JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 33, No. 2, March–April 2010

Improved Multi-Aircraft Ground Trajectory Prediction for Air Traffic Control

Ioannis Lymperopoulos* and John Lygeros†
Swiss Federal Institute of Technology, Zurich CH-8092, Switzerland

DOI: 10.2514/1.46190

Accurate trajectory prediction plays a fundamental role in advanced air traffic control operations, because it forms the basis for (among others) conflict detection and resolution schemes. It has been demonstrated that improved trajectory prediction accuracy can be achieved using radar measurements for a single aircraft, but the benefits are expected to be much greater if one can fuse measurements from multiple aircraft at different locations and time instants. It is shown here how this multi-aircraft sensor fusion problem can be formulated as a high-dimensional state estimation problem. A novel particle filtering algorithm is developed to solve it in realistic scale situations. By exploiting the structure of the problem, one can address the technical challenges that arise in the process: efficiently handling the information, dealing with the estimation of a very high-dimensional state, and dealing with the nonlinear dynamics of aircraft motion and control. The effectiveness of the novel algorithms is demonstrated on feasibility studies involving multiple aircraft (from one to several hundred). The studies show that in the presence of multiple aircraft the trajectory prediction results approach the theoretical limit of accuracy under these conditions.

Nomenclature

$AP(\cdot)$	=	atmospheric pressure as a function of altitude
C_L	=	aerodynamic lift coefficient
c_t, d_t, G_t, g_t, e_t	=	fit parameters for wind error correlation in time
c_{XY}, G_{XY}, b_{XY}	=	fit parameters for wind error correlation in horizontal plane
c_Z, G_Z	=	fit parameters for wind error correlation with altitude
g	=	gravitational acceleration, 9.81ms^{-2}
m	=	aircraft mass
$\widehat{R}$	=	total wind error covariance matrix
r_t, r_{XY}, r_Z	=	wind error correlation with time, horizontal, and altitude difference
S	=	wing surface area
T	=	engine thrust
w_X,w_Y	=	wind in the east (X) and north (Y) direction
V	=	true air speed
X,Y,Z	=	aircraft position in the east, north, direction and altitude
η	=	fuel flow coefficient
$\rho(\cdot)$	=	air density as a function of altitude
σ_x,σ_r	=	standard deviation of state evolution noise and radar measurement noise
$\sigma(\cdot)$	=	standard deviation of wind error as a function of altitude
ϕ	=	bank angle
ψ	=	heading angle

Presented at the AIAA Guidance, Navigation, and Control Conference, Chicago, IL, 10–13 September 2009; received 30 June 2009; revision received 20 November 2009; accepted for publication 2 December 2009.

*Ph.D. Student, Department of Information Technology and Electrical Engineering, Automatic Control Laboratory.

†Professor, Department of Information Technology and Electrical Engineering, Automatic Control Laboratory.

I. Introduction

AIRCRAFT trajectory prediction (TP) is central to many advanced air traffic management (ATM) operational concepts. For example, TP is at the heart of most conflict detection and resolution algorithms. By comparing the predicted trajectories of different aircraft against each other, accurate TP can help detect real threats while avoiding false alarms. Large TP uncertainty forces Air Traffic Controllers (ATC) to use larger separations between aircraft to ensure safety, thus reducing the total number of aircraft a given sector can accommodate. Moreover, changing a flight path for fear of a possible conflict, can decrease the fuel efficiency, or affect the time of arrival of the aircraft involved. Therefore, more accurate TP may help ATC to accommodate for increased traffic demand. Current forecasts predict a significant increase in air traffic for the near and long-term future [1,2].

TP accuracy is naturally limited by the uncertainty inherent in every aircraft flight. Much of the TP uncertainty is due to meteorological forecast inaccuracies, especially those concerning the wind. It has been shown that the deviation of wind forecasts from the real wind conditions is correlated in time and space [3], which implies that the effect of forecast errors accumulates over time, leading to large uncertainties about the predicted position of aircraft [4]. The design of the flight management systems (FMS) also contributes to the propagation of the error. Most aircraft do not correct along-track deviations over horizons of tens of minutes, mainly for fuel efficiency reasons. Even though this has already started to change with the introduction of 3.5-D FMS systems that are able to track way points with a required time of arrival (RTA), for the current fleet and operating procedures large along-track uncertainty is the norm [5].

In this paper it is demonstrated how to reduce TP inaccuracies related to wind-forecast errors. Roughly speaking, the aim is to build online a probabilistic wind map to capture the difference between the wind forecast and the real wind conditions. This wind map is then added to the wind forecast and used to improve the accuracy of TP. The key observation that enables this approach is that past aircraft trajectories, available to the air traffic controllers through radar measurements or datalink, encapsulate information about the wind the aircraft has experienced. In other words, each aircraft is treated as a moving, local sensor of the wind field. Then the spatio-temporal correlation of the error in the wind forecasts [3] is exploited, to fuse the information from the multiple aircraft and to interpolate in space/time to points where/when aircraft measurements are not available. An implicit advantage of the proposed method is that it provides

more accurate wind estimates in regions with dense air traffic, where accurate TP is needed the most.

The problem of extracting the wind-forecast error information from the trajectories of the aircraft is formulated as a filtering (or state estimation) problem. The state that has to be estimated is high-dimensional, since it comprises the states (position and heading) of all aircraft in the region of interest, as well as the wind-forecast error (projected on a grid). The situation is further complicated by the fact that the aircraft dynamics, through which the wind-forecast error is indirectly observed, are nonlinear. This implies that robust filtering methods (such as the Kalman filter) are inapplicable in the case that is examined here, whereas methods that could cope with nonlinear dynamics [such as the particle filter (PF)] have difficulties dealing with high-dimensional states. To solve the problem a novel particle filtering algorithm is developed [called sequential conditional particle filter (SCPF)] that can deal with both the nonlinear and the high-dimensional nature of the problem. For this, the special structure afforded by the filtering problem in question is exploited, namely, the fact that wind dynamics are linear and the fact that, conditional on the wind, the dynamics of the different aircraft are independent.

Related ideas for improving TP by reducing wind uncertainty are explored in [6–8]. The impact of wind uncertainty on the TP along-track error and the effect of initial conditions is investigated in [6]. Wind-field estimation is performed in [7], employing an extended Kalman filter, by linearizing the model dynamics. The accuracy of the method is demonstrated for a constant wind and requires a number of aircraft turns (with known turn rate) to be used. Finally, [8] makes use of multiple-aircraft wind measurements and builds a Kalman filter to estimate the wind error, during the descend phase. The key innovation of the methods developed here is the generality of the PF approach, which allows us to model wind varying in time and space and treat the nonlinear dynamics of the aircraft through which the wind information is obtained.

PFs are not new in aviation and have been used extensively in air traffic applications, especially for tracking [9–12]. Their main attraction is that they pose effectively no restrictions on the process dynamics and measurements and are suitable for nonlinear systems with non-Gaussian noise. Moreover, it can be shown that they converge to an optimal solution as the computational resources tend to infinity. Particle filters also have some drawbacks, however, which limit their applications to practical problems. The main one is degeneracy, a situation where the discrete approximation of the continuous probability density of interest degenerates to a single point, severely degrading accuracy. This problem is especially severe in high-dimensional spaces, a situation known as the curse of dimensionality [13,14]; in the case examined here degeneracy starts to emerge already after one or two aircraft for the classical PF. The SCPF aims to alleviate this drawback by exploiting the problem structure. The SCPF shares some of the insights of the marginalized particle filter (MPF) [11], in the sense, that both treat the linear and the nonlinear part of the state separately. The two main novelties compared with the MPF is the sequential incorporation of information from different aircraft and the substitution of particles carrying uncertainty realizations by particles carrying conditional distributions. A related idea was proposed in [15] for static instead of dynamic models.

The performance of the proposed algorithms is documented through simulation based feasibility studies. The studies treat the case of multiple aircraft flying level, possibly at different altitudes, with known airspeeds and aerodynamic equations (both depending on the aircraft type). This is an ideal situation in an air traffic control context, since airspeeds and aerodynamic coefficients are only partially known in general. The aim is to determine whether the proposed approach is viable, in particular, whether aircraft trajectories contain enough information to estimate wind-forecast errors and whether the proposed filtering algorithms can deal computationally with the problem. The results are very encouraging on both fronts. The proposed algorithms can deal with large numbers of aircraft (up to hundreds) while avoiding degeneracy. Moreover, in the presence of sufficient aircraft, the trajectory prediction results using the improved wind map produced by the novel algorithm are close to the theoretical limit corresponding to perfect knowledge of the current wind conditions in the entire region of interest. The studies reported in this paper assume that information is collected only through ground radar, but the methods can easily be adapted if datalink information (for example, onboard measured wind speed) is also available. Current studies concentrate on TP for nonlevel flights with unknown airspeed and unknown aircraft mass.

In the beginning the model for the aircraft dynamics and wind generation is presented in Secs. II and III. The nonlinear filtering and PF algorithms are outlined in Sec. IV, followed by the presentation of the novel PF algorithm for air traffic TP. Section V presents simulation results that demonstrate the effectiveness of the proposed method. Section VI summarizes the results and highlights possible drawbacks and directions for future work.

II. Aircraft Model

To support the subsequent development of PF algorithms, a model [16,17] that simulates the dynamics of commercial aircraft from the point of view of an air traffic controller is outlined. The model is framed in the context of stochastic hybrid systems [18] and is capable of capturing multiple instances of flights, each with a different flight plan, aircraft dynamics, and flight management system, representing different types of aircraft (Fig. 1). The physical motion of the aircraft gives rise to the continuous part of the model. The flight plan and the logic variables embedded in the FMS give rise to the discrete part. The weather uncertainty (mainly the wind-forecast error) is treated as a stochastic disturbance to the model, that has its own dynamics (Sec. III). The model presented below concentrates on level flights with constant airspeed, used in the feasibility studies. It is a simplified version of a full model (including varying true airspeed (TAS), altitude, and control of flight path angle) developed in earlier work [17], which was successfully used for TP [19] and conflict detection [20] in level, climb, and descend situations.

A. Aircraft Dynamics

From the point of view of air traffic controller (ATC) an aircraft can be adequately modeled using a point mass model (PMM). Even though PMM do not reflect all the complications of aircraft flight, they are reasonably accurate and very commonly used in ATM research [21–25]. Let g be the gravitational acceleration, m denote the mass of the aircraft, and Z denote its altitude (assumed to be constant for each aircraft but not the same for all aircraft). Figure 2 summarizes the major variables of a PMM: the horizontal position (East—X and North—Y) of the aircraft, the TAS[‡] (V), the heading angle (ψ) and bank angle (ϕ). It is assumed that each aircraft flies with known, constant TAS which depends on aircraft type and altitude. The forces applied to the aircraft are its weight (mg), the engine thrust (T), and the aerodynamic forces of lift (L) and drag (D). The aerodynamic forces in practice depend on the angle of attack (α not shown in the figure) and the side slip angle (β). From the point of view of ATC, however, trimmed flight conditions can be assumed and thus one can set $\alpha = \beta = 0$. Moreover by ignoring fast dynamics, T and ϕ are treated as inputs. The movement of the aircraft is also affected by the wind (not shown in the figure) which acts as a disturbance. The wind is modeled through its speed $(w_X, w_Y) \in \mathbb{R}^2$. Thus, the equations of motion for level flight become

$$\begin{bmatrix} \dot{X}(t) \\ \dot{Y}(t) \\ \dot{\psi}(t) \\ \dot{m}(t) \end{bmatrix} = \begin{bmatrix} V\cos(\psi(t)) + w_X(t) \\ V\sin(\psi(t)) + w_Y(t) \\ \dfrac{C_L S \rho(Z) V \sin(\phi(t))}{2m(t)} \\ -\eta T(t) \end{bmatrix} \tag{1}$$

C_L is the lift coefficient, S represents the surface of the wings, $\rho(\cdot)$ the air density depending on altitude (Z), and η the fuel burn coefficient. The values of the parameters (including the TAS) which depend on aircraft type, the phase of flight and aircraft configuration are obtained from the base of aircraft data (BADA) database [25].

[‡]True airspeed is the speed of the aircraft relative to the surrounding air.

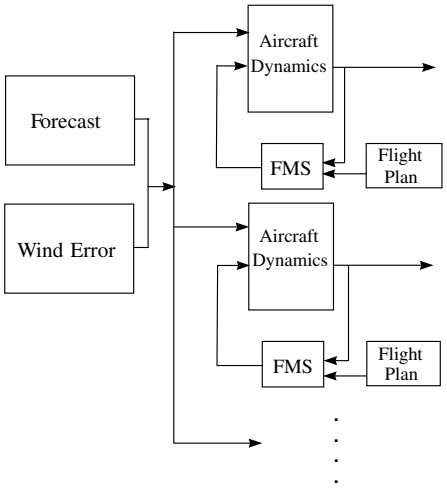


Fig. 1 Block diagram of the model components for multiple aircraft flying in the same wind field.

B. Flight Management System

The FMS can be thought as a controller that measures the state of the aircraft and uses it together with the flight plan to determine the values of the inputs. The FMS is divided into two components, one controlling along track and vertical motion (in this case maintaining constant altitude and airspeed) through the thrust and flight path angle and the other controlling cross-track motion through the bank angle. This arrangement is illustrated in Fig. 3.

The values of the inputs are determined to some extent by conventional continuous controllers. To ensure constant airspeed the thrust is set equal to the drag force, whereas to ensure level flight the flight path angle is assumed identically to be equal to zero. The bank angle is set using a nonlinear feedback controller which corrects cross-track deviations from the flight plan. Details of the design of these controllers are given in [16,17].

The discrete dynamics arise from the flight plan of the aircraft and the logic variables embedded in the FMS. Following [25], the discrete state of the FMS is stored in eight discrete variables: flight level (FL), way-point index (WP), acceleration mode (AM), climb mode (CM), speed hold mode (SHM), flight phase (FP), reduced power mode (RPM), and troposphere mode (TrM). Changes in the discrete state affect both the aircraft dynamics (e.g., through the lift and drag coefficients) and the control inputs applied by the FMS. In the simplified level flight model considered here, some of the discrete states (like AM and CM) remain constant. The remaining discrete states evolve by jumping from one value to another whenever certain conditions on the continuous state are met; the overall system is therefore hybrid. For more details the reader is referred to [16,17,25].

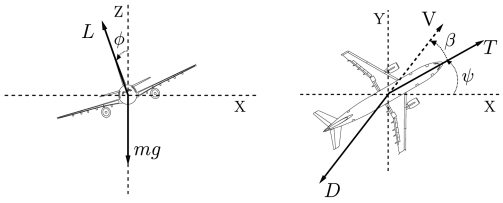


Fig. 2 Forces acting on an aircraft, flying level.

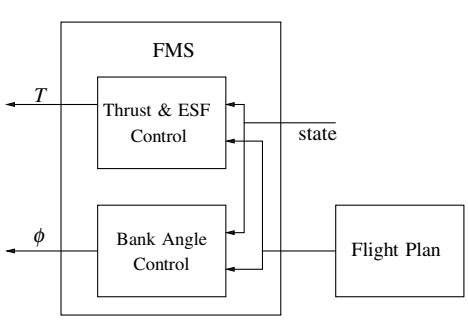


Fig. 3 FMS controllers.

C. Radar Model

The position of all aircraft is measured using an ATC ground radar. This represents only a partial observation of the aircraft state; for example, mass and aircraft heading are not directly measured. Moreover, the radar measurements are assumed to be corrupted by noise. In practice the accuracy of the radar usually decreases as an aircraft moves away from the radar location. For a more realistic model the polar description of the radar noise should be used. For simplicity the noise is added in the x and y position of the aircraft. Furthermore, the same measurement error statistics for all distances is used, and the variance is selected high enough ($\sigma_r = 80$ m) to err on the side of caution. Radar measurements are assumed to arrive sequentially at discrete time steps. Normally these would be every 6–12 s, but it is demonstrated below that the proposed algorithm works well even with measurements spaced every 30 s.

III. Wind Model

A large part of the uncertainty about the evolution of flights stems from the fact that meteorological forecasts are inherently inaccurate [26]. Among the different weather phenomena affecting an aircraft, wind speed is arguably one of the most important [27]. The wind is modeled as a sum of two components: a nominal component (representing weather forecasts) and a stochastic component (representing forecast errors).

A. Wind Forecast

The nominal part of the wind field represents the meteorological predictions that are available to the ATC. Those meteorological data are obtained mainly from the rapid update cycle (RUC), a numerical weather prediction model for the U.S.A. that incorporates aircraft measurements, balloon soundings and other sensor data [28]. The RUC model is run every three hours and each run produces a set of three hourly forecasts. For each run the data are stored in a 3-D grid with a horizontal resolution of 60 km and a vertical resolution of 50 mb.[§] The wind is computed for time and space points not corresponding to grid nodes by linear interpolation between the neighboring grid nodes.

B. Wind-Forecast Error Statistics

The stochastic component of the wind is modeled as a random field: w: $\mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}^2$ where $w(t, P)$ represents the wind at point $P \in \mathbb{R}^3$ and at time $t \in \mathbb{R}$. Since the scenarios discussed here concern level flights, wind in the vertical direction is ignored. In earlier work, a very general method for generating wind uncertainty scenarios that respect given statistics was developed [16,17]. For simplicity, here the attention is restricted to the case where $w(t, P) \in \mathbb{R}^2$ is Gaussian with zero mean and covariance matrix $R(t, P, t', P') \in \mathbb{R}^{2\times 2}$. The zero mean assumption reflects a hypothesis that weather forecasts contain all deterministic information about the wind. This hypothesis is not crucial and the novel algorithm can still be implemented even if forecast error contains a bias. Furthermore, the wind field is assumed isotropic (invariant under rotations) and that wind speeds in the north–south and east–west directions are uncorrelated. It should be noted that when the forecast error is not strictly Gaussian, the algorithm proposed here will exhibit a suboptimal performance depending on how severely this assumption is violated. Under these assumptions the covariance matrix R simplifies to

[§]With newer versions of RUC 40/20/13 km of horizontal resolution and hourly runs can be used.

$$R(t,P,t',P') = E[w(t,P)w^T(t',P')] = \begin{bmatrix} r(t,P,t',P') & 0 \\ 0 & r(t,P,t',P') \end{bmatrix} \quad (2)$$

where E denotes expected value, and for $P=(X,Y,Z)\in\mathbb{R}^3$, $P'=(X',Y',Z')\in\mathbb{R}^3$, $t,t'\in\mathbb{R}$,

$$r(t,P,t',P') = \sigma(Z)\sigma(Z')r_t(|t-t'|)r_{XY}\left(\left\| \begin{matrix} X-X' \\ Y-Y' \end{matrix} \right\|\right) r_Z(|AP(Z) - AP(Z')|) \quad (3)$$

$AP(Z)$ is the atmospheric pressure in millibar (mb) and $\sigma(Z)$ is the standard deviation of the wind error in m/s at altitude Z. In the simulations $AP(Z)$ is computed using the standard atmosphere model and $\sigma(Z)$ based on the data reported in [3]. Based on [3], for $s\geq 0$

$$r_t(s) = c_t + (1-c_t-d_t)e^{-s/G_t} + d_t\cos\left(2\pi\frac{s-e_t}{g_t}\right) \quad (4)$$

$$r_{XY}(s) = c_{XY} + (1-c_{XY})e^{-s/G_{XY}} \quad (5)$$

$$r_Z(s) = c_Z + (1-c_Z)e^{-s/G_Z} \quad (6)$$

Note that the correlation decays exponentially as horizontal distance, altitude, or time difference increase. The parameters σ, c_t, d_t, G_t, g_t, e_t, c_{XY}, G_{XY}, b_{XY}, c_Z, G_Z are set according to [3]. The parameter values suggest a strong correlation between wind errors in the same horizontal plane, a very strong correlation in time, and a weaker correlation across different altitudes.

C. Wind-Error Generation

For simulation purposes and for the implementation of the filtering algorithms, a method for generating wind realizations that respect the preceding correlation structure is needed. Different ways of doing this have been proposed [16,17] aimed primarily at simulation. Here, a new approach is proposed that is better suited for filtering, since it generates wind realizations efficiently and in a sequential manner at the expense of a small approximation in the correlation structure.

To describe the wind field the airspace is gridded into a lattice comprising 1) N_X points in the east—west direction, spaced every δ_X (e.g., $\delta_X = 60$ km); 2) N_Y points in the north—south direction, spaced every δ_Y (e.g., $\delta_Y = 60$ km); 3) N_Z points vertically, spaced every δ_Z (e.g., $\delta_Z = 1000$ ft); 4) N_{t+1} points in time, spaced every δ_t (e.g., $\delta_t = 30$ s).

For each vertex in the lattice two random numbers are generated, one for the north-south and one for the east-west direction of the wind error. Let $i\in\{1,\dots,N_X\}$, $j\in\{1,\dots,N_Y\}$, and $h\in\{1,\dots,N_Z\}$ denote the indices of a vertex in the lattice, $k\in\{0,\dots,N_t\}$ denote the current time step, and $w_X(i,j,h,k)\in\mathbb{R}$ and $w_Y(i,j,h,k)\in\mathbb{R}$ denote the wind in the two directions corresponding to the lattice point; in the notation of the preceding section

$$w\left[k\delta_t,\ \begin{pmatrix} i\delta_X \\ j\delta_Y \\ h\delta_Z \end{pmatrix}\right] = \begin{bmatrix} w_X(i,j,h,k) \\ w_Y(i,j,h,k) \end{bmatrix} \in \mathbb{R}^2$$

This is ordered lexicographically in h, j, i, to form two vectors

$$W_X(k) = \begin{bmatrix} w_X(1,1,1,k) \\ w_X(2,1,1,k) \\ \vdots \\ w_X(N_X,1,1,k) \\ w_X(1,2,1,k) \\ \vdots \\ w_X(N_X,N_Y,N_Z,k) \end{bmatrix} \in \mathbb{R}^{N_XN_YN_Z}$$

$$W_Y(k) = \begin{bmatrix} w_Y(1,1,1,k) \\ w_Y(2,1,1,k) \\ \vdots \\ w_Y(N_X,1,1,k) \\ w_Y(1,2,1,k) \\ \vdots \\ w_Y(N_X,N_Y,N_Z,k) \end{bmatrix} \in \mathbb{R}^{N_XN_YN_Z} \quad (7)$$

To derive the dynamics of the vectors $W_X(k)$ and $W_Y(k)$ the time correlation function is approximated by $r_t(s)\approx e^{-s/G_t}$. This is a very good approximation for the time horizons of interest (30 s–1 h). The quality of the approximation decreases for time differences of more than 3 h, since the effect of the cosine term in Eq. (4) becomes stronger (Fig. 4).

Let $\hat{R}\in\mathbb{R}^{N_XN_YN_Z\times N_XN_YN_Z}$ denote the covariance matrix of $W_X(k)$ (by the isotropic assumption, the matrix will be identical for $W_Y(k)$). Based on the discussion in the preceding section and the simplifying assumption on r_t this covariance matrix is constant in time. Moreover, its elements (describing the correlation between two spatial grid points, say (i,j,h) and (i',j',h'), at the same point in time k) can be calculated as

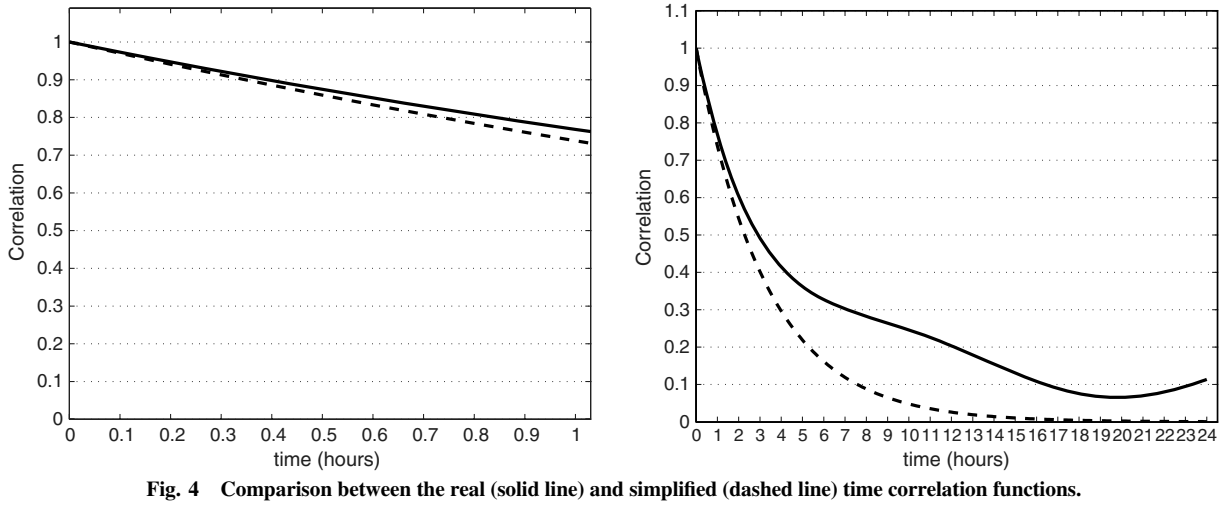


Fig. 4 Comparison between the real (solid line) and simplified (dashed line) time correlation functions.

$$r\left(k\delta_t, \begin{bmatrix} i\delta_X \\ j\delta_Y \\ h\delta_Z \end{bmatrix}, k\delta_t, \begin{bmatrix} i'\delta_X \\ j'\delta_Y \\ h'\delta_Z \end{bmatrix}\right)$$
$$= \sigma(h\delta_Z)\sigma(h'\delta_Z) r_{XY}\left(\sqrt{[(i-i')^2\delta_X^2 + (j-j')^2\delta_Y^2]}\right) r_Z(|AP(h\delta_Z) - AP(h'\delta_Z)|)$$

Samples that respect the simplified correlation structure can now be generated using the following linear Gaussian model:

$$W_X(0) = \hat{Q} v_X(0), \qquad W_X(k+1) = aW_X(k) + Q v_X(k)$$
$$W_Y(0) = \hat{Q} v_Y(0), \qquad W_Y(k+1) = aW_Y(k) + Q v_Y(k) \tag{8}$$

where $v_X(k)$, $v_Y(k) \in \mathbb{R}^{N_X N_Y N_Z}$ are standard (zero mean, identity covariance matrix) independent Gaussian random variables. Q and $\hat{Q}$ are derived by Cholesky decomposition from the covariance matrix $\hat{R}$ according to

$$QQ^T = (1-a^2)\hat{R} \quad \text{and} \quad \hat{Q}\hat{Q}^T = \hat{R} \tag{9}$$

Finally, it is set

$$a = e^{-d_t/G_t} \in \mathbb{R} \tag{10}$$

It is easy to show that the resulting vectors indeed respect the approximate correlation structure; for completeness a proof of this fact is given in the appendix. Linear interpolation of the wind at the neighboring grid vertices is used to compute the wind error between the grid points.

IV. Nonlinear Filtering

The sequential conditioning particle filter algorithm developed to estimate the wind field and perform trajectory prediction using the models described earlier is presented here. A brief overview of nonlinear filtering and particle filters and the analytical description of the novel structure of the proposed algorithm are provided.

A. General Problem Formulation

Filtering of states that evolve with nonlinear dynamics using inaccurate or partial observations is a recurrent problem in the engineering world. The starting point is typically a discrete time model of the dynamics of the process and the measurements of the form

$$x(k+1) = f(x(k), v(k), k) \qquad y(k) = h(x(k), n(k), k) \tag{11}$$

where $x(k) \in \mathbb{R}^n$ and $y(k) \in \mathbb{R}^p$ are the state and output of the system at time $k \in \mathbb{N}$; $f\colon \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{N} \to \mathbb{R}^n$ and $h\colon \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{N} \to \mathbb{R}^p$ are (possibly nonlinear) functions; and $v(k) \in \mathbb{R}^n$ and $n(k) \in \mathbb{R}^p$ are process and measurement noise, which are generally assumed to be independent, identically distributed stochastic processes but not necessarily additive or Gaussian. The initial state is assumed independent of the noise processes and its distribution is given through a probability density function (pdf) $p(x(0))$. If the distributions of the noise processes are known, the system can be equivalently represented using two pdfs

$$x(k) \sim p_x(\cdot|x(k-1), k) \qquad y(k) \sim p_y(\cdot|x(k), k) \tag{12}$$

Here, $p_x(\cdot|x(k-1), k)$ is a conditional pdf that models the stochastic dynamics of the state of the system, determined by f and the pdf of $v(k)$. Likewise $p_y(\cdot|x(k), k)$ is a conditional pdf that models the probability distribution of the measurements, determined by h and the pdf of $n(k)$.

Given k, $k' \in \mathbb{N}$ let $\mathbb{Y}(k') = \{y(i)\}_{i=0,\ldots,k'}$ denote the sequence of measurements up to time k' and $\mathbb{X}(k) = \{x(i)\}_{i=0,\ldots,k}$ denote the sequence of states up to time k. The aim is to estimate the pdf $p(\mathbb{X}(k)|\mathbb{Y}(k'))$. This density function embodies the best estimate of the state vector up to time k given all available information up to time k'. Depending on the relation of k to k' three different types of estimation problems can be formulated: 1) filtering ($k = k'$): estimate the state trajectory up to the current state, 2) prediction ($k > k'$): estimate the state trajectory up to a future state, 3) smoothing ($k < k'$): refine the estimate of an older part of the state trajectory.

All three of these problems can be seen to be special cases of the problem of estimating the expected value

$$E[g(\mathbb{X}(k), k)] = \int_{\mathbb{R}^{(k+1)\times n}} g(\mathbb{X}(k), k) p(\mathbb{X}(k)|\mathbb{Y}(k'))\, \mathrm{d}\mathbb{X}(k) \tag{13}$$

of an arbitrary integrable function $g\colon \mathbb{R}^{(k+1)\times n} \times \mathbb{N} \to \mathbb{R}$. [For example, think of taking $g(\mathbb{X}(k), k) = \mathbb{X}(k)$ for the filtering problem.]

The filtering, prediction, and smoothing problems can be solved recursively by invoking Bayes's theorem. For example, in the case of filtering, consider the density

$$p(\mathbb{X}(k)|\mathbb{Y}(k)) = p(\mathbb{X}(k)|y(k), \mathbb{Y}(k-1)) = \frac{p(y(k)|\mathbb{X}(k), \mathbb{Y}(k-1)) p(\mathbb{X}(k)|\mathbb{Y}(k-1))}{p(y(k)|\mathbb{Y}(k-1))} = \frac{p(y(k)|\mathbb{X}(k)) p(\mathbb{X}(k)|\mathbb{Y}(k-1))}{p(y(k)|\mathbb{Y}(k-1))}$$

where the Markov property is used in the second equation. Then $p(y(k)|\mathbb{Y}(k-1))$ and $p(x(k+1)|\mathbb{Y}(k))$ can be calculated using the Chapman–Kolmogorov equation [29]:

$$p(y(k)|\mathbb{Y}(k-1)) = \int_{\mathbb{R}^{(k+1)n}} p(y(k)|\mathbb{X}(k)) p(\mathbb{X}(k)|\mathbb{Y}(k-1))\, \mathrm{d}\mathbb{X}(k) \tag{14}$$

$$p(x(k+1)|\mathbb{Y}(k)) = \int_{\mathbb{R}^{(k+1)n}} p(x(k+1)|\mathbb{X}(k)) p(\mathbb{X}(k)|\mathbb{Y}(k))\, \mathrm{d}\mathbb{X}(k) \tag{15}$$

Now a recursive filter can be constructed, consisting of two steps, an update step [when a new measurement arrives the estimation of the state is updated, (14)] and a prediction step [predict the state one step in the future, (15)].

B. Particle Filters

Equations (14) and (15) represent the theoretically optimal Bayesian estimate. This conceptual solution is deceptively simple, however, since the integrals involved are seldom tractable. Two notable exceptions are the Kalman filter [30] for linear systems with additive Gaussian noise and the grid-based filter [31] for systems with a finite state space. In the general case the pdf of interest has to be approximated numerically. Particle filters (or sequential Monte Carlo methods [32]) are fast estimation techniques that perform this numerical approximation using simulation. The main idea is to approximate the continuous probability distribution of interest using a discrete distribution comprising weighted samples (known as particles, Fig. 5). To do this N independent identically distributed particles, $\mathbb{X}^1(k), \ldots, \mathbb{X}^N(k)$ are extracted from $p(\mathbb{X}(k)|\mathbb{Y}(k))$, and an empirical estimate of the distribution is constructed

$$\hat{p}(\mathbb{X}(k)|\mathbb{Y}(k)) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\mathbb{X}^i(k)}(\mathbb{X}(k)) \tag{16}$$

where $\delta_{\mathbb{X}^i(k)}$ denotes the Dirac mass at particle $\mathbb{X}^i(k)$. Then the expectation of any integrable function g can be estimated by

$$E[g(\mathbb{X}(k), k)] \approx \int g(\mathbb{X}(k), k) \hat{p}(\mathrm{d}\mathbb{X}(k)|\mathbb{Y}(k)) = \frac{1}{N}\sum_{i=1}^{N} g(\mathbb{X}^i(k), k) \tag{17}$$

It can be shown that this estimator is unbiased and (under weak assumptions) converges to the true expectation as the number of

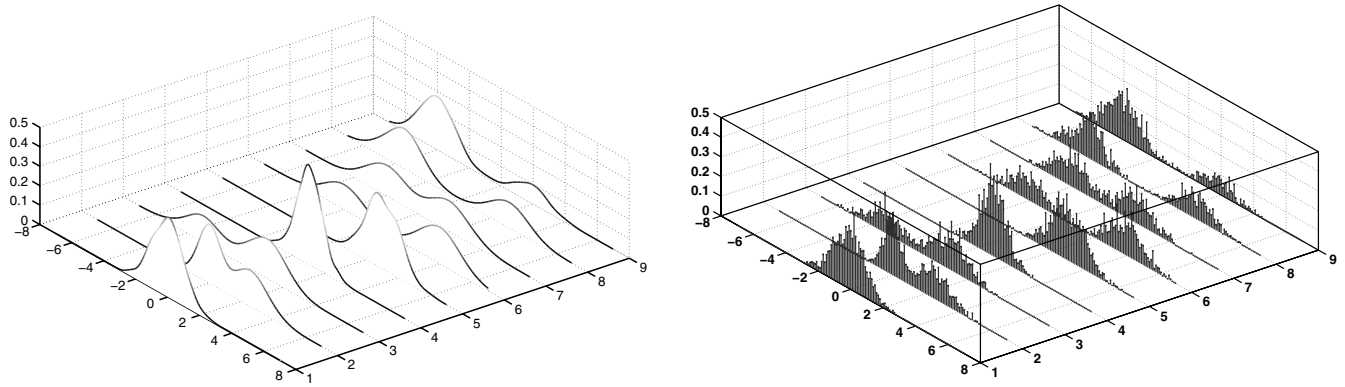


Fig. 5 Example of true continuous probability density function (left) and its particle approximation (right). The location and weight of the particles reflect the value of the probability density in that region of the state space.

particles N tends to infinity [14]. In cases where it is not possible to sample from $p(\mathbb{X}(k)|\mathbb{Y}(k))$ directly, because the analytical expression cannot be evaluated, a technique known as importance sampling [33] can be employed. An algorithm to perform these operations recursively is sequential importance sampling (SIS).

While it can be shown that as N tends to infinity this algorithm leads to correct estimation, for a finite number of particles SIS easily leads to degeneracy. This is a situation where all the particles have negligible weight, except for a single one; indeed it can be shown that in some cases as k increases the SIS algorithm is guaranteed to fail in this way [34]. A standard method for avoiding this problem is to introduce an additional resampling step. The idea is to stop propagating particles that have low importance weights and multiply particles with high importance weights, thus focusing computational resources in areas where the pdf is higher. This leads to the Bayesian bootstrap [35] or sequential importance resampling (SIR) algorithm.

A potential problem of the SIR algorithm is loss of diversity as particles with large weights are selected more and more often in step 3, eventually eliminating all other particles. The loss of diversity diminishes the exploration capabilities of the algorithm. Experiments suggest that even though this algorithm works satisfactorily for trajectory prediction for one or two aircraft, it fails to converge for more than two aircraft. Kernel density methods [36], which artificially add diversity to the particles at the resampling step work somewhat better but not substantially. Experiments suggest that they cannot handle more than four aircraft simultaneously.

C. Trajectory Prediction and Filtering

The trajectory prediction problem is reformulated so that its relation to nonlinear filtering becomes more apparent. To keep the notation simple from now on the nominal (forecast) wind is assumed zero and only the stochastic error in the wind forecast is treated here; extension to nonzero wind forecast is straightforward.

Assume that at a discrete time instant $k\delta_t$ there are $M(k)$ aircraft flying in a region of the airspace; this number can of course change over time as aircraft enter and exit the region of interest. Let $z(t,l) = (X(t,l), Y(t,l), \psi(t,l), m(t,l)) \in \mathbb{R}^4$ denote the state of aircraft $l = 1,\ldots,M(k)$ at time $t \in [k\delta_t, (k+1)\delta_t]$ and $W_X(k)$ and $W_Y(k)$ denote the wind vectors. Let

$$x(k) = \begin{bmatrix} z(k\delta_t, 1) \\ \vdots \\ z(k\delta_t, M(k)) \\ W_X(k) \\ W_Y(k) \end{bmatrix} \in \mathbb{R}^{4M(k)+2N_XN_YN_Z}$$

denote the overall state of the system [cf. Eq. (11)].

Using the information contained in $x(k)$ one can compute the state of each aircraft $l = 1,\ldots,M(k)$ at the next discrete time instant $(k+1)\delta_t$ by solving the differential equation

$$\dot{z}(t,l) = \begin{bmatrix} V(l)\cos(\psi(t,l)) + w_X(t,l) \\ V(l)\sin(\psi(t,l)) + w_Y(t,l) \\ \dfrac{C_L(l)S(l)\rho(Z(l))V(l)\sin(\phi(t,l))}{2m(t,l)} \\ -\eta(l)T(t,l) \end{bmatrix} \tag{18}$$

for $t \in [k\delta_t, (k+1)\delta_t]$. The values of the control inputs $T(t,l)$ and $\phi(t,l)$ are computed using the FMS model. Note that the feedback controller encoded in the FMS is time-varying, since it depends on the flight plan; in other words, the same $z(t,l)$ will lead to different $T(t,l)$ and $\phi(t,l)$ depending on how far along the flight plan aircraft are. The values of the wind $w_X(t,l)$ and $w_Y(t,l)$ are computed by linear interpolation of $W_X(k)$ and $W_Y(k)$ at the current position $X(t,l), Y(t,l), Z(l)$ of aircraft l. This implies a linear dependence of $w_X(t,l)$ and $w_Y(t,l)$ on $W_X(k)$ and $W_Y(k)$ of the form

$$\begin{aligned} w_X(t,l) &= A(X(t,l), Y(t,l), Z(l))W_X(k) \\ w_Y(t,l) &= A(X(t,l), Y(t,l), Z(l))W_Y(k) \end{aligned} \tag{19}$$

where the interpolating function $A(X(t,l), Y(t,l), Z(l))$ can be computed explicitly. To capture the approximation introduced by the simulation, independent Gaussian random variables with zero mean and standard deviation 5 m are added to the resulting $X((k+1)\delta_t, l)$ and $Y((k+1)\delta_t, l)$; let $v(k,l) \in \mathbb{R}^4$ denote the vector comprising these two random variables followed by two 0s.

The value of the state $x(k+1)$ is completed by updating $W_X(k)$ and $W_Y(k)$ according to Eq. (8). The overall result is a new value

$$x(k+1) = f(x(k), v(k), k)$$

where the process noise vector $v(k)$ comprises the aircraft state evolution errors $v(k,l) \in \mathbb{R}^4$, $l = 1,\ldots,M(k)$ and the wind innovation vectors $v_X(k) \in \mathbb{R}^{N_xN_YN_z}$ and $v_Y(k) \in \mathbb{R}^{N_xN_YN_z}$; note again the dependence of f on the time k entering through the flight plan. This defines a function f needed in Eq. (11), implicitly (through simulation) for the aircraft states and explicitly [through Eq. (8)] for the wind states.

At time instant $k\delta t$ the positions $y(k,l) = (X(k\delta t, l), Y(k\delta t, l)) \in \mathbb{R}^2$ of all aircraft $l = 1,\ldots,M(k)$ are measured through the radar; the altitude of each aircraft $Z(l)$ is assumed constant and known. This gives rise to a measurement vector

$$y(k) = \begin{bmatrix} y(k,1) \\ \vdots \\ y(k,M(k)) \end{bmatrix} = h(x(k)) + n(k) \in \mathbb{R}^{2M(k)}$$

The function h [cf. Eq. (11)] is linear, a projection to the X and Y coordinates of the aircraft positions. The vector $n(k)$ models the measurement noise and comprises the independent Gaussian random variables with zero mean and standard deviation 80 m added to the radar measurements of each aircraft.

Trajectory prediction using this model is executed every time new radar measurements are taken. Roughly speaking, the algorithm proceeds in two steps. The prediction step is straightforward both conceptually and computationally. The main difficulty lies in the filtering step, since the dimension of the state and the nonlinear nature of the dynamics make it impossible to apply conventional filtering techniques. It is shown how the structure of the problem can be exploited to develop specialized filtering algorithms to solve the problem at a realistic scale.

D. Sequential Conditional Particle Filter

Note that $x(k)$ comprises two classes of state variables: the aircraft states $z(k\delta t, l)$ and the wind states $W_X(k)$ and $W_Y(k)$. From the preceding discussion it is apparent that

1) The aircraft states evolve according to nonlinear dynamics, while the wind states evolve according to linear dynamics.

2) The evolution of the wind states is independent of the evolution of the aircraft states.

3) The evolution of the states of different aircraft [$z(k\delta_t, l)$ and $z(k\delta_t, l')$ for $l \neq l'$] are only coupled to each other through the wind states.

The first two observations imply that the wind states should be easier to estimate, since under Gaussianity assumptions storing and manipulating them only requires keeping track of their mean and covariance matrix. Moreover, given a probabilistic estimation of the wind at some points in the wind field, the conditional distribution of the wind at all other points can be explicitly derived. This distribution will also be Gaussian, hence easy to store and manipulate. The first novelty of the proposed algorithm is that, instead of using realizations for the wind states carried by the particles (as in conventional particle filtering) the entire conditional probability distribution is stored and manipulated.

The latter two observations imply that, conditional on the wind states, the states of different aircraft are independent of each other. This is exploited by the second novelty of the algorithm, which is the sequential incorporation of the information from different aircraft. Every radar measurement contains information about the positions, $y(k, l)$, of all aircraft in a region of the airspace, $l = 1, \ldots, M(k)$. New measurements are processed one aircraft at the time. The measurement $y(k, 1)$ of the position of aircraft 1 is used first to filter all the particles $x^i(k)$, updating the estimates of the states of aircraft 1 $z^i(k, 1)$ and the distributions of the wind states $W_X(k)$ and $W_Y(k)$. The updated $z^i(k, 1)$ is stored until the $k + 1$ measurement, and the distributions $W_X(k)$ and $W_Y(k)$ are filtered using the measurement $y(k, 2)$ of the position of aircraft 2, then aircraft 3, etc. Experiments suggest that the order in which the aircraft measurements are used is not particularly important. For consistency the order in which the aircraft entered the region of interest is used. The complete algorithm is reported below. A related idea was proposed in [15] for static instead of dynamic models.

These two modifications provide a very substantial improvement in the performance of the filtering algorithm. Conventional particle filtering methods would normally carry realizations of the wind states as part of the particles and would filter these (together with the states of the aircraft) using the measurements from all aircraft simultaneously. In theory this should not be a problem, provided the number of particles is high enough. In practice, due to computational limitations the number of particles is limited to a few thousands. Because of the large dimension of the wind states (which can be several hundreds or thousands) it is unlikely that any one of the finite number of wind realizations carried by conventional particles would match well the wind in all the locations. Therefore, as the aircraft explore the wind field the importance weights tend to shift to just a few particles (those that most closely match the wind field) leading to degeneracy. This problem is especially acute in the presence of many aircraft. Experiments suggest that none of the conventional particle filtering methods can handle more than four aircraft. Even with 10,000 particles degeneracy sets in only after two to three measurements. By contrast, the method proposed here appears to be stable with hundreds of aircraft and with 1000 (or even as few as 50) particles.

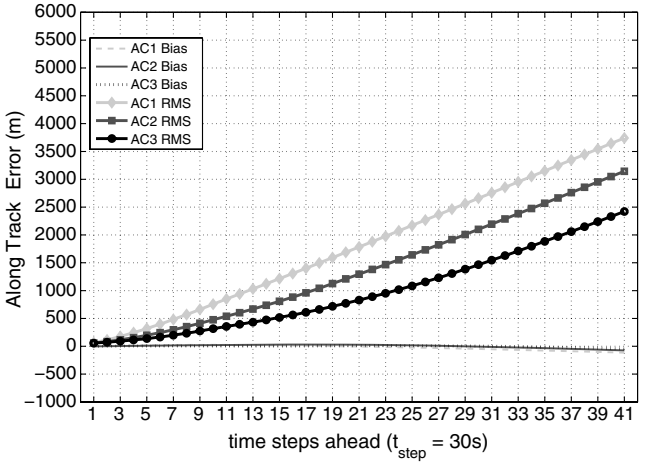


Fig. 6 Comparison between the bias and RMSE in TP for a three-aircraft scenario.

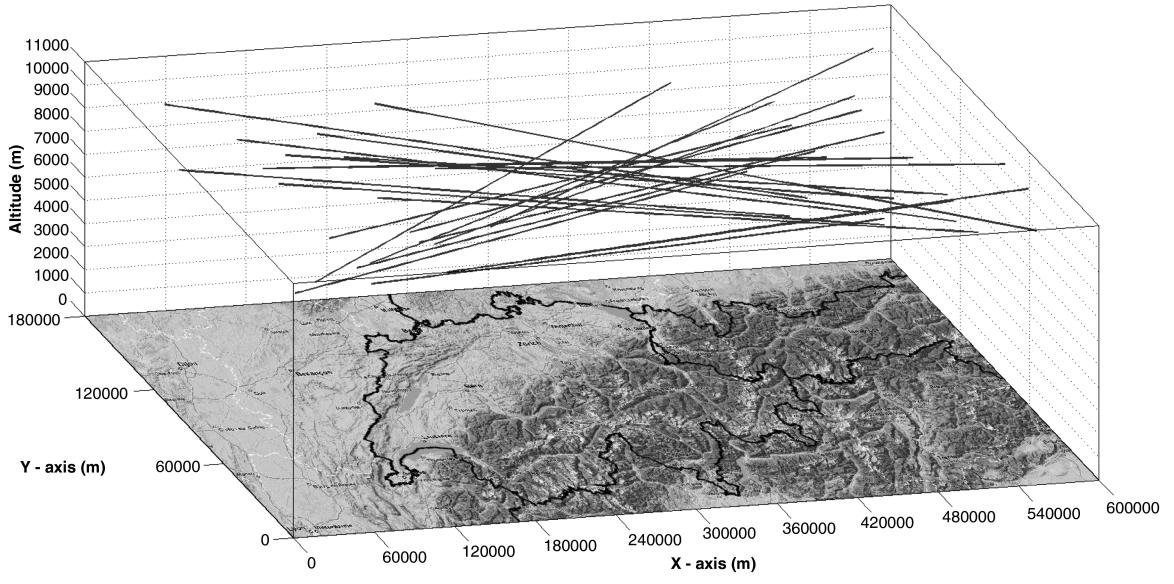


Fig. 7 Flight plans for the 24-aircraft scenario overlayed over a map of Switzerland.

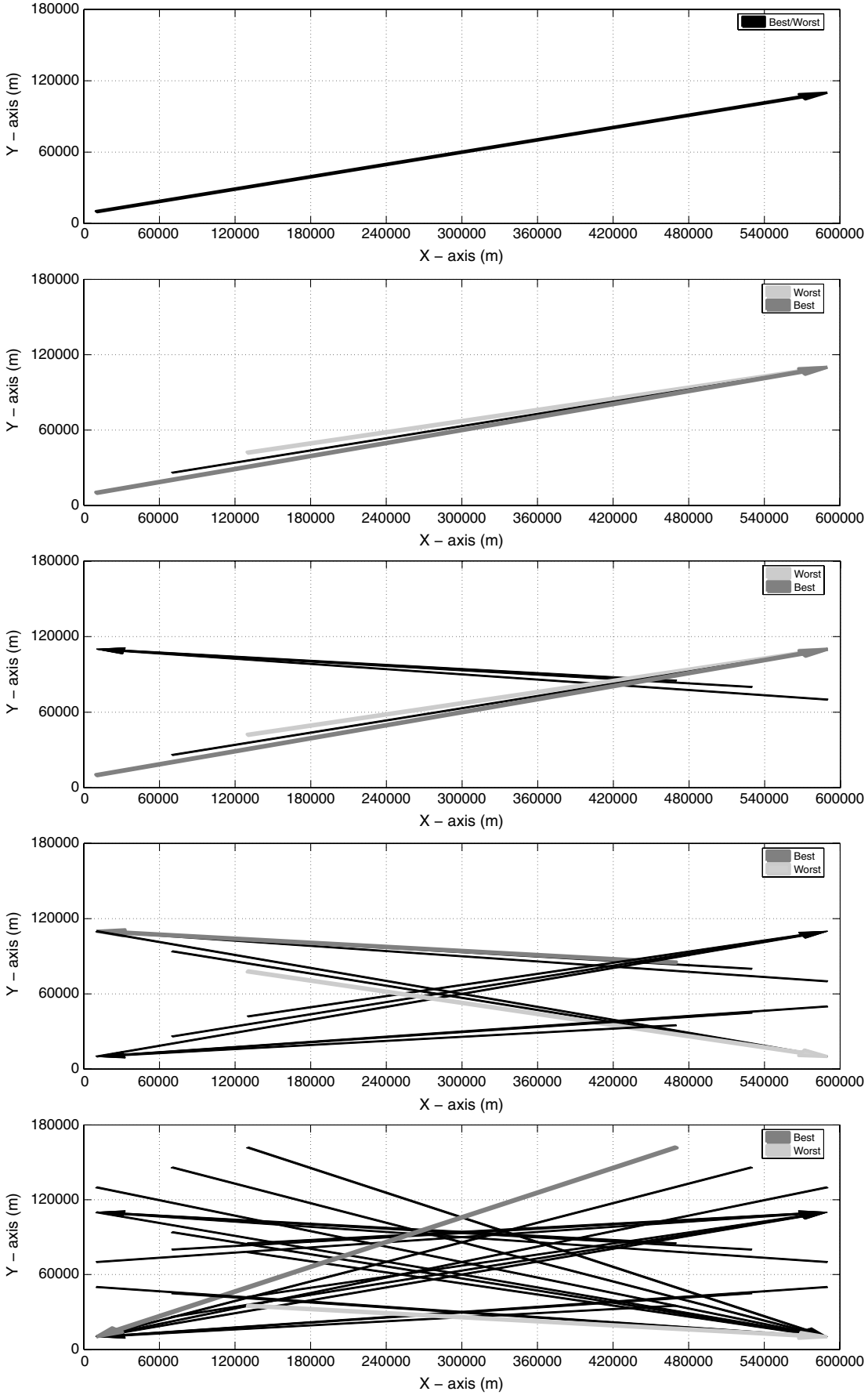


Fig. 8 Flight plans for 1, 3, 6, 12, and 24 aircraft. Arrows display the direction of motion. Gray and dark gray display the flight with the worst and best performance, respectively, in terms of uncertainty reduction.

V. Simulation Results

A. Benchmark Simulation Setup

To evaluate the limitations of the algorithm, a series of artificial flight plan scenarios was devised, with a flight plans of the 24-aircraft scenario were superimposed over a map of Switzerland in different number of aircraft (1, 3, 6, 12, and 24) in a sector of interest. To give an idea of the scale of the setup, the flight plans of the 24-aircraft scenario were superimposed over a map of Switzerland in Fig. 7. The flight plans for each scenario can be seen in Fig. 8; flight plan sets with more aircraft are supersets of those with fewer aircraft. The sector of interest is a square with dimensions 600×600 km. However, all flights are restricted in an area approximately 180×600 km. This is done to be able to assess the reduction in wind uncertainty even in areas where no aircraft have flown recently. All the aircraft fly level at 10,000 m with a nominal airspeed of 215.6 m/s, and there are no turns included in the flight plans. The parameters of the dynamical models for all aircraft represent a Boeing 737-200. The initial mass is considered known and is set to 46000 kg. Flights have a duration of approximately 30 m and radar measurements arrive to the ATC every 30 s.

As discussed earlier, the weather forecast component of the wind is set to 0 throughout the region of interest for simplicity. The wind-forecast error is evaluated in both the south–north and the west–east directions on a grid with 11×11 vertices (i.e., the planar surface is split into cells of 60×60 km). Note that for the largest scenario (24 aircraft) the dimension of the nonlinear state is $4 \times 24 = 96$ states, whereas the dimension of the linear part of the state is $2 \times 11 \times 11 = 242$ states.

To examine the uncertainty in midterm TP the along-track error is used as a metric. Cross-track error was also evaluated but was found to be considerably smaller, as one would expect due to the cross-track corrections introduced by the 3-D FMS model. The improvement in TP is therefore small in the cross-track direction as opposed to the along-track directions; the cross-track results are omitted in the interest of space. For a thorough overview of the derivation of the along-track and cross-track error metrics the reader is referred to [37,38].

Unless otherwise stated, the results presented below are for TP based on filtering 10 min of radar measurements (20 measurements total). The SCPF algorithm is running for this period, attempting to filter the aircraft and wind state. In all simulations 1000 particles are employed, independent of the number of aircraft. Having used the SCPF algorithm to filter 10 min of data the state estimate of all particles is extrapolated for a further 20 min (40 time steps) into the future to get an estimate of the future trajectory for each aircraft. The extrapolation is done by simulating all the particles for the required time. The mean between these particles is considered as a TP estimate ($\hat{\theta}$) for each flight. The real trajectory (θ) is then compared with this estimate to compute the along-track error metric.

To collect sufficient statistics, each scenario was simulated $\Lambda = 1000$ times, keeping the same flight plans, but different wind fields, producing different trajectory estimates ($\hat{\theta}_j$) and real trajectories (θ_j) for $j = 1, \ldots, \Lambda$. The estimator bias is then calculated as

$$\text{bias} = \frac{1}{\Lambda}\sum_{j=1}^{\Lambda}(\hat{\theta}_j - \theta_j)$$

while the root mean square error (RMSE) is calculated as

$$\text{RMSE} = \sqrt{\frac{1}{\Lambda}\sum_{j=1}^{\Lambda}((\hat{\theta}_j - \theta_j))^2}$$

Experiments suggest that the bias is generally very small compared with the RMSE (Fig. 6). Therefore, report only the RMSE statistics are reported below.

For comparison purposes in the figures two additional statistics are also reported. The first one (referred to as "agnostic" below) describes the growth of the TP error when no filtering is carried out. This is generally the case in current practice. In this case, the aircraft fly in an uncertain wind environment (except for the forecast provided to the ATC) and no attempt is made to estimate the wind-forecast errors from the radar measurements. The agnostic statistic constitutes a worst-case bound on the algorithm performance, at the very least the algorithm should improve on the current situation. The second statistic (referred to as "all-known" below) describes the growth of the TP error given perfect information about the wind field all over the airspace at the current point in time. The TP uncertainty in this case is only due to the time evolution of the wind error, which is unpredictable given the current state. This is clearly an unrealistic, perfect filtering situation and constitutes an optimal best-case performance bound for the algorithm.

B. Trajectory Prediction Results

The first scenario that is examined, involves a single aircraft that is cruising at constant altitude, with constant speed, across the airspace. In this case, the algorithm is not fully exercised, since there is no sequential incorporation of estimates between different aircraft. The results of the simulations for the one-aircraft scenario are presented in Fig. 9. The agnostic and all-known performance bounds are also shown in the figure (approximately 5000 and 1400 m, respectively, for a 20 min TP). The figure also includes a third statistic (referred to as "local-known" below) which also represents an ideal filter that perfectly estimates the position of the aircraft and uses ideal radar measurements (with no error) to estimate the wind. This is the best one can hope for with a single aircraft, since measurements are only available in the aircraft location and nowhere else in the airspace. SCPF is closely tracking the ideal local-known filter with an RMSE approximately at 3780 m after 20 min of prediction. The deterioration in performance is only marginal. This clearly suggests that the filtering algorithm has successfully managed to extract all the available information from the data. It is also interesting to note that the growth of the along-track error is almost linear in time, as expected, due to the accumulation of wind errors.

The algorithm is exercised fully in the three-aircraft scenario, as can be observed in Fig. 10a. It is interesting to note that now there

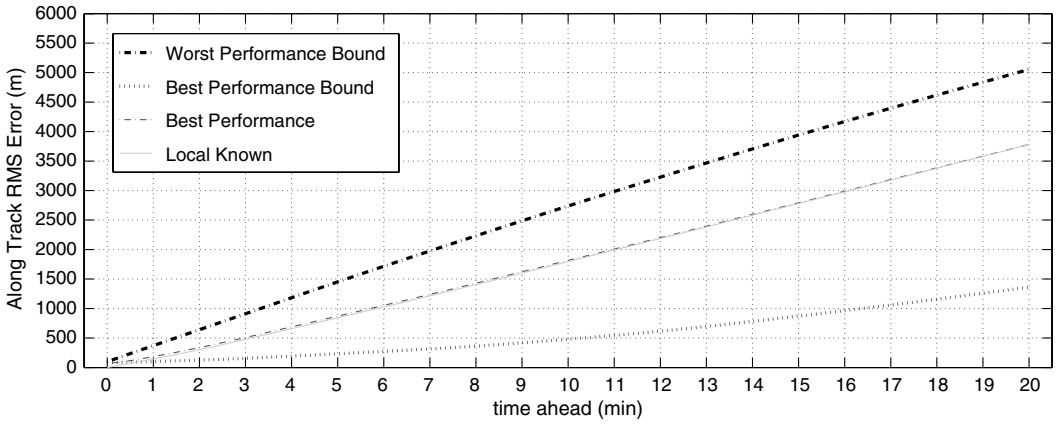
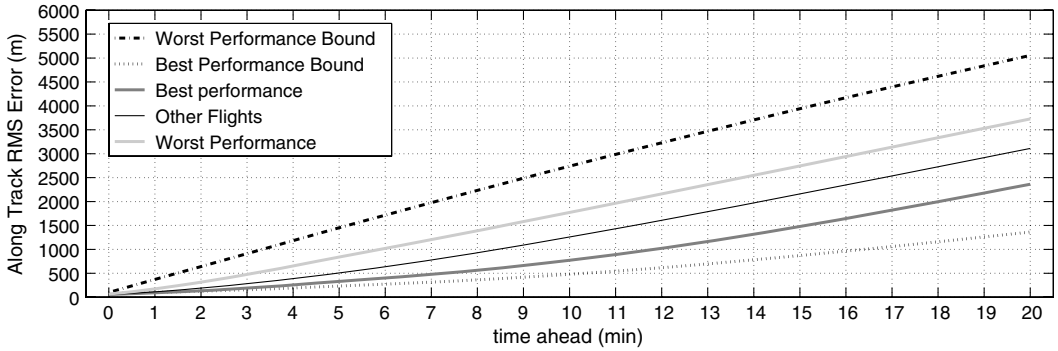
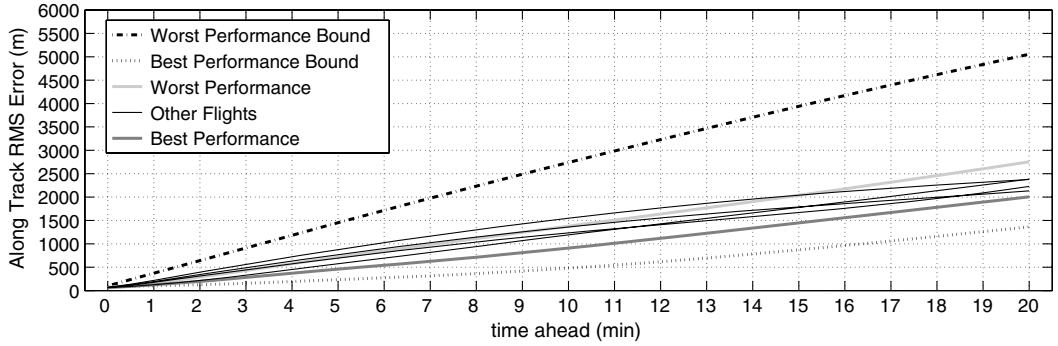


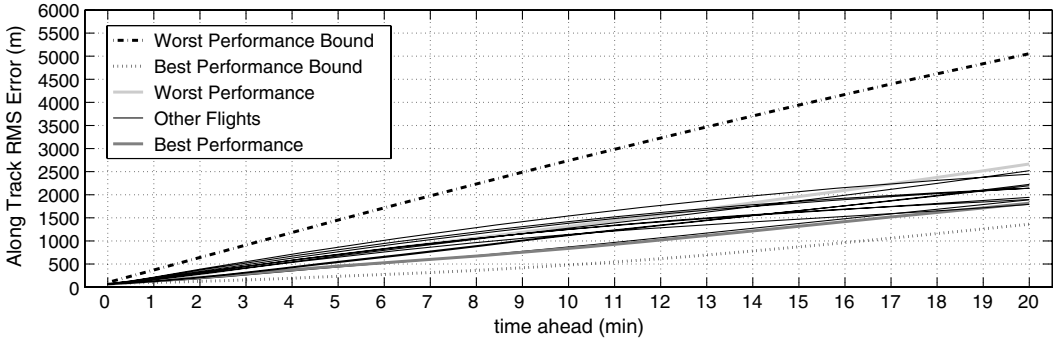
Fig. 9 Evolution of the along-track RMSE, for one aircraft. Comparison of SCPF performance with a perfect local filter (local known).



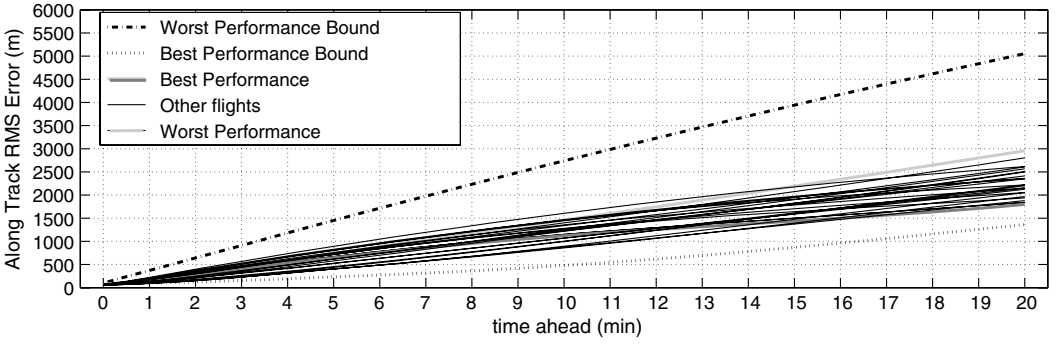
a) Flight scenario with 3 aircraft



b) Flight scenario with 6 aircraft



c) Flight scenario with 12 aircraft



d) Flight scenario with 24 aircraft

Fig. 10 Evolution of the RMS along-track error, for different flight scenarios, after performing filtering for 10 min of radar measurements.

exist performances that are better than the ideal local-known filter. This is expected, since in this case, measurements for the position of three rather than one aircraft are acquired. Those measurements, even if they are inaccurate, provide an information advantage over the single-aircraft case. Observing the flight plans of the three-aircraft case (Fig. 8) it is easy to note the advantage for the two trailing aircraft. The leading aircraft has no significant information for the wind error it is going to encounter except through the spatiotemporal correlation of the wind in its own position. This is the reason that the along-track error for this aircraft is similar to that of the one-aircraft scenario. The two trailing aircraft, on the other hand, take advantage of the radar measurements of the leading aircraft. The effect is stronger for the last aircraft which is favored by measurements from both other aircraft.

The flight plans of the six-aircraft scenario present slightly more favorable conditions for some of the aircraft involved. In this scenario another formation, consisting of three aircraft, moves with almost an opposite direction to the first three. Results are clearly better for the

worst and mean cases, since even the leading aircraft have now some informational advantage from each other. Along-track error growth for the six-aircraft scenario is presented in Fig. 10b.

When more aircraft enter the airspace, performance keeps improving in the along-track error growth, as can been seen in Figs. 10c and 10d. Both the maximum and minimum along-track error show a substantial reduction. There even exist some flights that come close enough to the ideal all-known performance bound. Those aircraft have favorable flight plans that allow them to take advantage from the measurements of most of the other flights. The slight increase of the maximum along-track error in the 24-aircraft scenario (compared with the 12-aircraft scenario) is due to the unfavorable conditions for one of the new flights. Even flights with unfavorable flight plans, however, have a significant reduction in uncertainty compared with the no-filtering and one-aircraft cases. It is important to note that after the 12-aircraft case there exists a saturation concerning the performance of the algorithm when it comes to the best-performing aircraft, even when doubling the number of flights. This is not a surprise, since the best performance is already close to the ideal all-known case.

Figures 11 and 12 collect the results from all different flight plan scenarios. The performance of the algorithm is examined after the incorporation of 5 and 10 min of radar measurements. The figure depicts the RMSE in the predicted along-track position of the aircraft 20 min into the future. It is clear that there exists a strong improving trend when multiple aircraft are involved. For the 12- and 24-aircraft scenarios, the mean TP along-track error for 20 min of prediction (after 5 min of filtering, Fig. 11), drops to just 2629 and 2506 m, compared with 3784 m for the single-aircraft case. When SCPF receives radar measurements for another 5 min, Fig. 12, the mean TP error drops to less than 2200 m. Considering the minimum along-track error, results are as low as 1799 m for the 24-aircraft case and 1792 m for the 12-aircraft case. This is a substantial reduction, since the ideal all-known RMSE bound for 20 min of prediction is approximately 1400 m. If multiple-aircraft fusion was not employed a significant number of such flights would experience as much as double the along-track TP uncertainty achieved here.

Figure 13 shows an estimate of a part of the covariance matrix of the wind-forecast error distribution for the 24-aircraft scenario after the incorporation of 20 min of measurements. It is apparent that where most aircraft fly uncertainty is very low. However, even in parts of the airspace where there are no flights, reduction in wind uncertainty can be significant.

An important feature of the multiple-aircraft fusion is that even the worst cases perform at least marginally (in some scenarios significantly) better than the single-aircraft case. If some flight, in a multi-aircraft scenario with SCPF, had an inferior performance compared with the single-aircraft case (or even worse the no-filtering case) it would indicate a significant drawback of the algorithm. It would actually suggest that the TP performance of some flights deteriorates in accuracy in order for some others to gain an information advantage, an unfavorable trade-off. Happily, this does not appear to be the case with the proposed algorithm.

C. Algorithm Analysis

In addition to the improvement in TP accuracy discussed earlier, the two novelties of the SCPF algorithm also provide a substantial improvement in numerical robustness. For all scenarios simulated and for all repetitions there was never a single case of particle

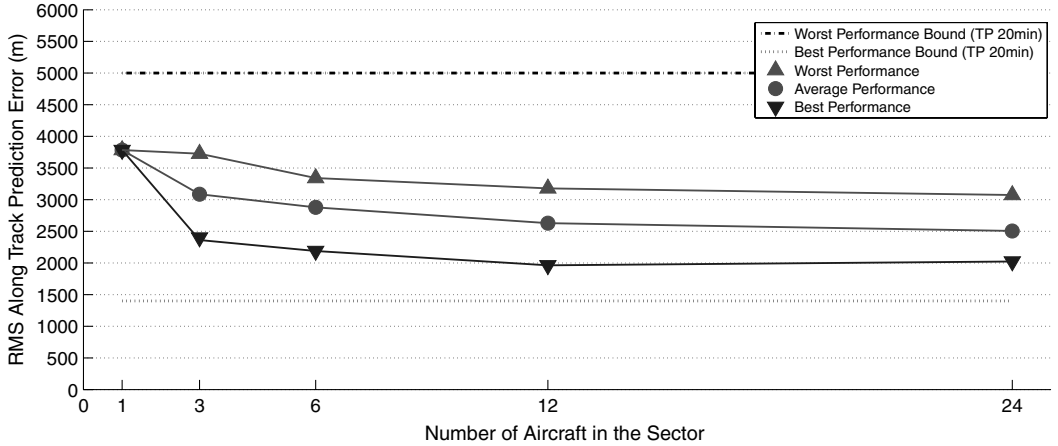


Fig. 11 Along-track error after 20 min of prediction for different numbers of aircraft in the region of interest, after the incorporation of 5 min of radar measurements.

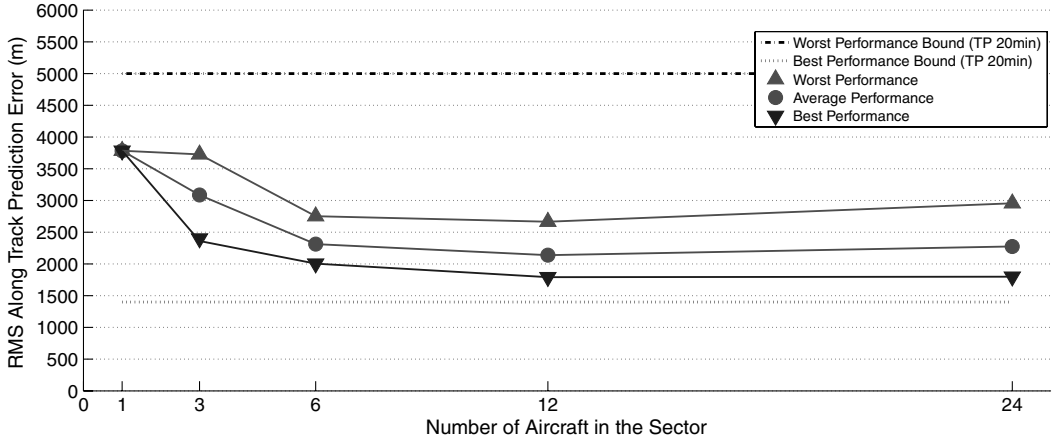


Fig. 12 Along-track error after 20 min of prediction for different number of aircraft in the sector, after the incorporation of 10 min of radar measurements.

collapse due to insufficient number of particles and extreme degeneracy.

To quantify this observation a common indicator of the performance of a PF algorithm known as the effective sample size N_{eff} is computed. N_{eff} is a measure of the degeneracy of the algorithm and is usually computed by

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} (\tilde{q}^i)^2} \tag{20}$$

(recall that $\tilde{q}^i$) is the normalized weight of particle i). N_{eff} will be equal to the number of particles, N, when all particles share the same weight and equal to 1, when all the particles have zero weight except a single one. The former is the ideal case, the latter indicates extreme degeneracy.

In the case of SCPF for each measurement the normalized weights are computed repeatedly, as the radar measurements of the different aircraft are sequentially conditioned. The average of the N_{eff} for the different aircraft for the 24-aircraft case scenario is displayed in Fig. 14, as a percentage of the 1000 particles used in the simulation. Typically in PF algorithms N_{eff} decays as time evolves. SCPF, however, manages to keep N_{eff} at a roughly constant level. This is an important advantage of the algorithm that contributes to its robustness and reliability.

The computational resources devoted to the execution of the algorithm were 1000 particles. The computational effort grows linearly with the number of aircraft involved in each scenario (the number of particles remains the same but the dimension of the state becomes higher). Even though generating the data reported in the figures required several days of computation time, one needs to keep in mind that each experiment was repeated 1000 times to collect statistics. A single run of the algorithm for a scenario including 12 different aircraft flying for 10 min (and filtering performed every 30 s) required on average 40 min of computational time. The same flights projected 20 min into the future require 30 min of computational time. It should be noted that the algorithm was implemented in MATLAB and not a high-performance language, while it was also not optimized for speed (for example, accelerating large matrix operations). Moreover, the algorithm is easy to parallelize, since most of the computation for the evolution of the particles can be done separately for each particle; only the computation of the normalized weights and resampling requires any communication among particles.

D. Maastricht Airspace

After examining the performance of the algorithm with artificial flight plans, an initial assessment for a more realistic setup was carried out. As a region of interest a 3-D box centered in Brussels National Airport (IATA: BRU) airport was used, with dimensions 600×600 km in the horizontal plane and 33000 to 36000 feet in altitude. Real flight plans from central flow management unit (CFMU) were used, more specifically all the flights that cross this sector throughout a single day and fly level for this part of their course. Figure 15 depicts the flight plans from a top down and side view. The total number of flights considered was 441, but aircraft were constantly entering and exiting the region of interest. The peak value of aircraft inside the 3-D box was 330 and the algorithm was operating with at least 150 aircraft for half of the simulation time. At the start and the end of the day, however, a single aircraft was left inside the sector.

This scenario puts a considerable stress on SCPF, since the dimension of the state increased dramatically. The nonlinear part of the state (aircraft dynamics), topped at 1320 states, but was varying depending on the number of aircraft in the sector. The linear part (wind-error dynamics), included the calculation of the wind-error

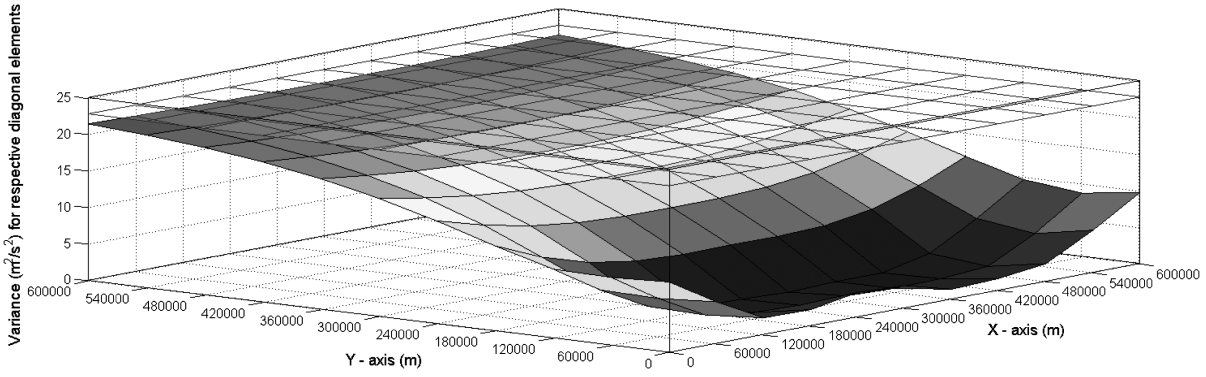


Fig. 13 Diagonal elements of the covariance matrix of a particle after 20 radar measurements for 24 aircraft. Elements are positioned on their respective place in the horizontal plane. The transparent flat surface displays the initial uncertainty, before any measurements are incorporated.

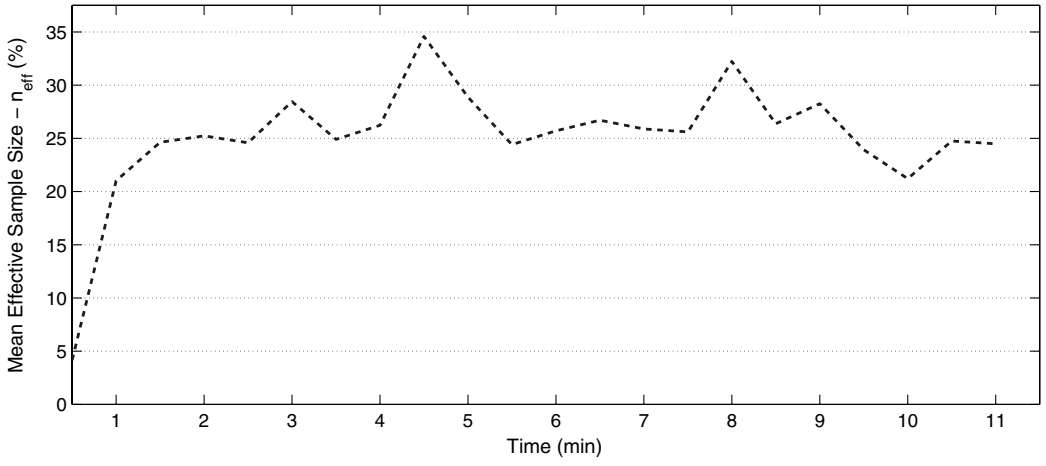


Fig. 14 Evolution of the mean N_{eff} as new measurements arrive.

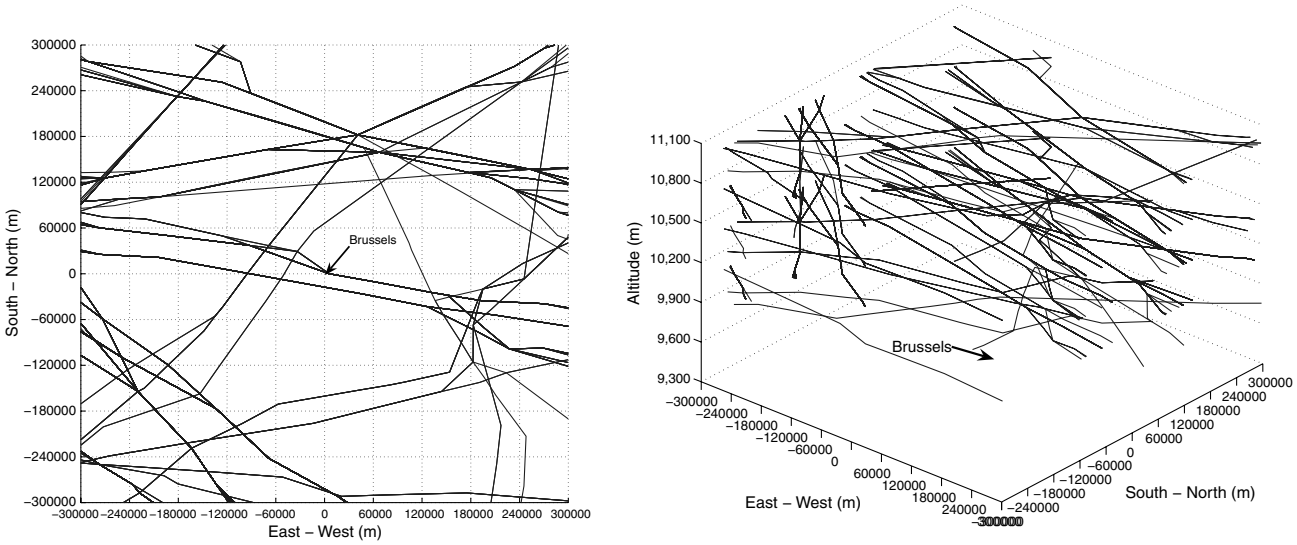


Fig. 15 Flight Plans centered at Brussels.

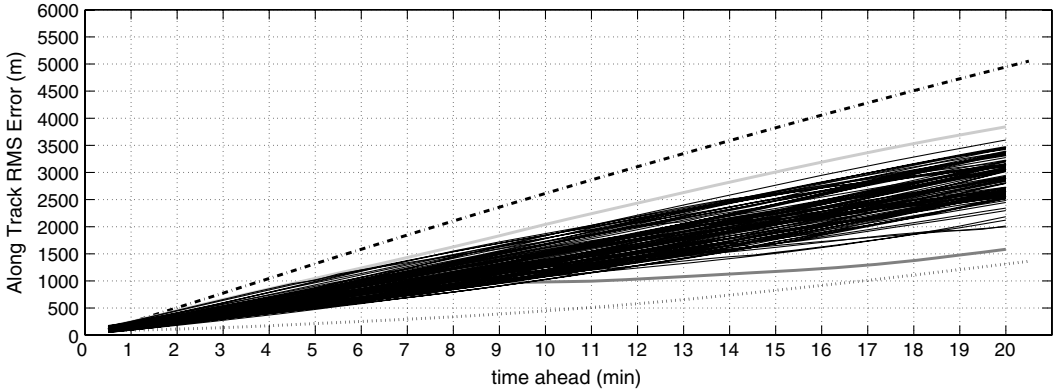


Fig. 16 Evolution of the RMS along-track error for flights in Brussels 3-D sector.

vector in the X and Y axes, in $11 \times 11 \times 6$ vertices, a total of 1452 states. The flight plans included turns and were positioned at different flight levels. Because computational limitations just 50 particles were used throughout the simulation.

The performance of SCPF for TP can be seen in Fig. 16 and bears many similarities with the 12- and 24-aircraft scenarios above. The average TP error over all aircraft is similar, but the best and worst aircraft performances are more extreme, since there exist so many different flights and possibly complex situations. The flight with the worst performance is still below the threshold of the agnostic- and single-aircraft cases. The flight with the best performance is considerably closer to the ideal all-known case.

Figure 17 is the equivalent of Fig. 13 for this case and shows an estimate of a part of the covariance matrix of the wind-forecast error distribution for one of the flight levels of the Maastricht scenario. Like for the 24-aircraft case, in the region of the airspace where most aircraft fly uncertainty is very low. In this case specifically, where there exist significantly more aircraft, wind uncertainty is dramatically reduced. This estimate is only after 20 min of flight, so most of the aircraft have not yet entered the sector or the aircraft that

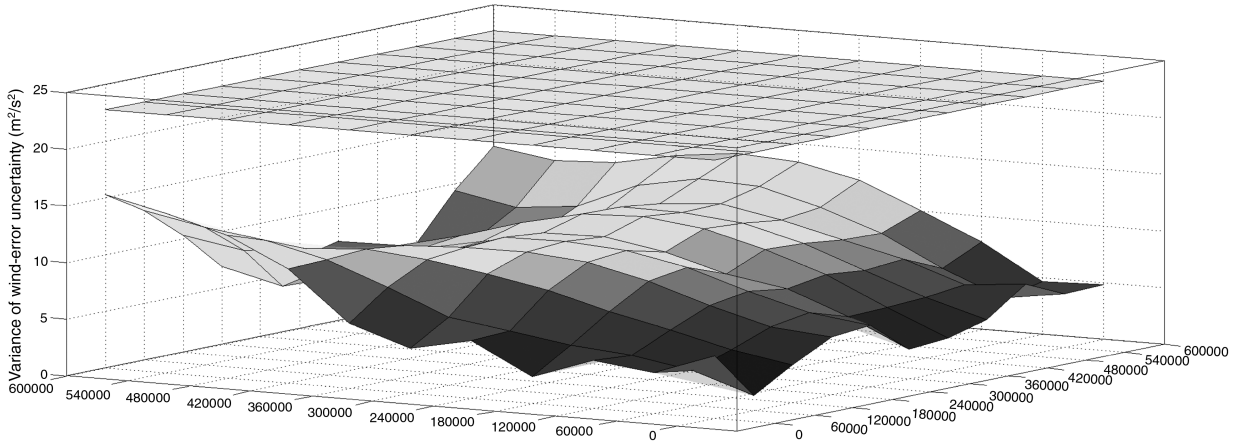


Fig. 17 Diagonal elements of the covariance matrix of a particle after 40 radar measurements for Brussels airspace. Elements are positioned on their respective place in the horizontal plane. The transparent surface displays the initial uncertainty, before any measurements are incorporated.

have still have not fully traversed their flight paths. The incorporation of data from more flights is expected to reduce the uncertainty even more.

Despite the complexity of the simulation and the computational limitations, this initial assessment suggests that SCPF has a viable potential for both realistic and demanding situations. Including flights at the multiple flight levels case offers some advantage, because information about the wind can be passed from aircraft flying at different altitudes. However, is has to be noted that wind-forecast error correlation in the vertical direction decays rather quickly. Moreover, the flight plans included multiple turns, which implicitly provided advantageous information to the algorithm because the dynamic behavior of the aircraft exhibits a strong dependence on wind while turning. Finally, despite the low number of particles, the algorithm never collapsed due to degeneracy or the high-dimensionality of the problem.

VI. Conclusions

A novel sequential Monte Carlo algorithm was developed, the sequential conditional particle filter to improve trajectory prediction (TP) performance in a multi-aircraft case from the perspective of a ground based air traffic management (ATM) system. Simulation results suggest that this methodology is viable in realistic situations, since it is robust and can offer a substantial improvement in the accuracy of TP.

Wind-forecast error, though important, is not the only uncertainty involved in TP. Missing parameters, partially known intent and airline settings contribute significantly in TP errors. The next step of testing the algorithm is to include climb and descend phases in flights and uncertainty about the mass of the aircraft and its nominal airspeed. This is an especially demanding and intriguing case. Because of the inclusion of the vertical direction, both the dimension of the nonlinear aircraft state and the dimension of the linear wind-field state are going to increase (reaching thousands of states in total) rendering filtering even more challenging. Furthermore, the wind-forecast error is weakly correlated across different altitudes, so the information advantage might not be similarly obvious from one flight level to the other.

In the same direction of improving TP by fusing measurements from multiple aircraft, work toward algorithms that incorporate direct wind measurements from the aircraft is performed, assuming that these can be datalinked to the ground. Initial results suggest that this additional source of information can significantly boost the performance of the algorithm, since direct measurements of the linear part of the state become available (albeit through a time-varying measurement matrix). However, a realistic approach would require advanced filters, since onboard wind measurements are not very accurate, while neither linear nor Gaussian. The combination of positional and wind-speed measurements provides a very large number of data that have to be treated properly to extract all the useful information despite the high complexity of the problem.

Appendix

W below represents either W_X or W_Y, because wind error has been already assumed an isotropic field.

Proposition 1: Let

$$W(0) = \hat{Q}n(0) \qquad W(k+1) = aW(k) + Qn(k) \tag{A1}$$

where $n(k)$ are independent standard (zero mean, identity covariance matrix) Gaussian random variables. Then

$$E[W(k)] = 0 \qquad E[W(k)W^T(k)] = \hat{R}$$
$$E[W(k)W^T(k')] = e^{-\frac{|k-k'|\delta_t}{G_t}}\hat{R} \tag{A2}$$

Proof: By induction. At $k = 0$

$$E[W(0)] = E[\hat{Q}n(0)] = \hat{Q}E[n(0)] = 0 \tag{A3}$$

and

$$E[W(0)W^T(0)] = E[\hat{Q}n(0)n^T(0)\hat{Q}^T] = \hat{Q}E[n(0)n^T(0)]\hat{Q}^T$$
$$= \hat{Q}\hat{Q}^T = \hat{R} \tag{A4}$$

Algorithm 1 Sequential importance sampling

1: Initialization. Set $k = 0$ and extract N particles $\{\mathbb{X}^i(0)\}_{i=1,\ldots,N} = \{x^i(0)\}_{i=1,\ldots,N}$ from the initial distribution $p(x(0))$.
2: Measurement update. Given the measurement $y(k)$, evaluate the importance weights $q^i(k) = p_y(y(k)|x^i(k), k)$ and normalize them

$$\tilde{q}^{\,i}(k) = \frac{q^i(k)}{\Sigma_{j=1}^N q^j(k)}$$

3: Prediction. For $i = 1, \ldots, N$, extract $x^i(k+1) \sim p_x(\cdot|x^i(k), k)$ and set $\mathbb{X}^i(k+1) = (\mathbb{X}^i(k), x^i(k+1))$.
4: Iteration. Increment k and return to step 2.

Algorithm 2 Sequential importance resampling

1: Initialization. Set $k = 0$ and extract N particles $\{\mathbb{X}^i(0)\}_{i=1,\ldots,N} = \{x^i(0)\}_{i=1,\ldots,N}$ from the initial distribution $p(x(0))$.
2: Measurement update. Given the measurement $y(k)$, evaluate the importance weights $q^i(k) = p_y(y(k)|x^i(k), k)$ and normalize them,

$$\tilde{q}^{\,i}(k) = \frac{q^i(k)}{\Sigma_{j=1}^N q^j(k)}$$

3: Resampling. Extract N particles $\{\tilde{\mathbb{X}}^i(k)\}$ from $\{\mathbb{X}^i(k)\}_{i=1,\ldots,N}$ with replacement according to importance weights $\tilde{q}^i(k)$.
4: Prediction. Extract $x(k+1)^i \sim p_x(\cdot|\tilde{x}^i(k), k)$ and set $\mathbb{X}(k+1)^i = (\tilde{\mathbb{X}}^i(k), x^i(k))$ for $i = 1, \ldots, N$.
5: Iteration. Increment k and return to step 2.

Algorithm 3 Trajectory prediction

1: Initialization. Set $k = 0$ and extract $i = 1, \ldots, N$ particles for the entire state $x^i(0)$
2: Filtering. Update the particles based on the measurement $y(k)$.
3: Prediction. For each aircraft and for each particle simulate the continuous dynamics of Eq. (18) and (19) over the prediction horizon, using the initial condition and wind vectors contained in $x^i(k)$.
4: Iteration. Increment k, and return to step 2.

Algorithm 4 Sequential conditional particle filter

1: Initialization. Set $k = 0$. Generate $i = 1, \ldots, N$ particles, each comprising:
1) An estimate $z^i(k\delta, l)$ of the state of each of the aircraft $l = 1, \ldots, M(k)$ present in the airspace. Set $(X^i(k\delta_t, l), Y^i(k\delta_t, l)) = y(k, l)$ (the measured position of aircraft l at time $k\delta_t$), $\psi^i(k\delta_t, l)$ the direction of the flight plan of aircraft l, and $m^i(k\delta_t, l)$ the nominal mass of aircraft l
2) An estimate of the mean μ_X^i and μ_Y^i and the covariance matrix Σ^i of the wind states. Set $\mu_X^i = 0$, $\mu_Y^i = 0$ and $\Sigma^i = \hat{R}$.

2: Increment k.

3: Update aircraft list. Drop from the particles the state of all aircraft that have left the region of interest between $(k-1)\delta_t$ and $k\delta_t$ and add aircraft that have entered, initializing their states as earlier.

4: **for** aircraft $l = 1, \ldots, M(k)$ **do**

5: Extract wind realizations. For each particle $i = 1, \ldots, N$ extract Gaussian random variables

$$W_X^i \sim N(\mu_X^i, \Sigma^i) \quad \text{and} \quad W_Y^i \sim N(\mu_Y^i, \Sigma^i)$$

.

6: Evolution of aircraft state. For each particle $i = 1, \ldots, N$ propagate $z^i((k-1)\delta_t, l)$ to $z^i(k\delta_t, l)$ using Eq. (18) and (19) with W_X^i and W_Y^i.

7: Measurement update. Evaluate the weight of each particle $i = 1, \ldots, N$ according to the radar measurement $q^i = p(y(k, l)|z^i(k\delta_t, l))$ and normalize $\tilde{q}^i = q^i / \Sigma_{i=1}^N q^i$.

8: Resampling. Draw N particles with replacement from among the existing particles, with probability of selecting particle $j = 1, \ldots, N$ equal to $\tilde{q}^j$.

9: Conditioning. For ease of notation let $A^i = A(z^i(k\delta_t, l))$, $b_X^i = A^i W_X^i$ and $b_Y^i = A^i W_Y^i$. For each particle $i = 1, \ldots, N$ update the wind means and covariance matrix according to

$$\mu_X^i = \mu_X^i + \Sigma^i (A^i)^T (A^i \Sigma^i (A^i)^T)^{-1} (b_X^i - A^i \mu_X^i)$$

$$\mu_Y^i = \mu_Y^i + \Sigma^i (A^i)^T (A^i \Sigma^i (A^i)^T)^{-1} (b_Y^i - A^i \mu_Y^i)$$

$$\Sigma^i = \Sigma^i - \Sigma^i (A^i)^T (A^i \Sigma^i (A^i)^T)^{-1} A^i \Sigma^i$$

to reflect the conditional distributions $p(W_X|A^i W_X^i = b_X^i)$ and $p(W_Y|A^i W_Y^i = b_Y^i)$.

10: end for

11: Evolution of wind dynamics. Propagate the wind distribution of each particle $i = 1, \ldots, N$ to the next time step according to

$$\mu_X^i = \alpha \mu_X^i, \qquad \mu_Y^i = \alpha \mu_Y^i, \qquad \Sigma^i = a^2 \Sigma^i + (1 - a^2)\hat{R}$$

12: Iterate. Return to step 2.

Assume (A2) holds for some $k \geq 0$, show that it also holds for $k + 1$. Clearly,

$$E[W(k+1)] = E[aW(k) + Qn(k)] = aE[W(k)] + QE[n(k)] = 0 \tag{A5}$$

Moreover

$$\begin{aligned} E[W(k+1)W^T(k+1)] &= E[(aW(k) + Qn(k))(aW^T(k) \\ &\quad + n^T(k)Q^T)] = a^2 E[W(k)W^T(k)] + aQE[n(k)W^T(k)] \\ &\quad + aE[W(k)n^T(k)]Q^T + QE[n(k)n^T(k)]Q^T \end{aligned}$$

Since $W(k)$ depends only on $n(0), n(1), \ldots, n(k-1)$ and is independent of $n(k)$,

$$E[W(k+1)W^T(k+1)] = a^2 \hat{R} + QQ^T = a^2 \hat{R} + (1 - a^2)\hat{R} = \hat{R} \tag{A6}$$

Assume that for some $k' = 0, 1, 2, \ldots, k+1$

$$E[W(k+1)W(k')^T] = e^{-\frac{|k+1-k'|\delta_t}{G_t}} \hat{R} \tag{A7}$$

Show that (backward induction toward $k' = 0$)

$$E[W(k+1)W(k'-1)] = e^{\frac{(k+1-(k'-1))\delta_t)}{G_t}} \hat{R} \tag{A8}$$

$$\begin{aligned} E[W(k+1)W(k'-1)^T] &= E[(aW(k) + Qn(k))W^T(k-1)] \\ &= aE[W(k)W^T(k'-1)] + QE[n(k)W^T(k'-1)] \end{aligned}$$

For $k' = 0, 1, 2, \ldots, k+1$, $W(k'-1)$ depends only on $n(0), n(1), \ldots, n(k-2)$ and is independent of $n(k)$. So, $E[n(k)W^T(k'-1)] = 0$. By the original induction hypothesis

$$E[W(k)W^T(k'-1)] = e^{-\frac{(k-(k'-1))\delta_t}{G_t}} \hat{R} \tag{A9}$$

Therefore,

$$E[W(k+1)W^T(k'-1)] = ae^{-\frac{(k-k'+1)\delta_t}{G_t}} \hat{R} \tag{A10}$$

$$= e^{-\delta_t/G_t} e^{-(k-k'+1)\delta_t/G_t} \hat{R} \tag{A11}$$

$$= e^{-(k-k'+2)\delta_t/G_t} \hat{R} \tag{A12}$$

which concludes the proof. □

Acknowledgments

Research was supported by the European Organization for the Safety of Air Navigation under contract C20051E/BM/03 and by the European Commission under project ERASMUS, FP6-TREN-518276 and through project iFLY, FP6-TREN-037180. The authors would like to thank E. Cinquemani, G. Chaloulos, F. Ramponi, and T. Schön for fruitful discussions and ideas.

References

[1] Long-Term Forecast of Air Traffic, European Organization for the Safety of Air Navigation, Rept. 2008 -2030, Feb. 2008.

[2] *Air Passenger Demand and CO2 Forecasts 2009*, Department for Transport, London, Jan. 2009.

[3] Cole, R. E., Richard, C., Kim, S., and Bailey, D., "An Assessment of the 60 km Rapid Update Cycle (RUC) with Near Real-Time Aircraft Reports," MIT Lincoln Laboratory, Report NASA/A-1, 15 July 1998.

[4] Chaloulos, G., and Lygeros, J., "Effect of Wind Correlation on Aircraft Conflict Probability," *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 6, 2007, pp. 1742–1752. doi:10.2514/1.28858.

[5] PHARE:EFMS Phase 1B, Eurocontrol, Report DOC 96- 70-15, Brussels, July 1996.

[6] Mondoloni, S., "A Multiple-Scale Model of Wind-Prediction Uncertainty and Application to Trajectory Prediction," 2006-7807, AIAA, 2006.

[7] Delahaye, D., Puechmorel, S., and Vacher, P., "Windfield Estimation by Radar Track Kalman Filtering and Vector Spline Extrapolation," *The 22nd Digital Avionics Systems Conference, DASC*, Vol. 1, IEEE Publications, Piscataway, NJ, 2003.

[8] Groenouwe, R. D., van Paassen, M. M., Mulder, M., and in 't Veld, A. C., "Pilot Support for Curved Decelerating Approaches in Actual Wind Conditions," *AIAA Guidance, Navigation and Control Conference and Exhibit*, Delft University of Technology, Delft, The Netherlands, 2008.

[9] Blom, H. A. P., and Bloem, E. A., "Joint IMMPDA Particle Filter," In *Sixth International Conference of Information Fusion*, IEEE Publications, Piscataway, NJ, 2003.

[10] Blom, H. A. P., and Bloem, E. A., "Tracking Multiple Maneuvering Targets from Possibly Unresolved, Missing or False Measurements," *8th International Conference of Information Fusion*, Vol. 1, Philadelphia, PA, 2005, p. 8.

[11] Schon, T., Gustafsson, F., and Nordlund, P. J., "Marginalized Particle Filters for Mixed Linear/Nonlinear State-Space Models," *IEEE Transactions on Signal Processing*, Vol. 53, No. 7, 2005, pp. 2279–2289.
doi:10.1109/TSP.2005.849151

[12] Lin, X., Kirubarajan, T., Bar-Shalom, B. Y., and Simon Maskell., "Comparison of EKF, Pseudomeasurement and Particle Filters for a Bearing-Only Target Tracking Problem," *Proceedings of SPIE: The International Society for Optical Engineering*, Vol. 4728, pp. 240–250.

[13] Daum, F., and Huang, J., "Curse of Dimensionality and Particle Filters," In *Aerospace Conference, 2003. Proceedings. 2003*, IEEE volume 4, Montreal, Sept. 2003, pp. 1979–1993.

[14] Crisan, D., and Doucet, A., "A Survey of Convergence Results on Particle Filtering Methods for Practitioners," *IEEE Transactions on Signal Processing*, Vol. 50, No. 3, 2002, pp. 736–746.
doi:10.1109/78.984773

[15] Chopin, N., "A Sequential Particle Filter Method for Static Models," *Biometrika*, Vol. 89, No. 3, 2002, pp. 539–552.
doi:10.1093/biomet/89.3.539

[16] Glover, W., and Lygeros, J., "A Stochastic Hybrid Model for Air Traffic Control Simulation," *Hybrid Systems: Computation and Control* edited by R. Alur and G. Pappas, Springer–Verlag, Berlin, 2004, pp. 372–386.

[17] Lymperopoulos, I., Lecchini, A., Glover, W., Maciejowski, J., and Lygeros, J., "A Stochastic Hybrid Model for Air Traffic Management Processes," University of Cambridge, England, U.K., Technical Report CUED/F-INFENG/TR.572, Feb. 2007.

[18] Cassandras, C. G., and Lygeros, J., *Stochastic Hybrid Systems*, Control Engineering Series, CRC Press, Boka Raton, FL, 2007.

[19] Lymperopoulos, I., and Lygeros, J., "Adaptive Aircraft Trajectory Prediction using Particle Filters," *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008-7387, AIAA, Reston, VA, Aug. 2008.

[20] Chaloulos, G., Roussos, G., Lygeros, J., and Kyriakopoulos, K., "Ground Assisted Conflict Resolution in Self-Separation Airspace," In *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008-6967, AIAA, Reston, VA, Aug. 2008.

[21] Schild, R. S., "Rule Optimization for Airborne Aircraft Separation," PhD thesis, Technical University of Vienna, Vienna, 1998.

[22] Erzberger, H., Paielli, R., Isaacson, D. R., and Eshow, M., "Conflict Detection and Resolution in the Presence of Prediction Error," In *1st USA/Europe Air Traffic Management R & D Seminar*, Saclay, France, 1997.

[23] Slattery, R., and Zhao, Y., "Trajectory Synthesis for Air Traffic Automation," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 2, 1997, pp. 232–238.
doi:10.2514/2.4056.

[24] Tomlin, C., Lygeros, J., and Sastry, S., "A Game Theoretic Approach to Controller Design for Hybrid Systems," *Proceedings of the IEEE*, Vol. 88, No. 7, 2000, pp. 949–969.
doi:10.1109/5.871303.

[25] Nuic, A., User Manual for the Base of Aircraft Data (BADA) revision 3.3, Eurocontrol Experimental Centre, Rept. 2000-034, 2000.

[26] Benjamin, S. G., Schwartz, B. E., and Cole, R. E., "Accuracy of ACARS Wind and Temperature Observations Determined by Collocation," *Weather and Forecasting*, Vol. 14, No. 6, 1999, pp. 1032–1038.
doi:10.1175/1520-0434(1999)014<1032:AOAWAT>2.0.CO;2

[27] Jackson, M. R., Zhao, Y. J., and Slattery, R. A., "Sensitivity of Trajectory Prediction in Air Traffic Management," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 2, 1999, pp. 219–228.
doi:10.2514/2.4398.

[28] *Earth System Research Laboratory. Rapid update cycle (RUC)*, National Oceanic & Atmospheric Administration (NOAA), [online database], http://ruc.noaa.gov [retrieved 25 Oct. 2006].

[29] Jazwinski, A., *Stochastic Process and Filtering Theory*, Vol 64, Mathematics in Science and Engineering, Academic Press, New York, 1970.

[30] Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *ASME -Journal of Basic Engineering*, Vol. 82, No. Series D1960, pp. 35–45.

[31] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, 2002, pp. 174–188.
doi:10.1109/78.978374

[32] Doucet, A., De Freitas, J. F. G., and Gordon, N., *Sequential Monte Carlo Methods in Practice*, Springer–Verlag, New York, 2001.

[33] Geweke, J., "Bayesian Inference in Econometric Models Using Monte Carlo Integration," *Econometrica : Journal of the Econometric Society*, Vol. 57, No. 6, Nov. 1989, pp. 1317–1339.

[34] Doucet, A., Godsill, S., and Andrieu, C., "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Journal of Statistics and Computing*, Vol. 10, No. 3, 2000, pp. 197–208.
doi:10.1023/A:1008935410038.

[35] Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *Radar and Signal Processing, IEE Proceedings F*, Vol. 140, No. 2, 1993, pp. 107–113.

[36] Liu, J., and West, M., "Combined Parameter and State Estimation in Simulation-Based Filtering," *Sequential Monte Carlo Methods in Practice*, edited by J. F. G. De Freitas, A. Doucet, and N. J. Gordon, Springer–Verlag, New York, 2000, pp. 197–217.

[37] Mondoloni, S., Swierstra, S., and Paglione, M., "Assessing Trajectory Prediction Performance-Metrics Definition," *DASC Digital Avionics Systems Conference*, Vol. 1, pp. 3.C.1–13, 2005.

[38] Ryan, H. F., Paglione, M. M., and Green, S. M., "Review of Trajectory Prediction Accuracy Methodology and Comparison of Error Measurement Metrics," *AIAA Guidance, Navigation, and Control, Providence*, 2004-4787, AIAA, Reston, VA, Aug. 2004.