Filtering and Smoothing Algorithms for Nonlinear Variable Rate Models with Applications in Tracking

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

***Abstract*—The abstract goes here.**

***Index Terms*—**

I. Introduction

CONVENTIONAL sequential state-space inference methods use discrete-time hidden Markov models to estimate an unknown quantity evolving over time. These models may be developed by discretisation of a continuous-time model, or by formulation directly in discrete time. In either case, it is common practice to use a grid of time points synchronous with the observations. Such "fixed rate" models may give poor representations of processes which exhibit structured behaviour over longer time scales. In such cases, a "variable rate" model may prove advantageous - the model is discretised onto a set of set of random times, which characterise the transitions in system behaviour.

Using a variable rate model, it is necessary to estimate both the set of changepoints and the (nonlinearly) evolving state. This is not analytically tractable so numerical approximations must be employed. The particle filter (introduced by [1]) and smoother (see [2], [3]) are such schemes which approximate the posterior state 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 [4], [5].

Target tracking algorithms are commonly based upon fixed rate models (see, e.g. [6] 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 [7]–[9], variable rate models were introduced for tracking, in which the state trajectory is divided up by a set of changepoints between which the motion follows a deterministic path governed by kinematic parameters (accelerations, etc.) which are fixed for that segment. The changepoint times and kinematic parameters may be estimated using a numerical approximation called the variable rate particle filter (VRPF).

In this paper we review the VRPF for piecewise-deterministic nonlinear models and introduce improvements using the resample-move method of [10]. We then describe a new variable rate particle smoother (VRPS) algorithm, which uses a novel Markov Chain Monte Carlo (MCMC) formulation to circumvent difficulties of the conventional methods. In section **??**, we adapt the intrinsic-coordinate models of [8], [9] for use with the new smoothing algorithm. Simulations are presented in section **??**.

II. Piecewise-Deterministic 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, K, of changepoints occur at times $\{\tau_1 \dots \tau_K\}$. Discrete sets containing multiple values over time will be written as, e.g. $y_{1:n} = \{y_1 \dots y_n\}$.

Changepoints are treated as random variables, dependent on the previous times. The sequence it assumed to be Markovian throughout this paper.

$$\tau_k \sim p(\tau_k|\tau_{k-1}) \tag{1}$$

In continuous time, the dynamics of the latent state, $x(t)$, may be written as a stochastic differential equation dependent on the sequence of changepoints which have occured so far, $\tau_{1:k}$ (i.e. $\tau_k < t < \tau_{k+1}$).

$$dx(t) = \mathfrak{f}(x(t), v(t), \tau_{1:k})dt \tag{2}$$

The dynamics depend on the changepoint sequence, $\tau_{1:k}$, and a continuously varying random innovation, $v(t)$. For the variable rate models considered in this paper, this stochastic differential equation will be solved by assuming that the random innovation is constant between changepoints, $v(t) = v_k \forall t : \tau_k < t < \tau_{k+1}$. This means that the state trajectory is a deterministic function dependent on the changepoint times and innovations.

$$x(t) = f(x_{k-1}, v_{k-1}, \tau_{1:k-1}, t), \tau_{k-1} < t < \tau_k \tag{3}$$

The innovations may be treated as random variables dependent on the previous values. This sequence will also be assumed to be Markovian throughout this paper.

$$v_k \sim p(v_k|v_{k-1}) \tag{4}$$

Once the next changepoint time is known, (3) may be used to calculate the state at this changepoint, $x_k = x(\tau_k)$. In addition, the state at the observation times will be given by $\hat{x}_n = x(t_n)$, which is needed to evaluate measurement likelihoods.

To keep the following derivations concise, an additional variable is introduced to denote each changepoint time-innovation pair $\theta_k = \{\tau_k, v_k\}$. It will often be assumed unless stated otherwise that the most recent changepoint occured at time τ_k and that the current innovation value is v_k.

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

Manuscript received January 01, 1901; revised January 02, 1901.

It will be useful to denote the set of changepoints and parameters (of unspecified size) which occur within some range of time. This will be written as $\theta_{[s,t]} = \{\theta_k \forall k : s < \tau_k < t\}$. Such a variable may be thought of as representing the number, k, of changepoints within the interval, plus the times and innovations of each changepoint. Note also that such a variable not only conveys where changepoints occur, but also where they do not within the specfied range.

III. The Variable Rate Particle Filter

The objective of an inference algorithm will be to estimate the sequence of changepoint times, $\tau_{[0,t_n]}$ and the state trajectory, $x(t) \forall t < t_n$, given some set of the observations, $y_{1:n}$. The state may be calculated deterministically using (3) once the changepoint time and innovation sequences, $\theta_{[0,t_n]}$, are known (plus the initial state, x_0).

In [7]–[9], the variable rate particle filter (VRPF) was introduced for performing this inference sequentially. At time t_n, the filter estimates the posterior changepoint distribution.

$$
\begin{aligned}
&p(\theta_{[0,t_n]}, x_0 | y_{1:n}) \\
&\quad = \frac{p(y_n|\theta_{[0,t_n]}, x_0, y_{1:n-1}) p(\theta_{[0,t_n]}, x_0 | y_{1:n-1})}{p(y_n|y_{1:n-1})} \\
&\quad = \frac{p(y_n|\hat{x}_n) p(\theta_{[t_{n-1},t_n]}|\theta_{[0,t_{n-1}]}) p(\theta_{[0,t_{n-1}]}, x_0 | y_{1:n-1})}{p(y_n|y_{1:n-1})}
\end{aligned}
\quad (5)
$$

This distribution cannot be calculated analytically, but may be approximated numerically.

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

$$
\hat{p}(\theta_{[0,t_n]}, x_0 | y_{1:n}) = \sum_j w_n^{(j)} \delta_{(\theta_{[0,t_n]}^{(j)}, x_0^{(j)})}(\theta_{[0,t_n]}, x_0) \quad (6)
$$

where $\delta_x(X)$ is a probability mass at the point where $X = x$.

The particle filter works recursively. At the nth step, a particle, $\theta_{[0,t_{n-1}]}^{(i)}$, is first resampled from those approximating the filtering distribution at the $(n-1)$th step, using an arbitrary set of proposal weights.

$$
q(\theta_{[0,t_{n-1}]}, x_0) = \sum_j \beta_n^{(j)} \delta_{(\theta_{[0,t_{n-1}]}^{(j)}, x_0^{(j)})}(\theta_{[0,t_{n-1}]}, x_0) \quad (7)
$$

An extension, $\theta_{[t_{n-1},t_n]}^{(i)}$, is then proposed from an importance distribution, $q(\theta_{[t_{n-1},t_n]}|\theta_{[0,t_{n-1}]}^{(i)}, y_n)$. Finally, the particle is weighted according to the ratio of the target distribution and the proposal.

$$
\begin{aligned}
w_n^{(i)} &= \frac{p(\theta_{[0,t_n]}^{(i)}, x_0^{(i)} | y_{1:n})}{q(\theta_{[0,t_{n-1}]}^{(i)}, x_0^{(i)} | y_{1:n-1}) q(\theta_{[t_{n-1},t_n]}^{(i)} | \theta_{[0,t_{n-1}]}^{(i)},)} \\
&\propto \frac{p(y_n|\theta_{[0,t_n]}^{(i)}, x_0^{(i)}, y_{1:n-1}) p(\theta_{[t_{n-1},t_n]}^{(i)}|\theta_{[0,t_{n-1}]}^{(i)}) p(\theta_{[0,t_{n-1}]}^{(i)}, x_0^{(i)}|}{q(\theta_{[0,t_{n-1}]}^{(i)}, x_0^{(i)} | y_{1:n-1}) q(\theta_{[t_{n-1},t_n]}^{(i)} | \theta_{[0,t_{n-1}]}^{(i)}, y_n)} \\
&\approx \frac{w_{n-1}^{(i)}}{\beta_{n-1}^{(i)}} \times \frac{p(y_n|\hat{x}_n) p(\theta_{[t_{n-1},t_n]}^{(i)}|\theta_{[0,t_{n-1}]}^{(i)})}{q(\theta_{[t_{n-1},t_n]}^{(i)}|\theta_{[0,t_{n-1}]}^{(i)}, y_n)}
\end{aligned}
$$

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

The first term, $p(y_n|\hat{x}_n)$, is simply an observation likelihood of the deterministically calculated state. The second term, $p(\theta_{[t_{n-1},t_n]}^{(i)}|\theta_{[0,t_{n-1}]}^{(i)})$, is a transition density, the form of which depends on whether or not a new changepoint occurs within the interval $[t_{n-1}, t_n]$. Let the occurence of k changepoints already have been estimated in the interval $[0, t_{n-1}]$. It is assumed that the probability of more than one changepoint occuring in the interval $[t_{n-1}, t_n]$ is negligible.

$$
\begin{aligned}
&p(\theta_{[t_{n-1},t_n]}|\theta_{[0,t_{n-1}]}) \\
&\quad = \begin{cases} p(\tau_{k+1}|\tau_k, \tau_{k+1} > t_{n-1}) p(\tau_{k+2} > t_n|\tau_{k+1}) p(v_{k+1}|v_k) & \tau_{k+1} < \\ p(\tau_{k+1} > t_n|\tau_{1:k}, \tau_{k+1} > t_{n-1}) & \tau_{k+1} > \end{cases}
\end{aligned}
$$

This can be calculated from transition models for τ_k and v_k specified by (1) and (4). For the most basic "bootstrap" [1] form of VRPF, this transition density may be used as the importance distribution, leading to the usual simplification in the weight formula.

The choice of proposal weights, $\{\beta_{n-1}^{(i)}\}$, requires particular attention in the design of VRPFs. In some models a changepoint may not have an immediate effect on the observations, especially if a jump occurs in some quantity which is only observed via its integral, e.g. if there is a jump in the acceleration of a moving object, yet only the position is measured, the change will not be apparent until several more observations have been made. In the meantime, particles which contain a changepoint at the correct time may all have been removed by the resampling process. A particle pays a debt in transition probability when a changepoint is proposed, and does not see it repaid in likelihood until later. To avoid this loss of good particles, proposal weights should be chosen which preserve a significant number of low-weight particles. One scheme which has been found to work well is described in [9], in which proposal weights are given by:

$$
\beta_{n-1}^{(i)} \propto \max(1, N_F w_{n-1}^{(i)}) \quad (10)
$$

where N_F is the number of filtering particles. The VRPF is summarised below.

```
Initialise particles $\{\theta_{[0,0]}^{(i)}\} \leftarrow \emptyset$.
Sample starting state prior, $\{x_0^{(i)}\} \sim p(x_0)$
for $n = 1 \ldots N$ do
  for $i = 1 \ldots N_F$ do
```

```
        Sample a history $\{\theta^{(i)}_{[0,t_{n-1}]}, x^{(i)}_0\} \sim \sum_j v^{(j)}_n \delta_{(\theta^{(j)}_{[0,t_n]}, x^{(j)}_0)}(\theta_{[0,t_n]}, x_0)$.
        Propose $\theta^{(i)}_{[t_{n-1},t_n]} \sim q(\theta_{[t_{n-1},t_n]}|\theta^{(i)}_{[0,t_{n-1}]}, y_n)$
        Calculate $\hat{x}_n$ using (3).
        Weight $w^{(i)}_n \propto \frac{w^{(i)}_{n-1}}{\beta^{(i)}_{n-1}} \times \frac{p(y_n|\hat{x}_n)p(\theta^{(i)}_{[t_{n-1},t_n]}|\theta^{(i)}_{[0,t_{n-1}]})}{q(\theta^{(i)}_{[t_{n-1},t_n]}|\theta^{(i)}_{[0,t_{n-1}]},y_n)}$
    end for
    Scale weights such that $\sum_i w^{(i)}_n = 1$
end for
```

A. Improved VRPF Using Resample-Move

The weakness of the ordinary VRPF is that it is very difficult to spot changepoints until after they have actually happened. In the basic filter, changepoints can only be added between the previous and current observation times, t_{n-1} and t_n. It is necessary to have a very large number of particles in order to ensure that some propose changepoints in the correct areas.

The filter performance can be greatly improved by the addition of resample-move steps [10] to adjust the time of previous changepoints, $\tau_{1:k}$ and innovations, $v_{1:k}$. Here we consider altering only the most recent values.

In a resample-move scheme, Metropolis-Hastings (MH) [11], [12] steps are included after resampling to regenerate the particle distribution. A proposal is made to change one or more of the previous states, and accepted with a probability that depends on the ratio of target and proposal distribution probabilities. Here, we propose a new value for the most recent changepoint time and the corresponding innovation, $\theta^*_k \sim q(\theta_k|\theta_{1:k-1}, y_{1:n})$, which are used to replace the current values, $\theta^{(m)}_k$, to construct a new changepoint sequence, $\theta^*_{[0,t_n]}$. Thus, the changepoint times and innovations may be changed many frames after they were first proposed, when more informative observations are available. The acceptance probability for such a MH move is the following ratio.

$$\alpha = \frac{p(\theta^*_{[0,t_n]}, x_0|y_{1:n})q(\theta^{(m)}_k|\theta_{1:k-1}, y_{1:n})}{p(\theta^{(m)}_{[0,t_n]}, x_0|y_{1:n})q(\theta^*_k|\theta_{1:k-1}, y_{1:n})} \tag{11}$$

$$= \frac{p(y_{s_k:n}|\theta^*_{[0,t_n]}, x_0)p(\theta^*_{[t_{s_k-1},t_n]}|\theta_{[0,t_{s_k-1}]})q(\theta^{(m)}_k|\theta_{1:k-1}, y_{1:n})}{p(y_{s_k:n}|\theta^{(m)}_{[0,t_n]}, x_0)p(\theta^{(m)}_{[t_{s_k-1},t_n]}|\theta_{[0,t_{s_k-1}]})q(\theta^*_k|\theta_{1:k-1}, y_{1:n})} \tag{12}$$

$$= \frac{p(y_{s_k:n}|\hat{x}^*_{s_k:n})p(\theta^*_k|\theta_{1:k-1})q(\theta^{(m)}_k|\theta_{1:k-1}, y_{1:n})}{p(y_{s_k:n}|\hat{x}^{(m)}_{s_k:n})p(\theta^{(m)}_k|\theta_{1:k-1})q(\theta^*_k|\theta_{1:k-1}, y_{1:n})} \tag{13}$$

Here, s_k is the index of the earliest observation to come after either τ_k or τ^*_k (Thus $y_{s_k:n}$ simply means “all the observations affected by the proposed move”.).

In addition, simpler moves can be conducted in which only v_k is changed, or more complex moves in which multiple past changepoints are altered. The computational complexity for such moves is likely to be high, however, because of the large number of likelihood evaluations to be made. It is also possible to included reversible jump moves [13] to add or remove changepoints entirely.

A proposal distribution for resample-move steps can be formulated using the Unscented Transform [14] to approximate the “optimal importance distribution” []*Cite someone who explains this term*. Ideally, new values for the random variables should be proposed from the conditional distribution:

$$p(v_k|\theta_{1:k-1}, \tau_k, y_{1:n}) \propto p(y_{s_k:n}|\theta_{1:k})p(v_k) \tag{14}$$

This may be approximated using an Unscented Kalman Filter (UKF) [14] with no prediction steps. If changepoints occurly sparsely, this can lead to large amounts of computation due to the number of observations in the set $y_{s_k:n}$. In this case, the proposal can be formed using only a subset of the available observations, thus limiting the computational load.

IV. The Variable Rate Particles Smoother

A filter conducts inference sequentially as new observations are introduced. The purpose of a smoother is to produce a second estimate once all the observations have been made, using future values to improve upon the filter performance. For ordinary fixed rate models, a particle approximation of the joint smoothing distribution may be constructed by resampling values from the filter approximations backwards through time [3].

Formulating an effective smoother for piecewise deterministic variable rate models proves to be a challenging task, because changing an estimate of the innovations or the state in the past will change the state, and thus the observation likelihood, in the future. This obstacle prevents the use of direct backward sampling smoothers, such as that of [3], for which an intractable integral would be required in the weight calculations. In this section, a smoother which uses MCMC to sample the changepoint history is developed, using similar principles to [**?**].

In the derivation of the VRPF, the filter targetted the joint distribution over changepoint times, $\tau_{[0,t_n]}$, and innovations, $v_{[0,t_n]}$. This was desirable because once these sequences have been estimated, the state can be calculated deterministically for all times up to t_n, an essential property for measurement likelihood evaluations. For a smoother, such a scheme is problematic because changing one of the innovations at an early time will alter the state at all subsequent times. Instead, the sequence of innovations, $v_{[0,T]}$, is replaced by the sequence of states occuring at the changepoint times, $x_{[0,T]} = \{x(\tau_k)\forall k\}$, which may be calculated deterministically from the changepoint time and innovation sequences. Targetting a distribution over this state sequence, if a change is made to an estimate of x_k, it is no longer necessary for the rest of the states to change, only to alter the preceeding and following innovations, v_{k-1} and v_k. For such a scheme to be workable, it must be possible to calculate an innovation from the states and times of the bounding changepoints. In other words, (3) must be “invertible”.

$$v_k = f^{-1}(x_{k-1}, x_k, \tau_{k-1}, \tau_k) \tag{15}$$

Analagously with θ, a new variable is introduced to represent changepoint time-state pairs, $\zeta_k = \{\tau_k, x_k\}$. Thus if $\zeta_{0:K}$ is known (with $\tau_0 = 0$), it is possible to calculate the innovation sequence using (15) and thence to calculate the state at any time. Thus, for piecewise-deterministic variable rate models, the posterior distribution targetted by a smoother is $p(\zeta_{[0,T]}|y_{1:N})$. An approximation to this distibution may be derived from the final filtering results, by tranforming from the innovation sequence, $v_{[0,T]}$, to the changepoint state sequence, $x_{[0,T]}$. This approximation will tend to suffer from particle degeneracy due to the resampling procedure, i.e. the number of unique particles decreases for earlier times, giving a poor representation of the distribution [15]. An improved smoothing algorithm is needed to rejeuvenate this particle diversity.

For an ordinary fixed rate model, particle smoothing may be conducted by sequentially resampling states from the filtering approximations backwards in time. With the nonlinear variable rate models under consideration here, this process is not tractable. Instead, particles are sampled from the joint smoothing distribution using MCMC. The posterior distribution is first expanded using Bayes theorem around observation time t_n.

$$\begin{aligned} p(\zeta_{[0,T]}|y_{1:N}) & \\ &= \frac{p(y_{1:N}|\zeta_{[0,T]})p(\zeta_{[0,T]})}{p(y_{1:N})} \qquad (16) \\ &\propto p(\zeta_{[t_n,T]}|\zeta_{[0,t_n]})p(\zeta_{[0,t_n]}) \\ &\quad \times p(y_{r_k^+ +1:N}|\hat{x}_{r_k^+ +1:N})p(y_{r_k^-:r_k^+}|\hat{x}_{r_k^-:r_k^+})p(y_{1:r_k^- -1}|\hat{x}_{1:r_k^- -1}) \qquad (17) \end{aligned}$$

The likelihood term has been split up into three parts, divided at r_k^- and r_k^+, the earliest and latest observations to occur between the successive changepoints τ_k and τ_{k+1}, which lie before and after t_n respectively.

A Markov chain is initialised by sampling a particle from the approximation generated by the final filtering step. Because this is an approximation of the target distribution, no burn-in is requiired. Proceeding backwards in time, MH steps are conducted by proposing a replacement for the changepoint sequence history from the appropriate filtering approximation. At the nth step, the future changepoints are considered to be fixed, $\tilde{\zeta}_{[t_n,T]}$. A replacement for the past changepoints is proposed from the filtering particles.

$$\zeta^*_{[0,t_n]} \sim q(\zeta_{[0,t_n]}|y_{1:n}) = \sum_j \gamma_n^{(j)} \delta_{(\zeta^{(j)}_{[0,t_n]})}(\zeta_{[0,t_n]}) \tag{18}$$

This proposed move is accepted with the following acceptance probability, where $\zeta^{(m)}_{[0,t_n]}$ is the previous sequence of past changepoints in the Markov chain.

$$\begin{aligned} \alpha &= \frac{p(\zeta^*_{[0,t_n]}, \tilde{\zeta}_{[t_n,T]}|y_{1:N})q(\zeta^{(m)}_{[0,t_n]}|y_{1:n})}{p(\zeta^{(m)}_{[0,t_n]}, \tilde{\zeta}_{[t_n,T]}|y_{1:N})q(\zeta^*_{[0,t_n]}|y_{1:n})} \\ &= \frac{\gamma_n^{(m)}}{\gamma_n^*} \times \frac{p(\tilde{\zeta}_{[t_n,T]}|\zeta^*_{[0,t_n]})p(\zeta^*_{[0,t_n]})}{p(\tilde{\zeta}_{[t_n,T]}|\zeta^{(m)}_{[0,t_n]})p(\zeta^{(m)}_{[0,t_n]})} \\ &\quad \times \frac{p(y_{r_k^-:r_k^+}|\hat{x}^*_{r_k^-:r_k^+})p(y_{1:r_k^- -1}|\hat{x}^*_{1:r_k^- -1})}{p(y_{r_k^-:r_k^+}|\hat{x}^{(m)}_{r_k^-:r_k^+})p(y_{1:r_k^- -1}|\hat{x}^{(m)}_{1:r_k^- -1})} \\ &= \frac{\gamma_n^{(m)}}{\gamma_n^*} \times \frac{p(\zeta^*_{[0,t_n]})p(y_{1:r_k^- -1}|\hat{x}^*_{1:r_k^- -1})}{p(\zeta^{(m)}_{[0,t_n]})p(y_{1:r_k^- -1}|\hat{x}^{(m)}_{1:r_k^- -1})} \\ &\quad \times \frac{p(\tilde{\zeta}_{k+1}|\zeta^*_k)p(y_{r_k^-:r_k^+}|\hat{x}^*_{r_k^-:r_k^+})}{p(\tilde{\zeta}_{k+1}|\zeta^{(m)}_k)p(y_{r_k^-:r_k^+}|\hat{x}^{(m)}_{r_k^-:r_k^+})} \end{aligned} \tag{19}$$

The first fraction is simply a ratio of proposal weights for the current and proposed changepoint histories. The second ratio comprises probabilities which depend solely on the past changepoints, and which are thus the same as values calculated during the filtering stage. These may be stored when first calculated, meaning they can simply be fetched from memory when calculating the acceptance probability.

The third ratio consists of the transition probability between two known changepoints, ζ_k and ζ_{k+1}, and the likelihood of observations occurring between them. These may be found by calculating the innovation, v_k, to get from ζ_k to ζ_{k+1}. The continuous state trajectory may then be calculated allowing the likelihood terms to be evaluated. The transition density term is given by:

$$p(\zeta_{k+1}|\zeta_k) = p(\tau_{k+1}|\tau_k)p(v_k|v_{k-1}) \tag{20}$$

The variable rate particle smoother (VRPS) is summarised below.

```
Run a particle filter to approximate $p(\theta_{[0,t_n]}, x_0|y_{1:n})$ for each $n$.
Convert each particle innovation sequence to a changepoint state sequence, to produce approximations of $p(\zeta_{[0,t_n]}|y_{1:n})$.
for $i = 1 \ldots S$ do
  Initialise sampler with $\tilde{\zeta}^{(i)}_{[0,T]} \sim \sum_j w_N^{(j)} \delta_{\zeta^{(j)}_{[0,T]}}(\zeta_{[0,T]})$
  for $n = N \ldots 1$ do
    $\zeta^{(i)(0)}_{[0,T]} \leftarrow \tilde{\zeta}^{(i)}_{[0,T]}$.
    for $m = 1 \ldots M$ do
      Propose a new past $\zeta^{(i)*}_{[0,t_n]} \sim \sum_j \gamma_n^{(j)} \delta_{(\zeta^{(j)}_{[0,t_n]})}(\zeta_{[0,t_n]})$.
      Calculate $\alpha$ using equation 19
      With probability $\alpha$, $\zeta^{(i)(m)}_{[0,t_n]} \leftarrow \zeta^{(i)*}_{[0,t_n]}$. Otherwise $\zeta^{(i)(m)}_{[0,t_n]} \leftarrow \zeta^{(i)(m-1)}_{[0,t_n]}$.
    end for
    $\tilde{\zeta}^{(i)}_{[0,t_n]} \leftarrow \zeta^{(i)(M)}_{[0,t]}$.
  end for
end for
```

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. In the simulations discussed in this report, $M = 1$ was used throughout.

V. Variable Rate Models for Tracking

In this section, a tracking model for use with variable rate particle filters and smoothers is presented. In [8], [9], a two-dimensional (2D) variable rate model was introduced in which an object experienced piecewise constant random accelerations tangential and normal to the velocity, and a drag force negatively proportional to the velocity. The latent state is a vector consisting of the position in Cartesian coordinates along with the bearing and speed. The resulting differential equations cannot be solved analytically, so the filter relies on numerical integration to calculate the position. This makes it unsuitable for use with the proposed VRPS algorithm, because it is not possible to calculate the accelerations (and thus the transition probabilities) given the state at two successive changepoints.

In addition, the basic intinisic coordinate model is degenerate; there are four state variables (two position, two velocity) but only two parameters governing the intervening motion (the two perpendicular accelerations). Consequently, for two successive changepoints there is likely to be no possible innovation value resulting in the required change. This property makes the model unsuitable for smoothing algorithms.

In this section, the kinematic state at a changepoint will be denoted $\mathbf{x}_k$, and the innovation as $\mathbf{v}_k$.

The simpler 2D model outlined here is analytic ($\mathbf{x}_{k+1}$ may be calculated without numerical integration given $\mathbf{x}_k$, $\mathbf{v}_k$ and the changepoint times), non-degenerate (there are as many independent random variables for each transition as state variables) and invertible ($\mathbf{v}_k$ may be calculated analytically given $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$.

The state vector consists of cartesian position, x_t and y_t, plus bearing, ψ_t, and speed, $\dot{s}$.

$$\mathbf{x}_t = [x_t, y_t, \psi_t, \dot{s}_t]^T \tag{21}$$

In addition to the tangential and normal accelerations, $a_{T,k}$ and $a_{N,k}$, we introduce two linear drift velocity terms, $d_{X,k}$ and $d_{Y,k}$. Together, these four variables make up the innovation vector:

$$\mathbf{v}_k = [a_{T,k}, a_{N,k}, d_{X,k}, d_{Y,k}]^T \tag{22}$$

The target dynamics are described by four differential equations.

$$\ddot{s}_t = a_{T,k} \tag{23}$$

$$\dot{s}_t \dot{\psi}_t = a_{N,k} \tag{24}$$

$$\dot{x}_t = \dot{s}_t \cos(\psi_t) + d_{X,k} \tag{25}$$

$$\dot{y}_t = \dot{s}_t \sin(\psi_t) + d_{Y,k}. \tag{26}$$

Solving these yields the following state equations (where $\Delta\tau = \tau_{k+1} - \tau_k$):

$$\begin{aligned}
\dot{s}_{k+1} &= \dot{s}_k + a_{T,k}\Delta\tau \\
\psi_{k+1} &= \psi_k + \frac{a_{N,k}}{a_{T,k}} \log\left(\frac{\dot{s}_{k+1}}{\dot{s}_k}\right) \\
x_{k+1} &= x_k + d_{X,k}\Delta\tau + \frac{\dot{s}_{k+1}^2}{4a_{T,k}^2 + a_{N,k}^2}\left[a_{N,k}\sin(\psi_{k+1}) + 2a_{T,k}\cos(\psi\right. \\
&\quad - \frac{\dot{s}_k^2}{4a_{T,k}^2 + a_{N,k}^2}\left[a_{N,k}\sin(\psi_k) + 2a_{T,k}\cos(\psi_k)\right] \\
y_{k+1} &= y_k + d_{Y,k}\Delta\tau + \frac{\dot{s}_{k+1}^2}{4a_{T,k}^2 + a_{N,k}^2}\left[-a_{N,k}\cos(\psi_{k+1}) + 2a_{T,k}\sin\right. \\
&\quad - \frac{\dot{s}_k^2}{4a_{T,k}^2 + a_{N,k}^2}\left[-a_{N,k}\cos(\psi_k) + 2a_{T,k}\sin(\psi_k)\right].
\end{aligned}$$

Particular care must be taken when $\lim a_{T,k} \rightarrow 0$ or $\lim a_{N,k} \rightarrow 0$ (or both), which can be handled using L'Hôpital's rule or by returning to the equations of motion and re-integrating.

By introducing the drift terms , $d_{X,k}$ and $d_{Y,k}$, the model becomes non-degenerate. These terms represent a linear velocity superimposed onto the nonlinear manoeuvre of the object. Physically, they could represent a genuine effect, e.g. the tide in a ship tracking model or the wind in an aircraft tracking model, or they may simply represent an expected degree of model error.

The system equations may easily be solved to yield $\mathbf{v}_k$ from $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$, which is essential for use with the VRPS algorithm.

For the simulations, the innovations are assumed to be Gaussian distributed.

A. Observation Model

A radar-type, polar observation model was used. Assuming that the coordinate system is centered on the sensor, the observations consist of bearing and range, with the optional addition of bearing and range rate. For this paper, the drift velocities, $d_{X,k}$ and $d_{Y,k}$ are assumed to represent genuine physical effects, and are thus included in the velocity measurements.

$$\mathbf{y}_n = \begin{cases} [b_n, r_n]^T & \text{position measurement only} \\ [b_n, r_n, \dot{b}_n, \dot{r}_n]^T & \text{position and velocity measurements} \end{cases} \tag{31}$$

$$b_n = \tan^{-1}(y_n/x_n) + \epsilon_{1,n} \tag{32}$$

$$r_n = \sqrt{y_n^2 + x_n^2} + \epsilon_{2,n} \tag{33}$$

$$\dot{b}_n = \frac{(\dot{s}_n\cos(\psi_n) + d_{X,k})y_n + (\dot{s}_n\sin(\psi_n) + d_{Y,k})x_n}{y_n^2 + x_n^2} + \epsilon_{3,n} \tag{34}$$

$$\dot{r}_n = \frac{(\dot{s}_n\cos(\psi_n) + d_{X,k})x_n + (\dot{s}_n\sin(\psi_n) + d_{Y,k})y_n}{\sqrt{y_n^2 + x_n^2}} + \epsilon_{4,n} \tag{35}$$

The observation noise terms, $\epsilon_{i,n}$ are assumed to be independent and Gaussian distributed.

VI. Simulations

The VRPF and VRPS algorithms were tested on simulated data. An example trajectory using the 2D intrinsic-coordinate

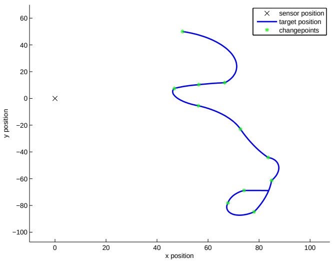


Fig. 1. Example trajectory simulated from 2D model.

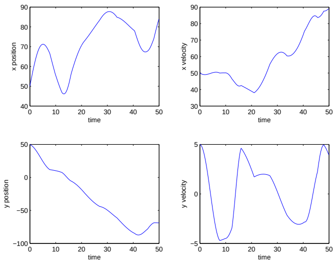


Fig. 2. Example states simulated from 2D model.

model is shown in figure 1. Figures **??** and 3 show individual states and observations.

The VRPF was run using 50 particles and a RM step with a UKF proposal whenever a particle is replicated through resampling. The VRPS was used to simulated 50 smoothing trajectories.

Figure 4 shows the particle position estimates from the final VRPF frame and the VRPS.

Figure 5 shows a kernel density estimate of the changepoint times for the VRPF and VRPS.

While the VRPF generally overestimates the number of jumps, the VRPS is remarkably accurate. In this example, the VRPF estimate 15.1 changepoints (average over all particles), while the VRPS estimates 10.9. The true value is 11.

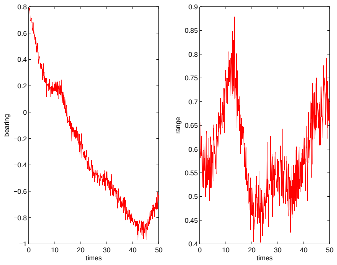
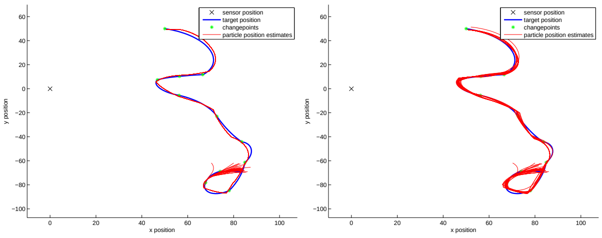


Fig. 3. Example observations simulated from 2D model.



(a) VRPF (b) VRPS

Fig. 4. VRPF and VRPS particle position estimates for an example 2D trajectory

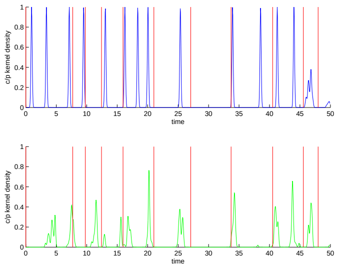


Fig. 5. VRPF (top) and VRPS (bottom) kernel density estimates of jump times.

State estimates were formed by averaging the state values from each particle at the observation times. Root mean square error values can then be calculated for the algorithms. By this metric, the VRPS does not improve upon the performance of the VRPF. RMSEs over time for the example case are shown in figure 6

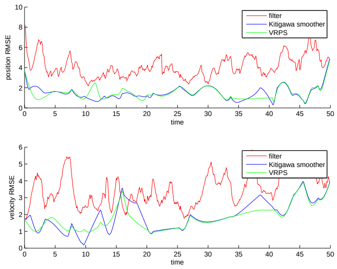


Fig. 6. RMSE over time in position and velocity estimates for the filter, KS and VRPS.

The benefit of the VRPS is in the improved particle diversity of the state estimates. This is illustrated in figure 7, which shows the number of unique state trajectories up to each time, i.e. the number of unique estimates of $\theta_{[0,t]}$ for each t.

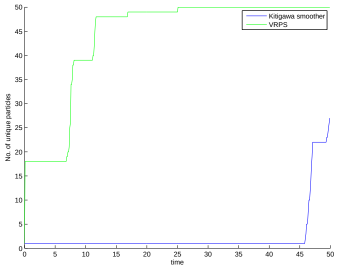


Fig. 7. Number of unique particles over time.

- *Presumably there should be some comparison of batch results. What?*
- *Label the graphs better*
- *Add VRPF without RM*

VII. Conclusion

Improved filtering and smoothing algorithms have been presented for inference with variable rate models. The addition of resample-move steps to the basic variable rate particle filter allows the algorithm to run with a fraction of the number of particles. A smoother for estimation of the joint posterior distribution is not tractable using conventional backwards resampling methods. The algorithm derived instead uses MCMC to resample particles from the filter approximations. The resulting smoother does not necessarily improve the RMSE of state estimates, but the number of unique particles in the approximation is increased significantly.

Appendix A

Appendix one text goes here.

Appendix B

Appendix two text goes here.

Acknowledgment

The authors would like to thank...

References

[1] 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, no. 2, pp. 107 –113, apr 1993.

[2] 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, 10.1023/A:1008935410038. [Online]. Available: http://dx.doi.org/10.1023/A:1008935410038

[3] S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for nonlinear time series," *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 156–168, 2004. [Online]. Available: http://pubs.amstat.org/doi/abs/10.1198/016214504000000151

[4] 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, no. 5, pp. 899–924, 2007.

[5] 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. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.157.772&rep=rep1&type=pdf

[6] X. Li and V. Jilkov, "Survey of maneuvering target tracking. part i: Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

[7] S. Godsill and J. Vermaak, "Models and algorithms for tracking using trans-dimensional sequential monte carlo," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, vol. 3, may 2004, pp. iii – 976–9 vol.3.

[8] S. Godsill, "Particle filters for continuous-time jump models in tracking applications," *ESAIM: Proc.*, vol. 19, pp. 39–52, 2007. [Online]. Available: http://dx.doi.org/10.1051/proc:071907

[9] 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, no. 5, pp. 925 –952, May 2007.

[10] 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, no. 1, pp. 127–146, 2001. [Online]. Available: http://dx.doi.org/10.1111/1467-9868.00280

[11] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The journal of chemical physics*, vol. 21, no. 6, p. 1087, 1953.

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

[13] P. Green, "Reversible jump Markov chain Monte Carlo computation and bayesian model determination," *Biometrika*, vol. 82, no. 4, p. 711, 1995.

[14] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *Proceedings of SPIE - The International Society for Optical Engineering*, K. Ivan, Ed., vol. 3068. Orlando, FL, USA: Society of Photo-Optical Instrumentation Engineers, Bellingham, WA, United States, 1997, pp. 182–193, cited By (since 1996) 144; Conference of Signal Processing, Sensor Fusion, and Target Recognition VI; Conference Date: 21 April 1997 through 24 April 1997; Conference Code: 23022. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-0031347068&partnerID=40&md5=fea6ea231297ae299a7f8932628c53b7

[15] G. Kitagawa, "Monte carlo filter and smoother for non-gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. pp. 1–25, 1996. [Online]. Available: http://www.jstor.org/stable/1390750

Pete Bunch Biography text here.

Simon Godsill Biography text here.