Particle filter for aircraft mass estimation and uncertainty modeling

Junzi Sun[a,*], Henk A.P. Blom[a,b], Joost Ellerbroek[a], Jacco M. Hoekstra[a]

[a] *Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629 HS, Delft, the Netherlands*
[b] *National Aerospace Laboratory, Anthony Fokkerweg 2, 1059 CM Amsterdam, the Netherlands*

Abstract

This article investigates the estimation of aircraft mass and thrust settings of departing aircraft using a recursive Bayesian method called particle filtering. The method is based on a nonlinear state-space system derived from aircraft point-mass performance models. Using only aircraft surveillance data, flight states such as position, velocity, wind speed, and air temperature are collected and used for the estimations. With the regularized Sample Importance Re-sampling particle filter, we are able to estimate the aircraft mass within 30 seconds once an aircraft is airborne. Using this short flight segment allows the assumption of constant mass and thrust setting. The segment at the start of the climb also represents the time when maximum thrust setting is most likely to occur. This study emphasizes an important aspect of the estimation problem, the observation noise modeling. Four observation noise models are proposed, which are all based on the native navigation accuracy parameters that have been obtained automatically from the surveillance data. Simulations and experiments are conducted to test the theoretical model. The results show that the particle filter is able to quantify uncertainties, as well as determine the noise limit for an accurate estimation. The method of this study is tested with a data-set consisting of 50 Cessna Citation II flights where true masses were recorded.

Keywords: aircraft, state estimation, point-mass model, observation noise, particle filter, Bayesian estimation

This is a post-print version of the accepted manuscript, self-archived on June 6, 2019.
Copyright ©2019. Licensed under the Creative Commons CC-BY-NC-ND 4.0 license
http://creativecommons.org/licenses/by-nc-nd/4.0/

The article is published by Elsevier in Journal *Transportation Research Part-C*. The publisher's version can be found at: https://doi.org/10.1016/j.trc.2019.05.030

*Corresponding author
Email address: j.sun-1@tudleft.nl (Junzi Sun)

Nomenclature

$\boldsymbol{\nu}$	process noise
$\boldsymbol{f}$	state transition function
$\boldsymbol{h}$	observation function
$\boldsymbol{n}$	observation noise
χ	track angle (°)
δ	Dirac delta function
δ_T	thrust setting
γ	flight path angle (°)
$\mathbf{x}$	state vector
$\mathbf{y}$	observation vector
ψ	heading angle (°)
τ	air temperature (K)
C_{D0}	zero drag coefficient
D	total drag (N)
g	gravitational acceleration (m/s^2)
k	lift-induced drag coefficient
m	aircraft mass (kg)
S	wing area (m/s^2)
T	thrust (N)
v_a	aircraft airspeed (m/s)
v_g	aircraft ground speed (m/s)
v_z	vertical speed (m/s)
w	particle weight
x	downrange position (m)
y	cross-range position (m)
z	altitude (m)

1. Introduction

Estimating aircraft mass using flight trajectory data has long been a topic of interest in ATM research. Aircraft mass not only is an important parameter for aircraft performance-related studies but is also a desired piece of knowledge for air traffic controllers in practice. Airlines, however, treat this data as confidential, and access is rarely granted to either researchers or air traffic controllers. In practice, mass information is, thus, not accessible nor actively used by the research community. Having an accurate estimate of aircraft mass at takeoff (or during the initial climb) can, for example, be beneficial for obtaining accurate trajectory and fuel consumption predictions during the flight.

Earlier studies that addressed this problem commonly involved deterministic methods based on the aircraft total energy model. A least-squares solution to this problem is proposed in Alligier et al. (2013), and an adaptive estimation method is presented in Schultz et al. (2012). Using radar track data, these two approaches are compared in Alligier et al. (2014). All these methods employ active radar data for estimation. With advancements in machine learning studies, supervised machine learning methods that require a well-established training set are also introduced to address the estimation problem in Alligier et al. (2015) and Chati and Balakrishnan (2017, 2018). With the growing adoption of Automatic Dependent Surveillance-Broadcast (ADS-B), estimation studies have been conducted around this open data source, such as Sun et al. (2016).

The approaches by Alligier et al. (2013) and Chati and Balakrishnan (2018) have in common that they make use of regression analysis. Chati and Balakrishnan (2018) focus on estimating aircraft mass during the takeoff roll, whereas Alligier et al. (2013) focuses on the climb phase. The advantage of regression analysis is that it avoids the need for an accurate model of the various noise levels. However, regression analysis requires the exploitation of model linearity. Recently Sun et al. (2018b) has shown that ADS-B data availability makes it worthwhile to use Bayesian inference methods for the estimation of aircraft mass. The goal of this article is to extend this Bayesian inference approach such that it can incorporate non-linear aircraft evolution equations for the nonlinear Bayesian filtering of aircraft mass, based on ADS-B and Enhanced Mode-S surveillance data.

From most of the aforementioned studies, a strong link between aircraft mass and thrust setting is evident, and it is not possible to estimate one of the two parameters without some knowledge of the other. Several of these studies essentially addressed the estimation process as a minimization problem. Solutions were obtained using a form of least-squares fitting. Aircraft mass estimated under these conditions could be unrealistic and even outside plausible physical boundaries. Often, the main cause for this relates to uncertainty in the trajectory data, as well as uncertainty and non-linearity in the system model. Although the effect of noise is an important aspect in all inference processes, previous studies have not yet comprehensively studied its relation to mass estimation. Therefore, alongside estimating aircraft mass, observation noise modeling for the nonlinear aircraft equations is one of the main focuses of this study.

The problem of mass estimation by a (ground-based) observer can be considered as having to solve an inverse non-linear multi-state system using noisy observation data. To approach this complex system, in this study, we construct a detailed point-mass flight performance model with 11 system states and 8 observable states. A regularized Sample Importance Re-sampling (SIR) particle filter is introduced to estimate the aircraft states from noisy observations. In addition, four models of different levels of observation noise are constructed, which are based on ADS-B navigation accuracy standards. These models are used in particle filtering.

This study is an extension of our previous preliminary research of Sun et al. (2018a), with an improved system model and new insights. Unlike Alligier et al. (2015) where the machine learning model uses mass estimations from regression methods as training input, or Chati and Balakrishnan (2018) where the machine learning model uses flight recorder data as training input, our method is a purely model-based state estimation approach. The remainder of this study is structured as follows. Section two describes the nonlinear point-mass flight dynamic system equations. Section three describes the theory of recursive Bayesian estimation and particle filtering. Section four develops models for the process and observation noise. Section five specifies the particle filter for aircraft mass estimation during initial take-off. Section six presents the experiments conducted with simulated and real flight trajectories. Section seven conducts sensitivity analyses of the proposed approach. Conclusions are presented in section eight.

2. System of point-mass aircraft performance model

2.1. Aircraft state

This section describes the system equations for aircraft locomotion, implemented as a point-mass model. To illustrate, Figure 1 shows the observable states of an aircraft, primarily based on aircraft ADS-B broadcasts. The wind can be obtained from a weather prediction model or estimated based on Enhanced Mode-S data, which is described in Sun et al. (2018d). The left figure shows a horizontal projection of a trajectory, and the right figure a vertical projection. The heading and track are relative to the true north. For ease of computation, latitudes, longitudes, and altitudes are converted to a Cartesian coordinate system, with its origin at the first observed position. The point-mass model used in this study ignores the angle of attack and assumes no vertical wind.

Denoting aircraft mass as m, thrust setting coefficient as δ_T, distance flown as $\vec{s}$, altitude as z, ground speed as $\vec{v}_g$, true airspeed as $\vec{v}_a$ (in horizontal projection), vertical rate as v_z, wind speed as $\vec{v}_w$, and air temperature as τ, the system state $\mathbf{x}$ vector can be written as:

$$\mathbf{x} = (m, \delta_T, \vec{s}, z, \vec{v}_a, v_z, \vec{v}_w, \tau) \tag{1}$$

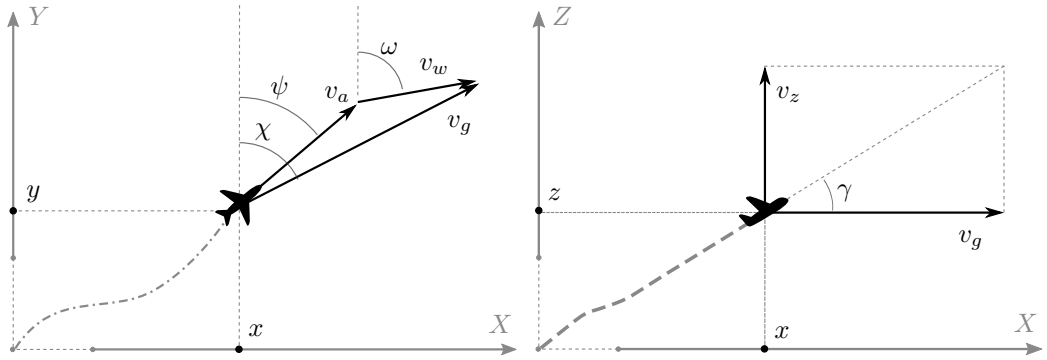

Figure 1: Observed states in the flight dynamic system

The vector variables corresponding to the x and y axes are:

$$\begin{aligned} \vec{s} &= (x, y) \\ \vec{v}_a &= (v_{\mathrm{ax}}, v_{\mathrm{ay}}) \\ \vec{v}_w &= (v_{\mathrm{wx}}, v_{\mathrm{wy}}) \end{aligned} \tag{2}$$

Additional angular parameters in Figure 1 are flight path angle γ, ground track χ, aircraft heading ψ, and wind direction ϕ. In addition, the measurement vector $\mathbf{y}$ is represented as:

$$\mathbf{y} = (\tilde{\vec{s}}, \tilde{z}, \tilde{\vec{v}}_g, \tilde{v}_z, \tilde{\vec{v}}_w, \tilde{\tau}) \tag{3}$$

where $\tilde{\vec{s}}$, $\tilde{z}$, $\tilde{\vec{v}}_g$, and $\tilde{v}_z$ are computed from ADS-B data. Wind $\tilde{\vec{v}}_w$ and temperature $\tilde{\tau}$ are obtained using Enhanced Mode-S data (Sun et al., 2018d).

2.2. State evolution

We want to model the system accurately in this point-mass performance model. The process noise is considered to be zero for states where exact process equations are known. These states are m_t, $\delta_{T,t}$, $\vec{s}_t$, z_t, and $\vec{v}_{a,t}$. For states in which a perfect process equation cannot be established accurately or are unknown, we use an auto-regressive model to describe their state evolution. These states are the vertical rate v_z, wind $\vec{v}_w$, and temperature τ. The complete state process equations can be described as follows:

$$m_t = m_{t-1} \tag{4}$$
$$\delta_{T,t} = \delta_{T,t-1} \tag{5}$$
$$\vec{s}_t = \vec{s}_{t-1} + (\vec{v}_{a,t-1} + \vec{v}_{w,t-1})\ \mathrm{dt} \tag{6}$$
$$z_t = z_{t-1} + v_{z,t-1}\ \mathrm{dt} \tag{7}$$
$$\vec{v}_{a,t} = \vec{v}_{a,t-1} + \vec{a}_{t-1}\ \mathrm{dt} \tag{8}$$
$$v_{z,t} = \alpha_{\mathrm{vz}} v_{z,t-1} + \varepsilon_{\mathrm{vz}} \tag{9}$$
$$\vec{v}_{w,t} = \vec{\alpha}_w \vec{v}_{w,t-1} + \vec{\varepsilon}_w \tag{10}$$
$$\tau_t = \alpha_\tau \tau_{t-1} + \varepsilon_\tau \tag{11}$$

where $\vec{a}$ is the horizontal acceleration. State updates for v_z, $\vec{v}_w$, and τ are expressed using autoregressive (AR) models with lag $p = 1$. The coefficients of the AR models are constructed using historical data, which is described in Section 4.2. It is worth noting that since we are working with

a short segment of the flight (within 30 seconds), constant mass (m) and thrust setting (δ_T) are assumed. Obviously, such an assumption cannot be applied to a longer period of the flight.

To compute the acceleration ($\vec{a}$), we need to consider the forces acting on the aircraft, which are thrust, drag, lift, and gravity. The equations used at each time step to compute the acceleration are listed as follows:

$$\vec{a}_t = (a_t, \psi_t) \tag{12}$$

$$\psi_t = \arctan_2(v_{ax,t},\ v_{ay,t}) \tag{13}$$

$$a_t = \frac{1}{m}(\delta_{T,t} T_t - D_t) - g\frac{v_{z,t}}{v_{a,t}} \tag{14}$$

$$T_t = f_{\text{thr}}(z_t, v_{a,t}, v_{z,t}) \tag{15}$$

$$D_t = f_{\text{drag}}(m_t, v_t, z_t, C_{D0}, k, S) \tag{16}$$

where T and D are maximum thrust and drag forces expressed as functions of other state parameters. S is the aircraft wing surface. The thrust is calculated based on the empirical model proposed by Bartel and Young (2008). The drag is calculated based on the drag polar of the aircraft modeled in Sun et al. (2018c).

In general, only the maximum thrust provided by the engines can be calculated from aircraft speed and altitude. During operation, aircraft engines only stay close to the maximum thrust for a short period of time at takeoff and at the start of the climb. Sometimes, a reduced thrust is applied, where the reduction of thrust is constrained by the actual aircraft mass.

In the particle filter, such a constraint is reflected at the initialization of the particles. The relation of mass and thrust setting is expressed using the following equation adapted from the Base of Aircraft Data (BADA) reduced climbing power model from Nuic (2014): [1]

$$\delta_{T\min} = 1 - \Delta\delta_T \frac{m_{\max} - m}{m_{\max} - m_{\min}} \tag{17}$$

$$0 \leq \Delta\delta_T \leq 0.2 \tag{18}$$

Here, $\Delta\delta_T$ is the minimum derated thrust ratio, which is aircraft type specific and commonly small. $m_{\max}$ and $m_{\min}$ are considered as the maximum take-off mass and minimum operational mass respectively.

Equation 17 shows the relationship between mass and the possibly reduced thrust ratio. It indicates that the higher the aircraft mass, the smaller the range in which the thrust can be reduced. Aircraft mass is assumed to be uniformly distributed between the operational empty mass and maximum takeoff mass. The thrust setting is assumed to be uniformly distributed between this minimum possible derated thrust ratio and one:

$$m \sim \mathcal{U}\Big[m_{\text{oew}}, m_{\text{mtow}}\Big] \tag{19}$$

$$\delta_T \sim \mathcal{U}\Big[\delta_{T\min}, 1\Big] \tag{20}$$

[1] Note that a similar equation in BADA is used to determine power setting. In this study, the form of the equation is adopted for the thrust reduction ratio.

3. Recursive Bayesian estimation

In the recursive Bayesian estimation, an estimate of the system state is obtained using all previous observations. To this end, a recursive system with additive noise can be generalized by a discrete-time state model and observation model:

$$\begin{aligned} \mathbf{x}_t &= \boldsymbol{f}(\mathbf{x}_{t-1}) + \boldsymbol{\nu}_{t-1} \\ \mathbf{y}_t &= \boldsymbol{h}(\mathbf{x}_t) + \boldsymbol{n}_t \end{aligned} \tag{21}$$

where $\mathbf{x}_t$ represents the set of system states and $\mathbf{y}_t$ the set of observations at time t ($t \in \mathbb{N}$), $\boldsymbol{f}$ and $\boldsymbol{h}$ represent the state transition function and observation function, $\boldsymbol{\nu}_t$ is the process noise, and $\boldsymbol{n}_t$ is the observation noise. $\boldsymbol{\nu}_t$ and $\boldsymbol{n}_t$ are assumed to be mutually independent sequences of independently and identically distributed variables. In the particular case of this study, we assume an additive Gaussian model for both process noise and observation noise as a common practice.

The goal of filtering is to compute the probability of the system state at a given time t, based on the observations from time 1 to time t, denoted as $p(\mathbf{x}_t|\mathbf{y}_{1:t})$:

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})} \tag{22}$$

where the first part of the numerator, $p(\mathbf{y}_t|\mathbf{x}_t)$, is the observation probability that follows the observation function and observation noise $\boldsymbol{n}$ from Equation 21:

$$p(\mathbf{y}_t|\mathbf{x}_t) = \mathcal{N}\Big\{\mathbf{y}_t;\ \boldsymbol{h}(\mathbf{x}_t), \boldsymbol{\Sigma}_{\boldsymbol{n}}\Big\} \tag{23}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{n}}$ is the covariance of the observation noise. Because the system described in Equation 21 is a first-order Markov process, the second part $p(\mathbf{x}_t|\mathbf{y}_{1:t-1})$ becomes:

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})\ d\mathbf{x}_{t-1} \tag{24}$$

where the first term $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ on the right-hand side of the equation is the state transition probability. It follows the state transition function and the process noise model $\boldsymbol{\nu}$ in Equation 21:

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}\Big\{\mathbf{x}_t;\ \boldsymbol{f}(\mathbf{x}_{t-1}), \boldsymbol{\Sigma}_{\boldsymbol{v}}\Big\} \tag{25}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{v}}$ is the covariance of the process noise. Combining Equation 22 and Equation 24, the recursive form becomes:

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \int \frac{p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})} p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})\ d\mathbf{x}_{t-1} \tag{26}$$

where the denominator of the fraction is a normalizing factor which does not need to be computed explicitly. The challenge is to resolve Equation 26 in a recursive way using input data $\mathbf{y}_{1:t}$. This is where Sequential Monte Carlo (SMC) simulation-based particle filtering can be used.

3.1. Particle filtering

The specific Sequential Monte Carlo simulation method considered in this study is known as particle filtering (Doucet et al., 2001). A particle filter is a recursive Bayesian estimator based on importance sampling that computes the posterior density in a Monte Carlo fashion. It approximates the target distribution (denoted as $p(x)$) using a large number of samples (particles), drawn from a proposal distribution (denoted as $q(x)$) that updates it recursively.

To describe the SMC process, at time t, let $\{\mathbf{x}_t^i, \mathrm{w}_t^i\}_{i=i}^N$ be a set of particles that can represent the posterior density $p(\mathbf{x}_t|\mathbf{y}_{1:t})$, where $\mathbf{x}_t^i$ is the ith possible state with weight w_t^i. Henceforth, the posterior density is approximated with the empirical probability density:

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{i=1}^{N} \mathrm{w}_t^i \delta(\mathbf{x}_t^i) \tag{27}$$

where $\delta(\cdot)$ is a Dirac delta function centered at $\mathbf{x}_t^i$ and w^i is the normalized weight of a particle which satisfies $\mathrm{w}^i = p(x^i)/q(x^i)$. The most important part of particle filtering is the weight updating. Sequentially, the particle weight w_t^i is updated in a recursive form. The solution is presented in Equation 28, as derived by Arulampalam et al. (2002) :

$$\begin{aligned} \mathbf{x}_t^i &\sim q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \mathbf{y}_{1:t}) \\ \tilde{\mathrm{w}}_t^i &\propto \mathrm{w}_{t-1}^i \frac{p(\mathbf{y}_t|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \mathbf{y}_{1:t})} \\ \mathrm{w}_t^i &= \frac{\tilde{\mathrm{w}}_t^i}{\sum_{i=1}^N \tilde{\mathrm{w}}_t^i} \end{aligned} \tag{28}$$

At each iteration, the sum of all weights is normalized to one as shown in the last part of Equation 28. The posterior filtered density is approximated using Equation 27. We can compute the expected state values at each time step using the obtained particle weights:

$$\mathbb{E}[\mathbf{x}_t] = \sum_{i=1}^{N} \mathbf{x}_t^i \mathrm{w}_t^i \tag{29}$$

There are different ways to choose the proposal distribution $q(x)$. A specific particle filter - *Sample Importance Re-sampling* (SIR) - is used for solving the problem of this study. The SIR particle filter uses the state transition distribution $p(\mathbf{x}_t|\mathbf{x}_{t-1}^i)$ as the proposal distribution $q(\mathbf{x}_t|\mathbf{x}_{t-1}^i, \mathbf{y}_k)$. Therefore, the particle update equations in Equation 28 can be simplified to:

$$\begin{aligned} \mathbf{x}_t^i &\sim p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) \\ \mathrm{w}_t^i &\propto \mathrm{w}_{t-1}^i p(\mathbf{y}_t|\mathbf{x}_t^i) \end{aligned} \tag{30}$$

For a SIR particle filter, an optional additional re-sampling process at each iteration is included. The re-sampling is useful for systems which involve a large number of states (commonly more than 3) and prevents the impoverishment of particles. The re-sampling step generates a new set of particles based on the approximated $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$. Once particles are re-sampled, all weights are assigned to $1/N$.

The re-sampling step is essentially a redistribution of particles, which replaces low-weight particles with high-weight particles. The standard re-sampling algorithm is called *residual re-sampling*, which was proposed by Liu and Chen (1998). Other forms of re-sampling such as *systematic re-sampling* and *stratified re-sampling* were summarized by Douc and Cappé (2005). In this study, the commonly used residual re-sampling is applied.

The process of flight state estimation using the particle filter is summarized in Figure 2. In this figure, we can see that ADS-B and Enhanced Mode-S data are decoded and used as state observations which support the update of the particle filter. The system dynamic equations define the particle state evolution. In section 5, different elements of the SIR particle filtering algorithm are detailed. Before this can be implemented, we first need to introduce the system equations for describing the aircraft performance model.

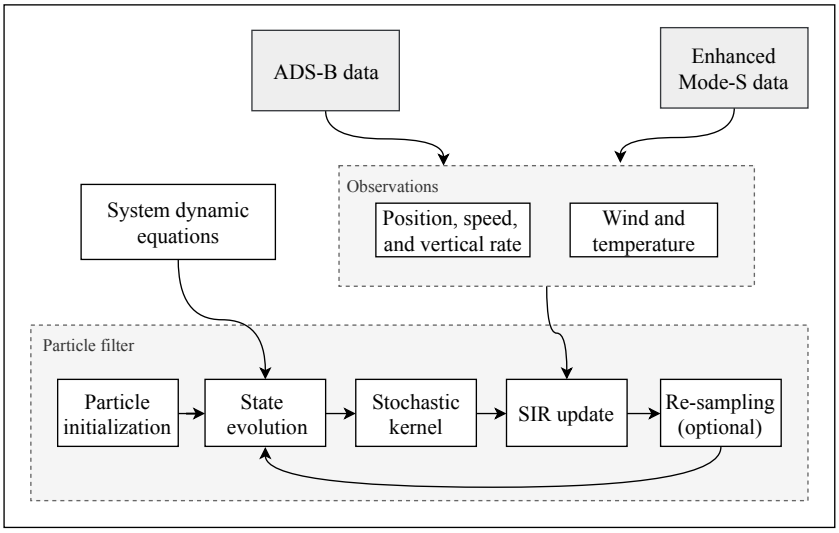

Figure 2: Estimation diagram including the source of observations and process of SIR particle filter. Further details are given in section 5.

3.2. Application to the aircraft mass and thrust estimation

The observable states are flight distance and ground speed in the horizontal and vertical direction, as well as the atmospheric conditions. The measurement vector is denoted as $\mathbf{y}$. Recall the measurement function from Equation 21:

$$\mathbf{y}_t = \boldsymbol{h}(\mathbf{x}_t) + \boldsymbol{n}_t; \quad \boldsymbol{n}_t \sim \mathcal{N}(0, \boldsymbol{\Sigma_n}) \tag{31}$$

where the state and measurement vectors at time t are:

$$\begin{aligned} \mathbf{x}_t &= [m_t, \delta_{T,t}, \vec{s}_t, z_t, \vec{v}_{a,t}, v_{z,t}, \vec{v}_{w,t}, \tau_t] \\ &= [m_t, \delta_{T,t}, (x_t, y_t), z_t, (v_{ax,t}, v_{ay,t}), v_{z,t}, (v_{wx,t}, v_{wy,t}), \tau_t] \\ \mathbf{y}_t &= [\tilde{\vec{s}}_t, \tilde{z}_t, \tilde{\vec{v}}_{g,t}, \tilde{v}_{z,t}, \tilde{\vec{v}}_{w,t}, \tilde{\tau}_t] \\ &= [(\tilde{x}_t, \tilde{y}_t), \tilde{z}_t, (\tilde{v}_{gx,t}, \tilde{v}_{gy,t}), \tilde{v}_{z,t}, (\tilde{v}_{wx,t}, \tilde{v}_{wy,t}), \tilde{\tau}_t] \end{aligned} \tag{32}$$

The observation functions for all states can be written in detail as follows:

$$\tilde{\vec{s}}_t = \vec{s}_t + \vec{n}_{s,t} \tag{33}$$

$$\tilde{z}_t = z_t + n_{z,t} \tag{34}$$

$$\tilde{\vec{v}}_{g,t} = \vec{v}_{a,t} + \vec{v}_{w,t} + \vec{n}_{vg,t} \tag{35}$$

$$\tilde{v}_{z,t} = v_{z,t} + n_{vz,t} \tag{36}$$

$$\tilde{\vec{v}}_{w,t} = \vec{v}_{w,t} + \vec{n}_{vw,t} \tag{37}$$

$$\tilde{\tau}_t = \tau_t + n_{\tau,t} \tag{38}$$

As previous mentioned, the observation noise is assumed to be uncorrelated additive Gaussian noise. The noise $\boldsymbol{n}$ from Equation 21 can then be considered a a multi-variate Gaussian distribution. The diagonal covariance matrix $\boldsymbol{\Sigma}_n$ for these noise models is expressed as:

$$\boldsymbol{\Sigma}_n = \mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_z^2, \sigma_{\mathrm{vg}}^2, \sigma_{\mathrm{vg}}^2, \sigma_{\mathrm{vz}}^2, \sigma_{\mathrm{vw}}^2, \sigma_{\mathrm{vw}}^2, \sigma_\tau^2) \tag{39}$$

where σ^2_{vg} and σ^2_{vw} apply to both x and y directions. Since the noise follows a multivariate normal distribution, the associated weight of the particles, $p(\mathbf{y}_t|\mathbf{x}^i_t)$, can be conveniently computed as follows:

$$\begin{aligned} \mathrm{w}^i_t &\propto \mathrm{w}^i_{t-1}\, p(\mathbf{y}_t|\mathbf{x}^i_t) \\ &\propto \mathrm{w}^i_{t-1}\, \exp\left(-\frac{1}{2}\left[\mathbf{y}_t - \boldsymbol{h}(\mathbf{x}^i_t)\right]^T \boldsymbol{\Sigma}^{-1}\left[\mathbf{y}_t - \boldsymbol{h}(\mathbf{x}^i_t)\right]\right) \end{aligned} \tag{40}$$

In this study, the particles are re-sampled at each time step and normalized to $1/N$. Hence, the weight calculation is simplified to:

$$\mathrm{w}^i_t \propto \exp\left(-\frac{1}{2}\left[\mathbf{y}_t - \boldsymbol{h}(\mathbf{x}^i_t)\right]^T \boldsymbol{\Sigma}^{-1}\left[\mathbf{y}_t - \boldsymbol{h}(\mathbf{x}^i_t)\right]\right) \tag{41}$$

4. Observation and process noise models

4.1. The observation noise model

Observation noise is closely related to sensor errors. For example, GPS errors affect position measurements (related to the state $\tilde{x}$) and altimeter errors affect altitude measurements ($\tilde{z}$). ADS-B transponders operate under regulations that define the minimum accuracy of sensors RTCA (2011). Different categories of uncertainty indicators are transmitted through ADS-B. In this study, these *Navigation Accuracy Categories* (NAC) of ADS-B (version 1 and 2) are considered for the construction of observation noise models.

Table 1 illustrates the *Navigation Accuracy Category - velocity* (NACv), which defines the level of accuracy in terms of horizontal and vertical speed. The NACv indicator is broadcast along with the airborne velocity (message Type Code 18). HFOM and VFOM, short for the horizontal and vertical figure of merit, indicate the 95% confidence interval which corresponds to twice the standard deviation in the observation noise model. The HFOM and VFOM can therefore be used to obtain values for σ^2_{vg}, σ^2_{va}, and σ^2_{vz}.

Table 1: Navigation Accuracy Category / Velocity

NACv	HFOM	VFOM
4	<0.3 m/s	<0.46 m/s
3	<1 m/s	<1.52 m/s
2	<3 m/s	<4.57 m/s
1	<10 m/s	<15.24 m/s
0	>10 m/s or unkown	>15.24 m/s or unkown

Similarly, Table 2 shows the *Navigation Accuracy Category - position* (NACp), which defines the level of accuracy in terms of horizontal and vertical position. For each NACp level, an Estimated Position Error (EPU) and a Vertical Estimated Position Error (VEPU) are defined. Similarly, they indicate a 95% confidence interval for the horizontal and vertical bounds. From these values, we can obtain σ^2_x, σ^2_y, and σ^2_z for the observation noise model.

Four sets of observation noise models are proposed based on the ADS-B specifications, each corresponding to a different combination of NACp and NACv (see Table 3). These four models are also the foundation for the experiments carried out later on in this study.

The uncertainties in the wind and temperature measurements depend on the Enhanced Mode-S data that was used to generate weather information. The variances for wind and temperature are set to be 2.5^2 and 1^2 for the noise model $\boldsymbol{\Sigma}_{n3}$. These two values are based on the variance of the model results obtained in Sun et al. (2018d). For other noise models, they are adjusted

Table 2: Navigation Integrity Category

NACp	EPU	VEPU
11	<3 m	<4 m
10	<10 m	<15 m
9	<30 m	<45 m
8	<0.05 NM	n/a
7 - 0	not used in this study	

Table 3: Noise models

Noise model	NACp	NACv
$\mathbf{\Sigma}_{n1}$	11	4
$\mathbf{\Sigma}_{n2}$	10	3
$\mathbf{\Sigma}_{n3}$	9	2
$\mathbf{\Sigma}_{n4}$	8	1

with respect to the scale of the velocity uncertainty. The exact values of all four noise models for Equation 39 used in the experiments are listed as follows:

$$\begin{aligned}
\mathbf{\Sigma}_{n1} &= \mathrm{diag}(1.5^2, 1.5^2, 2^2, 0.15^2, 0.15^2, 0.23^2, 0.2^2, 0.2^2, 0.1^2) \\
\mathbf{\Sigma}_{n2} &= \mathrm{diag}(5^2, 5^2, 7.5^2, 0.5^2, 0.5^2, 0.76^2, 0.8^2, 0.8^2, 0.3^2) \\
\mathbf{\Sigma}_{n3} &= \mathrm{diag}(15^2, 15^2, 22.5^2, 1.5^2, 1.5^2, 2.28^2, 2.5^2, 2.5^2, 1^2) \\
\mathbf{\Sigma}_{n4} &= \mathrm{diag}(48^2, 48^2, 68^2, 5^2, 5^2, 7.62^2, 7.5^2, 7.5^2, 3^2)
\end{aligned} \tag{42}$$

4.2. Process equations for vertical rate, wind, and temperature

As described in Equations 9 and 10, vertical rate, wind, and temperature along the climb path are modeled as autoregressive (AR) models. This is because the underlying process model is unknown, and these time series exhibit a strong correlation between consecutive data points.

AR is also an approach that models the process error of certain variables that do not have explicit process expressions. The purpose is not to have an accurate process error, but the magnitude of the process error models for these parameters. As long as the process error is large enough to capture and track the changes in the variables, it is sufficient for the estimator. To obtain these error models, we can make use a large quantity of historical data.

For simplification, we will use a first-order AR model (AR1) to describe state evolution. In general, the AR1 model (without the bias term) can be expressed in the following form:

$$\begin{aligned}
x_t &= \alpha x_{t-1} + \varepsilon \\
\varepsilon &\sim \mathcal{N}(0, \sigma^2)
\end{aligned} \tag{43}$$

where α is the model parameter and ε is the white noise with variance of σ^2. Using real flight data, these parameters can be estimated. For a given flight, α and σ can be estimated in Equation 44 using the least-squares regression :

$$\begin{aligned}
\alpha &= \frac{\sum_{t=1}^{n} \tilde{x}_{t-1}\tilde{x}_t}{\sum_{t=1}^{n} \tilde{x}_{t-1}^2} \\
\sigma^2 &= \mathrm{Var}(x_t - \alpha\tilde{x}_{t-1})
\end{aligned} \tag{44}$$

where $\tilde{x}$ represent the wind, temperature, or vertical rate data extracted from real flights. This real flight data source should be different from the data source $\mathbf{y}_{1:t}$ that is used for mass estimation

by the particle filter. It is worth clarifying that the AR approach is only used to find out the parameters of process equations using a different set of flights than the ones used for particle filter mass estimation.

The reason that wind and temperature can be modeled in this way is that locally they tend to be homogeneously distributed with some degree of variability. There is often a gradual increase in wind magnitude and a lapse of temperature with increasing altitude. On the other hand, the vertical rate is often a controlled variable in point mass flight models. Without more information from the aircraft, an AR model can capture the change in vertical rate from the observer's point of view.

To determine representative values for α and σ^2, ADS-B and Enhanced Mode-S data was collected for climbing flights in a period of one month (around 10,000 flights) using our receiver. α and σ^2 were computed for each state parameter (see Figure 3).

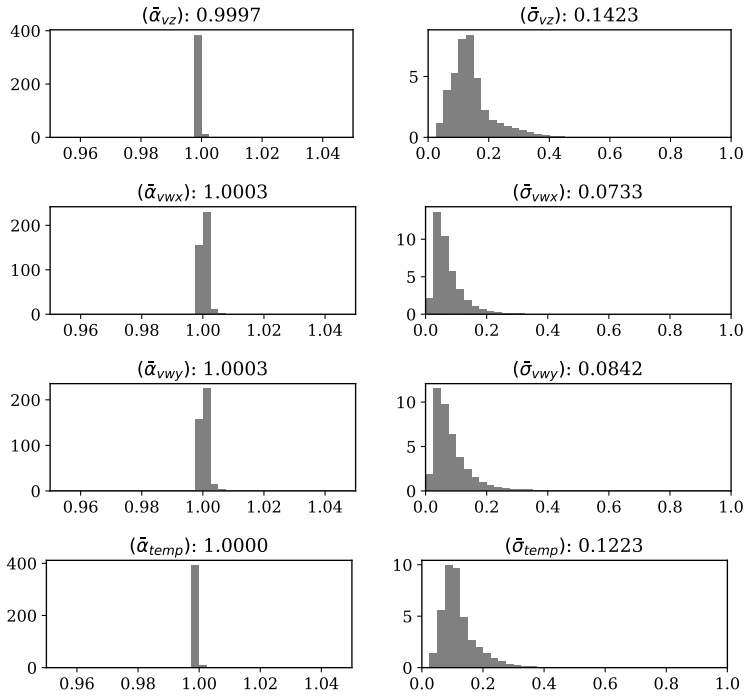

Figure 3: First order AR model parameters for v_z, v_{wx}, and v_{wy}

The mean value of each parameter is then used for the AR models. Table 4 summarizes the obtained parameters for v_z, $\vec{v}_w$, and τ (used in Equation 9 to 11).

Table 4: Parameter summary for the first order AR models

	α	σ
v_z	0.9997	0.1423
v_{wx}	1.0003	0.0733
v_{wy}	1.0003	0.0842
τ	1.0000	0.1223

5. The particle filter algorithm

The detailed algorithm is specified in Algorithm 1, which is adapted from Blom and Bloem (2007). The most essential part of the algorithm is the *SIR update*, where the weights of the

particles are updated based on the observations. This step reflects Equation 40 that is described in the previous section. In addition, the *state evolution* is defined by the aircraft performance that is described by Equations 4 to 11.

Algorithm 1 The SIR particle filter: $p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \rightarrow p(\mathbf{x}_t|\mathbf{y}_{1:t})$

for i=1 to N **do** ▷ Initialize particles
  draw $m_0^i \sim \mathcal{U}(m_{\text{oew}}, m_{\text{mtow}})$
  draw ${\delta_T}_0^i \sim \mathcal{U}\Big(1 - \delta_{T\min}(m_{\text{mtow}} - m_0^i)/(m_{\text{mtow}} - m_{\text{oew}}),\ 1\Big)$
  draw $(\vec{s}, z_0^i, \vec{v}_{a0}^i, v_{z0}^i, \vec{v}_{w0}^i, \tau_0^i) \sim \mathcal{N}(\bar{\mathbf{x}}_0, \mathbf{\Sigma}_n)$
  $\mathbf{x}_0^i := (m_0^i, {\delta_T}_0^i, \vec{s}_0^i, z_0^i, \vec{v}_{a0}^i, v_{z0}^i, \vec{v}_{w0}^i, \tau_0^i)$
  $\mathrm{w}_0^i := 1/N$
let $\bar{\mathbf{x}}_0 := h^{-1}(\mathbf{y}_0)$
let $\mathbf{\Sigma}_n$ be observation noise covariance

for t=t_0 to t_N **do**
  for i=1 to N **do** ▷ SIR update
    $\mathrm{w}_t^i := \mathrm{w}_{t-1}^i p(\mathbf{y}_t|\mathbf{x}_t^i)$
    $\bar{\mathrm{w}}_t^i := \mathrm{w}_t^i / \sum_{i=1}^N \mathrm{w}_t^i$
  for i=1 to N **do** ▷ re-sampling
    draw $\bar{\mathbf{x}}_t^i \sim \sum_{i=1}^N \bar{\mathrm{w}}_t^i \delta(\mathbf{x} - \mathbf{x}_t^i)$
    $\mathrm{w}_t^i := 1/N$
  for i=1 to N **do** ▷ state evolution
    draw $\nu_t \sim \mathcal{N}(0, \mathbf{\Sigma}_\varepsilon)$
    $\mathbf{x}_{t+1}^i := f(\bar{\mathbf{x}}_t^i) + \nu_t$
  for i=1 to N **do** ▷ apply kernel
    draw $dm^i \sim \mathcal{N}(0, \sigma_{k,m}^2)$ as kernel noise for m
    $m_t^i := m_t^i + dm^i$
    draw $d{\delta_T}^i \sim \mathcal{N}(0, \sigma_{k,\delta_T}^2)$ as kernel noise for δ_T
    $\delta_{T,t}^2 := \delta_{T,t}^2 + d{\delta_T}^i$
    draw $d\psi^i \sim \mathcal{N}\{0, \sigma_{k,\psi}^2\}$ as kernel noise for heading
    compute $d\vec{v}_a^i$ from $d\psi^i$
    $\vec{v}_{a,t}^i := \vec{v}_{a,t}^i + d\vec{v}_a^i$

In this study, we use the regularized SIR particle filter proposed in Musso et al. (2001). Specifically, a stochastic kernel filter (KF) is used as an effective technique to prevent degeneration and impoverishment of the set of particles that are maintained by the particle filter. The stochastic kernel is applied to the states of interests (m and δ_T), as well as a hidden state, aircraft heading (ψ). It adds a small random Gaussian noise to the state variables of all particles after re-sampling. For m and δ_T, the kernel helps to prevent the degeneration of particles and maintain a local diversity of values. In the case of aircraft heading ψ, the kernel is essential for the functioning of the particle filter to track small changes in heading. These kernels are defined as:

$$\begin{aligned} k_m &\sim \mathcal{N}(0, \sigma_{k,m}^2) \\ k_{\delta_T} &\sim \mathcal{N}(0, \sigma_{k,\delta_T}^2) \\ k_\psi &\sim \mathcal{N}(0, \sigma_{k,\psi}^2) \end{aligned} \tag{45}$$

In this study, the $\sigma_{k,m}$ is chosen to be 0.5% of the maximum m range , which is $0.005 \times (m_{\text{mtow}} - m_{\text{oew}})$. The σ_{k,δ_T} is chosen to the 0.5% of the maximum δ_T range, which is less than 0.005×0.2. The choice of $\sigma_{k,\psi}$ is 2 degrees.

From a nonlinear system point of view, this noise has the same effect as the process noise, since they both introduce uncertainty to the state evolution (see Equation 21). However, in the context of particle filtering, this approach should be considered as a form of regularization (Musso et al., 2001). This is a simple but effective technique to prevent the degeneration and impoverishment

of particles, which lead to run-time execution errors in practice. To accomplish this, the kernel produces only a tiny variation of the states. Overall, it also compensates, to some extent, the limitation of the constant mass and thrust assumption.

6. Experiments

This section describes three different experiments that are designed to test the proposed method. In the first experiment, a simulated flight is generated with a known mass and thrust setting. Four rounds of estimations are carried out under four different noise models. The simulation is undertaken to ensure the validity of the particle filter based on the proposed system equations. In the second experiment, a real flight is chosen, and the estimation is undertaken with the same four noise models. This is to ensure that the results from a real flight are in line with the simulation results. The number of particles is set to one million. This choice reflects a balance between accuracy and computational speed. The third experiment is based on a number of flights with a Cessna Citation II laboratory aircraft, with known mass.

6.1. Experiment I: Simulation of a Boeing B737-700 flight

In this experiment, the aircraft mass m is set to be 60,000 kg, and the thrust setting δ_T is set to be 0.96. The actual observation noise for the simulated trajectory is $\mathbf{\Sigma}_{n2}$. The wind is absent in the simulation, and the climb is performed with a constant climb rate. Using the simulated trajectory, particle filters with four distinct noise levels are applied to estimate the mass and thrust settings.

In Figure 4, the convergence of particles under assumed observation noise $\mathbf{\Sigma}_{n2}$ (the same as in the simulation) is illustrated. In each convergence plot, the green line represents the true state value, the black dots are simulated observations,[2] and the gray or colored area is bounded by the minimum and maximum state values at each iteration. Under this condition, it can be seen that the mass and thrust settings nicely converge to their true values. On the other hand, convergence is less important for other states. The main goal is to be able to track the changes in these states during the filtering process.

In Figure 5, the convergence of m and δ_T under all four observation noise levels is shown. In each plot, the left-hand side red distribution corresponds to mass, and the right-hand side blue distribution represents the thrust setting. It is apparent that with increasing assumed observation noise, the uncertainty of the final results increases.

The Figure also shows that the estimates can become trapped in incorrect states when the assumed noise is lower than actual noise (in the first two plots). When the assumed noise level is much higher than the actual noise, the uncertainty in the final estimate becomes large (as demonstrated in the last two plots). These observations are consistent with the characteristics of a particle filter.

6.2. Experiment II: Example of a real Boeing B737-700 flight

In this second experiment, an actual climb trajectory of a Boeing 737-700 is used to demonstrate the application of the SIR particle filter to real flights. The trajectory data is gathered from

[2]Note that the airspeed is calculated as the difference between ground speed and wind speed.

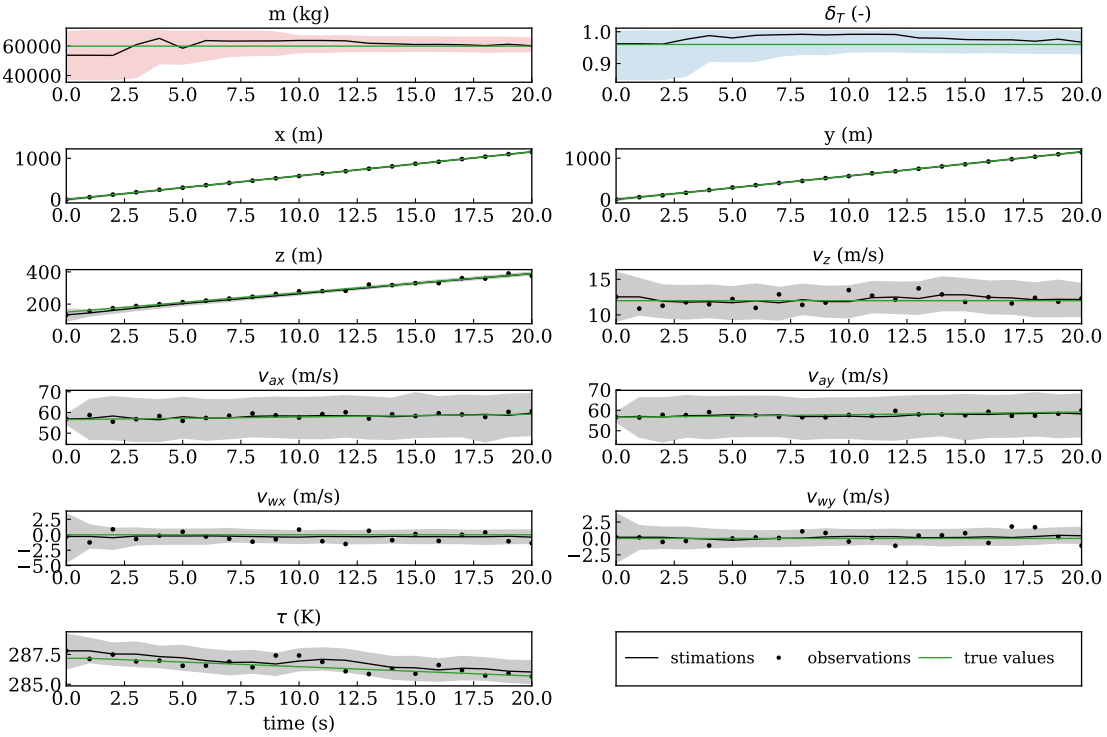

Figure 4: Convergence of the particle filter states (noise level $\mathbf{\Sigma}_{n2}$)

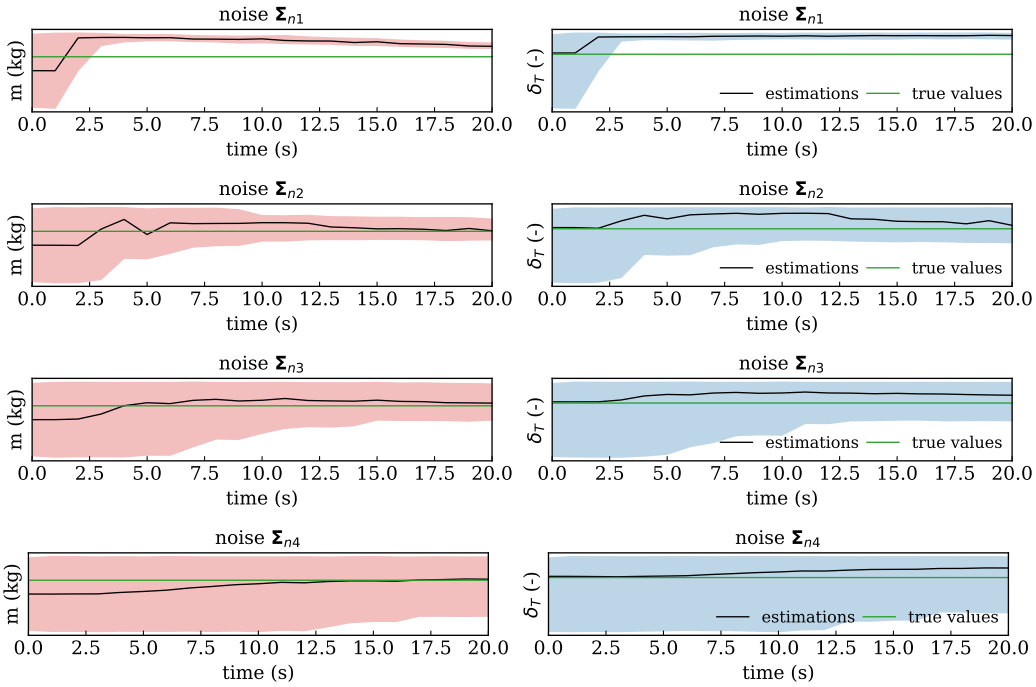

Figure 5: Final distribution of m and δ_T under different observation noise levels

ADS-B and decoded. The wind data are computed as the combination of ADS-B and Enhanced Mode-S data.

The trajectory and convergence of the particles under noise level $\mathbf{\Sigma}_{n2}$ are shown in Figure 6. Similar to the previous simulated case, the black dots represent actual observed data, while the

solid lines represent weighted average state values of particles at each iteration. In these plots, not only the mass and thrust setting exhibit convergence, but we can also see that the changes in airspeed, vertical rate, wind, and temperature are tracked by the particle filter. It is also clear that the particle filter can cope with missing data points, as shown in the plots.

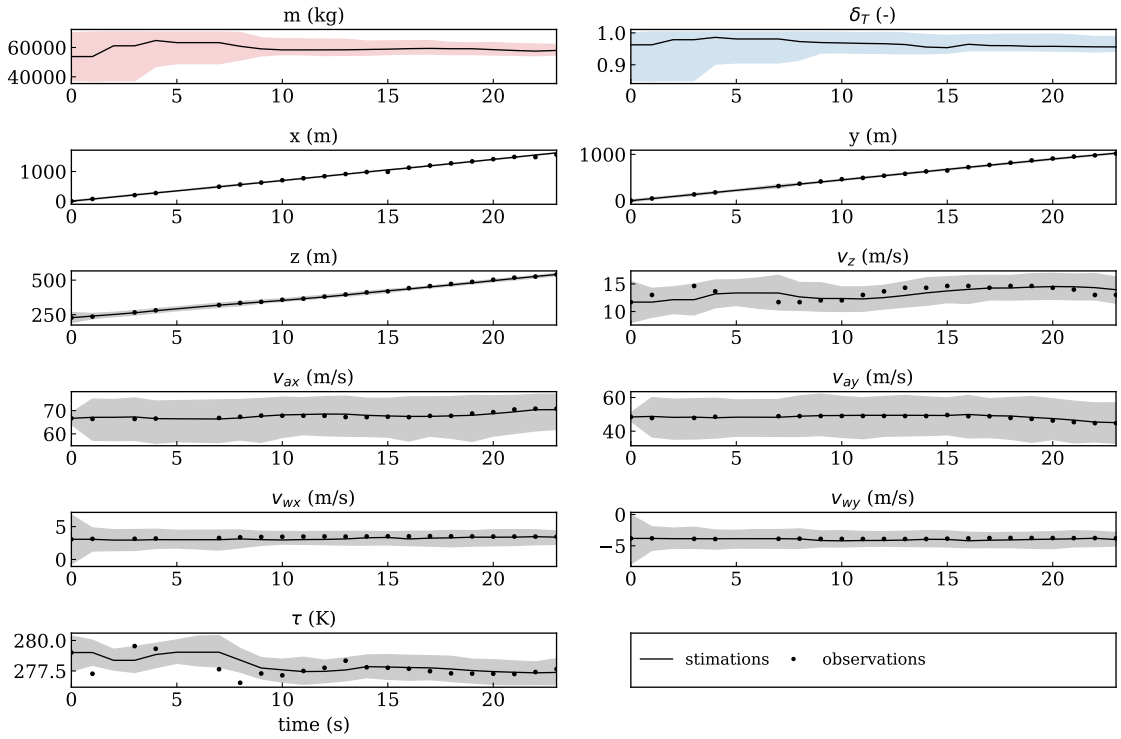

Figure 6: Convergence of the particle filter states (noise level $\mathbf{\Sigma}_{n2}$)

Figure 7 illustrates the convergence of m and δ_T under different noise levels, showing uncertainties with similar characteristics to the previous simulated case.

The use of these simulated and real flights demonstrates the applicability of the SIR particle filter for mass and thrust setting estimations. However, these experiments are based on a single sequence of filtering under each noise level. We can observe that the estimation uncertainty increases with the observation noise in both experiments, but it is not sufficient to quantify the estimation uncertainty yet.

6.3. Experiment III: Practical demonstration with Cessna Citation II flights

A set of 50 real flights with known mass is used to test the proposed system model and the particle filter. These flights were carried out by a Cessna Citation II aircraft that is operated by TU Delft for student practicals. The mass of the aircraft is obtained accurately by weighing all passengers and measuring the exact amount of fuel on board prior to the start of each flight.

Although accurate FMS trajectory data is available, we used the collected Mode-S data for the experiment. In this way, the validation is closer to real cases. In addition, NACp and NACv values are decoded from raw ADS-B data to automatically select the observation noise model. In the validation, the $\Delta\delta_T$ is assumed to be 0.1 for this aircraft.

To obtain a stable estimation, each trajectory is executed 30 times with the particle filter. This is similar to using 30 million particles in one round. The implication of this will be discussed later on. The averaged final results are shown in Figure 8. Estimated mass (indicated with crosses) is

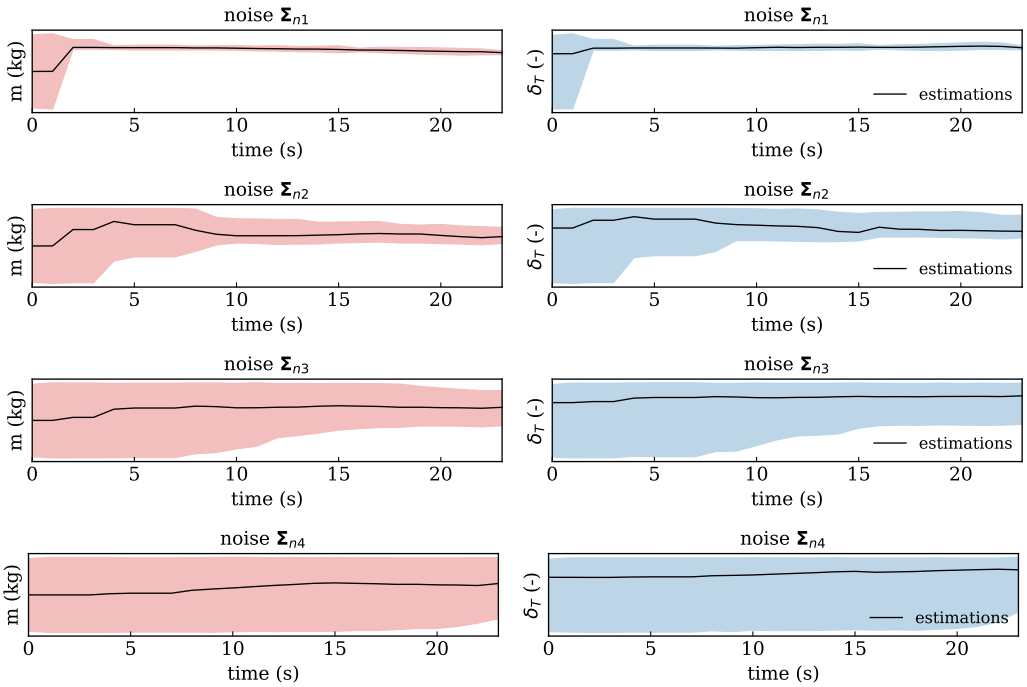

Figure 7: Final distribution of m and δ_T under different observation noise levels

plotted against the real mass of the aircraft (indicated with circles) in the top plot of the figure. The estimated thrust setting is plotted in the bottom plot.

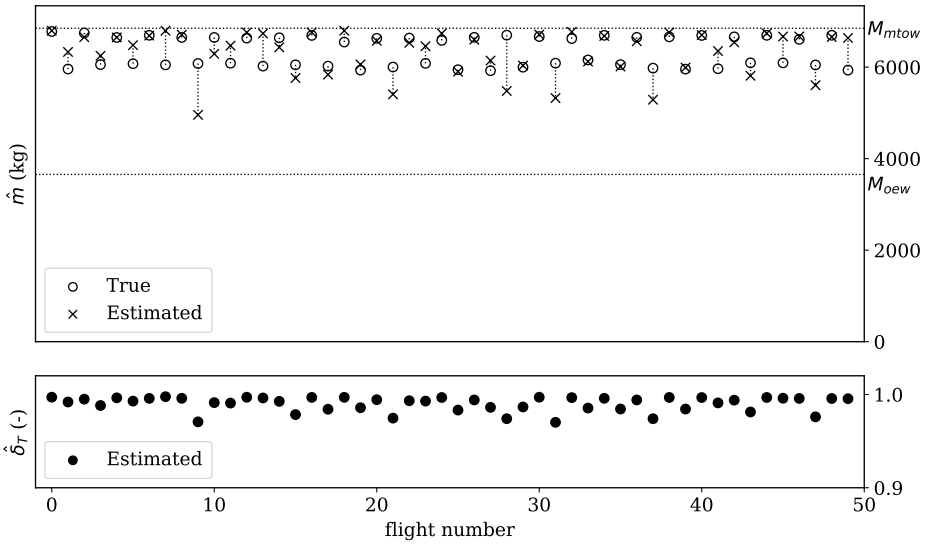

Figure 8: Estimated results for all measurement flights

The estimation errors are plotted in Figure 9. The plots on the left-hand side show the estimation errors ($\Delta m/m_{\text{true}}$), where the mean and median error are both around -0.5% of the true mass. There is a very small but not significant negative bias in the results. On the right hand side, the absolute estimation errors ($|\Delta m|/m_{\text{true}}$) are computed and shown. As a result, the mean absolute error (MAE) is found to be 4.3%, while the median of the absolute error is 2.6%.

Compared to the result from Alligier et al. (2015), a higher error is shown. However, in this ar-

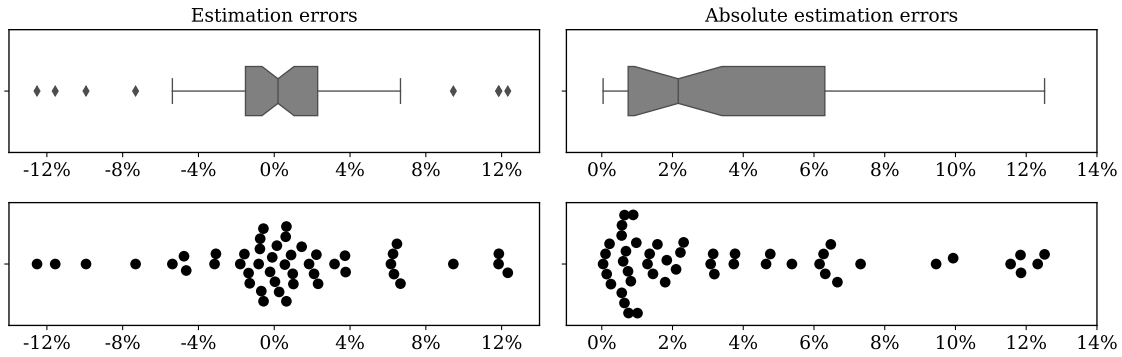

Figure 9: Distribution of estimation errors

ticle, we obtain the error by comparing the estimates with the aircraft true masses, while the error from Alligier et al. (2015) is obtained by comparing estimates from machine learning algorithms with the least-squares estimates.

7. Sensitivity analyses and discussions

This section presents sensitivity analyses of the fundamental aspects of aircraft mass estimation. This yields insight into the recommendations on choosing a noise model and practical particle filtering.

7.1. Determination the estimation uncertainty

A large number of additional simulations are performed to study the uncertainty of the estimation. 500 rounds of estimations under each defined noise model are performed for the simulated flight. The simulated Boeing B737-700 flights are generated with a mass of 60,000 kg, a thrust setting of 0.96, and a small simulation noise of $\mathbf{\Sigma}_{n1}/4$.

First, we want to illustrate the estimation accuracy using the particle filter. In Figure 10, the distributions of estimates at the final time step of all rounds are shown. The results are grouped by observation noise. On the left-hand side of the figure, the results of m are indicated, with the y-axis ranging from m_{oew} to m_{mtow}. The horizontal black line indicates the actual mass (60,000 kg) used in the simulations. On the right-hand side, the thrust settings are plotted in the same fashion. We can see that the particle filter yields a high level of accuracy with the simulated trajectory.

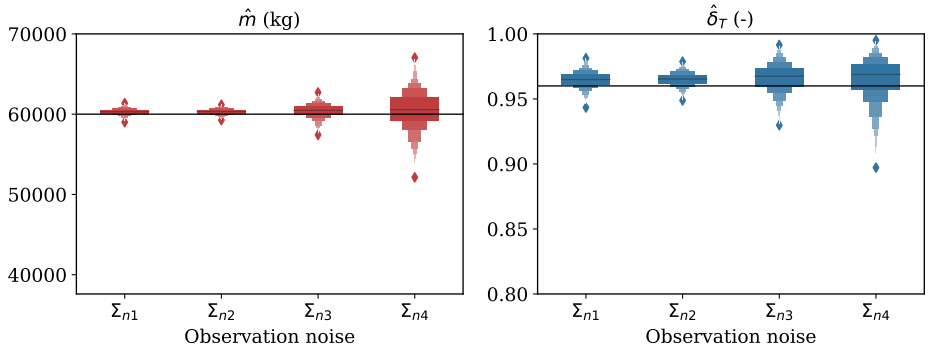

Figure 10: The distributions of estimated mass and thrust setting based on 500 rounds of simulated trajectory under different observation noise models

Next, we want to study the uncertainty of the estimation. By computing the standard deviations of the mass and thrust values from all particles at the end of each run, we are able to visualize estimation uncertainty under different noise models. On the left-hand side of Figure 11, distributions of particle mass standard deviation are shown. It can be observed that with increasing observation noise, uncertainty also increases. The right-hand side of the plot indicates the same trends for the thrust setting.

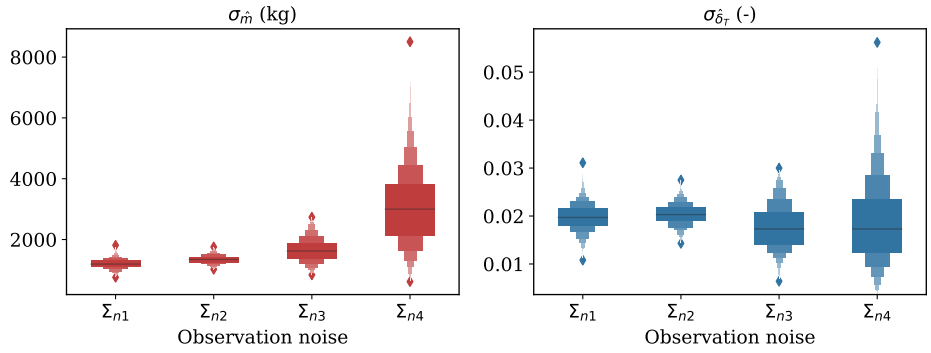

Figure 11: Standard deviation of particles states on mass and thrust setting among 500 rounds of simulations under different observation noise models

Figure 10 and Figure 11 display similarities, but show different aspects of the estimation result. The earlier figure shows the final estimates by using the mean state values from particles at last time steps of all runs. In the latter figure, the spread of particles by the measurement of standard deviation is shown.

To quantify the uncertainty under different noise models, twice the standard deviation is used as an indicator. This is then translated into the percentage of the aircraft maximum take-off mass and thrust settings, which are shown in Table 5.

Table 5: Uncertainties related to observation noise obtained using 500 round of estimations with simulated trajectories under different observation noise levels

Noise	$2\bar{\sigma}_{\hat{m}}$	% of m_{mtow}	$2\bar{\sigma}_{\hat{\delta}_T}$	%
Σ_{n1}	2403 kg	3.0%	0.040	4.0%
Σ_{n2}	2694 kg	3.4%	0.041	4.1%
Σ_{n3}	3307 kg	4.2%	0.035	3.5%
Σ_{n4}	6079 kg	7.7%	0.037	3.7%

Similarly, the accuracy of the estimation can also be obtained by comparing the estimation results with the input mass the thrust setting of these simulated trajectories. The results are listed in Table 6.

Table 6: Mean accuracy of the estimation based on 500 round of estimations with simulated trajectories under different observation noise levels

Noise	$2\bar{\sigma}_{\hat{m}}$	% of m_{mtow}	$2\bar{\sigma}_{\hat{\delta}_T}$	%
Σ_{n1}	304 kg	0.4%	0.005	0.5%
Σ_{n2}	308 kg	0.4%	0.005	0.5%
Σ_{n3}	437 kg	0.6%	0.006	0.6%
Σ_{n4}	509 kg	0.6%	0.006	0.6%

However, we have to be cautious when extrapolating the accuracy listed in Table 6. It is based on simulated trajectories based on the dynamic model that is also used by the particle filter. In

practice, the result from Section 6.3 better reflects the accuracy of the estimations. In addition, we also see that the uncertainty results are obtained based on one aircraft type. However, the percentage values from this table can still be an important indicator to quantify the uncertainties of estimations. It can be used as an indicator for the missing uncertainty component for other estimation methods that are based on ADS-B surveillance data.

In order to ensure the functioning of the SIR particle filter, we are required to have an accurate understanding of the observation noise. If the assumed noise is much larger than the actual observation noise level, the states of particles would have large variances. This, in turn, produces estimations with greater uncertainties. Examples can be seen in the last plots of Figure 5 and 7. On the other hand, if the assumed noise is much smaller than the actual noise, the particles would converge to a set of incorrect states. With the assumed noise model, all other possible states have extremely low probabilities. As such, the convergence to the real states becomes difficult. Under such a noise level assumption, the estimator will result in high confidence (low variance) but with possibly incorrect (biased) estimates. Examples can be seen in the first plots of Figure 5 and 7.

7.2. The choice of a proper observation noise model

As a general rule, the observation noise model (the covariance matrix $\mathbf{\Sigma}_{n*}$) should reflect the actual noise in the measurement data. As such, the particles would be able to track the changes in states accurately. However, in real life, actual noise from the data is not often known directly.

One way to find out the noise level in ADS-B data is to decode the NACp and NACv values as described in Section 4.1 and then use the corresponding noise models proposed in Table 3 to identify the proper noise model. This is only possible, however, when access to raw ADS-B data is available. When using other data sources, one can possibly construct the proper noise model using the vector auto-regression (VAR) method (Lütkepohl, 2005, p15).

As a rule of thumb, when raw NACp and NACv values are not available, the noise models of $\mathbf{\Sigma}_{n2}$ or $\mathbf{\Sigma}_{n3}$ are generally good starting points. This conclusion is based on our observation that the majority of the ADS-B messages meet the accuracy levels of NACp 9/10 and NACv 3/2.

7.3. The influence of variation in mass and thrust setting

From experiments in previous sections, it is noticeable that the thrust setting (δ_T) does not always converge. The same trend is not encountered with mass estimations until a larger noise model has been introduced. To understand the reason for this, the variation of these two parameters and their influence on the (simulated) flight trajectory is studied. In line with this goal, two sets of simple simulations are conducted.

In the first set of simulations, the aircraft mass is fixed at $(m_{\mathrm{oew}} + m_{\mathrm{mtow}})/2$, and the thrust setting varies from 0.85 to 1. Results are shown on the left two plots of Figure 12. In the second set of simulations, the thrust setting is kept at 0.9, and aircraft mass varies from m_{oew} to m_{mtow}. Results are shown on the right-hand side of Figure 12. Here, only the horizontal flight distances and speed profiles are illustrated.

The influence of different thrust settings on the flight trajectory is much smaller than the influence of differences in mass. This is shown in both distance traveled and the speed profile. Next, noise is added to the simulation. For simplification, only the resulting speed profiles are shown in Figure 13.

Here, we can see that the trajectories become hard to distinguish for thrust settings when noise is larger than $\mathbf{\Sigma}_{n2}$. In the case of mass, the trajectories only become hard to distinguish when noise reaches $\mathbf{\Sigma}_{n3}$. This offers an intuitive demonstration of the noise effects on trajectories.

From a statistical point of view, we can analyze the relationships between mass and thrust settings through particle distributions when convergence is reached. In Figure 14, the resulting distributions of m and their δ_T are shown. The particles are grouped by the final mass distribution.

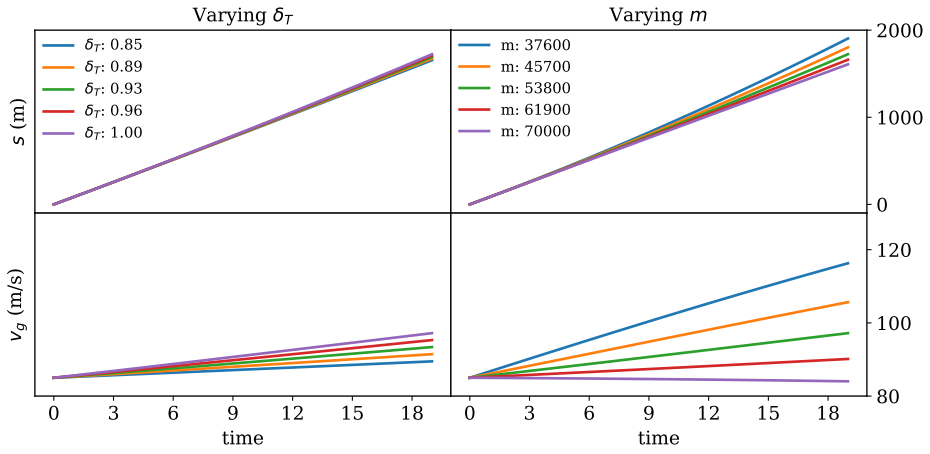

Figure 12: Varying δ_T versus varying m

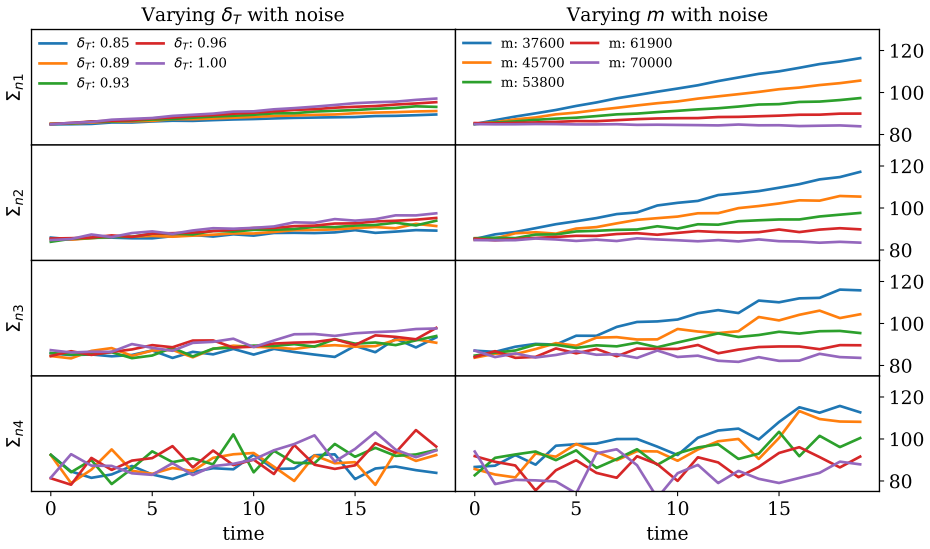

Figure 13: Simulated speeds with different noise model

Here, m values are separated in small bins, and corresponding δ_T values are shown in the right-hand plots. m and δ_T from the same groups are plotted with the same color. It is possible to see that the diversity of the particles disappears with noise level $\mathbf{\Sigma}_{n1}$. With noise level $\mathbf{\Sigma}_{n4}$, a large range of estimates is considered possible. With noise level $\mathbf{\Sigma}_{n2}$ and $\mathbf{\Sigma}_{n3}$, the true values of the states are represented. It is also clear to see the corresponding mass and thrust setting values in this visual representation.

The results from Figure 5 can be confirmed with the distributions of the particles. The estimation of mass can only be possible when the noise is smaller than or equal to $\mathbf{\Sigma}_{n3}$. The equivalent NACp and NACv values are 9 and 2 respectively. With the first two plots of Figure 5, we can also observe that when the assumed observation noise ($\mathbf{\Sigma}_{n1}$) is smaller than the actual observation noise ($\mathbf{\Sigma}_{n2}$), the estimation results are biased.

8. Conclusion

In this study, steps were taken to construct a comprehensive model and estimation method to derive aircraft mass and thrust setting solely using ADS-B and Enhanced Mode-S observations. The

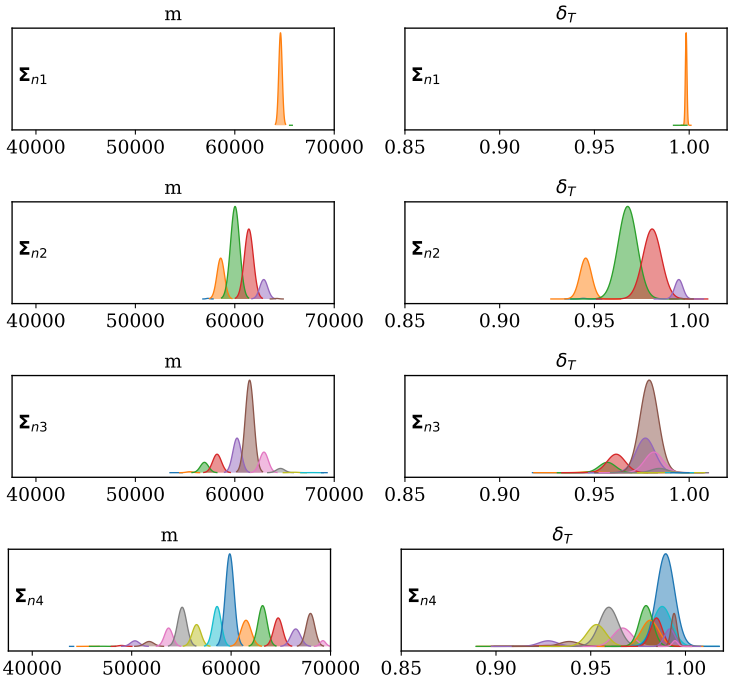

Figure 14: Relationship between deviations in estimated thrust setting and mass

complexity of such a problem was discussed at the start of this study. In summary, the difficulty of the estimation lies in solving an inverse non-linear system that consists of eleven states.

In order to address this challenging nonlinear estimation problem, this article developed a stochastic recursive Bayesian approach using a particle filter. The recursive Bayesian approach not only provided estimates but also allowed for quantifying uncertainties under different noise levels. If realistic observation noise levels can be identified, the proposed particle filter is able to estimate aircraft mass and thrust setting. Simulated, real, and measurement flights were used to test the method proposed in this study, specifically to test the method, define the uncertainty, and validate the accuracy of the method. With the measurement dataset, this particle filter approach yielded a mean absolute error of around 4.3% of the true mass. Finally, the convergence speeds were swift. In most cases, the estimation could be obtained using around 30 seconds of the flight data at the start of the climb.

In this study, we have selected the flight segment where the thrust setting is most likely to be at the maximum, which is in the initial climb phase. The maximum thrust is computed using the model proposed by Bartel and Young (2008). Even with the reduction setting (δ_T) modeled, we still cannot completely mitigate the uncertainty in the underlying thrust model. However, for the errors of the particle filter, estimated mass and thrust setting during initial take-off a strong positive correlation is known, i.e., a too high assumed thrust setting leads to a too high estimated aircraft mass. When testing the particle filter with reference flights from a Cessna Citation II, we were able to increase the accuracy of the estimation with better knowledge of the thrust setting. Changing from 0.85 to 0.95 allows the particle filter to converge closer to the real mass. However, this assumption is specific to our test flight. For common commercial airlines, we still recommend using the value of 0.85 unless there is evidence indicating otherwise.

By linking the ADS-B native uncertainty reporting system with the observation noise, we were able to construct realistic observation noise models. At the same time, we were also able to automatically select the appropriate noise model for each flight using raw ADS-B data when available. Knowing the effect of these uncertainties by using a particle filter can be beneficial for future

ADS-B-related studies. Better understanding the noise model in surveillance data is the key to improving the accuracy of the dynamic model used for estimation. This can be an interesting area for follow-on studies.

In the current model, the bank angle is left out of the equations, which is, of course, a simplification. The consequence is that we are only able to use non-turning flight segments for the estimations. Introducing an additional parameter for the bank angle could enable the estimation to use data from turning trajectories. However, this may increase the complexity of the model, as well as the estimation uncertainty.

A simplified form of the observation noise is assumed in this study. That is, the noise for different observable states is assumed to be uncorrelated. However, this may not be true in real situations. Auto-correlation of errors may exist as suggested in Mohleji and Wang (2010). If such knowledge can be discovered, it may further improve the estimation. The noise models that are defined in this study correspond to Version 1 and Version 2 of the ADS-B implementation. For older ADS-B Version 0 transponders (equipped on older aircraft types), the noise should be defined based on its Navigation uncertainty Categories (NUC). The NUC levels are less refined than NAC in Version 1 and 2. Future implementations of our method should take this into consideration.

The proposed Bayesian filtering framework has shown its potential for parameter estimation. It was also noticeable that the designed particle filter can be a good candidate for aircraft trajectory state filtering. When mass can be estimated at the start of a flight, other state parameters along the entire flight can be better estimated or predicted. This will bring more insight to, for example, fuel consumption and trajectory prediction during the rest of the flight. Now that the estimation of mass using a particle filter has been demonstrated, one of the logical follow-on steps is to investigate applications of other non-linear estimation approaches. If in such case the state space would further grow, then it might also useful to consider other nonlinear estimation methods such as combinations of particle filtering and Markov Chain Monte Carlo simulation.

Finally, to allow open access to the method developed in this study, the source code of the particle filter is published alongside this article. [3]

[3] Available at: `https://github.com/junzis/acsmc`

References

Alligier, R., Gianazza, D., Durand, N., 2013. Ground-based estimation of aircraft mass, adaptive vs. least squares method. In: 10th ATM Seminar for Air Traffic Management Research and Development.

Alligier, R., Gianazza, D., Durand, N., 2015. Machine learning and mass estimation methods for ground-based aircraft climb prediction. IEEE Transactions on Intelligent Transportation Systems 16 (6), 3138–3149.

Alligier, R., Gianazza, D., Hamed, M. G., Durand, N., 2014. Comparison of two ground-based mass estimation methods on real data. In: 6th International Conference on Research in Air Transportation.

Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. IEEE Transactions on signal processing 50 (2), 174–188.

Bartel, M., Young, T. M., 2008. Simplified thrust and fuel consumption models for modern two-shaft turbofan engines. Journal of Aircraft 45 (4), 1450–1456.

Blom, H. A. P., Bloem, E. A., 2007. Exact bayesian and particle filtering of stochastic hybrid systems. IEEE Transactions on Aerospace and Electronic Systems 43 (1).

Chati, S., Balakrishnan, H., 2018. Modeling of aircraft takeoff weight using gaussian processes. Journal of Air Transportation, 1–10.

Chati, Y. S., Balakrishnan, H., 2017. Statistical modeling of aircraft takeoff weight. In: 12th ATM Seminar for Air Traffic Management Research and Development.

Douc, R., Cappé, O., 2005. Comparison of resampling schemes for particle filtering. In: Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on. IEEE, pp. 64–69.

Doucet, A., De Freitas, N., Gordon, N., 2001. An introduction to sequential monte carlo methods. In: Sequential Monte Carlo methods in practice. Springer, pp. 3–14.

Liu, J. S., Chen, R., 1998. Sequential monte carlo methods for dynamic systems. Journal of the American statistical association 93 (443), 1032–1044.

Lütkepohl, H., 2005. New introduction to multiple time series analysis. Springer Science & Business Media.

Mohleji, S. C., Wang, G., 2010. Modeling ads-b position and velocity errors for airborne merging and spacing in interval management application. MITRE release, 10–3026.

Musso, C., Oudjane, N., Le Gland, F., 2001. Improving regularised particle filters. In: Sequential Monte Carlo methods in practice. Springer, pp. 247–271.

Nuic, A., 2014. User manual for the base of aircraft data (bada) revision 3.12. EUROCONTROL 2014.

RTCA, 2011. Minimum operational performance standards for 1090mhz extended squitter automatic dependent surveillance-broadcast (ads-b) and traffic information services-broadcast (tis-b)[j]. RTCA DO-260B with Corrigendum 1 (1), 1365–1372.

Schultz, C., Thipphavong, D., Erzberger, H., 2012. Adaptive trajectory prediction algorithm for climbing flights. In: AIAA Guidance, Navigation, and Control (GNC) Conference. p. 2.

Sun, J., Blom, H. A., Ellerbroek, J., Hoekstra, J. M., 2018a. Aircraft mass and thrust estimation using recursive bayesian method. In: 8th International Conference on Research in Air Transportation.

Sun, J., Ellerbroek, J., Hoekstra, J., 2016. Modeling and inferring aircraft takeoff mass from runway ads-b data. In: 7th International Conference on Research in Air Transportation.

Sun, J., Ellerbroek, J., Hoekstra, J. M., 2018b. Aircraft initial mass estimation using bayesian inference method. Transportation Research Part C: Emerging Technologies 90, 59–73.

Sun, J., Hoekstra, J. M., Ellerbroek, J., 2018c. Aircraft drag polar estimation based on a stochastic hierarchical model. In: 8th International Conference on Research in Air Transportation.

Sun, J., Vû, H., Ellerbroek, J., Hoekstra, J. M., 2018d. Weather field reconstruction using aircraft surveillance data and a novel meteo-particle model. PloS one 13 (10), e0205029.