle filter to approximate $p(\theta_n^-|y_{1:n})$ with weighted particles $\{\theta_n^{-(i)}, w_n^{(i)}\}$ and $\{p(x_n|\theta_n^{-(i)}, y_{1:n})\}$ as normal distributions with moments $\{m_n^{(i)}\}$ and $\{P_n^{(i)}\}$. Store all results.
for $i = 1 \dots \aleph_S$ **do**
  Initialise particle using $\theta^{(i)} \sim \sum_j w_N^{(j)} \delta_{\theta^{(j)}}(\theta)$.
  Initialise sufficient statistics $\tilde{m}_N^{(i)}$ and $\tilde{P}_N^{(i)}$ (see text).
  for $n = N-1 \dots 1$ **do**
    Discard $\theta_n^{-(i)}$.
    Predict state mean and covariance $\tilde{m}_n^{-(i)}$ and $\tilde{P}_n^{-(i)}$ using (53).
    for $j = 1 \dots \aleph_F$ **do**
      Calculate weight $\tilde{w}_n^{(j)}$ using (55).
    end for
    Sample $\theta_n^{-(i)} \sim \sum_j \tilde{w}_n^{(j)} \delta_{\theta_n^{-(j)}}(\theta_n^-)$.
    Join $\theta^{(i)} \leftarrow \theta_n^{-(i)} \cup \theta_n^{+(i)}$.
    Update state mean and covariance $\tilde{m}_n^{(i)}$ and $\tilde{P}_n^{(i)}$ using (54).
  end for
end for

Fig. 3. Rao-Blackwellised Variable Rate Particle Smoother

B. Conditionally Deterministic Smoothing

In the conditionally deterministic models of section II-B, the entire continuous-time state trajectory, $x(t)$, is a function of the changepoint sequence θ and the initial state, x_0. When the changepoint sequence is split into past, θ_n^-, and future θ_n^+, and the past is altered, this will alter the state trajectory in the future. This is undesirable in practice, as each sampling operation will require the recalculation of state values and likelihoods for all future observation times.

In order to achieve conditional independence between past and future observations, consider the augmented changepoint sequence, $\tilde{\theta} = \{\tau_{0:K}, u_{0:K}, x_{0:K}\}$ ($\tilde{\theta}_n^-$ and $\tilde{\theta}_n^+$ defined in the same manner as before). The changepoint state values, $x_{0:K}$ may be calculated deterministically from θ for each particle. The smoother is then formulated by factorising the augmented sequence posterior distribution,

$$p(\tilde{\theta}|y_{1:N}) = p(\tilde{\theta}_n^+|y_{1:N})p(\tilde{\theta}_n^-|\tilde{\theta}_n^+, y_{1:N}). \tag{56}$$

As before, this suggests a backwards sequential sampling algorithm for smoothing. The second term may be expanded as,

$$p(\tilde{\theta}_n^-|\tilde{\theta}_n^+, y_{1:N}) \propto p(y_{n+1:N}|\tilde{\theta}_n^-, \tilde{\theta}_n^+) \times p(\tilde{\theta}_n^+|\tilde{\theta}_n^-)p(\tilde{\theta}_n^-|y_{1:n}). \tag{57}$$

A difficulty is now encountered. We cannot directly substitute the particle approximation for the filtering distribution because the transition density term for the augmented sequences is given by,

$$p(\tilde{\theta}_n^+|\tilde{\theta}_n^-) \propto p(\tau_{K_n+1}, u_{K_n+1}|\tau_{K_n}, u_{K_n}) \times \delta_{f(x_{K_n}, u_{K_n}, \tau_{K_n}, \tau_{K_n+1})}(x_{K_n+1}), \tag{58}$$

which will be 0 for all but one unique filtering particle, the one from which the next selected future changepoint originated. Intuitively, we are trying to join two deterministic state trajectories together, but they do not meet up in the middle.

The solution to this problem is provided by an idea from the Sequential Monte Carlo (SMC) samplers of [12]. Rather than sampling $p(\tilde{\theta}_n^-|\tilde{\theta}_n^+, y_{1:N})$ directly, an augmented distribution is targeted. A value of $\tilde{\theta}_n^-$ can be proposed from the filtering particles, after which a replacement, u'_{K_n}, is proposed for the existing final changepoint parameter, with a value chosen such that the past and future paths meet up. The modified sequence is denoted $\tilde{\theta}_n^{-'}$. Such a proposal density may be written as,

$$p(\tilde{\theta}_n^-|y_{1:n})q(u'_{K_n}|\tilde{\theta}_n^-, \tilde{\theta}_n^+, y_{1:N}) \propto p(y_{1:n}|\tilde{\theta}_n^-)p(\tilde{\theta}_n^-)q(u'_{K_n}|\tilde{\theta}_n^-, \tilde{\theta}_n^+, y_{1:N}). \tag{59}$$

The augmented target distribution over the the new sequence, $\tilde{\theta}_n^{-'}$, and the discarded parameter, u_{K_n}, is constructed by introducing an artificial density,

$$p(\tilde{\theta}_n^{-'}|\tilde{\theta}_n^+, y_{1:N})L(u_{K_n}|\tilde{\theta}_n^{-'}, \tilde{\theta}_n^+, y_{1:N}) \propto p(y_{1:N}|\tilde{\theta}_n^{-'}, \tilde{\theta}_n^+)p(\tilde{\theta}_n^+|\tilde{\theta}_n^{-'})p(\tilde{\theta}_n^{-'}) \times L(u_{K_n}|\tilde{\theta}_n^{-'}, \tilde{\theta}_n^+, y_{1:N}). \tag{60}$$

Clearly this new target distribution admits the desired conditional posterior, $p(\tilde{\theta}_n^{-'}|\tilde{\theta}_n^+, y_{1:N})$, as a marginal.

For this method to work, a new condition must be imposed on the state dynamics. For any pair of adjacent changepoints, (τ_k, x_k), and (τ_{k+1}, x_{k+1}) it must be possible to calculate a parameter value, u_k, which results in the transition from the former to the latter. Generally, this requires there to be as many degrees of freedom in the selection of u_k as there are state dimensions of x_k. This condition is satisfied by the tracking model of section II-B only through the introduction of the linear drift parameters.

The augmented target distribution may be sampled using Markov chain Monte Carlo (MCMC) [33], in a similar manner to the fixed rate smoother of [32]. Metropolis-Hastings (MH) steps are conducted to draw samples, $\{\tilde{\theta}_n^{-'}, u_{K_n}\}$, from the target distribution by sampling the proposal and accepting the new values with a given probability. Note that IS could be used instead of MCMC, but this would require resampling steps, which would lead to a loss of path-space diversity similar to that suffered by the particle filter.

The ratio of target (60) and proposal (59) densities is given by,

$$\beta_n \propto \frac{p(y_{r_n^-:r_n^+}|\tilde{\theta}_n^{-'})}{p(y_{r_n^-:n}|\tilde{\theta}_n^-)} \times \frac{p(u'_{K_n}|\tau_{K_n}, \tau_{K_n-1}, u_{K_n-1})}{p(u_{K_n}|\tau_{K_n}, \tau_{K_n-1}, u_{K_n-1})} \times p(\tilde{\theta}_n^+|\tilde{\theta}_n^{-'}) \times \frac{L(u_{K_n}|\tilde{\theta}_n^{-'}, \tilde{\theta}_n^+, y_{1:N})}{q(u'_{K_n}|\tilde{\theta}_n^-, \tilde{\theta}_n^+, y_{1:N})}, \tag{61}$$

where $r_n^- = \min(m : t_m > \tau_{K_n})$ and $r_n^+ = \max(m : t_m < \tau_{K_n+1})$ are the indexes of the earliest and latest observations to fall between the changepoints each side of t_n.

The proposal density $q(u'_{K_n}|\tilde{\theta}_n^-, \tilde{\theta}_n^+, y_{1:N})$ is constructed so as to be non-zero only where $p(\tilde{\theta}_n^+|\tilde{\theta}_n^{-'})$ is non-zero. This could require it to be a delta function if there is only one feasible value. The artificial conditional term, $L(u_{K_n}|\tilde{\theta}_n^-, \tilde{\theta}_n^+, y_{1:N})$, may be any arbitrary density, but the simplest choice is a uniform distribution over the set of parameter values, $\{u_{K_n}^{(i)}\}$, which appear in the filtering approximation. With this choice, the artificial density terms will always cancel out in the MH acceptance probability. The remaining terms are likelihoods and parameter transition density terms, defined by the system equations.

If $\beta_n^{(m-1)}$ is the target-proposal ratio for the current state in the chain, and β_n^* that for the new state, then the acceptance probability is given by,

$$\alpha_n^{(m)} = \min\left(1, \frac{\beta_n^*}{\beta_n^{(m-1)}}\right). \tag{62}$$

The algorithm is summarised in Fig. 4.

Each Markov chain may be initialised with the output from the previous stage. This value is itself a particle from the target distribution (albeit one from an approximation with poor path-space diversity), meaning that no burn-in period is required. However, longer chains are likely to result in less dependence between samples and a reduction in Monte Carlo error. This is the same reasoning which justifies the absence of a burn-in when using a resample-move scheme [23].

The number of MH steps, M, at each observation time, n, allows a trade-off of performance against time. Larger values of M will ensure more unique samples in the smoothing approximation, but will also take longer to execute. See [32] for analysis of this trade-off on a fixed rate model. The resulting computational complexity of the algorithm is $O(M \times \aleph_S \times N)$.

Run variable rate particle filter to approximate $p(\tilde{\theta}_n^-|y_{1:n})$ with weighted particles $\{\tilde{\theta}_n^{-(i)}, w_n^{(i)}\}$. Store all results.
for $i = 1 \dots \aleph_S$ **do**
  Initialise particle using $\tilde{\theta}^{(i)} \sim \sum_j w_N^{(j)} \delta_{\tilde{\theta}^{(j)}}(\tilde{\theta})$.
  for $n = N \dots 1$ **do**
    Initialise chain with $\tilde{\theta}_n^{-(i)(0)} \leftarrow \tilde{\theta}_n^{-(i)}$.
    for $m = 1 \dots M$ **do**
      Propose a new history $\tilde{\theta}_n^{-(i)*} \sim \sum_j w_n^{(j)} \delta_{\tilde{\theta}_n^{-(j)}}(\tilde{\theta}_n^-)$.
      Propose a parameter $u'^*_{K_n} \sim q(u_{K_n}|\tilde{\theta}_n^-, \tilde{\theta}_n^+, y_{1:N})$.
      Replace $u^*_{K_n}$ with $u'^*_{K_n}$ to form $\tilde{\theta}_n^{-'(i)*}$.
      Calculate $\alpha_n^{(m)}$ using (61) and (62).
      With probability $\alpha_n^{(m)}$, $\tilde{\theta}_n^{-(i)(m)} \leftarrow \tilde{\theta}_n^{-'(i)*}$.
      Otherwise, $\tilde{\theta}_n^{-(i)(m)} \leftarrow \tilde{\theta}_n^{-(i)(m-1)}$.
    end for
    Store final sample $\tilde{\theta}_n^{-(i)} \leftarrow \tilde{\theta}_n^{-(i)(M)}$.
  end for
end for

Fig. 4. MCMC piecewise-deterministic variable rate particle smoother

V. Simulations

The smoothers and the final step of the filters each produce a particle approximation to the changepoint sequence posterior, $p(\theta|y_{1:N})$. Calculating a single estimator from these is no simple task. Each particle consists of a list of changepoint times and parameters, and there is no obvious way to take an average over the set to produce a single estimate. Furthermore, if a best estimate could be calculated (e.g. by finding the maximum a posteriori sequence), it is not trivial to define a distance metric between the estimated and true sequences to assess relative accuracy. (One possible choice of metric has been considered for this comparison, the optimal sub-pattern assignment (OSPA) distance of [34]. However, the relative performance of the algorithms is liable to vary significantly with the parameters of the metric.)

Comparisons of accuracy are most easily conducted by comparing state estimates. These can be calculated by simply taking a mean of the particle values (using the Gaussian means for the linear-Gaussian case). The accuracy is then assessed by comparing the size of estimation errors of the resulting estimators. In addition, the benefit of employing a smoothing algorithm is that it ensures a more diverse particle representation of the posterior state distribution. To demonstrate this effect, a measure of estimator consistency is calculated by which to compare the algorithms, the mean empirical normalised estimation error squared (MENEES). This is defined as,

$$\begin{aligned}
\text{MENEES}_k &= (\hat{x}_k - x_k^*)^T \hat{P}_k (\hat{x}_k - x_k^*) \\
\hat{x}_k &= \sum_i x_k^{(i)} \\
\hat{P}_k &= \sum_i (x_k^{(i)} - x_k^*)(x_k^{(i)} - x_k^*)^T,
\end{aligned}$$

where x_k^* is the true state, and $x_k^{(i)}$ are the individual particle states. This statistic takes a value between 0 and 1, where 1 is scored by a degenerate particle estimate (fewer linearly independent particles than state dimensions). For a non-degenerate estimate, smaller values correspond to smaller estimation errors relative to the estimated covariance. Thus, it incorporates the diversity of the particle set as well the accuracy of the estimate and covariance.

In the following examples, we compare the state estimates from the "filter-smoother" [25] and the VRPS. Filter-smoother estimates use the particle approximation from the final step of the VRPF for the changepoint distribution. In the linear-Gaussian case, an RTS smoother is used to find the state posterior distribution for each particle, $p(x_n|\theta^{(i)}, y_{1:N})$, from

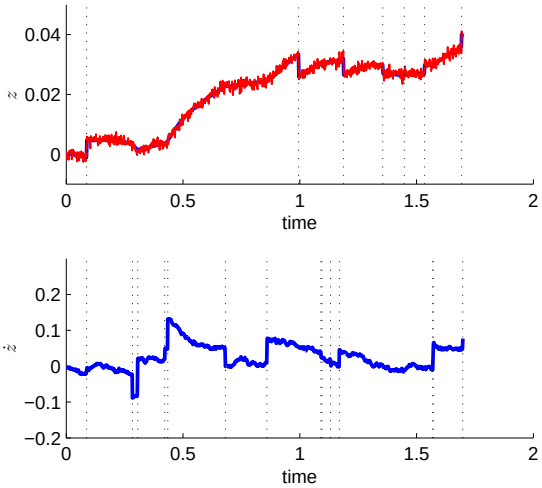


Fig. 5. An example simulated data set, showing value (top) and trend (bottom). Value observations are overlayed. Jump times are shown as dotted vertical lines.

which state estimates are derived.

A. Conditionally Linear-Gaussian Model

The Rao-Blackwellised variable rate algorithms were tested on the jump-diffusion model outlined in section II-A using artificial simulated data. The times between changepoints are assumed to be exponentially distributed with mean $1/\alpha$, and equal probability of value and trend jumps. Observations were made on a regular time grid with a spacing of $0.0017s$. Other parameters used were: $\alpha = 20$, $\lambda = 5$, $\sigma = 0.05$, $\sigma_{J1} = 0.005$, $\sigma_{J2} = 0.05$, $\sigma_y = 0.001$.

The filter used $\aleph_F = 100$ particles, and the smoother resampled $\aleph_S = 100$ sequences. Bootstrap proposals were used for the filter. An example forward realisation simulated from the model is shown in figure 5. The sampled trend trajectories produced by the filter-smoother and VRPS are shown in figure 6 (the equivalent graphs for value are not particularly informative as this quantity is observed, and the errors produced by both algorithms are small). Notice that the path diversity is dramatically improved by the VRPS.

Performance was evaluated by testing on 100 scenarios, each with 1000 observations. At each observation time, an estimate of the value and trend is made by taking the average of Gaussian means generated by an RTS smoother for each particle. The resulting mean absolute error over all scenarios is shown in figure 7, demonstrating a small improvement in accuracy from the smoother. More significantly, MENEES is shown in figure 8, showing a similar consistency between filter and smoother. The filter-smoother produces an MENEES of 1 at almost all time instants, because there is only one unique particle in the set. As well as providing more accurate and more diverse estimates, the smoother is also capable of preserving multi-modality in its representation of the posterior distribution, which is lost by the filter smoother due to the particle collapse to a single value. This is highlighted in figure 9, which shows a segment of the trend estimates from figure 6 containing a small jump. In the filter-smoother results, none of the particles have a jump at this point. In contrast, a significant number of the smoother particles contain a jump around about the right place. This ability to represent multi-modality is a key advantage of good particle smoothing algorithms.

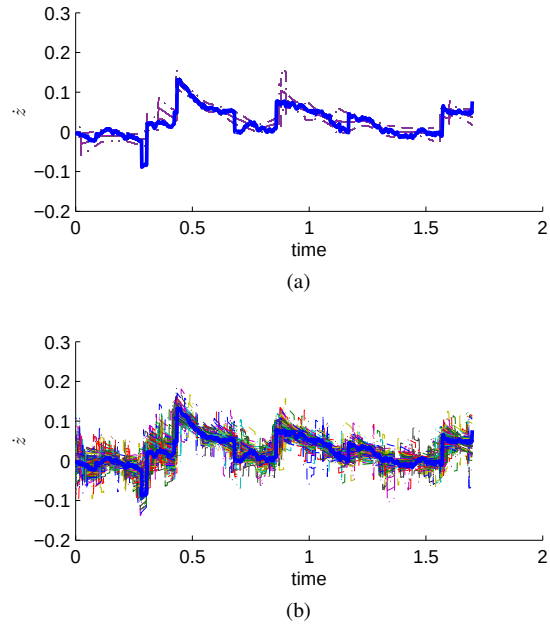


Fig. 6. Trend estimates from (a) the filter-smoother and (b) smoother. Solid lines show means of each particle. Dashed lines show mean ± 2 standard deviations.

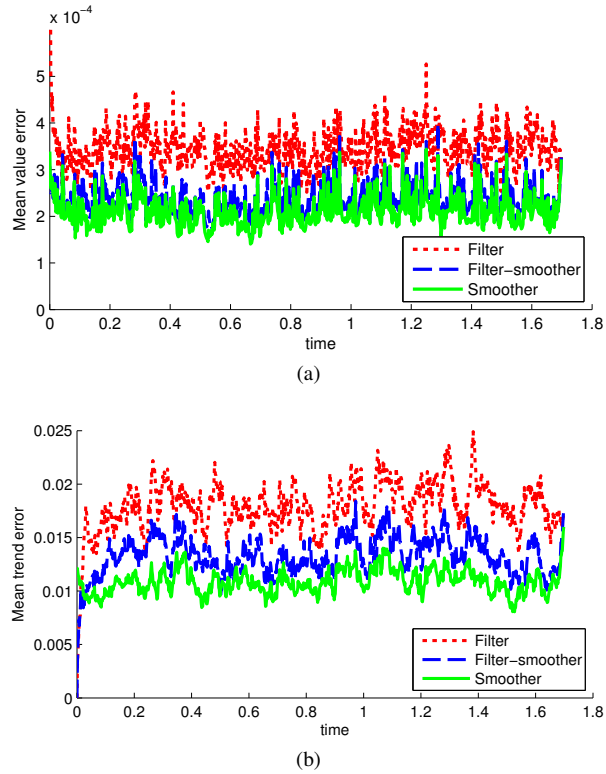


Fig. 7. Value (a) and trend (b) mean absolute error over 100 scenarios using filter, filter-smoother, and smoother.

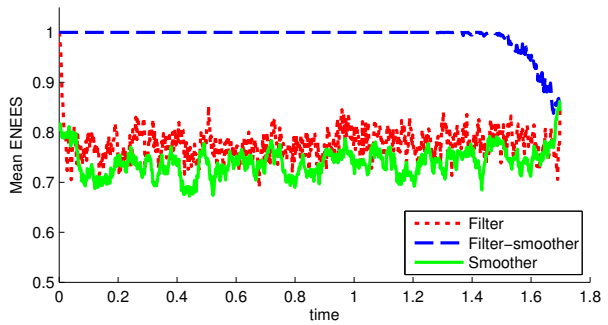
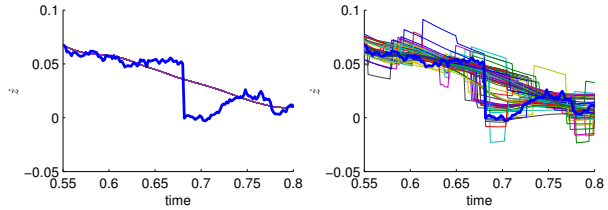


Fig. 8. MENEES over 100 scenarios using filter, filter-smoother, and smoother.



(a) (b)

Fig. 9. Filter-smoother (a) and smoother (b) estimates of trend around a jump, highlighting the ability of the smoother to represent multi-modal distributions.

B. Conditionally Deterministic Model

The piecewise-deterministic variable rate algorithms were tested on the intrinsic coordinate tracking model outlined in section II-B using artificial simulated data. The changepoint motion parameters are assumed to be independent and zero-mean Gaussian distributed. The times between changepoints are modelled as gamma distributed. Observations are made via a radar-style bearing and range measurement model with Gaussian noise.

The filtering and smoothing algorithms were tested on artificial data simulated from the model. Observations were generated every $0.1s$. The variances of tangential and normal accelerations were set to $0.01m/s^2$ and $1m/s^2$ respectively, and those for the drift velocities to $1m/s$. The parameters of the Gamma distribution for inter-changepoint times were 5 and $1s$ for the shape and scale respectively. Observation noise standard deviations were $0.1m$ and $0.25°$ respectively.

The filter and smoother were each used to generate 50 particles, with the filter employing resample-move steps with proposals based on an unscented transform [35] approximation to the optimal proposal density, and the smoother using $M = 5$ steps in each Markov chain.

An example trajectory is shown in Fig. 10, with the results from the filter-smoother and smoother in Fig.11. The improved particle diversity achieved by the smoothing algorithm is readily apparent. Performance was evaluated by testing on 100 independently generated scenarios, each with 500 observations. At each observation time, an estimate of the position and velocity is made by taking the average of values from the array of particles. The resulting mean absolute error over all scenarios is shown in figure 7, again demonstrating a small improvement in accuracy from the smoother. The smoother also provides a more accurate estimate of the number of changepoints in each test, with an average error of 0.6 compared to 2.0 from the filter-smoother. MENEES is displayed in figure 13, showing a similar consistency between filter and smoother. The filter-smoother produces an MENEES of 1 at almost all time

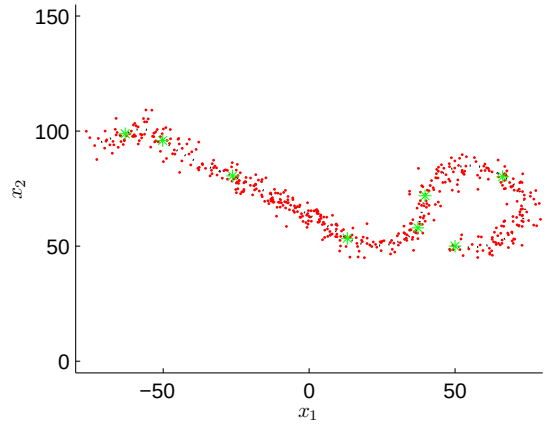
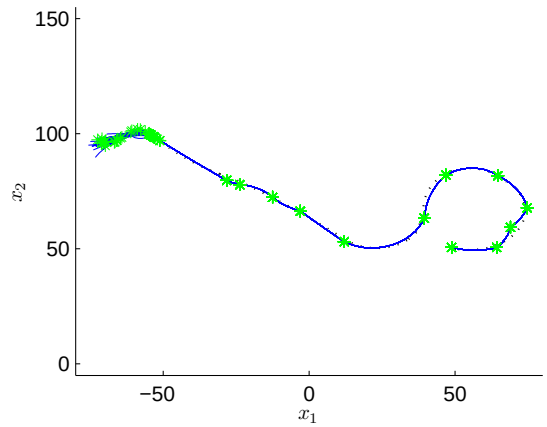
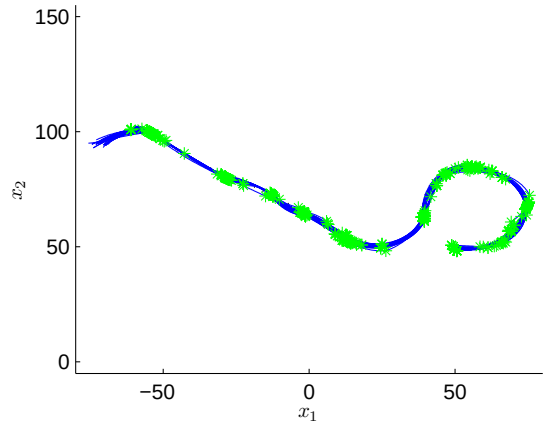


Fig. 10. An example simulated data set, showing true position (dashed line), observations (dots) and changepoint positions (crosses).

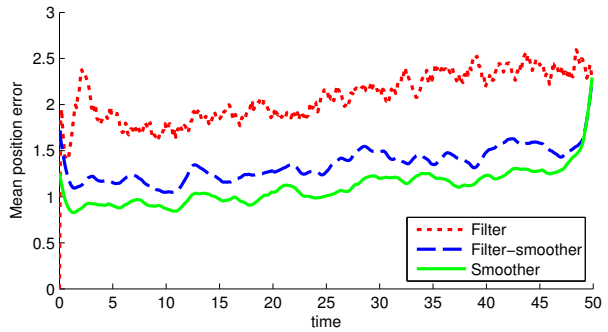


(a)

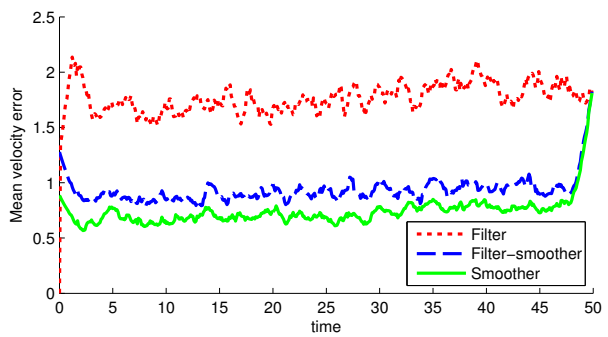


(b)

Fig. 11. Filter-smoother (a) and smoother (b) particle estimates of position (solid lines), with particle changepoint positions (crosses).



(a)



(b)

Fig. 12. Position (a) and velocity (b) mean absolute error over 100 scenarios using filter, filter-smoother, and smoother.

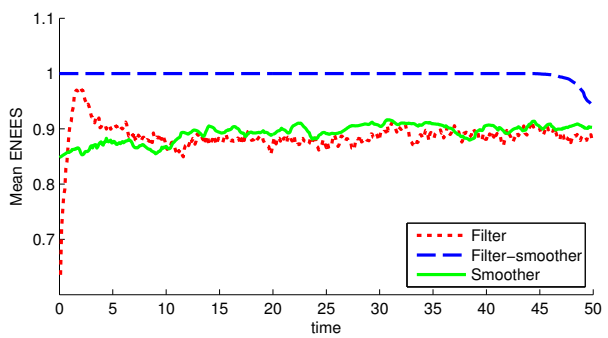
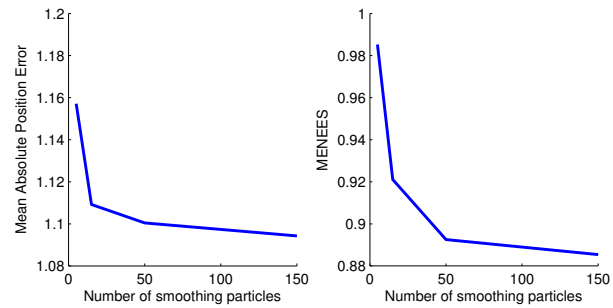


Fig. 13. MENEES over 100 scenarios using filter, filter-smoother, and smoother.

instants, because there is only one unique particle in the set. Finally, in figure 14 the variation in accuracy and consistency of the smoothing particle approximation is demonstrated as the number of smoothing particles varies. As expected, the rate of improvement diminishes and becomes negligible as the number of smoothing particles exceeds the number of filtering particles.

VI. Conclusions

New smoothing algorithms have been introduced for variable rate models which have either linear-Gaussian or deterministic dynamics conditional on a set of unknown changepoints. For the linear-Gaussian case, the algorithm employs Rao-Blackwellisation, using particle methods to estimate the nonlinear changepoint sequence, and Kalman filtering/smoothing to estimate the linear state components. For



(a) (b)

Fig. 14. Average position error (a) and MENEES (b) over 100 scenarios with varying number of smoother particles and 50 filter particles.

the deterministic case, smoothing is achieved by sampling an augmented target distribution using MCMC. Simulations demonstrate that the smoothers generate approximations with much improved particle diversity and small increases in accuracy when compared to the equivalent filter-smoothers, which produce only a degenerate particle representation.

Appendix A
Discretisation of the Jump-Diffusion Model

State evolution is governed by the following stochastic differential equation,

$$dx(t) = \begin{bmatrix} 0 & 1 \\ 0 & -\lambda \end{bmatrix} x(t)dt + \begin{bmatrix} 0 \\ \sigma \end{bmatrix} d\beta(t) + dJ(t)$$

where λ is the mean reversion constant, σ is the diffusion constant, $\beta(t)$ is standard Brownian motion, $dJ(t)$ is a jump term defined according to,

$$dJ(t) = \begin{cases} J_k & t \in \{\tau_k\} \\ 0 & \text{elsewhere} \end{cases}$$
$$J_k \sim \mathcal{N}(J_k|0, Q_{J,u_k}).$$

$\{\tau_k\}$ is the set of jump times, and $\{u_k\}$ the set of jump types, where $u_k = 1$ indicates a value jump, and $u_k = 2$ a trend jump. The jump covariance matrices are,

$$Q_{J,u_k} = \begin{cases} \begin{bmatrix} \sigma_{J1}^2 & 0 \\ 0 & 0 \end{bmatrix} & u_k = 1 \\ \begin{bmatrix} 0 & 0 \\ 0 & \sigma_{J2}^2 \end{bmatrix} & u_k = 2 \end{cases}.$$

Discretisation proceeds by separating the diffusion and jump terms and integrating separately. The jump term is straightforward to integrate. The diffusion term is a standard linear time invariant stochastic differential equation, and may be solved using the methods described in, for example, [14], [36]. Matrix fraction decomposition is used for solving the covariance propagation equation [14], [37]. The resulting discrete-time system is governed by a difference equation,

$$x_n = Ax_{n-1} + w_n$$

where the w_n is a Gaussian random variable with covariance matrix Q_n. The inter-observation time is denoted $\Delta t = t_n - t_{n-1}$, and,

$$A = \begin{bmatrix} 1 & \frac{1}{\lambda}(1 - e^{(-\lambda \Delta t)} \\ 0 & e^{(-\lambda \Delta t)} \end{bmatrix}$$

$$Q_n = \begin{cases} Q_D + Q_{J,u_k} & \exists k : \tau_k \in [t_{n-1}, t_n] \\ Q_D & \text{otherwise} \end{cases}$$

$$Q_D = \frac{\sigma^2}{2\lambda} \begin{bmatrix} q_1 & q_2 \\ q_2 & q_3 \end{bmatrix}$$

$$q_1 = \frac{1}{\lambda^2}(2\lambda\Delta t - (3 - e^{(-\lambda\Delta t)})(1 - e^{(-\lambda\Delta t)}) \ )$$

$$q_2 = \frac{1}{\lambda}(1 - e^{(-\lambda\Delta t)})^2$$

$$q_3 = 1 - e^{(-2\lambda\Delta t)}.$$

Assuming only the value is observed, corrupted by Gaussian noise with standard deviation σ_y^2, the measurement model is defined by,

$$y_n = Cx_n + v_n$$

where v_n is a Gaussian random variable with covariance matrix R.

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$
$$R = \sigma_y^2.$$

REFERENCES

[1] M. Jacobsen, *Point process theory and applications: Marked point and piecewise deterministic processes*, C. Newman and S. Resnick, Eds. Birkhauser, 2006.

[2] S. Godsill, J. Vermaak, W. Ng, and J. Li, "Models and algorithms for tracking of maneuvering objects using variable rate particle filters," *Proceedings of the IEEE*, vol. 95, pp. 925–952, 2007.

[3] N. Whiteley, A. M. Johansen, and S. Godsill, "Monte Carlo filtering of piecewise deterministic processes," *Journal of Computational and Graphical Statistics*, vol. 20, pp. 119–139, 2011.

[4] M. Morelande and N. Gordon, "Rao-blackwellised variable rate particle filters," in *Proc. 12th Int. Conf. Information Fusion*, july 2009, pp. 1 –8.

[5] S. Godsill, "Particle filters for continuous-time jump models in tracking applications," *ESAIM: Proc.*, vol. 19, pp. 39–52, 2007.

[6] H. L. Christensen, J. Murphy, and S. J. Godsill, "Forecasting high-frequency futures returns using online Langevin dynamics," *IEEE Special Topics in Signal Processing, to be published*, 2012.

[7] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, pp. 107–113, 1993.

[8] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, pp. 197–208, 2000.

[9] S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for nonlinear time series," *Journal of the American Statistical Association*, vol. 99, pp. 156–168, 2004.

[10] O. Cappé, S. Godsill, and E. Moulines, "An overview of existing methods and recent advances in sequential Monte Carlo," *Proceedings of the IEEE*, vol. 95, pp. 899–924, 2007.

[11] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," in *The Oxford Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovsky, Eds. Oxford University Press, 2009.

[12] P. Del Moral, A. Doucet, and A. Jasra, "Sequential Monte Carlo samplers," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, pp. 411–436, 2006.

[13] P. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, p. 711, 1995.

[14] M. S. Grewal and A. P. Andrews, *Kalman filtering: Theory and practice using MATLAB*. John Wiley & Sons, Inc., 2002.

[15] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal Of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[16] H. Rauch, F. Tung, and C. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA journal*, vol. 3, pp. 1445–1450, 1965.

[17] D. Fraser and J. Potter, "The optimum linear smoother as a combination of two optimum linear filters," *IEEE Trans. Automatic Control*, vol. 14, pp. 387–390, 1969.

[18] X. Li and V. Jilkov, "Survey of maneuvering target tracking. Part I: Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, pp. 1333–1364, 2003.

[19] P. Bunch and S. Godsill, "Dynamical models for tracking with the variable rate particle filter," in *Proc. 15th Int. Conf. Information Fusion, to be published*, 2012.

[20] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, pp. 590–599, 1999.

[21] G. Casella and C. P. Robert, "Rao-Blackwellisation of sampling schemes," *Biometrika*, vol. 83, pp. 81–94, 1996.

[22] A. Doucet, N. De Freitas, K. Murphy, and S. Russell, "Rao-Blackwellised particle filtering for dynamic Bayesian networks," in *Proc. 16th Conf. Uncertainty in Artificial Intelligence*, 2000, pp. 176–183.

[23] W. R. Gilks and C. Berzuini, "Following a moving target – Monte Carlo inference for dynamic Bayesian models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, pp. 127–146, 2001.

[24] K. Gilholm and N. O. Everett, "A novel algorithm for tracking targets using manoeuvre models," in *IET Seminar on Target Tracking and Data Fusion: Algorithms and Applications*, 2008, pp. 143–148.

[25] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian non-linear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, pp. 1–25, 1996.

[26] R. Douc, A. Garivier, E. Moulines, and J. Olsson, "Sequential Monte Carlo smoothing for general state space hidden markov models," *The Annals of Applied Probability*, vol. 21, no. 6, pp. 2109–2145, 2011.

[27] M. Briers, A. Doucet, and S. Maskell, "Smoothing algorithms for statespace models," *Annals of the Institute of Statistical Mathematics*, vol. 62, pp. 61–89, 2010.

[28] P. Fearnhead, D. Wyncoll, and J. Tawn, "A sequential smoothing algorithm with linear computational cost," *Biometrika*, vol. 97, pp. 447–464, 2010.

[29] S. Sarkka, P. Bunch, and S. Godsill, "A backward-simulation based Rao-Blackwellized particle smoother for conditionally linear Gaussian models," in *16th IFAC Symp. System Identification, to be published*, 2012.

[30] G. Kitagawa, "The two-filter formula for smoothing and an implementation of the Gaussian-sum smoother," *Annals of the Institute of Statistical Mathematics*, vol. 46, pp. 605–623, 1994.

[31] A. Gelb, Ed., *Applied optimal estimation*. MIT Press, 1974.

[32] P. Bunch and S. Godsill, "Improved particle approximations to the joint smoothing distribution using Markov chain Monte Carlo," *IEEE Trans. Signal Processing, to be published*, 2012, submitted for publication.

[33] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[34] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Processing*, vol. 56, pp. 3447–3457, 2008.

[35] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, pp. 401–422, 2004.

[36] B. Øksendal, *Stochastic differential equations: An introduction with applications*. Springer Verlag, 2003.

[37] S. Sarkka, "Recursive Bayesian inference on stochastic differential equations," Ph.D. dissertation, Helsinki University of Technology Laboratory of Computational Engineering, 2006.