

Variable Rate Models for Tracking

Pete Bunch

and Simon Godsill

Signal Processing and Communications Laboratory

Cambridge University Engineering Dept., UK

Email: {pb404, sjg30}@eng.cam.ac.uk

Abstract—

I. INTRODUCTION

Tracking is the task of inferring kinematic state of a target (position, velocity, etc.) over time from a set noisy or partial observations. The state of the target is a continuous process and is likely to be highly structured. However, many tracking systems model target dynamics as a discrete-time Markov process [1]. This assumption has the advantage of simplicity – by discretising the state at observation times, we arrive at a hidden Markov model, with which standard Kalman [2] or particle [3], [4] filtering and smoothing methods may be used. The disadvantage of such an assumption is that it may be a poor reflection of the system dynamics, disregarding long-term temporal structure in the state trajectory.

Recently, variable rate models have been developed for tracking, [5]–[7]. These models treat the kinematic state as a continuous process which is deterministic conditional upon an underlying sequence of changepoint times and motion parameters. This underlying changepoint sequence may be considered to be a realisation from marked point process (MPP), with the resulting state trajectory as a piecewise-deterministic process (PDP). (The relationship between MPPs and PDPs is developed in [8], along with a rigorous description of associated probability measures.) The sequence of changepoints may be estimated numerically, using a variable rate particle filter, leading to a particle approximation of the current posterior state distribution.

In a tracking scenario, changepoints represent the beginning and end of target manoeuvres, each of which is governed by an associated parameter vector (e.g. accelerations). In [7], a 2D dynamic model is considered in which the accelerations along the Cartesian axes are fixed for the duration of each manoeuvre. In [6], a variation is used in which the target experiences tangential and normal accelerations whose magnitudes are fixed throughout each manoeuvre, but whose directions are specified relative to the target velocity. This intrinsic coordinate model is more complex but also more realistic – the two components can be thought of as throttle and steering controls.

The aforementioned variable rate tracking models suffer from a lack of degrees of freedom. Both the Cartesian and intrinsic versions have four state variables (two position, two velocity), but only two governing accelerations. The result is that from a given manoeuvre start point, it is not possible to

reach every combination of state variables. If the state is fully observed (for example, using radar with Doppler measurements of velocity), this leads to poor tracking performance.

In this paper, we present extensions to the intrinsic coordinate tracking model. First, an augmented version of the 2D model is outlined which addresses the problem of underparametrisation. Secondly, 3D extensions of the 2D model are discussed for manoeuvring aircraft.

II. VARIABLE RATE PARTICLE FILTER

The main focus of this paper is the development of dynamics models for variable rate systems, so only a brief overview of the variable rate particle filter is included here, to motivate the following sections. For a more detailed exposition, see [5]–[7].

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 an 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 up until the current time $\theta_n = \{\tau_j, u_j \forall j : 0 \leq \tau_j < t_n\}$. It will also be useful to define a variable for the changepoints which occur in the interval $[t_{n-1}, t_n]$, $\theta_{n \setminus n-1} = \{\tau_j, u_j \forall j : t_{n-1} \leq \tau_j < t_n\}$.

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

$$K(t) = \max(k : \tau_k < t) \quad (1)$$

$$K_n = K(t_n). \quad (2)$$

The changepoint sequence is assumed to be a Markov process,

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

From this, a prior density for the changepoint sequence, $p(\theta_n)$ and the sequence extension $p(\theta_{n \setminus n-1})$ may be written down.

The state dynamics will be governed by a differential equation depending upon the most recent changepoint.

$$dx(t) = f(x(t), \tau_{K(t)}, u_{K(t)}). \quad (4)$$

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

$$x(t) = f(x_{K_n}, v_{K_n}, \tau_{K_n}, t), \tau_{K_n} \leq t \leq \tau_{K_n+1}. \quad (5)$$

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 calculated. These points will be denoted \hat{x}_n . To complete the model, a probabilistic measurement model must be chosen for the observation process, $p(y_n|\hat{x}_n)$.

The variable rate particle filter (VRPF) sequentially estimates the following changepoint posterior distribution,

$$p(\theta_n|y_{1:n}) \propto p(y_n|\theta_n, y_{1:n-1})p(\theta_{n \setminus n-1}|\theta_{n-1})p(\theta_{n-1}|y_{1:n-1}). \quad (6)$$

Each particle in the approximation is a sequence of changepoints between 0 and t_n . A particle is generated by proposing from the importance distribution,

$$q(\theta_n) = \sum_j v_{n-1}^{(j)} \delta_{\theta_{n-1}^{(j)}}(\theta_{n-1}) q(\theta_{n \setminus n-1}|\theta_{n-1}, y_n). \quad (7)$$

The resampling weights $v_{n-1}^{(j)}$ are chosen appropriately to achieve no resampling, normal resampling or auxiliary resampling.

The particle is then weighted according to the ratio of target and proposal densities,

$$\begin{aligned} w_n^{(i)} &= \frac{p(\theta_n^{(i)}|y_{1:n})}{q(\theta_n^{(i)})} \\ &= \frac{w_{n-1}^{(i)}}{v_{n-1}^{(i)}} \times \frac{p(y_n|\hat{x}_n)p(\theta_{n \setminus n-1}|\theta_{n-1})}{q(\theta_{n \setminus n-1}|\theta_{n-1}^{(i)}, y_n)} \end{aligned} \quad (8)$$

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

For the bootstrap VRPF, the sequence extension transition density is used as the proposal, $q(\theta_{n \setminus n-1}|\theta_{n-1}, y_n) = p(\theta_{n \setminus n-1}|\theta_{n-1})$, resulting in the usual simplification of the importance weights.

Filter performance may be improved by allowing previously proposed changepoints to be modified, either adding resample-move steps [9], or by using the sequential Monte Carlo sampler method of [7].

The rest of the paper is concerned with suitable choices for the state differential equation (4) and the resulting form of the transition function (5).

III. 2D INTRINSIC COORDINATE MODEL

For the simplest 2D intrinsic coordinate model, the target is treated as a particle subject to two accelerations, one tangential, $a_{T,k}$, and one normal, $a_{N,k}$, to the current velocity. The accelerations are fixed for the duration of a manoeuvre. The continuous-time kinematic state of the target is described

by the bearing, $\psi(t)$ (anticlockwise relative to the x axis), and speed $\dot{s}(t)$, as well as the Cartesian coordinates, $x(t)$ and $y(t)$.

Thus, the state and parameter vectors are,

$$\mathbf{x}(t) = [x(t), y(t), \psi(t), \dot{s}(t)]^T \quad (9)$$

$$\mathbf{u}_k = [a_{T,k}, a_{N,k}]^T \quad (10)$$

The target dynamics described by the following standard differential equations for curvilinear motion,

$$\ddot{s}_t = a_{T,K(t)} \quad (11)$$

$$\dot{s}_t \dot{\psi}_t = a_{N,K(t)} \quad (12)$$

$$\dot{x}_t = \dot{s}_t \cos(\psi_t) \quad (13)$$

$$\dot{y}_t = \dot{s}_t \sin(\psi_t). \quad (14)$$

Solving this system of equations yields the following state transition function, where $\Delta t = t - \tau_{K(t)}$,

$$\dot{s}(t) = \dot{s}_{K(t)} + a_{T,k} \Delta t \quad (15)$$

$$\psi(t) = \psi_{K(t)} + \frac{a_{N,k}}{a_{T,k}} \log \left(\frac{\dot{s}(t)}{\dot{s}_{K(t)}} \right) \quad (16)$$

$$\begin{aligned} x(t) &= x_{K(t)} \\ &+ \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2} [a_{N,k} \sin(\psi(t)) + 2a_{T,k} \cos(\psi(t))] \\ &- \frac{\dot{s}_{K(t)}^2}{4a_{T,k}^2 + a_{N,k}^2} [a_{N,k} \sin(\psi_{K(t)}) + 2a_{T,k} \cos(\psi_{K(t)})] \end{aligned} \quad (17)$$

$$\begin{aligned} y(t) &= y_{K(t)} \\ &+ \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2} [-a_{N,k} \cos(\psi(t)) + 2a_{T,k} \sin(\psi(t))] \\ &- \frac{\dot{s}_{K(t)}^2}{4a_{T,k}^2 + a_{N,k}^2} [-a_{N,k} \cos(\psi_{K(t)}) + 2a_{T,k} \sin(\psi_{K(t)})] \end{aligned} \quad (18)$$

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

A similar dynamic model was presented in [10]. In [6], an additional drag term is included in (11) proportional to the current velocity. This renders the equations unsolvable, and it is necessary to use numerical integration to calculate $x(t)$ and $y(t)$. Although this works acceptably well for a bootstrap implementation, it is computationally expensive. It also causes severe difficulties for resample-move and smoothing algorithms, because it is not possible to calculate the particular accelerations which result in a known state.

The weakness of this model is its under-parametrisation. Two fixed acceleration govern the evolution of four state variables. In one manoeuvre, not all values of $\mathbf{x}(t)$ are achievable (the achievable points form a 2D subspace of the 4D state space). As a result, a VRPF based on this model is not at all resilient to model error. It becomes necessary to include changepoints very regularly in order to match the estimated trajectories to the observations. The problem is especially bad if the state is fully observed, for example by using Doppler measurements of velocity as well as radar for position.

IV. THE AUGMENTED 2D MODEL

The solution to the degeneracy of the basic 2D intrinsic coordinate model is to add some more parameters. Here we propose altering the target dynamics by adding a fixed linear velocity,

$$\mathbf{u}_k = [a_{T,k}, a_{N,k}, d_{X,k}, d_{Y,k}]^T \quad (19)$$

$$\dot{x}_t = \dot{s}_t \cos(\psi_t) + d_{X,K(t)} \quad (20)$$

$$\dot{y}_t = \dot{s}_t \sin(\psi_t) + d_{Y,K(t)}. \quad (21)$$

These drift velocities could be considered merely as relaxation terms to account for model error, or they might instead reflect a real effect in the physical system under consideration. For example, in the case of tracking a ship or an aeroplane, they could represent the current or the wind respectively.

The new state transition function is given by,

$$\begin{aligned} x(t) &= x_{K(t)} + d_{X,K(t)} \Delta t \\ &+ \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2} [a_{N,k} \sin(\psi(t)) + 2a_{T,k} \cos(\psi(t))] \\ &- \frac{\dot{s}_K(t)^2}{4a_{T,k}^2 + a_{N,k}^2} [a_{N,k} \sin(\psi_{K(t)}) + 2a_{T,k} \cos(\psi_{K(t)})] \end{aligned} \quad (22)$$

$$\begin{aligned} y(t) &= y_{K(t)} + d_{Y,K(t)} \Delta t \\ &+ \frac{\dot{s}(t)^2}{4a_{T,k}^2 + a_{N,k}^2} [-a_{N,k} \cos(\psi(t)) + 2a_{T,k} \sin(\psi(t))] \\ &- \frac{\dot{s}_K(t)^2}{4a_{T,k}^2 + a_{N,k}^2} [-a_{N,k} \cos(\psi_{K(t)}) + 2a_{T,k} \sin(\psi_{K(t)})] \end{aligned} \quad (23)$$

Any value of $\mathbf{x}(t)$ can now be achieved by an appropriate choice of $\mathbf{u}_{K(t)}$.

V. 3D INTRINSIC COORDINATE MODEL

The intrinsic coordinate dynamic model is a good reflection of some targets, but so far it has only been described in 2D. In this section, we extend it to 3D.

It is always possible to specify the direction of a tangential acceleration as parallel to the velocity vector. In 2D, the normal acceleration occurs in the direction perpendicular to this, which is also unique. In 3D, there is a whole plane perpendicular to the velocity, so an additional parameter is required to specify the direction of the normal acceleration. Furthermore, it might seem natural to extend the speed-bearing description of velocity to 3D by adding an elevation angle. In fact, this becomes messy and the resulting equations are not solvable. A new approach is required.

We first devise a 3D intrinsic coordinate variable rate model by assuming that motion between changepoints occurs within a plane. Thus, the path followed by the target is a piecewise-planar trajectory.

The state vector is redefined using Cartesian coordinates,

$$\mathbf{x}(t) = [x(t), y(t), z(t), \dot{x}(t), \dot{y}(t), \dot{z}(t)]^T \quad (24)$$

$$= [\mathbf{r}(t)^T, \mathbf{v}(t)^T]^T. \quad (25)$$

We proceed by defining a set of orthogonal unit vectors in the tangential (parallel to the velocity), normal (parallel to the component of the acceleration perpendicular to the velocity), and binormal (perpendicular to the the previous two) directions. These are written as $\mathbf{e}_T(t)$, $\mathbf{e}_N(t)$, and $\mathbf{e}_B(t)$ respectively. The values at the changepoint times are denoted $\mathbf{e}_{T,k} = \mathbf{e}_T(\tau_k)$, and similarly for the normal and binormal cases.

The tangential unit vector is defined first as,

$$\mathbf{e}_T(t) = \frac{\mathbf{v}(t)}{|\mathbf{v}(t)|}. \quad (26)$$

The binormal unit vector specifies the plane in which motion occurs. It is this which is assumed to be fixed between changepoints,

$$\mathbf{e}_B(t) = \mathbf{e}_{B,K(t)}. \quad (27)$$

As noted above, there is an ambiguity in the directions of the normal and binormal vectors. We introduce a new parameter, ϕ_k , which is the angle between $\mathbf{e}_{B,k}$ and the vertical plane containing $\mathbf{e}_{T,k}$. Thus, $\mathbf{e}_{B,k}$ is determined by the following three equations,

Perpendicular to the	$\mathbf{e}_{T,k} \cdot \mathbf{e}_{B,k} = 0$
tangential vector:	
Unit magnitude:	$ \mathbf{e}_{B,k} = 1$
Angle ϕ_k to the	
vertical plane:	$\mathbf{e}_{B,k} \cdot \frac{\mathbf{e}_{T,k} \times \mathbf{k}}{ \mathbf{e}_{T,k} \times \mathbf{k} } = \sin(\phi_k)$

where \mathbf{k} is the z axis unit vector. Solving these, it may be shown that if $\mathbf{e}_{T,k} = [e_{T1,k}, e_{T2,k}, e_{T3,k}]^T$, then the binormal unit vector is given by,

$$\mathbf{e}_{B,k} = \begin{bmatrix} \frac{e_{T2,k} \sin(\phi_k) - e_{T1,k} e_{T3,k} \cos(\phi_k)}{\sqrt{e_{T1,k}^2 + e_{T2,k}^2}} \\ \frac{-e_{T1,k} \sin(\phi_k) - e_{T2,k} e_{T3,k} \cos(\phi_k)}{\sqrt{e_{T1,k}^2 + e_{T2,k}^2}} \\ \cos(\phi_k) \sqrt{e_{T1,k}^2 + e_{T2,k}^2} \end{bmatrix}. \quad (28)$$

Finally, the normal unit vector is given by,

$$\mathbf{e}_{N,k} = \mathbf{e}_{B,k} \times \mathbf{e}_{T,k} \quad (29)$$

It will also be useful to define the speed as the magnitude of the velocity,

$$\dot{s}(t) = |\mathbf{v}(t)|. \quad (30)$$

The system differential equations may now be written in terms of the speed and the unit vectors,

$$\ddot{s}(t) = a_{T,K(t)} \quad (31)$$

$$\dot{\mathbf{e}}_T(t) = \frac{a_{N,K(t)}}{\dot{s}(t)} \mathbf{e}_N(t) \quad (32)$$

$$\dot{\mathbf{e}}_N(t) = -\frac{a_{N,K(t)}}{\dot{s}(t)} \mathbf{e}_T(t) \quad (33)$$

$$\dot{\mathbf{e}}_B(t) = \mathbf{0} \quad (34)$$

Writing these as a matrix differential equation and solving using the integrating factor method yields the following solution,

$$\dot{\mathbf{s}}(t) = \dot{\mathbf{s}}_{K(t)} + a_{T,K(t)}\Delta t \quad (35)$$

$$\begin{bmatrix} \mathbf{e}_T(t)^T \\ \mathbf{e}_N(t)^T \\ \mathbf{e}_B(t)^T \end{bmatrix} = \begin{bmatrix} \cos(\Delta\psi) & \sin(\Delta\psi) & 0 \\ -\sin(\Delta\psi) & \cos(\Delta\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{e}_{T,K(t)}^T \\ \mathbf{e}_{N,K(t)}^T \\ \mathbf{e}_{B,K(t)}^T \end{bmatrix} \quad (36)$$

$$\Delta\psi = \frac{a_{N,K(t)}}{a_{T,K(t)}} \log\left(\frac{\dot{\mathbf{s}}(t)}{\dot{\mathbf{s}}_{K(t)}}\right). \quad (37)$$

The position and velocity may now be recovered from this solution using the following,

$$\mathbf{v}(t) = \dot{\mathbf{s}}(t)\mathbf{e}_T(t) \quad (38)$$

$$\begin{aligned} \mathbf{r}(t) &= \int \mathbf{v}(t)dt \\ &= \mathbf{r}_{K(t)} + \Xi(t) \times \end{aligned} \quad (39)$$

$$\begin{aligned} &[(2a_{T,K(t)}\cos(\Delta\psi) + a_{N,K(t)}\sin(\Delta\psi))\mathbf{e}_T(t) \\ &+ (-a_{N,K(t)}\cos(\Delta\psi) + 2a_{T,K(t)}\sin(\Delta\psi))\mathbf{e}_N(t)] \\ \Xi(t) &= \frac{\dot{\mathbf{s}}(t)^2}{a_{N,K(t)}^2 + 4a_{T,K(t)}^2}. \end{aligned} \quad (40)$$

This completes the model. It is straightforward to recover the 2D version as a special case. It is possible to add linear drift velocities to the parameter vector in exactly the same manner as the 2D model.

Because of the assumption of piecewise-planar motion, the model devised here does not allow for targets with continually varying angle of normal acceleration. On an aeroplane, this corresponds to the roll angle. This system is thus unlikely to cope well with a highly manoeuvrable aeroplane executing a barrel-roll.

VI. 3D MODEL FOR HIGHLY MANOEUVRABLE TARGETS

If space and time, put the complicated model here.

VII. SIMULATIONS

The new intrinsic coordinate models were tested with a VRPF implemented in MATLAB on a benchmark fighter aeroplane trajectory taken from [11], discarding the altitude components for the 2D model. For these tests, the parameters (perpendicular accelerations and drift velocities) are assumed to be independent Gaussian variables. However, any distribution could be used. The standard deviations are $\sigma_T = 10m/s^2$ and $\sigma_N = 50m/s^2$ for the tangential and normal accelerations respectively. For the augmented model, the drift velocities use standard deviation $\sigma_X = \sigma_Y = 10m/s$. The changepoint intervals are assumed to be gamma distributed with shape and scale parameters 7 and 5 respectively.

A radar observation model is used, measuring bearing and range in the 2D case and also elevation in the 3D case, each subject to Gaussian noise. The standard deviation of the angle measurement errors is $\sigma_B = \sigma_E = 0.5^\circ$ and that for the range, $\sigma_R = 100m$. In addition, we consider cases in which the range rate (velocity component in the direction towards the sensor) is

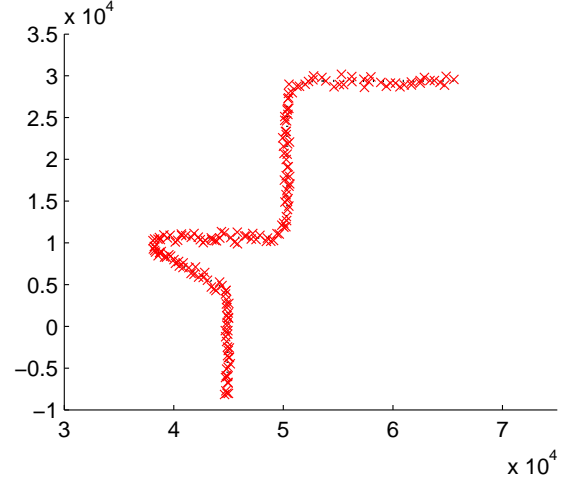


Fig. 1. Benchmark trajectory (dashed) and observations (crosses). Sensor is at (0,0).

measured, also with Gaussian errors with standard deviation $\sigma_{RR} = 10m/s$. Measurements are taken every second. An example realisation of the problem is shown in Fig. 1.

All tests were conducted using resample-move steps. 50 filter particles were used for 2D tests and 200 for 3D tests.

A. 2D Model with Observed Velocity

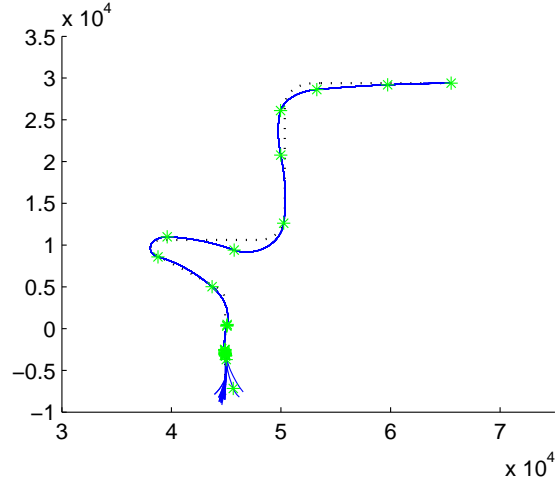
The combination of the basic 2D model and a VRPF has been demonstrated in [6] and has been proven effective. However, when the velocity is observed as well as the position (for example using Doppler radar), this can cause the VRPF to produce worse estimates or even to lose track, despite the fact that there is now more information available. This phenomenon is shown in Fig. 2. By introducing additional drift velocity parameters, the problem is alleviated. The result of applying the augmented 2D model to the same problem is shown in Fig. 3.

The basic and augmented models were tested on 100 realisations of the problem, with and without velocity observations (i.e. using different seeds for the pseudo-random number generator for producing the observations and running the algorithms). The mean error performance in position and velocity is shown in table ???. Two measures of error are compared: one of the filter state estimate and the other of a basic smoothing estimate - calculating the entire state trajectory from the final filter estimate, in the similar manner to the filter-smoother of [12].

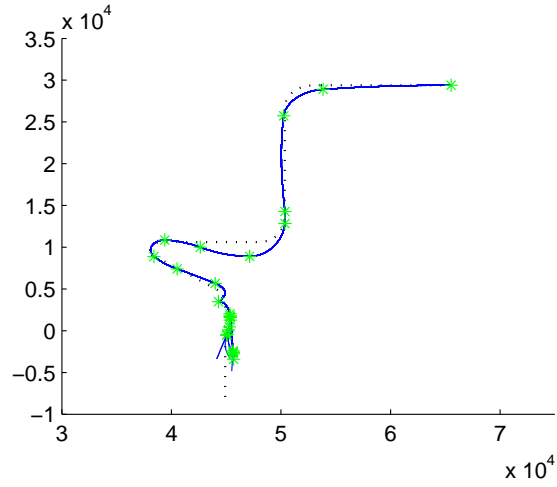
The augmented model results in a significant reduction in estimation errors, especially when the velocity is observed. Note that if the precision of the range rate measurements is increased then track loss becomes more common, and begins to affect the augmented model as well.

B. 3D Intrinsic Coordinate Model

Here we demonstrate the 3D intrinsic coordinate model, and compare it to the Cartesian equivalent used in [7], [13]. The same trajectory is used as before, but this time the



(a) Without velocity observations



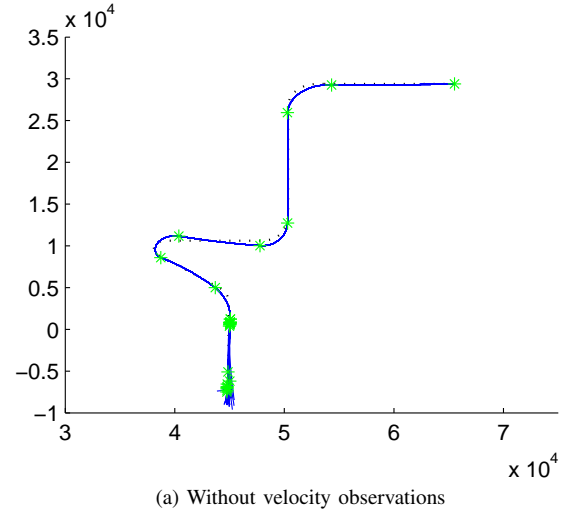
(b) With velocity observations

Fig. 2. Variable rate particle filter results using basic 2D model, with and without range rate observations. Dotted line shows ground truth. Solid lines shown particle estimates. Stars show particle changepoints.

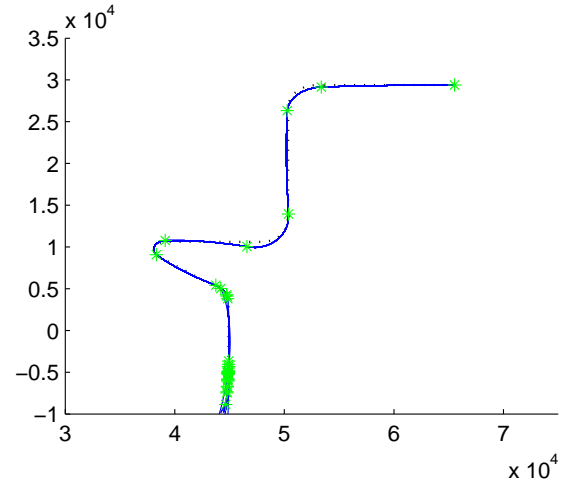
TABLE I

2D VRPF TRACKING PERFORMANCE OF BASIC AND AUGMENTED INTRINSIC COORDINATE MODELS WITH AND WITHOUT VELOCITY OBSERVATIONS

Model	Observed Velocity?	Filter RMSEs		Smoother RMSEs	
		Position (m)	Velocity (m/s)	Position (m)	Velocity (m/s)
Basic	no	884	188	476	65
Basic	yes	1715	174	1156	105
Augmented	no	773	191	314	52
Augmented	yes	1392	170	734	76



(a) Without velocity observations



(b) With velocity observations

Fig. 3. Variable rate particle filter results using augmented 2D model, with and without range rate observations. Dotted line shows ground truth. Solid lines shown particle estimates. Stars show particle changepoints.

TABLE II

3D VRPF TRACKING PERFORMANCE OF BASIC INTRINSIC COORDINATE AND CARTESIAN MODELS

Model	Observed Velocity?	Filter RMSEs		Smoother RMSEs	
		Position (m)	Velocity (m/s)	Position (m)	Velocity (m/s)
Intrinsic	no				
Cartesian	no				

altitude components are not discarded. An example realisation is shown in Fig. 4.

The results of applying the basic 3D intrinsic coordinate and Cartesian models to this data set are shown in Fig. 5.

Testing on 100 realisations, the average error performance is shown in table ??.

The intrinsic coordinate model consistently outperforms the Cartesian version.

There is plenty of scope for improving these models.

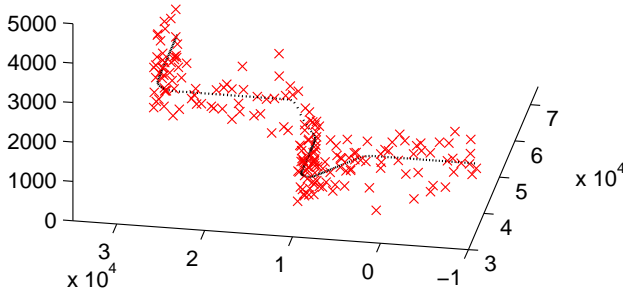
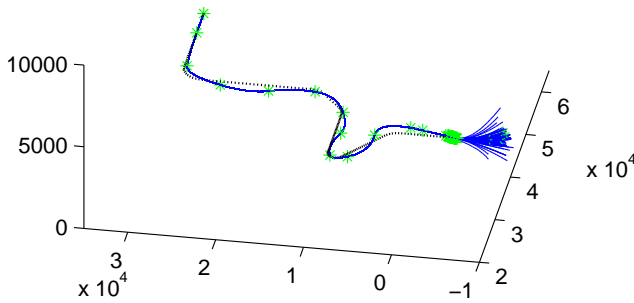
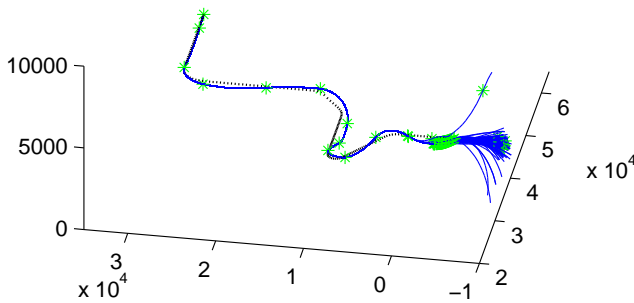


Fig. 4. Benchmark trajectory (dashed) and observations (crosses). Sensor is at (0,0).



(a) Cartesian



(b) Intrinsic

Fig. 5. Variable rate particle filter results using Cartesian and intrinsic coordinate dynamic models. Dotted line shows ground truth. Solid lines shown particle estimates. Stars show particle changepoints. Note the different scale in the z axis leading to greatly exaggerated errors in altitude.

Clearly, there is more prior information about aircraft in flight which could be incorporated. For example, a predilection for level flight could be quantified through the choice of the distribution of ϕ_k , the angle of the plane of motion, which is currently uniform.

VIII. CONCLUSION

Dynamic tracking models for use with variable rate particle filters have been developed. Variable rate models treat the state as a deterministic continuous-time process conditional on a sequence of changepoints and motion parameters. Previously, a system has been developed which models targets as particles moving under the influence of tangential and normal accelerations. These have been developed in two ways. First, additional parameters have been introduced to improve the filter performance when the velocity is observed, for example through the use of Doppler measurements. Second, the model has been extended for use in 3D, where it achieves better state estimation performance than a Cartesian model.

REFERENCES

- [1] 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.
- [2] B. Anderson and J. Moore, *Optimal filtering*. Prentice-Hall (Englewood Cliffs, NJ), 1979.
- [3] 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.
- [4] 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>
- [5] 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>
- [6] 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.
- [7] N. Whiteley, A. M. Johansen, and S. Godsill, "Monte carlo filtering of piecewise deterministic processes," *Journal of Computational and Graphical Statistics*, vol. 20, no. 1, pp. 119–139, 2011. [Online]. Available: <http://pubs.amstat.org/doi/abs/10.1198/jcgs.2009.08052>
- [8] M. Jacobsen, *Point process theory and applications: marked point and piecewise deterministic processes*, C. Newman and S. Resnick, Eds. Birkhauser, 2006.
- [9] 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>
- [10] R. Best and J. Norton, "A new model and efficient tracker for a target with curvilinear motion," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 33, no. 3, pp. 1030–1037, July 1997.
- [11] W. Blair, G. Watson, T. Kirubarajan, and Y. Bar-Shalom, "Benchmark for radar allocation and tracking in ecm," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 34, no. 4, pp. 1097–1114, Oct 1998.
- [12] 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. 1–25, 1996. [Online]. Available: <http://www.jstor.org/stable/1390750>
- [13] N. Whiteley, A. M. Johansen, and S. Godsill, "Efficient monte carlo filtering for discretely observed jumping processes," in *Statistical Signal Processing, 2007. SSP '07. IEEE/SP 14th Workshop on*, Aug. 2007, pp. 89–93.