

PARTICLE FILTERS FOR CONTINUOUS-TIME JUMP MODELS IN TRACKING APPLICATIONS *

SIMON GODSILL¹

Abstract. In this article we summarise recent work in modelling and estimation of continuous-time jump models for application in tracking scenarios. The models are constructed such that random jumps occur in the driving function (typically the applied force on an object being tracked) at random times, and in general asynchronously with the observation times. The sojourn times between jumps are modelled as general distributions (here gamma or shifted gamma), and hence we are in the class of semi-Markov models, since the arrival times do not form a Poisson point process (see [21] for an overview of such models in the tracking setting). In contrast with other such models in the tracking literature we allow a fully continuous random set of manoeuvre parameters, rather than a discrete set of switching models, and deterministic paths during the sojourns, obeying a set of nonlinear kinematic equations for point mass motion, thus modelling the path of the object in a smooth and parsimonious fashion. These models are aimed at capturing the highly random manoeuvres of real objects in a simple way. Estimation is carried out using a Variable Rate Particle Filter (VRPF) that parameterises the model explicitly in terms of the jump times and their parameters [12, 13]. Extensions to the models and algorithms are also presented that allow for a diffusion component of the model, which captures continuous random disturbances to the object in addition to jumps. These are illustrated in a 3-dimensional linear Gaussian setting where the entire path, except for jump times, may be marginalised, hence making a more efficient and effective particle filter.

INTRODUCTION

In this paper we give a brief overview of recent work in tracking of highly manoeuvrable objects by the use of nonlinear jump processes and particle filters. The models developed are expressed in a natural intrinsic coordinate system which, it is postulated, represent realistic conditions more accurately (see [21] for evidence to back up this claim). New models and algorithms are presented for linear jump processes that also contain a Brownian driving component, and a fully Rao-Blackwellised particle filtering implementation is demonstrated for this setup. Initial proof-of-principle results are given for simulated trajectories.

* The work of the author is partially funded by the EPSRC project *Bayesian Inference for Continuous Time Diffusions*, and by the UK's Data and Information Fusion Defence Technology Centre

¹ Signal Processing and Communications Group
University of Cambridge

1. MODELS - GENERAL NONLINEAR CASE

1.1. Dynamical models

The class of models we consider initially comprises continuous-time trajectories having piecewise deterministic sections. The type of models proposed fits closely with the general class of piecewise deterministic processes, see [5], although we do not need to restrict to the Markovian case. In our models the k th section, $k = 0, 1, \dots$, is parameterised by a time τ_k at which the section commences ($\tau_k > \tau_{k-1} > \dots$), and the parameters of the manoeuvre θ_k . A Markovian update on the k scale is defined for the combined k th state $x_k = [\theta_k, \tau_k]$:

$$x_k \sim f(x_k | x_{k-1}) = f_\theta(\theta_k | \theta_{k-1}, \tau_k, \tau_{k-1}) f_\tau(\tau_k | \theta_{k-1}, \tau_{k-1}). \quad (1)$$

The distribution $f_\tau(\tau_k | \theta_{k-1}, \tau_{k-1})$ is quite general, subject to the constraint $\tau_k > \tau_{k-1}$, although we do constrain ourselves to *finite activity* processes, i.e. those having a finite number of manoeuvre points τ_k in any finite time interval. We have typically used gamma, or shifted gamma, distributions for the inter-arrival times. Note that the parameters of the time distributions can explicitly depend on the manoeuvre parameters θ_k , and could themselves be made non-Markovian if required.

Having defined a skeleton for the process in this way in terms of a countable collection of states $\{x_k\}_{k=0}^\infty$, the continuous-time path of the process is defined as a deterministic function of these states. In general for models of this type the path could be a function of several neighbouring state variables, see [12, 13]. Here however we restrict ourselves to causal constructions in which the path at any time $t \geq \tau_0$ depends only upon the two closest state points, $\mathcal{N}_t = \{k, k-1; \tau_{k-1} < t \leq \tau_k\}$. Note that the underlying model here is a type of *Marked point process*, and hence the inference schemes proposed are for filtering of marked point process dynamical models, see [5, 6, 19] for example.

1.1.1. Example: Intrinsic Coordinates Motion Model [13]

To give a specific example, we construct the state variable θ_k for an object undergoing planar motion, subject to applied forces $T_{T,k}$ (tangentially to motion) and $T_{P,k}$ (perpendicular to the motion), as

$$\theta_k = [T_{T,k}, T_{P,k}, v(\tau_k), \psi(\tau_k), z(\tau_k)]$$

where $v(\tau_k)$ is the speed at time τ_k , $\psi(\tau_k)$ is the heading angle relative to a fixed coordinate axis, and $z(\tau_k)$ is the Cartesian position of the object in the fixed Cartesian coordinate frame. This model is distinct from others in the area in that forces are applied relative to the *body frame* of the object, which is regarded as a more realistic model for motion of objects under self-forcing [21]. These models are however avoided in standard settings owing to the high degree of nonlinearity involved. In the particle filter implementation this is not an issue and we may go for greater realism. The kinematic equations for the object between times τ_k and τ_{k+1} are then obtained from standard considerations as

$$T_{T,k} = \lambda \frac{ds}{dt} + m \frac{d^2s}{dt^2}, \quad T_{P,k} = m \frac{ds}{dt} \frac{d\psi}{dt}, \quad \tau_k \leq t < \tau_{k+1}$$

where $s(t)$ is the total distance moved along the path from time 0. The parameters m and λ denote the mass of the object and the linear coefficient of resistance, respectively. Note that a linear resistance model is unrealistic for many scenarios, but has been found to be a reasonable approximation for the trajectories that we have so far encountered. Current research is investigating extensions that model a more realistic *square law* effect in the resistance, for which closed form solutions to the differential equation also exist. The model as posed above falls within the general class of *curvilinear models* [21], which are some of the most powerful generic models¹ of target motion currently available.

¹By ‘generic’ we mean that the kinematics are not obtained from a direct modelling of the dynamics of a particular aircraft or boat, for example.

In order to obtain the interpolated value of the path $z(t)$, $\tau_{k-1} \leq t < \tau_k$, the kinematic equations are solved as standard coupled differential equations. The tangential equation is readily integrated from time τ_k to $\tau_k + \Delta\tau$ to give the speed $v(t)$ along the path at time $t = \tau_k + \Delta\tau$:

$$v(\tau_k + \Delta\tau) = \frac{1}{\lambda} \left(T_{T,k} - (T_{T,k} - \lambda v(\tau_k)) e^{-\Delta\tau\lambda/m} \right) \quad (2)$$

and the distance moved along along the path:

$$s(\tau_k + \Delta\tau) = s(\tau_k) + \frac{\Delta\tau}{\lambda} T_{T,k} + \frac{m}{\lambda^2} (T_{T,k} - \lambda v(\tau_k)) (e^{-\Delta\tau\lambda/m} - 1) \quad (3)$$

Then using the above expression for the speed $v(t)$, the perpendicular equation is rearranged and integrated to give:

$$\begin{aligned} \frac{d\psi}{dt} &= \frac{T_{P,k}}{mv(t)}, \quad \tau_k \leq t < \tau_{k+1}, \quad v(t) \neq 0 \\ \psi(\tau_k + \Delta\tau) &= \psi(\tau_k) + \int_{\tau_k}^{\tau_k + \Delta\tau} \frac{T_{P,k} dt}{mv(t)} \\ &= \psi(\tau_k) + T_{P,k}/T_{T,k} \left(\Delta\tau\lambda/m - \log \left| \frac{v(\tau_k)}{v(\tau_k + \Delta\tau)} \right| \right) \end{aligned} \quad (4)$$

The results thus far are exact and in closed form, subject to the assumption that the forcing is constant between time points τ_k and τ_{k+1} . To determine the Cartesian position $z(t) = [x(t) \ y(t)]^T$ of the object, however, appears to require a numerical integration procedure². We achieve this by a simple Euler approximation on a fine time grid, calculating the changes in x - and y - coordinates over a time interval δt as:

$$\delta x \approx v(t) \cos(\psi(t)) \delta t, \quad \delta y \approx v(t) \sin(\psi(t)) \delta t, \quad z(t + \delta t) \approx z(t) + [\delta x, \ \delta y]^T \quad (5)$$

The model is then completed by assigning a probability distribution to the thrust parameters $T_{T,k}$ and $T_{P,k}$. We have used independent Gaussian distributions for these, although a correlated Gaussian may also have useful properties (small forward thrust might imply small sideways thrust, for example).

See Figs. 1 and 2 for simulated data from our dynamical model, comparing the more standard fixed rate setup (jumps arrive at regularly spaced time points) with the variable rate setup described here.

An example trajectory from the model is shown in Fig. 3, showing its clear ability to generate elaborate manoeuvres containing persistent turns of various types.

1.2. Observation Models

The variable rate dynamical models are then linked with an observation process, whose arrival times are not chosen by us in this application. For convenience assume regularly spaced data at times $t = 0, 1, 2, \dots$. The likelihood is then specified in terms of the interpolated data z_t :

$$y_t \sim g(y_t | x_{0:\infty}) = g(y_t | z_t).$$

where $g()$ is some known probability distribution for measurements given the path of the process at time t .

The distribution $g()$ is determined by the sensor characteristics. In standard tracking for radar, sonar, etc., the sensor is characterised by false alarms as well as missed detections, combining to give a point process of some kind. Similar features occur in video tracking when tracking is based on point features extracted from the raw image data.

²Under the simpler model where $\lambda = 0$, i.e. there is no resistance force, the Cartesian position can also be obtained in closed form.

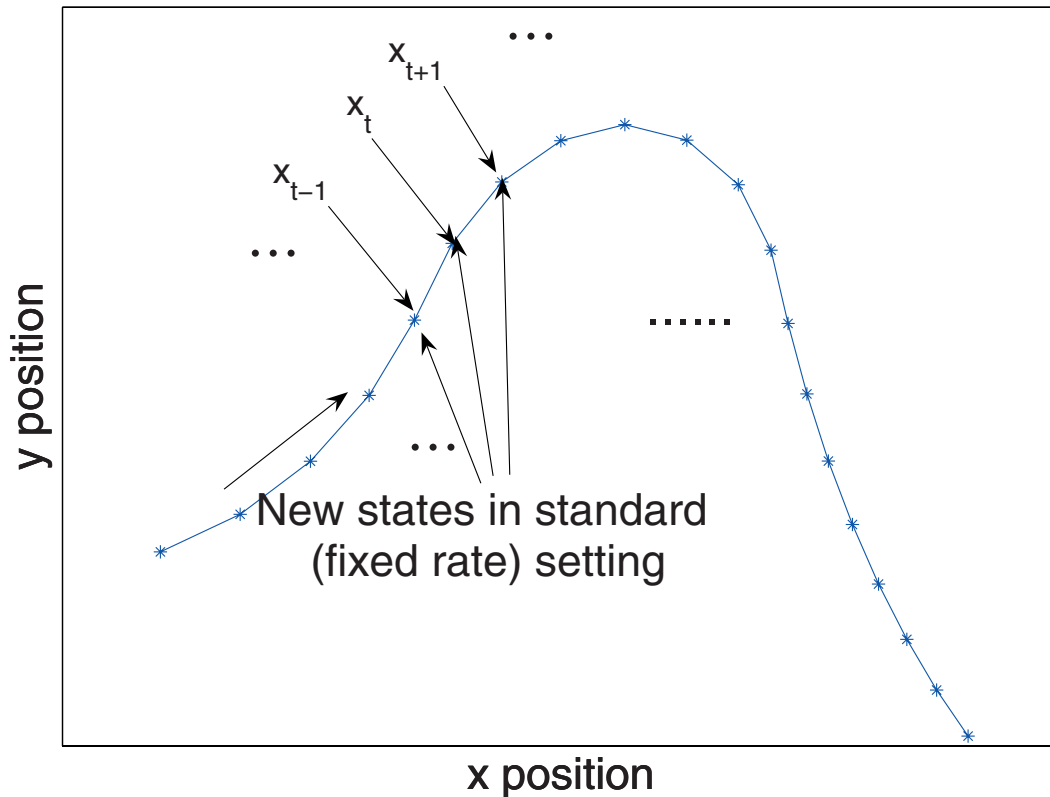


FIGURE 1. Standard (fixed-rate) state sequence

1.2.1. Example: Poisson Measurement Models [9, 10]

In this model a nonhomogeneous spatial Poisson point process is assumed as the model for both object-originating measurements and false-alarm ('clutter') measurements. In this way certain classical problems with data association [1] are bypassed, since the union property of two independent Poisson processes leads straightforwardly to a third Poisson process with well-defined intensity function. In this model the number of measurements from each target at each time point are assumed drawn randomly from a Poisson distribution having mean λ_T . Each such target measurement, say y , is assumed drawn from a sensor distribution $p_T(y|z)$ which is typically centered on the target state z , and may model the spatial extent and shape of the object/group (time index omitted for simplicity). Random clutter measurements are also included, whose number is again a Poisson random variable having mean λ_C and uniform or non-uniform spatial distribution $p_C(y)$. A useful consequence of this model is that the resulting random set of target and clutter measurements is also a Poisson point process having mean number of points $\lambda_T + \lambda_C$ and distribution of individual points given by

$$\frac{(\lambda_T p_T(y|z) + \lambda_C p_C(y))}{\lambda_T + \lambda_C}$$

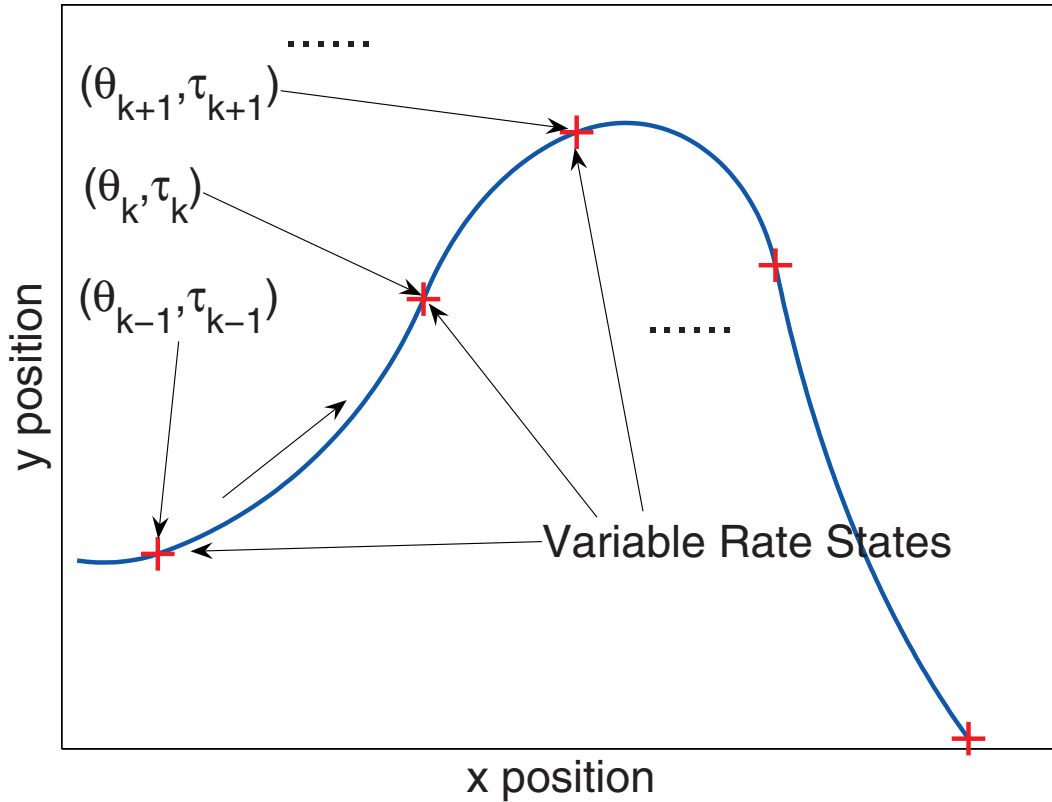


FIGURE 2. Variable rate state sequence

(see e.g. [4, 16]). Hence one may immediately write down the likelihood function for the observation values $y_t = [y_{1,t}, \dots, y_{M_t,t}]$ and their number M_t as [9]

$$g(y_t, M_t | z_t) = \frac{\exp(-(\lambda_T + \lambda_C))}{M_t!} \prod_{i=1}^{M_t} (\lambda_T p_T(y_{i,t} | z_t) + \lambda_C p_C(y_{i,t}))$$

where M_t is the total number of target and clutter measurements at time t . This model thus avoids any explicit treatment of the data association problem inherent in many tracking scenarios, but can potentially handle very high clutter densities. The model is typically used for tracking of group or extended targets, where the target is expected to make a random number of returns from random scatterer positions, or randomly from different objects in the group. The target distribution $p_T(y|z)$ can model the shape and extent of the group, and could itself have parameters that evolve over time. With small mean target detection number λ_T the models may also be used successfully for modelling of small or spatially concentrated targets that exhibit random multipath effects or occasional multiple returns.

In the models considered thus far, the target distributions $p_T(y|z)$ are Gaussian, with fixed mean and covariance, but may be parameterised as nonlinear functions of the state variable (e.g. bearings-only or range-and-bearings measurements).

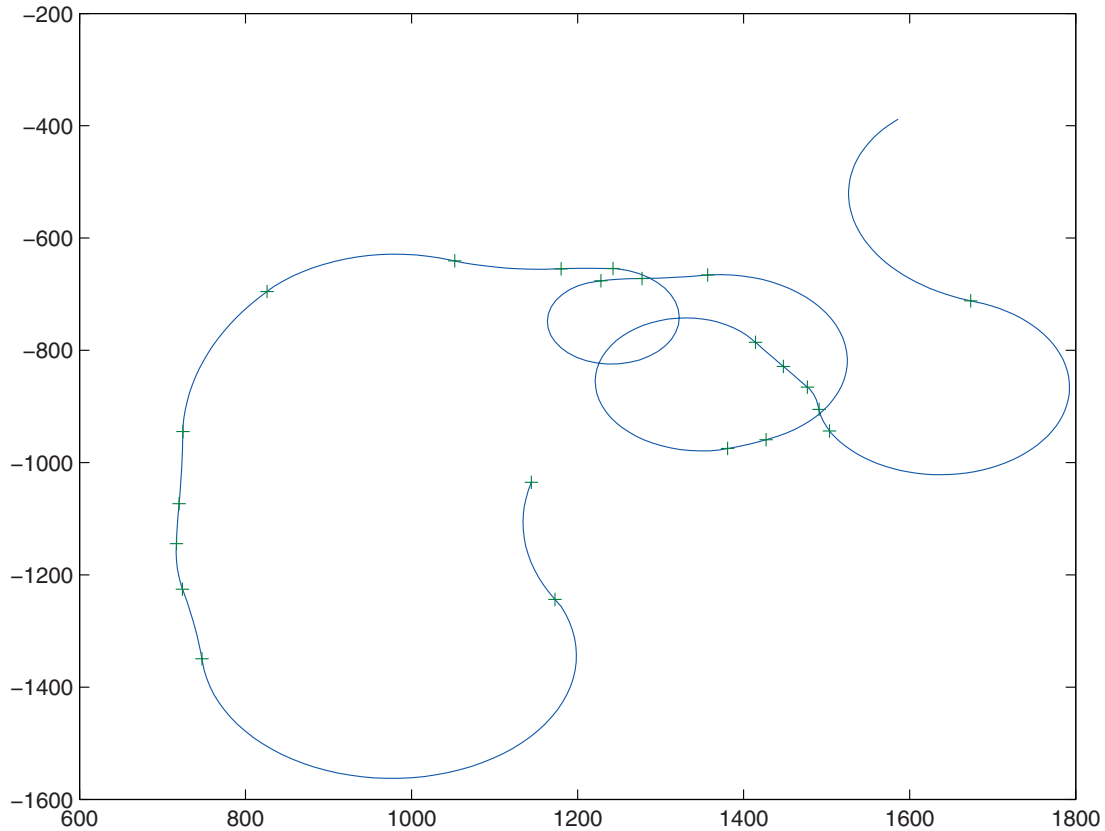


FIGURE 3. Example trajectory from dynamical model, $t = 0, \dots, 400$. Jump instants shown as '+'.

2. LINEAR SDE MODELS AND RAO-BLACKWELLISATION

The above models are inherently nonlinear, which has advantages for realism, however disadvantages in computational algorithms (many particles may be required to infer the filtering distributions). The question remains as to whether we can construct powerful continuous-time tracking models that obey linear dynamics. If so, then Rao-Blackwellisation (marginalisation) techniques can profitably be used to enhance the inference algorithms [8, 17, 22]. Here we propose a hybrid model that incorporates jumps at random times with a linear dynamical model for the jump parameters. Within the linear framework it is also straightforward to include a diffusion element in the model, allowing for external random disturbances on the object as well as discrete jump-type manoeuvres. The hybrid model is still able to produce an interesting range of behaviours, suitable for modelling of manoeuvring objects. A related set of models have been studied by [24], in which large manoeuvres are modelled as short bursts of high variance driving noise to a continuous-time (near constant-velocity) model, according to a two-state semi-Markov model. In [24] the models are being used to classify the type of object according to the sojourn-time distribution, which varies across different object types. Our models are distinct from these in that we model new manoeuvres explicitly by the jumps of a continuous-time process rather than by

a new sojourn having high driving noise. Nevertheless, the generic structure of the resulting Rao-Blackwellised particle filter algorithm has much in common with ours presented here³.

Consider first a 1-dimensional motion. 2- or 3-dimensional motions will be obtained by cascading several one-dimensional processes. A basic 1-dimensional motion model takes the following form:

$$T_t = \lambda \dot{z} + m \ddot{z}$$

where $\lambda \geq 0$ is a motion resistance term and $m > 0$ is the mass of the object as before, and $\{T_t\}$ is a stochastic forcing term. In this section $z(t)$ is interpreted as the Cartesian displacement of the object from the origin. We now assume that the stochastic forcing function is comprised of Brownian motion plus pure jumps:

$$T_t = \sigma_z B_t + W_t$$

where B_t is Brownian motion and W_t is a pure jump process constructed from a marked point process $\{\tau_k, J_k\}$, where $\{\tau_k\}$ are jump arrival times and $\{J_k\}$ are the corresponding jump sizes, respectively. Then, W_t is obtained from the point process in the following way,

$$W_t = \sum_{\tau_k \leq t} J_k, \quad \tau_k \leq t < \tau_{k+1}$$

We will require shortly the increments of this process, which are given by:

$$dW_t = \begin{cases} J_k, & t = \tau_k \\ 0, & \text{otherwise} \end{cases}$$

We will assume for now that the jump sizes $\{J_k\}$ are independent and normally distributed,

$$J_k \sim \mathcal{N}(\mu_J, \sigma_J^2)$$

(there are interesting possible extensions of the algorithms to scale mixtures of normals by including a latent scale parameter along with each jump parameter [7, 11, 23]). Note that this type of model has been proposed in the finance area for the pricing of options [25, 20], although analyses in that area tend to rely on Poisson point process arrivals and univariate continuous time models.

Jump times are drawn from a finite activity process, i.e. finitely many jumps occur in finite time intervals almost surely (we will in fact ensure this by using a gamma or shifted gamma distribution for the inter-arrival time distribution for jumps).

We thus have a linear SDE, as follows

$$d\ddot{z} = -\frac{\lambda}{m} \ddot{z} dt + \frac{1}{m} dT_t$$

which is seen to be an Ornstein-Ühlenbeck (OU) process in the acceleration \ddot{z} .

We now write this as a continuous-time state-space model,

$$d\mathbf{z}_t = A\mathbf{z}_t dt + \mathbf{h} dT_t$$

where:

$$\mathbf{z}_t = \begin{bmatrix} z_t \\ \dot{z}_t \\ \ddot{z}_t \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{\lambda}{m} \end{bmatrix} \quad \mathbf{h} = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m} \end{bmatrix}$$

³We thank Simon Maskell for helpful suggestions regarding the regeneration of variable rate states that lie beyond the current time t

This linear model has a simple solution, starting at state \mathbf{z}_s at time $s < t$:

$$\begin{aligned} \mathbf{z}_t &= e^{A(t-s)}\mathbf{z}_s + \int_s^t e^{A(t-u)}\mathbf{h}dW_u, \quad t > s. \\ &= e^{A(t-s)}\mathbf{z}_s + \int_s^t e^{A(t-u)}\mathbf{h}dW_u + \int_s^t e^{A(t-u)}\mathbf{h}\sigma_z dB_u \\ &= e^{A(t-s)}\mathbf{z}_s + \sum_{k:s < \tau_k \leq t} J_k e^{A(t-\tau_k)}\mathbf{h} + \int_s^t e^{A(t-u)}\mathbf{h}\sigma_z dB_u \end{aligned}$$

where the middle term in the last line arises as the integral of the increments in the pure jump component.

Since the $\{J_k\}$ are independent and Gaussian, and $\{B_u\}$ is Brownian motion, the Markovian state transition density, conditional upon the jump times, is Gaussian and may be obtained directly. Calculations are relatively straightforward, but note that A has repeated eigenvalues at zero; hence some special consideration is required and matrix exponentials are performed directly using Taylor series expansions. Then we obtain

$$f(\mathbf{z}_{t+\delta t}|\mathbf{z}_t, \{\tau_k; t < \tau_k \leq t + \delta t\}) = N(\mu_{t+\delta t|t}, C_{t+\delta t|t})$$

with

$$\begin{aligned} \mu_{t+\delta t|t} &= e^{A\delta t}\mathbf{z}_t + \sum_{k:t < \tau_k \leq t+\delta t} \mu_j e^{A(t+\delta t-\tau_k)}\mathbf{h} \\ C_{t+\delta t|t} &= \sigma_j^2 \sum_{k:t < \tau_k \leq t+\delta t} e^{A(t+\delta t-\tau_k)}\mathbf{h}\mathbf{h}^T e^{A(t+\delta t-\tau_k)T} + \sigma_z^2 \int_s^t e^{A(t-u)}\mathbf{h}\mathbf{h}^T e^{A(t-u)T} du \end{aligned}$$

Note again that this is in linear Gaussian form. When this is connected to a linear Gaussian observation model the system may thus be solved using linear Gaussian state-space theory, i.e. the Kalman filter. The jump times are not known, however, so these can be inferred using a marginalised, or *Rao-Blackwellised* particle filter.

3. INFERENCE USING PARTICLE FILTERS

3.1. General Nonlinear Case

The processes described above are semi-Markov since the inter-arrival distribution for jumps is not exponential in general. Particle filter inference is thus performed directly in terms of sequential updating of the jump times and their parameters, which we term a *variable rate particle filter* [12, 13, 15]. We consider as state variable at time t the current neighbourhood of states $x_{\mathcal{N}_t} = \{x_k, x_{k-1}; \tau_{k-1} < t \leq \tau_k\}$. The particle filter samples new states as required in order to maintain a complete neighbourhood for all particles at time t . Since it is only necessary in fact to know that $\tau_k^{(i)} > t$ for all particles $i = 1, \dots, N$, new values of τ_k and θ_k are only generated when required, in order to save computation and to introduce diversity between different particles. Within the different setting of a discrete time Hidden Markov model a related scheme has been proposed by [3]. The algorithm is presented from the selection step up to the weighting step. The selection step has deterministically produced N_i offspring for particle i and the time update is from $t-1$ to t , and hence the weight for each is diluted by N_i for replications of particle i . Other more standard fully or partly randomised schemes can of course be used instead, with appropriate weight calculations. The version here uses the prior dynamical model for proposing new state values, although more general proposals are possible as in standard particle filters. The drawing of Binomial random variables to choose how many particles will have a new τ_k value generated between $t-1$ and t can be replaced by a deterministic step to reduce Monte Carlo variance, appropriately weighted as necessary. The algorithm takes as its starting point the particle representation at time $t-1$ having particles $x_{\mathcal{N}_{t-1}}^{(i)}$ and weights $w_{t-1}^{(i)}$. The notation \mathcal{N}_t^+ denotes the final element of the neighbourhood at time t ,

i.e. $\mathcal{N}_t^+ = \min\{k; \tau_k > t\}$. The term $f_\tau(\tau_k | \tau_{k-1}^{(i)}, t-1 < \tau_k \leq t)$ denotes the dynamical model truncated so that $t-1 < \tau_k \leq t$. Algorithm 1 summarises the algorithm as it applies to a general nonlinear dynamical model and sensor distribution $g(\cdot)$.

3.2. Linear Gaussian Case

In the special case that the dynamical model is linear and Gaussian, as in Section 2, and also the observation model, i.e.

$$y_t = G_t \mathbf{z}_t + v_t, \quad v_t \sim N(0, C_v),$$

the same basic filtering structure applies but now we only simulate τ_k and not the manoeuvre parameter variable θ_k . Instead the path of the process is marginalised conditional on the jump times $\{\tau_k\}$ and we store at each time, for each particle, the sufficient statistics for the Gaussian distributed state variable \mathbf{z}_t :

$$p(\mathbf{z}_t^{(i)} | y_{0:t-1}, \tau_{0:\mathcal{N}_t^+}^{(i)}) = N(\mathbf{z}_t | \mu_{t|t-1}^{(i)}, C_{t|t-1}^{(i)})$$

where $\mu_{t|t-1}^{(i)}$ and $C_{t|t-1}^{(i)}$, the predictive means and covariance, are sequentially updated for each particle using Kalman filtering recursions [8, 2]. The weight function is then updated according to the modified formula:

$$w_t^{(i)(j)} = w_{t-1}^{(i)(j)} p(y_t | y_{0:t-1}, \tau_{0:\mathcal{N}_t^+}^{(i)(j)})$$

The predictive density $p(y_t | y_{0:t-1}, \tau_{0:\mathcal{N}_t^+})$ is obtained from the prediction error decomposition of the Kalman filter (see e.g. [18]), i.e.

$$p(y_t | y_{0:t-1}, \tau_{0:\mathcal{N}_t^+}) = N(y_t | G_t \mu_{t|t-1}, G_t C_{t|t-1} G_t^T + C_v)$$

4. EXAMPLES

We present here brief examples for the tracking of an object in three dimensions using the linear Gaussian model of Section 2. For examples in the nonlinear models of Section 1.1.1, see our previous work [13, 9, 14, 15]. Consider a dynamical model according to Section 2 running independently in each of the three dimensions but coupled by identical $\{\tau_k\}$ values in all dimensions. The parameters of the models in each dimension are: $\sigma_z = 1000$, $\mu_J = 0$, $\sigma_J = 50000$. The jump process obeys a shifted gamma inter-arrival time, i.e.

$$\tau_k - \tau_{k-1} - \tau_{\min} \sim G(\alpha_\tau, \beta_\tau)$$

with $\tau_{\min} = 4$, $\alpha_\tau = 0.5$ and $\beta_\tau = 4$. The initial state in each dimension is centered at $\mathbf{z}_0 = [0, 0, 0]^T$, with a Gaussian random perturbation of covariance $100^2 I$ and with initial jump time $\tau_0 = 0$. An example trajectory from such a process over a time interval $[0, 200]$ is given in Fig. 4. The total number of manoeuvres is 33 and it is clear that many manoeuvres have generated sharp turns and behaviour not characteristic of a standard linear Gaussian model (i.e. without jumps). Observations are generated from an additive Gaussian model in each dimension as

$$y_t = z_t + v_t, \quad v_t \sim N(0, \sigma_v^2), \quad \sigma_v = 5 \times 10^8$$

The observations are shown in Fig. 5. Rao-Blackwellised particle filtering using $N = 400$ particles is now performed to estimate the value of the state with time. The filtered (posterior mean) estimate is presented in Fig. 6, showing a good agreement with the true trajectory (shown dotted), considering the high levels of noise. The three two-dimensional projections are presented in panels 2-4 of this figure, including confidence ellipses from the Rao-Blackwellised particle filter. While it was found to be possible to track the data using a model without jumps ($\sigma_J = 0$), significantly greater mean-squared tracking errors resulted, even when the value of σ_z was carefully 'retuned'. Note that much smoother and potentially more accurate estimates can be obtained if a Kalman smoother is run for each particle to generate smoothed posterior mean estimates for the

ALGORITHM 1. Variable Rate Particle Filter

Select number of offspring N_i for particle i , $\forall i \in \{1, 2, \dots, N\}$.

for $i = 1, \dots, N$ **do**

Draw a binomial random variable N_0 according to N_i trials, each with probability

$$\alpha = \frac{\int_t^\infty f_\tau(\tau_k | \theta_{k-1}^{(i)}, \tau_{k-1}^{(i)}) d\tau_k}{\int_{t-1}^\infty f_\tau(\tau_k | \theta_{k-1}^{(i)}, \tau_{k-1}^{(i)}) d\tau_k}$$

▷ Choose how many particles require a new $\tau_k \in [t-1, t)$.

$N_1 = N_i - N_0$. ▷ N_1 is the number of particles for which a new $\tau_k \in [t-1, t)$ should be generated.

Reset $N_i = N_1 + 1$. ▷ Collapse particles for which no new τ_k is generated.

Assign N_i offspring to each particle:

$$x_{\mathcal{N}_{t-1}}^{(i)(j)} = x_{\mathcal{N}_{t-1}}^{(i)}, \quad w_{t-1}^{(i)(j)} = \frac{w_{t-1}^{(i)}}{N_i}, \quad \forall j \in \{1, \dots, N_i\}.$$

Increase weighting for ‘collapsed’ particle:

$$w_{t-1}^{(i)(N_1+1)} \leftarrow N_0 \times w_{t-1}^{(i)(N_1+1)}$$

for $j = 1, \dots, N_1$ **do**

▷ Sample paths that enforce $\tau_k \in [t-1, t)$.

Set $k = \mathcal{N}_{t-1}^+{}^{(i)}$.

Sample $\tau_k^{(i)(j)} \sim f_\tau(\tau_k | \tau_{k-1}^{(i)}, t-1 < \tau_k \leq t)$ (use rejection sampling, or inverse cdf mapping).

Sample $\theta_k^{(i)(j)} \sim f_\theta(\theta_k | \tau_k^{(i)(j)}, \mathbf{x}_{k-1}^{(i)})$.

while $(\tau_k^{(i)(j)} < t)$ **do**

$k \leftarrow k + 1$

Sample $x_k^{(i)(j)} \sim f(x_k | x_{k-1}^{(i)(j)})$

end while

end for

for $j = 1, \dots, N_i$ **do**

Calculate interpolated state value, $z_t^{(i)(j)}$ from $x_{\mathcal{N}_t}^{(i)(j)}$.

Update weight:

$$w_t^{(i)(j)} = w_{t-1}^{(i)(j)} g(y_t | z_t^{(i)(j)})$$

end for

end for

Restack the particles and weights from the replicates so that particle $(i)(j)$ becomes particle $(j + \sum_{i' < i} N_{i'})$.

Renormalise the weights such that $\sum_{i=1}^{\sum_{i=1}^N N_i} w_t^{(i)} = 1$.

entire trajectory. This is not carried out here but would form a useful addition to future results. An interesting diagnostic obtainable from the particles is an estimate of the probability of number of manoeuvres in the entire time interval, $p(\mathcal{N}_t^+ | y_{0:t})$. This is obtained by histogramming the values of $\mathcal{N}_t^+{}^{(i)}$, as shown in Fig. 7, which is in remarkably good agreement with the true value of 33.

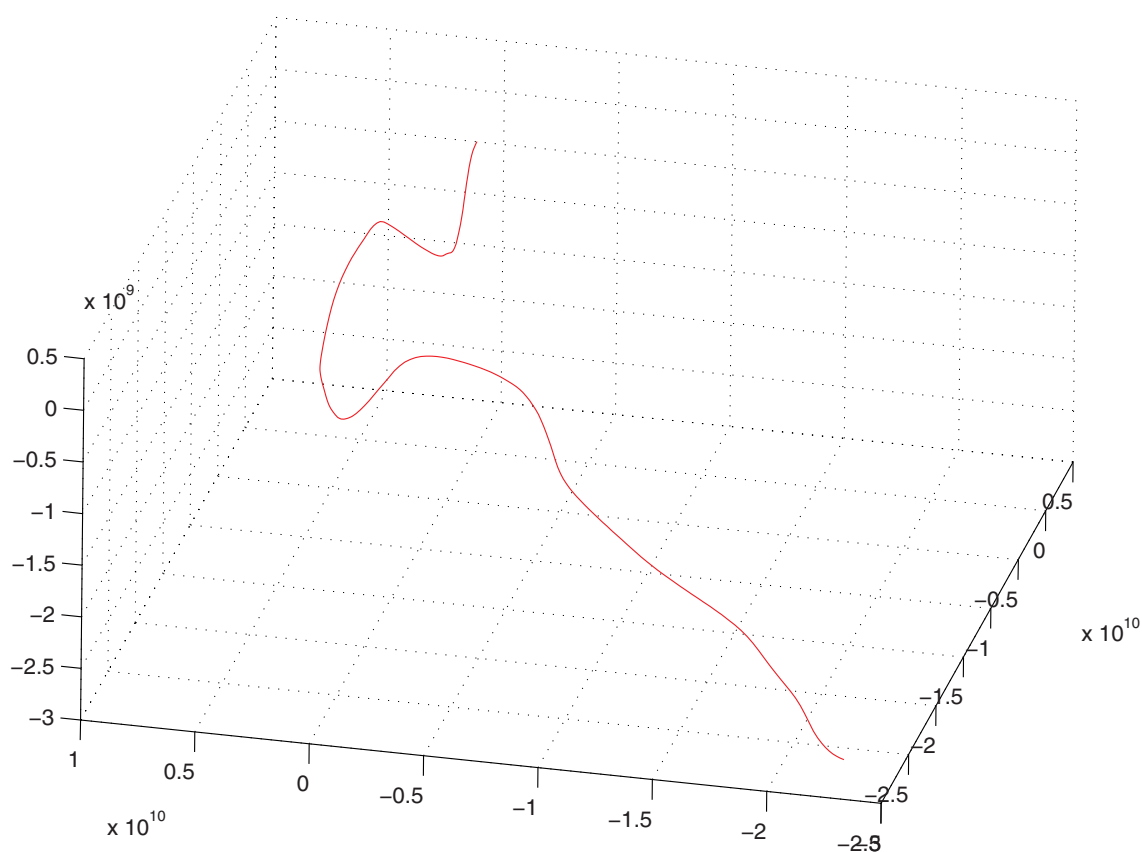


FIGURE 4. Three-dimensional trajectory drawn from linear Gaussian jump model

5. DISCUSSION

We have reviewed existing methods for tracking using piecewise deterministic jump models, and presented new methods for linear Gaussian models having both jumps and a diffusion element. Interestingly it is possible to marginalise the entire path of the process in this setting, conditioned only on the arrival times of manoeuvres, allowing for a highly efficient Rao-Blackwellised particle implementation. Initial results look promising for the new methods but a full evaluation will be carried out in future work on the topic. The new models can potentially be linked in with nonlinear observation processes and data with clutter by partial Rao-Blackwellisation of the unobserved states (typically the velocities and accelerations) and by simulating data associations as part of the particle filtering process (see [27, 26] for examples of such ‘soft-gating’ procedures in other tracking models).

REFERENCES

- [1] Y. Bar-Shalom and T.E. Fortmann. *Tracking and Data Association*. Academic Press Inc., U.S., 1988.
- [2] R. Chen and J. S. Liu. Mixture Kalman filter. *J. Roy. Statist. Soc. Ser. B*, 62(3):493–508, 2000.
- [3] N. Chopin and F. Pelgrin. Bayesian inference and state number determination for Hidden Markov models: an application to the information content of the yield curve about inflation. *Journal of Econometrics*, 123(2), 2004.

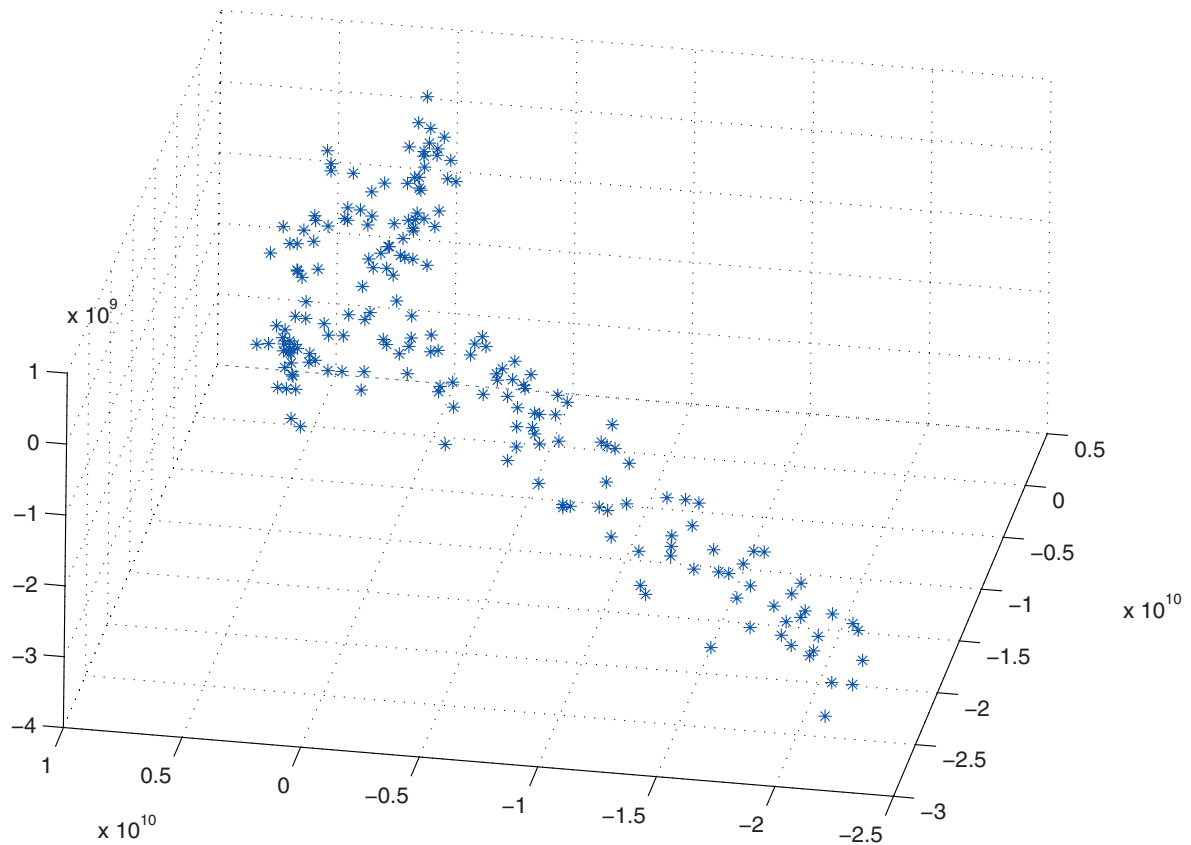


FIGURE 5. Data from three-dimensional trajectory

- [4] D. Cox and V. Isham. *Point Processes*. Chapman & Hall, 1980.
- [5] M. Davis. Piecewise-deterministic Markov processes: a general class of non-diffusion stochastic models (with discussion). *J. Roy. Statist. Soc. Ser. B*, (46), 1984.
- [6] M.H.A. Davis. *Markov Models and Optimization*. Chapman and Hall, 1993.
- [7] P. De Jong and N. Shephard. The simulation smoother for time series models. *Biometrika*, 82(2):339–350, 1995.
- [8] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte-Carlo sampling methods for Bayesian filtering. *Stat. Comput.*, 10:197–208, 2000.
- [9] K. Gilholm, S.J. Godsill, S. Maskell, and D. Salmond. Poisson models for extended target and group tracking. In *Proc. SPIE: Signal and Data Processing of Small Targets*, 2005.
- [10] K. Gilholm and D. Salmond. A spatial distribution model for tracking extended objects. *IEE Proc. on Radar, Sonar and Navigation*, 152(5), 2005.
- [11] S. J. Godsill. MCMC and EM-based methods for inference in heavy-tailed processes with alpha-stable innovations. In *Proc. IEEE Signal processing workshop on higher-order statistics*, June 1999. Caesarea, Israel.
- [12] S. J. Godsill and J. Vermaak. Models and algorithms for tracking using trans-dimensional sequential Monte Carlo. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2004.
- [13] S. J. Godsill and J. Vermaak. Variable rate particle filters for tracking applications. In *Proc. IEEE Stat. Sig. Proc., Bordeaux*, July 2005.
- [14] S.J. Godsill, J.F. Li, and W. Ng. Multiple and extended object tracking with Poisson spatial processes and variable rate filters. In *IEEE CAMSAP*, 2005.

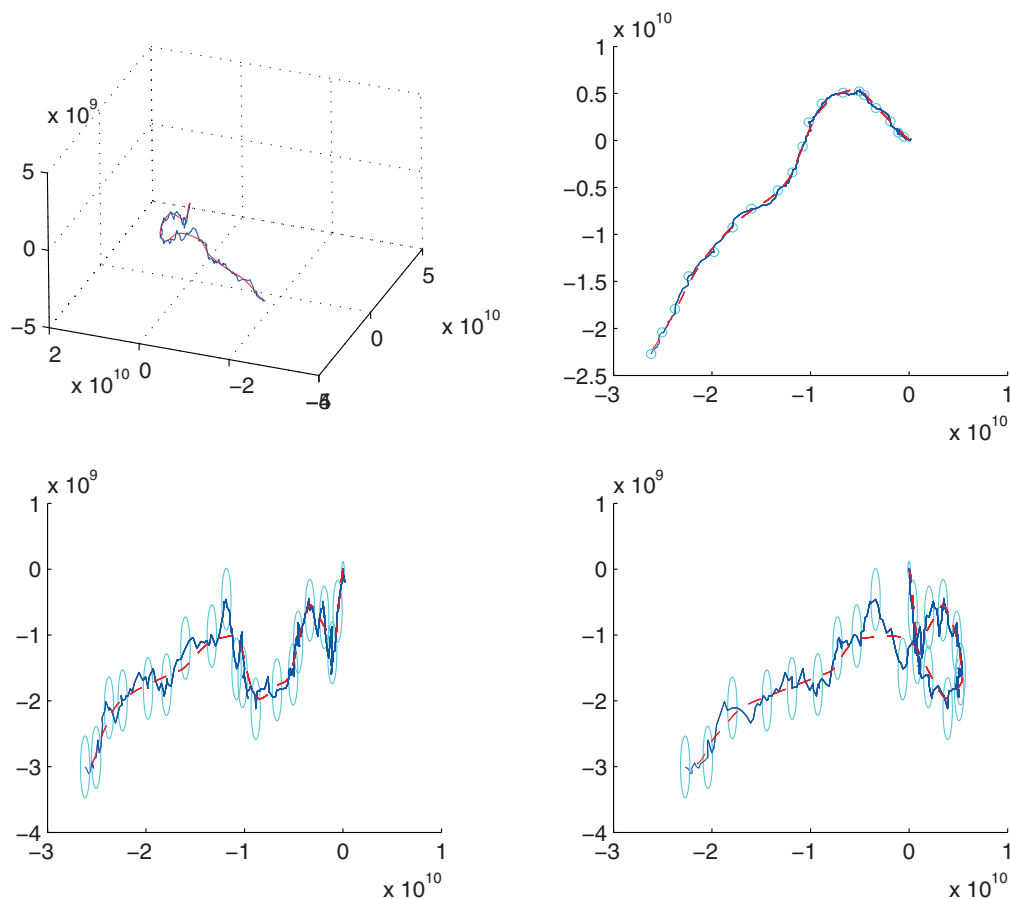


FIGURE 6. Posterior mean filtering estimates (true trajectory shown dotted). Top left: 3-d plot. Top right: y vs. x with posterior confidence ellipses; bottom left: z vs. x with posterior confidence ellipses; bottom right: z vs. y with posterior confidence ellipses.

- [15] S.J. Godsill, J. Vermaak, K-F. Ng, and J-F. Li. Models and algorithms for tracking of manoeuvring objects using variable rate particle filters. *Proc. IEEE*, April 2007. (To Appear).
- [16] G. Grimmett and D. Stirzaker. *Probability and Random Processes*. Oxford University Press, third edition edition, 2001.
- [17] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P-J. Nordlund. Particle filters for positioning, navigation and tracking. *IEEE Transactions on Signal Processing*, 50(2):425–437, February 2002.
- [18] A.C. Harvey. *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press, 1989.
- [19] M. Jacobsen. *Point Process Theory and Applications. Marked Point and Piecewise Deterministic Processes*. Birkhauser, 2006.
- [20] S.G. Kou. A jump-diffusion model for option pricing. *Management Science*, 48(8), August 2002.
- [21] X.R. Li and V.P. Jilkov. Survey of maneuvering target tracking. part 1: Dynamic models. *IEEE Trans. Aerospace and Electronic Systems*, 39(4), 2003.
- [22] J.S. Liu. *Monte Carlo strategies in scientific computing*. Berlin: Springer, 2001.
- [23] M. Lombardi and S.J. Godsill. On-line Bayesian estimation of AR signals in symmetric alpha-stable noise. *IEEE Trans. on Signal Processing*, 2006.
- [24] S. Maskell. Tracking manoeuvring targets and classification of their manoevrability. *EURASIP Journal of Applied Signal Processing*, 15, 2004.
- [25] R.C. Merton. Option pricing when the underlying stock returns are discontinuous. *Journal of Financial Econ.*, 1976.

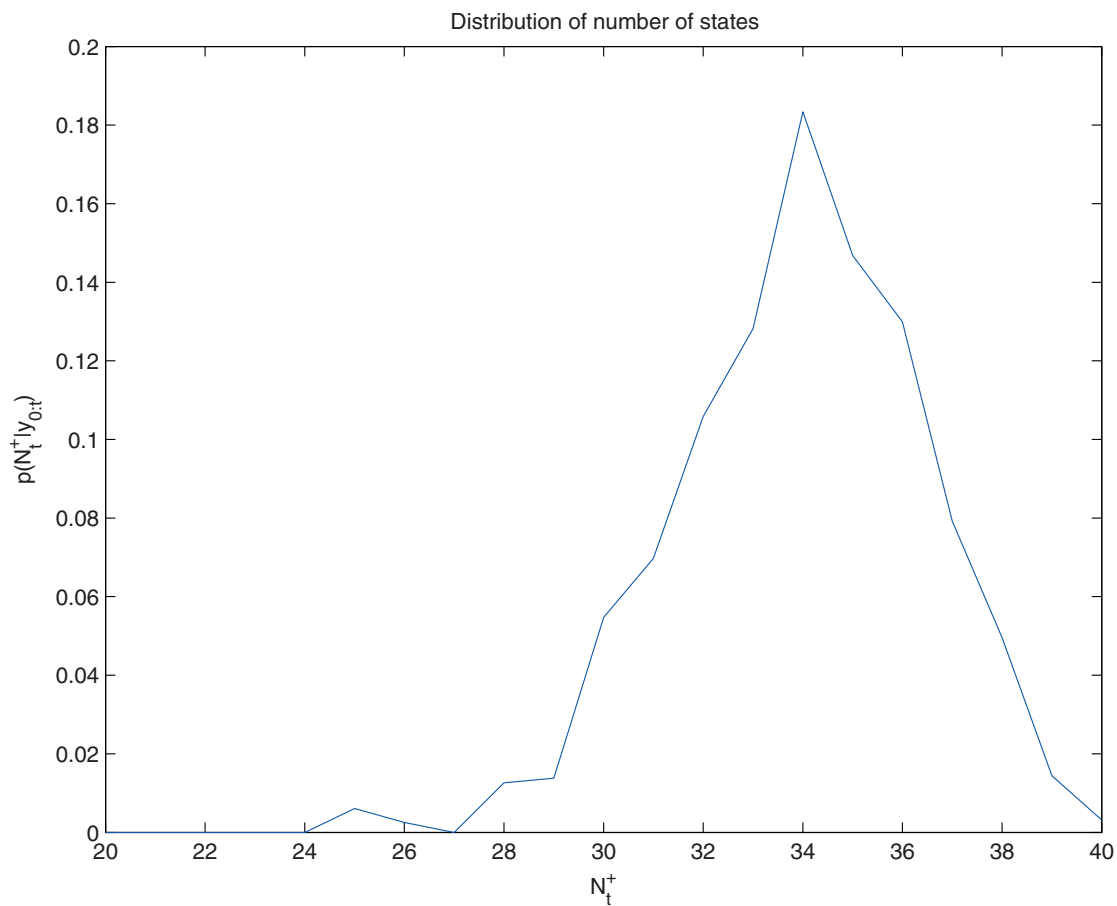


FIGURE 7. Estimate of probability of number of manoeuvres within time interval $t = 0, \dots, 200$ (true value=33).

- [26] J. Vermaak, S. Godsill, and P. Perez. Monte Carlo filtering for multi-target tracking and data association. *IEEE Tr. Aerospace and Electronic Systems*, 41(1):309–332, January 2005.
- [27] J. Vermaak, N. Ikoma, and S.J. Godsill. Sequential Monte Carlo framework for extended object tracking. *IEE Proc.-Radar Sonar Navig.*, 152(5):353–363, October 2005.