

# Chapter 8

## Particle Filter Implementation

Solution of the Bayesian estimation method described in Chap. 3 requires one to recursively integrate the aircraft dynamics pdf (3.6) and multiply it by the likelihood (3.7). Since the measurement model is highly nonlinear and the dynamics model is hybrid discrete-continuous, there is no way to produce a closed form posterior distribution. An alternative is to approximate the distribution numerically. As introduced in Sect. 3.2, the Sample-Importance-Resample (SIR) particle filter draws random samples from the dynamics model and weights them according to the measurement likelihood. This amounts to approximating the posterior distribution as

$$p(\mathbf{x}_k | \mathbf{Z}_k) \approx \sum_{p=1}^P w_k^p \delta(\mathbf{x}_k - \mathbf{x}_k^p), \quad (8.1)$$

where the  $w_k^p$  are referred to as weights (and sum to unity) and the  $\mathbf{x}_k^p$  are referred to as particles. The convergence properties of this approximation in the limit as the number of particles  $P$  increases have been well studied, e.g. [14, 21]. In the SIR version of the particle filter, the particles are randomly generated from the dynamics model and the weights are

$$w_k^p \propto \prod_{k': t_{k'} \leq t_k} p(\mathbf{z}_{k'} | \mathbf{x}_{k'}^p). \quad (8.2)$$

A problem with sampling from the dynamics is that this can be a very diffuse distribution. In the MH370 case, the model allows for turns and speed and altitude changes, and potentially several of each can be sampled between measurements. The proportion of particles that sample a trajectory close to the measurements will be small and a very large number of samples will be required to capture the high probability regions. This is a well known issue for filtering in high dimensional state spaces.

Resampling is one strategy that is used to improve the number of particles following trajectories with relatively high likelihood. It does this in a sequential manner, in which at each time step unlikely particles are replaced with copies of high likely particles through a random sampling process. Initial approaches used these conventional techniques, but it was found to be preferable to be able to process very high particle counts and to adaptively increase the number of particles used until it was possible to identify an adequate number of likely paths, rather than processing a pre-specified number of particles for each time step sequentially. To achieve this particles were propagated and weighted individually; this also reduced the size of the data structures required and allowed preliminary results to be extracted as the filter was executing. The approach adopted was a form of branching mechanism which repeatedly constructs full trajectories.

The method resampled each particle separately, branching a new set of particles from each parent instead of resampling a fixed number of particles across all of the particles at a given time. The branching naturally leads to an exponential growth in the number of particles with time and this was mitigated by pruning extremely unlikely paths when their likelihood became too low. This approach is not necessarily computationally efficient, but in this particular application it was more important to broadly explore the enormous state space than to minimise computational effort.

To motivate the approach, suppose that the distribution prior to resampling is approximated by (8.1), such that integrals can be approximated as

$$\int f(\mathbf{x}_k) p(\mathbf{x}_k) d\mathbf{x}_k \approx \sum_{p=1}^P w_k^p f(\mathbf{x}_k^p), \quad (8.3)$$

where it is assumed that  $\sum_{p=1}^P w_k^p = 1$ . For each particle  $p$ , draw  $n_k^p \geq 0$  copies of that particle, where  $n_k^p$  is a random variable, and for each new particle  $\tilde{\mathbf{x}}_k^{\tilde{p}} = \mathbf{x}_k^p$ . To each we apply the weight

$$\tilde{w}_k^{\tilde{p}} = \frac{w_k^p}{\mathbb{E}[n_k^p]}. \quad (8.4)$$

Assuming that  $\tilde{p}$  indexes the full set of  $\tilde{P} = \sum_{p=1}^P n_k^p$  new particles, it can easily be shown that:

$$\mathbb{E} \left[ \sum_{\tilde{p}=1}^{\tilde{P}} \tilde{w}_k^{\tilde{p}} f(\tilde{\mathbf{x}}_k^{\tilde{p}}) \right] = \sum_{p=1}^P \mathbb{E}[n_k^p] \tilde{w}_k^p f(\mathbf{x}_k^p) = \sum_{p=1}^P w_k^p f(\mathbf{x}_k^p) \quad (8.5)$$

Thus resampling can also be implemented through a randomised branching procedure, recursively adapting the number of particles. This permits a form of depth-first search, which adaptively performs more branching when likely paths result, and tends to prune paths which have low probability.

For our experiments, we chose a procedure which branches quite aggressively when likely paths are discovered, and prunes extremely unlikely paths. Likely paths are duplicated to form  $\bar{n}$  branches. This is implemented by setting

$$p(n_k^p) = \begin{cases} \delta(n_k^p - \bar{n}), & w_k^p \geq \eta \\ w_k^p \delta(n_k^p - 1) + (1 - w_k^p) \delta(n_k^p), & \text{otherwise} \end{cases} \quad (8.6)$$

for  $\eta \ll 1$ . Thus, for particles  $\tilde{p}$  sampled from parent particle  $p$  with  $w_k^p \geq \eta$ ,  $\tilde{w}_k^{\tilde{p}} = w_k^p / \bar{n}$ , and wide branching will occur, while for particles  $\tilde{p}$  sampled from parent particle  $p$  with  $w_k^p < \eta$ ,  $\tilde{w}_k^{\tilde{p}} = 1$ , but most commonly the sub-tree will be pruned.

## 8.1 BFO Bias

The BFO measurement has a bias term that was not able to be adequately calibrated, as discussed in Sect. 5.3. The model treats this bias as an random variable with a given prior density. It is possible to sample the bias along with the aircraft states but a more efficient implementation is to use a Rao-Blackwellised particle filter [15, 29, 38]. Conditioned on the other states we can write a simplified BFO measurement model

$$z_k^{\text{BFO}} = \hat{z}_k^{\text{BFO}} + \delta f^{\text{bias}} + w_k^{\text{BFO}}, \quad (8.7)$$

where  $\hat{z}_k^{\text{BFO}}$  is constant because all of the other states, such as aircraft location and velocity, are known because of the conditioning. This conditional measurement equation is clearly linear in the bias and the noise is modelled as Gaussian, so the posterior distribution of the bias can be determined using a Kalman filter update.

## 8.2 Algorithm

In practical terms, the algorithm proceeds by repeating the following process for each particle:

1. Randomly sample an average time between manoeuvres  $\tau$ .
2. Randomly sample a starting state  $\mathbf{x}_0$  (position, Mach, control angle and altitude) from the prior at  $t_0 = 18:01:49$ , which is described in Chap. 4.
3. Initialise the BFO bias Kalman filter.
4. Perform the following recursion, starting with the sample at  $\mathbf{x}_0$  and measurement time index  $k = 1$ :
  - a. Draw a sample of the trajectory from  $\mathbf{x}_{k-1}$  to  $\mathbf{x}_k$  using the hyperparameter  $\tau$  for selection of turns, speed changes and altitude changes.

- b. Calculate the measurement likelihood  $p(\mathbf{z}_k|\mathbf{x}_k)$  and use it to update the trajectory weight  $w_k^p = p(\mathbf{z}_k|\mathbf{x}_k) w_{k-1}^p$ .
- c. Use the sampled trajectory to update the BFO bias Kalman filter.
- d. If we have reached the final measurement  $k = K$ , store the trajectory and weight.
- e. Otherwise, if the accumulated weight is too low, i.e.,  $w_k^p < \eta$  then branch a single time with probability  $w_k^p$  and weight  $\tilde{w}_k^p = 1$ , terminating the recursion branch with probability  $(1 - w_k^p)$ ; otherwise, branch  $\bar{n}$  times to process remaining time steps with weight  $\tilde{w}_k^p = w_k^p / \bar{n}$ .

The particle weights constructed by the method are not normalised. A normalisation step is performed when the final set of weights at the last time point is used to construct the required pdf. The process in step 4a, namely sampling a trajectory, is critical and is realised through a finite time difference implementation given by the following steps:

1. Randomly sample times to make the next turn, speed change and altitude change
2. While the current sample time is before the next measurement time,  $t_k$

**Table 8.1** State vector elements

State vector	
Latitude	Degrees
Longitude	Degrees
Mach number set point	The OU parameter, denoted $\bar{m}_k$ in Sect. 6.6
Instantaneous Mach number	Mach, denoted $m_k$ in Sect. 6.6
Control angle set point	The OU parameter, denoted $\bar{\theta}_k$ in Sect. 6.6
Instantaneous control angle	Control angle, denoted $\theta_k$ in Sect. 6.6
Altitude	In discrete 1,000 feet steps
Instantaneous wind error, north	The difference between the wind and the tabulated value, denoted $v_{k,\text{wind}}^{\text{north}}$ in Sect. 6.6
Instantaneous wind error, east	The difference between the wind and the tabulated value, denoted $v_{k,\text{wind}}^{\text{east}}$ in Sect. 6.6
BFO bias mean	Mean of the Kalman filter used to estimate the BFO bias $\delta f^{\text{bias}}$
BFO bias variance	Variance of the Kalman filter used to estimate the BFO bias $\delta f^{\text{bias}}$
Mean time to manoeuvre	$\tau$
Autopilot control angle mode	Used to choose between constant true/magnetic heading/track and lateral navigation
Cost index value, when used	An integer between 0 and 100

- a. If the current sample time is the time of a manoeuvre (turn, speed change or altitude change), then execute the manoeuvre and sample a new time to make the next manoeuvre
- b. Otherwise predict ahead 10s (or to the next manoeuvre or measurement, whichever occurs first)

A manoeuvre is executed by making a sequence of 1 s steps. For each step the angle, speed or altitude is incremented and the aircraft position is predicted ahead. The increments continue until the new desired angle, speed or altitude is achieved. The procedure for state prediction under cruise dynamics is summarised in Sect. 6.6. The state vector used for the model is given in Table 8.1. There are a large number of parameters involved with this model and the full description of these is provided in Table 8.2.

**Table 8.2** Summary of filter parameters

Initialisation		
Latitude	Gaussian	s.d. 0.4 min
Longitude	Gaussian	s.d. 0.4 min
Control Mach	Gaussian	s.d. 0.03
Control angle	Gaussian	s.d. 1°
BFO bias	Gaussian	s.d. 25 Hz
Mach deviation	Gaussian	s.d. 0.00311
Angle deviation	Gaussian	s.d. 0.0826°
North wind deviation	Gaussian	s.d. 5.68 kn
East wind deviation	Gaussian	s.d. 5.68 kn
Cruise		
Mach	Reversion rate $\beta_{\text{mach}}$	$1.06 \times 10^{-2}$
	Noise strength $q_{\text{mach}}$	$2.05 \times 10^{-7} \text{ s}^{-1}$
Control angle	Reversion rate $\beta_{\text{angle}}$	$9.8 \times 10^{-3}$
	Noise strength $q_{\text{angle}}$	$4.07 \times 10^{-8} \text{ rad}^2 \text{ s}^{-1}$
North/East wind	Reversion rate $\beta_{\text{wind}}$	$1.09 \times 10^{-3}$
	Noise strength $q_{\text{wind}}$	$7.02 \times 10^{-2} \text{ kn}^2 \text{ s}^{-1}$
Manoeuvres		
All	Mean manoeuvre time $\tau$	~Jeffreys(0.1, 10)
Mach	Uniform new Mach	0.73–0.84
Control angle	Uniform turn angle	$\pm 180^\circ$
Altitude	Uniform new altitude	25,000 to 43,000 ft
Implementation		
Branching rate	Constant $\bar{n}$ within a flight	3–10
Likelihood threshold	Constant $\eta$ within a flight	$e^{-25}$ or $e^{-30}$

### 8.3 Assumptions

The key assumptions used by the filter are:

1. The radar data provides an accurate estimate of the aircraft trajectory up to 18:01:49. If, for example, the radar track used to build the prior were actually from a different aircraft, the predicted pdf would be invalid. Discarding the radar data leads to a significant broadening of the search zone, and accident investigators believe the radar data to be correctly associated with MH370. Chapter 10.6 considers an alternative analysis which ignores the radar data.
2. The measurement error characteristics are known. The pdf of BTO and BFO measurements, in particular the standard deviation of each, is provided to the algorithm as a known input. Extensive study of the statistics of these measurements has been undertaken and the distributions assumed are well characterised, subject to the caveats discussed in Chap. 5. Incremental changes, such as minor inflation of the assumed BTO variance would lead to incremental changes in the filter output.
3. The aircraft cruises in one of five prescribed modes and does not change between them (other than a single possible change from lateral navigation to constant magnetic/true heading). It is possible that the whole flight was continually under manual control but it is highly unlikely. The use of typical autopilot modes is reasonable.
4. Infinite fuel: the fuel constraints on the aircraft can be applied to the pdf afterwards. In the simplest case, maximum reachable ranges could be used to censor impossible trajectories. However, analysis of candidate trajectories has indicated that the majority are feasible. Broad information about the fuel consumption rate of the aircraft has been used to inform the range of allowable Mach numbers.
5. The fluctuations in speed, angle and the error in wind velocity are well-modelled by the OU process. The parameters of the OU model were selected to model these quantities based on recorded data from real flights.
6. The random turn and speed change model is rich enough to describe the real aircraft dynamics and the implicit preferred path for the model does not bias prediction. Validation results in the next chapter show that the model successfully produces pdfs containing the true aircraft location for the available instrumented flights that include air speed changes, altitude changes and angle changes.
7. The aircraft air speed is limited to the range Mach 0.73–0.84. Fuel consumption becomes very inefficient at speeds higher than this and at lower speeds the aircraft is not able to match the measurements. In practice it is likely that the viable range of speeds is actually much narrower than this.
8. In Chap. 10, the pdf of the location of the aircraft at 00:19 is combined with a distribution of aircraft translation during descent, to give a final search zone.

This distribution was developed by ATSB [5] and largely determines the width of the search area along the 00:19 arc. It is assumed that this distribution adequately models the true descent scenario.

**Open Access** This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, duplication, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, a link is provided to the Creative Commons license and any changes made are indicated.

The images or other third party material in this chapter are included in the work's Creative Commons license, unless indicated otherwise in the credit line; if such material is not included in the work's Creative Commons license and the respective action is not permitted by statutory regulation, users will need to obtain permission from the license holder to duplicate, adapt or reproduce the material.