

Flight-Mode-Based Aircraft Conflict Detection using a Residual-Mean Interacting Multiple Model Algorithm *

Inseok Hwang, Jesse Hwang, and Claire Tomlin
Hybrid Systems Laboratory
Department of Aeronautics and Astronautics
Stanford University, Stanford, CA 94305
ishwang, jdhwang, tomlin@stanford.edu

Based on the trajectory prediction error model proposed by Paielli and Erzberger, we propose nominal and probabilistic conflict detection algorithms using *flight mode estimates* as well as the aircraft current state estimates. This is different from previous conflict detection algorithms which use current state estimates only. Our algorithms are therefore based on *hybrid* models of aircraft, which allow for both continuous dynamics and discrete mode switching. To obtain accurate state and mode estimates, we propose a modified version of the Interacting Multiple Model (IMM) algorithm designed by Bar-Shalom et al. called the Residual-Mean Interacting Multiple Model (RMIMM) method. RMIMM is a multiple-model-based estimation algorithm based on a new likelihood function which uses the mean of the residual produced by each mode matched filter (usually Kalman filter), producing better mode estimates, and therefore better state estimates, than in the IMM case. We demonstrate our algorithm on multiple aircraft scenarios, and in the latter part of the paper, the probabilistic conflict detection algorithm is combined with the protocol-based conflict resolution algorithm, designed by the authors in earlier work.

KEYWORDS: hybrid state estimation, Interacting Multiple Model, conflict detection, Air Traffic Control.

Introduction

We consider in this paper the problem of accurate hybrid state estimation for aircraft conflict detection. The contributions of this paper are: a new variant of a hybrid state estimation scheme, and the application of hybrid estimation methods in general to conflict detection. The problem is applicable to both ground and airborne control scenarios in Air Traffic Control (ATC), with ground control using radar information, or airborne using information from Automatic Dependent Surveillance - Broadcast (ADS-B) data.

To the best of our knowledge, all conflict detection algorithms available are based on aircraft continuous state information (see Kuchar and Yang¹ for a comprehensive survey). For example, Yang et al.² propose a conflict alerting logic based on sensor and trajectory uncertainties, with conflict probability based on Monte Carlo simulation; Paielli et al.³ and Prandini et al.⁴ propose analytic algorithms for computing probability of conflict. The performance of the conflict detection algorithms depends strongly on the accuracy of these state estimates. A difficulty in aircraft state estimation lies in precision tracking when the aircraft makes

an unanticipated change of flight mode (e.g., mode change from the straight flight at constant speed to a coordinated turn). Thus, a challenge in aircraft tracking is to provide high quality state estimates of the aircraft in such situations. The current algorithms in use for ATC tracking are based on constant gain Kalman filters, known as $\alpha - \beta$ or $\alpha - \beta - \gamma$ filters.^{5,6}

Because a single Kalman filter cannot provide good estimates for a hybrid system with various modes, many adaptive state estimation algorithms have been proposed.⁷⁻¹⁰ The algorithm considered in this paper is a modified form of the Interacting Multiple Model (IMM) algorithm.^{11,12} The IMM algorithm is a multiple-model-based estimation algorithm which computes the state estimate using a weighted sum of the estimates from a bank of Kalman filters that are matched to different modes of the system. The weights are based on the *mode probability*, or the probability that an aircraft is in a given mode, computed from the *mode likelihood function*. It is the determination of this likelihood function that we focus on in this paper. The complexity of IMM has been shown to be approximately linear in the number of modes.⁷ IMM and its variants have been applied to single and multiple aircraft tracking problems.⁷

Hybrid system state estimation is an area of recent research activity. Balluchi et al.¹³ present the design for a location observer for discrete state estimation

*This research is supported by DARPA under the Software Enabled Control Program (AFRL contract F33615-99-C-3014 and Boeing contract Z20040), by ONR under MURI contract N00014-02-1-0720, and by an NSF Career Award (ECS-9985072).

combined with a Luenberger observer for continuous state estimation and give a condition for the estimation error to converge exponentially. Vidal et al.¹⁴ derive observability conditions for jump linear systems based on a rank test inspired by the familiar observability of linear systems. Hwang et al.¹⁵ propose that state error covariances may be used to improve these estimation results.

In this paper, we consider 2D aircraft conflict detection problems in which aircraft are assumed to fly at the same altitude. We consider two flight modes: constant heading - constant velocity, and coordinated turn. Thus, an aircraft trajectory is assumed to be composed of straight lines and circular arcs. For conflict detection, we use the nominal and probabilistic conflict detection algorithms from Paielli and Erzberger³ (which are based on the continuous state only) and extend them so that they use both the continuous state and flight mode estimates. Our algorithm for state and mode estimation is a modified IMM, called the Residual-Mean IMM (RMIMM), which has a likelihood function that uses the mean of the residual produced by each Kalman filter. Using the idea, proposed in Hanlon and Maybeck¹⁶ that the residual produced by each Kalman filter in the IMM algorithm should have a non-zero mean if the Kalman filter is not the correct one, we design a likelihood function that gives clearer and sharper differences between the correct mode and the other modes, so that the number of false mode estimates decreases relative to the standard IMM. This property is advantageous since the standard IMM may produce incorrect mode estimates frequently as the number of modes in the model increases. We then combine the probabilistic conflict detection algorithm with the protocol-based conflict resolution algorithm designed by the authors.^{17,18} We show that, even in cases in which the turn mode is a small portion of the whole aircraft flight trajectory, the accuracy of this hybrid conflict detection algorithm is improved over continuous schemes, especially in the airspace around waypoints and airports where several airways converge. In addition, the flight mode estimate could also be used for *blunder detection*. That is, aircraft 1 (or a ground controller) detects a conflict early enough to take (or issue) a safe resolution maneuver if aircraft 2 (blunderer) starts a maneuver which might cause a conflict with aircraft 1.

The paper is organized as follows: Section 2 presents the Residual-Mean Interacting Multiple Model algorithm. In Section 3, we derive the aircraft models for tracking and show the performance of the Residual-Mean Interacting Multiple Model algorithm through an example. In Section 4, nominal and probabilistic conflict detection algorithms and conflict detection and resolution examples are presented. Conclusions are presented in Section 5.

Residual-Mean Interacting Multiple Model Algorithm

In this section, we describe the general structure of the IMM algorithm and propose a modified IMM algorithm which uses information about the mean of the residual. We call this modified IMM algorithm the Residual-Mean IMM (RMIMM).

We consider a stochastic linear hybrid system with discrete-time, continuous-state dynamics:

$$\begin{aligned} x(k+1) &= A_j x(k) + w_j(k) \\ z(k) &= C_j x(k) + v_j(k) \end{aligned} \quad (1)$$

and a Markov transition of the discrete state (mode) given by:

$$P[j(k+1)|i(k)] = H_{ij} \quad i, j \in M = \{1, 2, \dots, N\} \quad (2)$$

where $x \in \mathbb{R}^n$ and $z \in \mathbb{R}^p$ are the state and the output respectively. M is the set of discrete states. The terms w and v are respectively the mode-dependent, uncorrelated, white Gaussian process noise and measurement noise with zero means and covariances Q_j and R_j . H_{ij} is the Markov mode transition probability from mode i to mode j . Thus, given the above system parameters, hybrid estimation estimates both the continuous state and the discrete state at time k from the measurement sequence up to time $k-1$ ($k = 1, 2, \dots$).

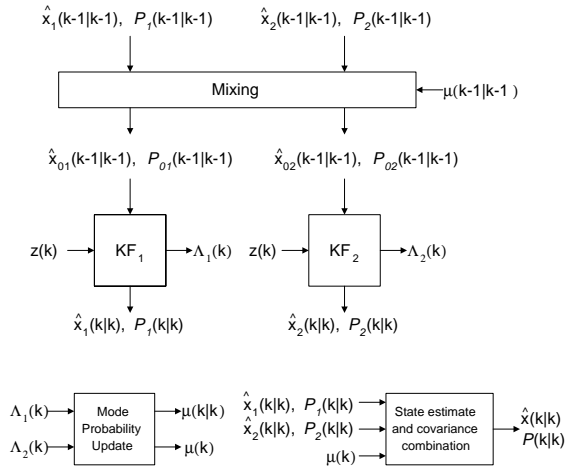


Fig. 1 Structure of the IMM algorithm (for two modes).¹²

The IMM algorithm is a multiple-model-based state estimation algorithm which computes the state estimate using a weighted sum of estimates from a bank of Kalman filters matched to different modes of the system. The general structure of the IMM algorithm as shown in Figure 1 is as follows:

- Mixing probability:** This is the probability that the system is in mode i at time $k-1$, given that it is in mode j at time k :

$$\mu_{ij}(k-1|k-1) = \frac{1}{c_j} H_{ij} \mu_i(k-1) \quad (3)$$

where c_j is a normalization constant, and where $\mu_i(k)$ is the mode probability of mode i at time k , i.e., a measure of how probable it is that the system is in mode i at time k . The initial condition $\mu_i(0)$ is assumed given, and is usually obtained from properties of the system.

2. **New initial states and covariances:** The input to each Kalman filter is adjusted by weighting the output of each Kalman filter with the mixing probability as the weight:

$$\begin{aligned} & \hat{x}_{0j}(k-1|k-1) \\ = & \sum_i \hat{x}_i(k-1|k-1)\mu_{ij}(k-1|k-1) \\ & P_{0j}(k-1|k-1) \\ = & \sum_i \{P_i(k-1|k-1) + [\hat{x}_i(k-1|k-1) \\ & - \hat{x}_{0j}(k-1|k-1)][\hat{x}_i(k-1|k-1) \\ & - \hat{x}_{0j}(k-1|k-1)]^T\} \mu_{ij}(k-1|k-1) \end{aligned}$$

where $\hat{x}_i(k-1|k-1)$ and $P_i(k-1|k-1)$ are the state estimate and its covariance produced by Kalman filter i at time $k-1$ after the measurement update.

3. **Kalman Filter:** N Kalman filters run in parallel (multiple-model-based (hybrid) estimation).
4. **Mode likelihood functions:** The likelihood function of mode j is a measure of how likely it is that the model used in Kalman filter j is the correct one; it is computed with the residual and its covariance produced by Kalman filter j :

$$\Lambda_j(k) = \mathcal{N}(r_j(k); 0, S_j(k)) \quad (4)$$

where $r_j(k) := z(k) - C_j \hat{x}_j(k|k-1)$ is the residual of Kalman filter j , $\hat{x}_j(k|k-1)$ is a state estimate by Kalman filter j at time k before the measurement update, and $S_j(k)$ is its covariance.

5. **Mode probabilities:** The probability of mode j is a measure of how probable it is that the system is in mode j :

$$\mu_j(k) = \frac{1}{c} \Lambda_j(k) \sum_i H_{ij} \mu_i(k-1) =: \frac{1}{c} \Lambda_j(k) \hat{\mu}_j(k) \quad (5)$$

where c is a normalization constant. The probability of each mode is updated using the likelihood function.

6. **Combination (output of the IMM):** The state estimate is a weighted sum of the estimates from N Kalman filters and the mode estimate is

the mode which has the highest mode probability:

$$\begin{aligned} \hat{x}(k|k) &= \sum_j \hat{x}_j(k|k) \mu_j(k) \\ P(k|k) &= \sum_j \{P_j(k|k) + [\hat{x}_j(k|k) - \hat{x}(k|k)] \\ & \quad \times [\hat{x}_j(k|k) - \hat{x}(k|k)]^T\} \mu_j(k) \\ \hat{m}(k|k) &= \arg \max_j \mu_j(k) \end{aligned}$$

where $\hat{m}(k|k)$ is the mode estimate at time k .

As can be seen from the standard IMM algorithm, the mode probability in (5) depends on the likelihood function Λ_j . Thus, if the likelihoods of the modes are close to each other, the mode estimate may be inaccurate. Inaccurate mode estimates could produce poor state estimates, thus degrading the tracking accuracy. Because we are interested in using this for conflict detection, we propose a method which reduces false mode estimation by increasing the difference between the likelihood of the correct mode and the likelihoods of the other modes, using the fact that if the Kalman filter corresponding to mode j is the correct one, then the residual in (4) should be a white Gaussian process with a zero mean. Otherwise, its mean should not be zero. We review from¹⁶ the mean and covariance of the residual. The mean of the residual is

$$\begin{aligned} E[r_j(k)|Z^{k-1}] &= E[z(k) - C_j \hat{x}_j(k|k-1)|Z^{k-1}] \\ &= E[z(k)|Z^{k-1}] - E[C_j \hat{x}(k|k-1)|Z^{k-1}] \end{aligned} \quad (6)$$

where Z^{k-1} is the measurement sequence up to time $k-1$. If the measurement sequence is given, the state estimates are not random. So, (6) becomes

$$E[r_j(k)|Z^{k-1}] = E[z(k)|Z^{k-1}] - C_j \hat{x}(k|k-1) \quad (7)$$

The covariance of the residual is

$$\begin{aligned} & Cov[r_j(k)|Z^{k-1}] \\ = & E[r_j(k)r_j(k)^T|Z^{k-1}] - E[r_j(k)|Z^{k-1}]E[r_j(k)|Z^{k-1}]^T \\ = & E[z_j(k)z_j(k)^T|Z^{k-1}] - E[z_j(k)|Z^{k-1}]E[z_j(k)|Z^{k-1}]^T \\ = & Cov[z_j(k)|Z^{k-1}] \end{aligned} \quad (8)$$

As outlined in Hanlon and Maybeck,¹⁶ the conditional covariance matrix of the residual is not dependent on the Kalman filter model. It is only dependent on the covariance of the measurements. Thus, all of the Kalman filter residuals have the same covariance since they all use the same measurement. The following notation is used:¹⁶

$$\Delta A_j := A_t - A_j, \quad \Delta C_j := C_t - C_j \quad (9)$$

where the subscript t denotes the true (or correct) model. The dimensions of the model in (1) for all j are assumed to be the same. If the model for Kalman filter j is not matched with the true model, the residual

is

$$\begin{aligned}
r_j(k) &:= z(k) - C_j \hat{x}_j(k|k-1) \\
&= C_t A_t e_j(k-1|k-1) + (C_t \Delta A_j + \Delta C_j A_t \\
&\quad - \Delta C_j \Delta A_j) \hat{x}_j(k-1|k-1) + C_t w_t(k-1) \\
&\quad + v_t(k)
\end{aligned} \tag{10}$$

where $e_j(k-1|k-1) = x_t(k-1) - \hat{x}_j(k-1|k-1)$ is the state estimation error. The detailed derivation is in Hanlon and Maybeck.¹⁶ The mean of the residual is

$$\begin{aligned}
&E[r_j(k)|Z^{k-1}] \\
&= C_t A_t E[e_j(k-1|k-1)|Z^{k-1}] + (C_t \Delta A_j \\
&\quad + \Delta C_j A_t - \Delta C_j \Delta A_j) \hat{x}_j(k-1|k-1)
\end{aligned} \tag{11}$$

To compute the mean of the residual, we first compute the expectation of the state estimation error:

$$\begin{aligned}
&E[e_j(k|k)|Z^{k-1}] \\
&= E[x_t(k-1) - \hat{x}_j(k-1|k-1)|Z^{k-1}] \\
&= (I - K_j(k)C_t)A_t E[e_j(k-1|k-1)|Z^{k-1}] \\
&\quad + ((I - K_j(k)C_t)\Delta A_t - K_j(k)\Delta C_j A_j) \\
&\quad \times \hat{x}_k(k-1|k-1)
\end{aligned} \tag{12}$$

where K_j is the Kalman filter gain for Kalman filter j . This is a recursive equation with respect to the state estimation error. Thus, the mean of the residual is computed from (11) and (12).

To the best of our knowledge, all multiple-model-based estimation and learning algorithms including various IMM algorithms use a likelihood function whose mean is zero to determine the current mode in which the system lies.^{7,9,10} We propose the RMIMM, which uses the mean of the residual to increase the difference between the likelihood of the correct mode and those of the other modes, thereby decreasing the number of false mode estimates. Since in the IMM framework there is no true model for sure, i.e., with probability 1, we propose a new definition of the mean of the residual: a weighted sum of the mean of the residual computed by each Kalman filter with the mode probability estimate in (5) as the weight. Similarly, a new definition of the mean of the state estimation error is proposed as a weighted sum of the mean of the state estimation error corresponding to Kalman filter j with the same weight.

$$\begin{aligned}
&E[r_j(k)|Z^{k-1}] \\
&\triangleq \sum_{j=1}^N \{C_t A_t E[e_j(k-1|k-1)|Z^{k-1}] \\
&\quad + (C_t \Delta A_j + \Delta C_j A_t - \Delta C_j \Delta A_j) \hat{x}_j(k-1|k-1)\} \\
&\quad \times \hat{\mu}_j(k) \\
&E[e_j(k)|Z^{k-1}] \\
&\triangleq \sum_{j=1}^N \{(I - K_j(k)C_t)A_t E[e_j(k-1|k-1)|Z^{k-1}] \\
&\quad + ((I - K_j(k)C_t)\Delta A_t - K_j(k)\Delta C_j A_j) \\
&\quad \times \hat{x}_k(k-1|k-1)\} \hat{\mu}_j(k)
\end{aligned} \tag{13}$$

If the mode probability of mode j is large, the mean of the residual becomes small (i.e., close to zero) because the other mode probabilities $\mu_i(k)$ for $\forall i \neq j$ are

small. This corresponds to the fact that the residual has a zero mean if the Kalman filter is the correct one. Since the proposed mean of the residual is small if the mode probability of the corresponding Kalman filter is small, and large if the mode probability of the corresponding Kalman filter is big, we can use the mean of the residual in (13) to make the likelihood of the correct mode more distinct from those of the other modes. Therefore, using the mean of the residual provided by each Kalman filter, we propose a new likelihood function:

$$\Lambda_j^{new}(k) = \begin{cases} \frac{\|E[r_j(k)|Z^{k-1}]\|^{-1} \Lambda_j(k)}{\sum_{j=1}^N \|E[r_j(k)|Z^{k-1}]\|^{-1} \Lambda_j(k)} & \text{if } E[r_j(k)|Z^{k-1}] \neq 0 \\ \Lambda_j(k) & \text{otherwise} \end{cases} \tag{14}$$

Proposition 1 *The differences between the new likelihood function (14) for the correct mode and those for the incorrect mode, is greater than the corresponding differences using the previous likelihood function from (4).*

Proof: If the model in Kalman filter j is incorrect, the mean of residual is not zero and the likelihood of mode j from the new likelihood function in (14) is less than that of the standard likelihood function in (4). If the model in Kalman filter j is correct, the likelihood of mode j from the new likelihood function is the same as that of the standard likelihood function in (4). Thus, the differences between the likelihood of the correct mode and those of incorrect modes are greater and the result follows. ■

Aircraft model for tracking

In this section, we derive aircraft models for aircraft tracking and conflict detection in the plane. Since our conflict detection is based on both current state and flight mode, the conflict detection algorithm needs the aircraft's current position, velocity, and yaw rate. If an aircraft is in the coordinated turn mode, the future trajectory of the aircraft for conflict detection is constructed along a circular arc. Thus, the aircraft's yaw rate should be estimated accurately. We found that the coordinated turn model with known yaw rates could not estimate the actual yaw rate and flight mode accurately if the aircraft's actual yaw rate is not matched to those used in the model. This is because the yaw rate estimate comes from the finite set of predetermined yaw rates. To solve this problem, the yaw rate is included as a state component, to be estimated. Thus, a new model has an additional equation for the yaw rate ω :

$$\omega(k+1) = \omega(k) + w_\omega(k) \tag{15}$$

where $w_\omega(k)$ is a zero-mean white Gaussian noise. This model is nonlinear, so the extended Kalman filter

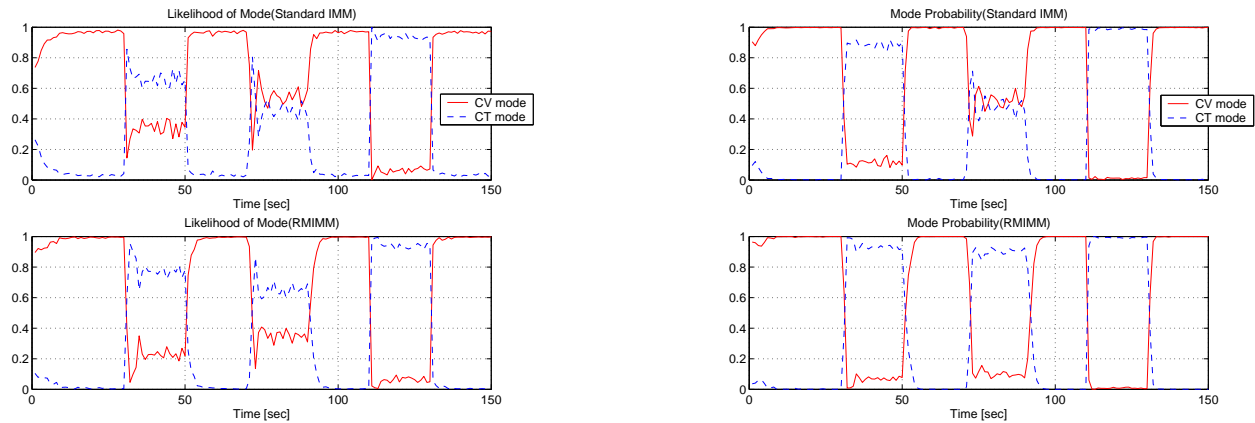


Fig. 2 100 times Monte Carlo simulation results using the IMM and the RMIMM for the aircraft tracking example. (left) Likelihood of each mode. (right) Mode probability of each mode. (The RMIMM gives more distinct likelihoods of modes than those of the IMM.)

is used for state estimation.^{6, 12, 19} However, the accuracy of yaw rate estimates is sensitive to the design parameters used for state estimation. The estimate of yaw rate oscillates severely around the true rate at the starting of a maneuver. This model has been used for aircraft tracking widely,^{6, 12, 19} however, because it produces good position and velocity estimates.^{6, 12, 19}

In order to obtain accurate yaw rate estimates, we use the *Wiener-sequence acceleration model*²⁰ as the aircraft model for the coordinated turn mode. It assumes that the acceleration increment is an independent, zero-mean, white Gaussian noise process, i.e. $a(k) - a(k-1) = w(k)$ (where a is an acceleration and w is a zero-mean, white Gaussian noise). Standard Kalman filters are used for state estimation because this model is linear. Since ADS-B information is assumed to be used for measurements, we use both aircraft's position and velocity information in ADS-B to increase the estimation accuracy. However, there is no restriction in using the aircraft's position information only to estimate the aircraft's current states with the RMIMM. If the state of an aircraft is defined as $\xi = [x \dot{x} \ddot{x} y \dot{y} \ddot{y}]^T$, the aircraft model for constant velocity (CV) mode is

$$\xi(k) = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \xi(k-1) + \begin{bmatrix} T^2/2 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & T^2/2 & T & 0 \end{bmatrix}^T w_{cv}(k) \quad (16)$$

$$z(k) = C\xi(k) + v_{cv}(k) \quad (17)$$

For the coordinated turn (CT) mode, the aircraft model (Wiener-sequence acceleration model) is

$$\xi(k) = \begin{bmatrix} 1 & T & T^2/2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & T^2/2 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \xi(k-1) + \begin{bmatrix} T^2/2 & T & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & T^2/2 & T & 1 \end{bmatrix}^T w_{ct}(k) \quad (18)$$

$$z(k) = C\xi(k) + v_{ct}(k) \quad (19)$$

where $C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$ is the output matrix, T is the sampling interval, and w_{cv} , w_{ct} , v_{cv} , and v_{ct} are zero-mean, uncorrelated, white Gaussian process noise and measurement noise for CV mode and CT mode, respectively.

The yaw rate estimate $\hat{\omega}$ is computed with a state estimate from (18) as follows:

$$\hat{\omega}(k) = \text{sign}(\hat{x}(k)\hat{y}(k) - \hat{x}(k)\hat{y}(k)) \sqrt{\frac{\hat{x}(k)^2 + \hat{y}(k)^2}{\hat{x}(k)^2 + \hat{y}(k)^2}} \quad (20)$$

To get the smooth estimate of ω , we design a first-order, low-pass filter. Therefore, we use a two-mode stochastic linear hybrid system model for aircraft tracking.

Example: Tracking a single aircraft

We show the performance of the proposed RMIMM through an aircraft tracking example, using (16)-(17) and (18)-(19) and assuming a 1 Hz ADS-B update. The RMS error of position and velocity measurements of the Global Positioning System (GPS) and Inertial Navigation System (INS) combined are assumed to be 50m and 3m/sec respectively.^{2, 21} The following Markov discrete state (mode) transition matrix de-

defined in (2) was used

$$H = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix} \quad (21)$$

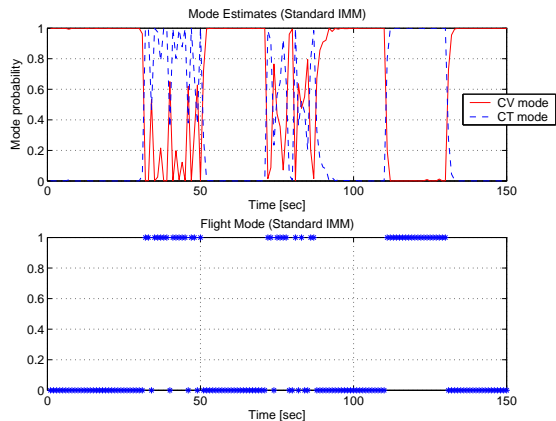


Fig. 3 A simulation result using the IMM for the aircraft tracking example.

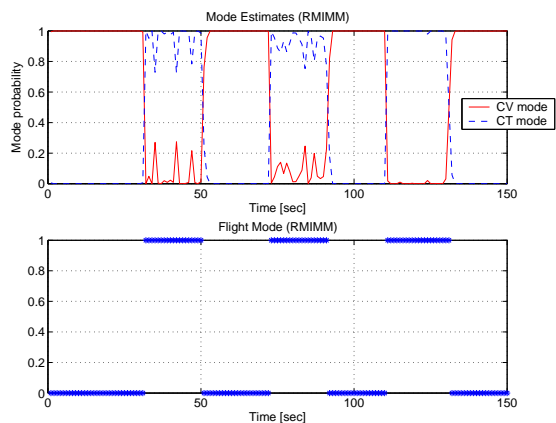


Fig. 4 A simulation result using the RMIMM for the aircraft tracking example.

We design a test flight trajectory with constant aircraft speed $v = 480$ knots (kn), composed of seven segments: straight flight from 0 to 30 seconds, a coordinated turn with $\omega = 3^\circ/sec$ from 31 to 50 seconds, straight flight from 51 to 70 seconds, a coordinated turn with $\omega = 1.5^\circ/sec$ from 71 to 90 seconds, straight flight from 91 to 110 seconds, a coordinated turn with $\omega = -4.5^\circ/sec$ from 111 to 130 seconds, and straight flight from 131 to 150 seconds. 100-trial Monte Carlo simulation results in Figure 2 show that the RMIMM gives more distinct likelihoods of modes than those of the standard IMM. In addition, a single simulation result in Figure 3 and Figure 4 shows how the RMIMM reduces the number of false mode estimates. The RMS estimation errors of position and velocity using the RMIMM are $15m$ and $2.1m/sec$. The RMS estimation errors of position and velocity using the IMM are $18m$ and $2.3m/sec$. The RMS estimation errors of the RMIMM are slightly better than those of the IMM, yet both algorithms give smaller RMS errors than those of

the raw measurements. Thus, the main advantage of the RMIMM, that it gives better mode estimates than the IMM, is demonstrated.

Conflict detection

In this section, we use the RMIMM to estimate current states and flight mode with the aircraft models derived in Section 3. Then, we predict conflicts within a look-ahead time ($T_{LA} < \infty$) using these estimates. A conflict is defined as an event in which the relative separation of two aircraft becomes less than a predefined safety distance (R). We set $R \triangleq 5$ nautical miles (nm) in this paper. For conflict detection, we project each aircraft's current state into the future in two ways: a nominal state projection which constructs a single trajectory based on the current state vector without prediction uncertainties, and a probabilistic state projection in which uncertainties in the model are used to construct a set of possible future trajectories, each weighted by its probability of occurring. In this paper, the nominal state projection method is used for comparison only.

Nominal conflict detection

We perform conflict detection by projecting the aircraft's current velocity estimate, obtained by the RMIMM, into the future. We project along a single trajectory which is a straight line if the current flight mode estimate is CV mode or a circular arc if it is CT mode. To validate the nominal conflict detection algorithm, we use the prediction uncertainty model in³. The trajectory prediction error is modelled as normally distributed with a zero mean, and a covariance with eigenvectors in the along-track and cross-track directions, as shown in Figure 5. The along-track error is modelled as a $15kn$ standard deviation speed uncertainty. The cross-track error grows from its initial uncertainty of $15m$ at the aircraft's current position (which is the position estimation error bound of the RMIMM) to a steady-state error with a standard deviation of $1nm$. The growth rate is $\frac{v}{57}nm/min$ (where v is the aircraft's speed). This growth rate is equivalent to a lateral deviation error with a standard deviation of 1° .² Thus, the prediction error covariance matrix in the body-fixed frame aligned with the aircraft heading is a diagonal matrix.

Probabilistic conflict detection

For probabilistic conflict detection, we should compute the conflict probabilities of all possible conflicts. In this paper, we use the conflict prediction model in Paielli and Erzberger³ to compute conflict probability. The trajectory prediction errors are modelled as Gaussian random variables with zero means and time-varying covariances. Through coordinate transformation, an analytical solution for conflict probability is derived based on the assumption that aircraft fly straight with constant velocity.³ Since our con-

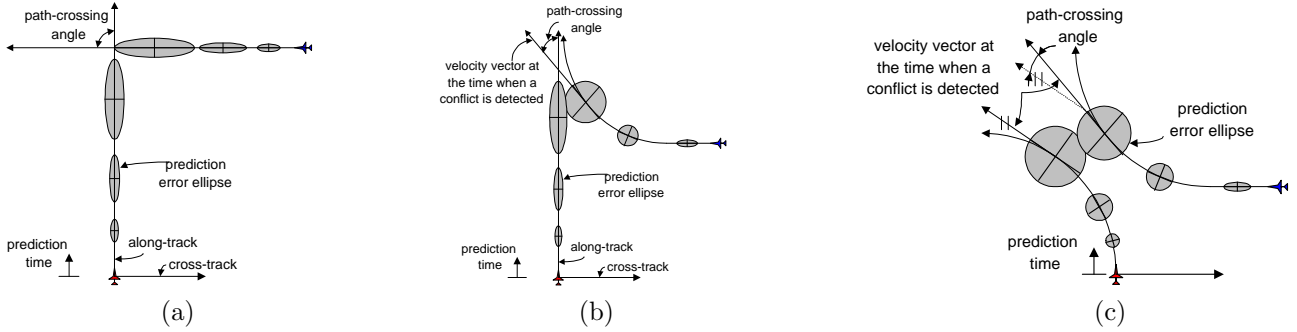


Fig. 5 Trajectory prediction error ellipses. (a) Both aircraft are in the constant velocity mode. (b) An aircraft is in constant velocity mode and the other is in the coordinated turn mode. (c) Both aircraft are in the coordinated turn mode.

conflict detection algorithm allows aircraft to be in CT mode, we suggest a modified conflict probability computation when aircraft are in CT mode. First, we summarize the conflict detection algorithm in Paielli and Erzberger³ (in which the authors considered only the case in which both aircraft are in CV mode). The trajectory prediction error is modelled as a zero-mean normal distribution, with diagonal covariance matrix S in the body-fixed frame aligned with the aircraft heading. If a rotation matrix R_{rot} transforms the heading-aligned body-fixed frame to the reference frame, the transformed covariance matrix is

$$Q \triangleq R_{rot} S R_{rot}^T \quad (22)$$

Two error covariances for an aircraft pair can be combined into a single equivalent covariance which is assigned to one of the aircraft called the “stochastic aircraft”. The other aircraft called the “reference aircraft” is assumed to have no position uncertainty. Then, the combined prediction error covariance is

$$M \triangleq Q_s + Q_r - Q_{sr} \quad (23)$$

where Q_s and Q_r are the covariance of the stochastic aircraft and the covariance of the reference aircraft, respectively. Q_{sr} is the cross-correlation between the two aircraft.

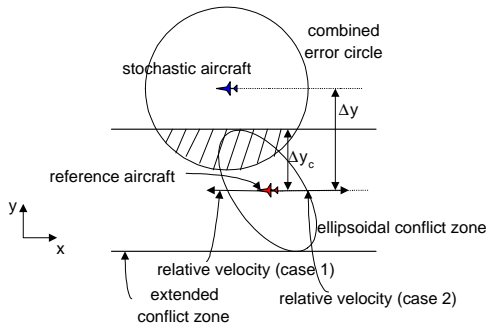


Fig. 6 Transformed conflict geometry.³

Through coordinate transformation, we make the combined covariance ellipse centered on the stochastic aircraft a unit circle and the relative velocity vector aligned in the x -direction as shown in Figure 6. The

elliptical conflict zone is centered on the reference aircraft. If the relative heading angle of two aircraft is constant during the conflict, the total conflict probability is the volume of the probability density function over the shaded area. Since the combined error covariance ellipse is a unit circle, the two dimensional Gaussian probability density function decouples into the product of two identical one-dimensional functions: $p(x, y) = p(x)p(y)$ where $p(x) = \exp(-x^2/2)/\sqrt{2\pi}$. Thus, the conflict probability P_c is³

$$\begin{aligned} P_c &= \int_{-\Delta y + \Delta y_c}^{-\Delta y - \Delta y_c} \int_{-\infty}^{\infty} p(x, y) dx dy \\ &= \int_{-\Delta y + \Delta y_c}^{-\Delta y - \Delta y_c} p(y) dy \int_{-\infty}^{\infty} p(x) dx \\ &= \int_{-\Delta y + \Delta y_c}^{-\Delta y - \Delta y_c} p(y) dy \end{aligned} \quad (24)$$

(24) is valid only in the case in which both aircraft are in CV mode (Figure 5-(a)). Therefore, for the conflict cases in Figure 5-(b) and Figure 5-(c), we change the prediction model (i.e., prediction error covariance S). If an aircraft is in CT mode, both the along-track error and the cross-track error are modelled as a 15 kn standard deviation speed uncertainty because the aircraft’s acceleration is composed of tangential (along-track) acceleration and centripetal (cross-track) acceleration. The relative heading is assumed to be constant during the conflict. We use the relative heading angle at time t_{md} , when the predicted minimum distance between aircraft is achieved, as the pass-crossing angle. The latter assumption is coarse, but since typical civilian aircraft yaw rates are not large ($\approx 3^\circ/\text{sec}$) and of short duration, the computation error may be small. So, it would be reasonable to use the analytical solution in (24) instead of a numerical one which is computationally intensive. However, to reduce the error in conflict probability computation, we introduce a time interval around the time t_{md} such as $[t_{md} - \delta, t_{md} + \delta]$ ($\delta > 0$), compute the conflict probability within this interval, and choose the maximum as the conflict probability.

Two main points about this algorithm warrant additional conditions for conflict detection. First, the conflict probability computation algorithm in Paielli and Erzberger³ is based on the conflict geometry, and

uses the extended conflict zone as shown in Figure 6 to compute the conflict probability. Thus, the conflict probability when the relative velocity of the reference aircraft is pointing towards the stochastic aircraft (case 1) is the same as that of the case in which the relative velocity of the reference aircraft is pointing away from the stochastic aircraft (case 2). Second, the proposed conflict resolution algorithms can be applied to both on-board applications and ground control applications. If the conflict detection algorithms are applied in ground control applications, they may handle large numbers of aircraft at a time. Thus, to eliminate the above error and to reduce the number of aircraft pairs for conflict detection computation, we propose the following conditions:

(Conditions for Conflict Detection Computation)

1. (a) The conflict detection algorithm is executed every τ ($0 < \tau < \infty$) seconds.
- (b) If mode changes are observed within the time interval τ , the conflict detection algorithm is executed instantly for the corresponding aircraft pairs.
- (c) For the aircraft pairs with conflicts with $P_c \geq \bar{P}$, the conflict detection is executed every τ_c ($0 < \tau_c < \tau$) seconds (where \bar{P} is a threshold for conflict detection and a design parameter).
2. (a) The conflict detection algorithm is executed for aircraft pairs satisfying $(\forall i, j)$
 - i. CASE I: Two aircraft fly towards each other.

$$[(\vec{r}_{ij}(k) \cdot \vec{v}_i(k) \geq 0) \wedge (\vec{r}_{ij}(k) \cdot \vec{v}_j(k) \leq 0)] \wedge [sign(\vec{r}_{ij}(k) \times \vec{v}_i(k)) \neq sign(\vec{r}_{ji}(k) \times \vec{v}_j(k))]$$
 - ii. CASE II: An aircraft follows the other aircraft.

$$[(\vec{r}_{ij}(k) \cdot \vec{v}_i(k) \geq 0) \wedge (\vec{r}_{ij}(k) \cdot \vec{v}_j(k) \geq 0) \wedge (\|\vec{v}_i(k)\| > \|\vec{v}_j(k)\|)]$$

$$\wedge [(\vec{r}_{ij}(k) \cdot \vec{v}_i(k) \leq 0) \wedge (\vec{r}_{ij}(k) \cdot \vec{v}_j(k) \leq 0) \wedge (\|\vec{v}_i(k)\| < \|\vec{v}_j(k)\|)]$$

where \vec{r}_{ij} is a relative position vector from aircraft i to aircraft j and \vec{v}_i is a velocity vector of aircraft i .
- (b) The conflict probability is computed only when the above Condition 2-(a) is satisfied and the predicted minimum distance within the look-ahead time is less than the test safety distance ($\bar{R} := \alpha R, \alpha \geq 1$).

In this paper, we set $\tau = 20sec$ and $\tau_c = 5sec$ for simulation. Condition 1-(b) comes from event-driven simulation in which simulation (or computation) is triggered not by a time instant but by an event (mode change in this case).^{22,23} For safety, even though there

is no mode change, we perform conflict detection in every τ seconds because measurements contain noise. Therefore, conflict detection may have prediction errors. We also pay more attention to aircraft pairs which have conflict probabilities over the threshold for safety. Condition 2 tests whether an aircraft pair may have a conflict based on the current aircraft's state and mode estimates. We can then filter out many aircraft pairs before applying the conflict detection algorithms. Since the predicted minimum distance is not exact due to prediction uncertainties and current state estimation errors, we compute conflict probability when it is less than the "test safety distance" ($\bar{R} \geq R$) so that we do not miss possible conflicts. In this paper, we set $\alpha = 1.5$.

Numerical examples

We apply these conflict detection algorithms to the scenarios of Figure 5, through many simulations. The RMIMM is used to estimate the aircraft's current state and flight mode. We assume the aircraft speed is constant at $v = 480kn$ for all simulations unless otherwise stated. The predefined safety distance is $5nm$. For prediction, the along-track and the cross-track RMS errors grow linearly with the rate of $0.25nm/min$ and $\frac{v}{57}nm/min$ from the initial uncertainty $15m$. Yet, the cross-track RMS error saturates when it reaches $1nm$.³ The RMS errors of position and velocity measurements are assumed to be $50m$ and $3m/sec$ respectively. Wind-error cross-correlation between aircraft (Q_{sr}) in (23) is assumed to be zero. The first-order low-pass filter $G(s) = \frac{1}{11.9s+1}$ is used for smoothing the yaw rate estimate.

In Figure 7, the deterministic conflict detection probability refers to the probability of successful conflict detection. In Figure 7-(a), both aircraft are in CV mode. The top graph shows the conflict geometry with the path-crossing angle at 90° . Middle and bottom plots show the average conflict probability as a function of the average time required to get to the predicted minimum separation point, where the minimum separation is 0 (collision) and 3 nautical miles respectively. The conflict probability of the probabilistic prediction is close to that of Paielli and Erzberger's algorithm.³ Thus, we conclude that the RMIMM used in the proposed probabilistic conflict detection algorithm produces good state estimates since the conflict probability of Paielli and Erzberger's algorithm assumes that there is no uncertainty in the current state.

In Figure 7-(b), only one aircraft is in CV mode, with a heading angle of 210° . The other aircraft begins in CV mode for a little bit and then switches to CT mode for most of its trajectory. The turn rate of the aircraft in CT mode is $1.5^\circ/sec$. There is an exact collision at the origin. The bottom plot shows the computed conflict probability versus the average time to minimum separation. The computation starts af-

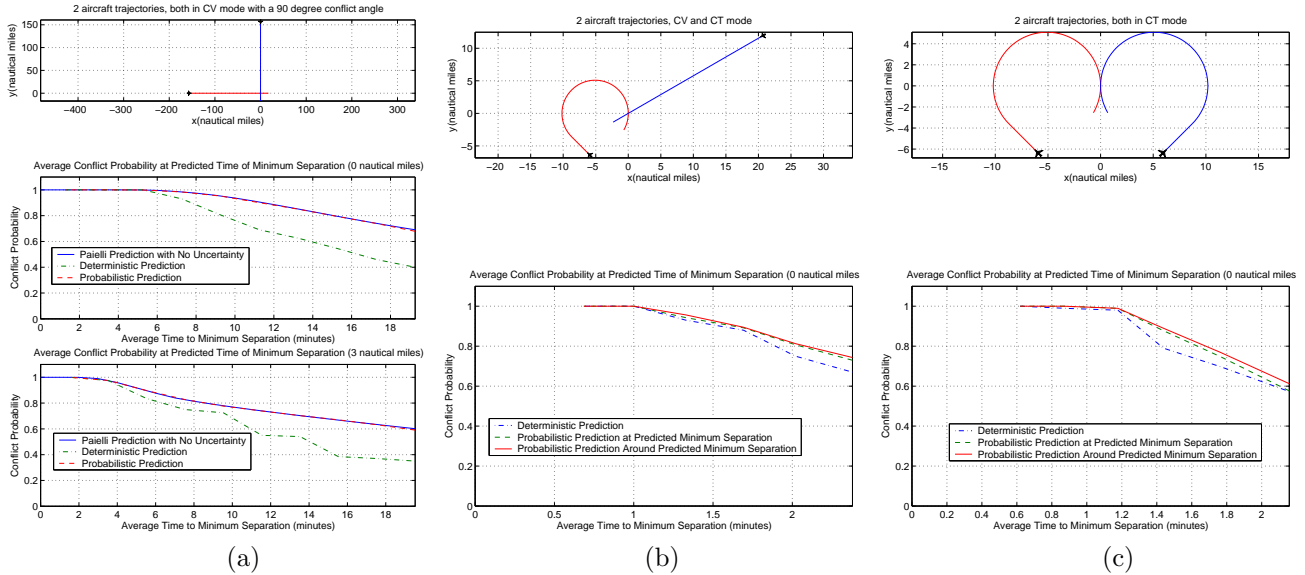


Fig. 7 200 Monte Carlo simulation results using the nominal and the probabilistic conflict detection algorithms. (a) Both aircraft are in the constant velocity mode. (b) An aircraft is in constant velocity mode and the other is in the coordinated turn mode. (c) Both aircraft are in the coordinated turn mode.

ter the second aircraft switches to CT mode so that both aircraft predict that there is some conflict in the future. At 6 regularly spaced points on the trajectory, we project the current state estimate to find a predicted minimum separation point and compute the conflict probability using the deterministic and probabilistic method. The dotted line represents the conflict probability at the predicted minimum separation. The continuous line comes from the maximum conflict probability in an interval with half width 5 seconds (i.e., $\delta = 5$) around the time of predicted minimum separation. The difference is small, but we can reduce the conflict probability computation error due to the assumption that the heading angle of aircraft is constant during the conflict. In Figure 7-(c), both aircraft are in CT mode and the plots are analogous to those in Figure 7-(b). Figure 7 shows the probabilistic conflict detection is more accurate than the deterministic one, especially when the look-ahead time is long.

We can interpret the conflict scenarios in Figure 5-(b) and in Figure 5-(c) as blunder detection because two aircraft are close to each other and the conflict time is very short compared to the look-ahead time. If an aircraft can detect that another aircraft (blunderer) starts a sudden maneuver that may cause a conflict within a short time, the aircraft can take a safe resolution maneuver. Blunder detection could be useful for conflict detection in congested airspace such as around waypoints and airports, and for parallel landings.²⁴ This is one of the advantages of using flight mode information for conflict detection over existing conflict detection methods. Figure 8 shows a blunder detection scenario for two aircraft. Initially two aircraft fly in parallel. Then one aircraft (blunderer) changes its heading. The trajectory interval in which the aircraft

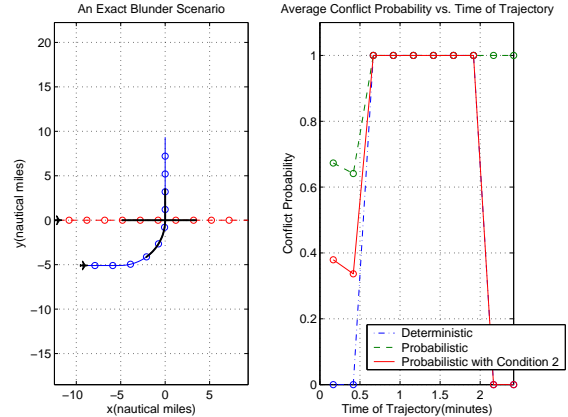


Fig. 8 Blunder detection scenario for two aircraft.

violate the $5nm$ safety zone is shown by the thick black lines. On the right figure, we plot the conflict probability as a function of the time of trajectory. The circles here correspond to the same time instants as the circles in the left figure. Thus, we can see how the conflict probability changes at points along the trajectory. This result was obtained with 200 simulations. One of the problems we observed was that even though the aircraft are no longer in conflict and heading away from each other at the end of the trajectory, the conflict probability algorithm in Paielli and Erzberger³ still returns a conflict probability of one. Thus, we use Condition 2-(b) to eliminate this phenomenon and the continuous line in Figure 8 shows this correction. Figure 9 shows multiple-aircraft conflict detection scenarios. We set the look-ahead time 20 minutes and the conflict probability threshold (\bar{P}) to 0.7. The conditions for conflict detection computation are applied to reduce the number of aircraft pairs for conflict detection. We perform conflict detection every 20 seconds

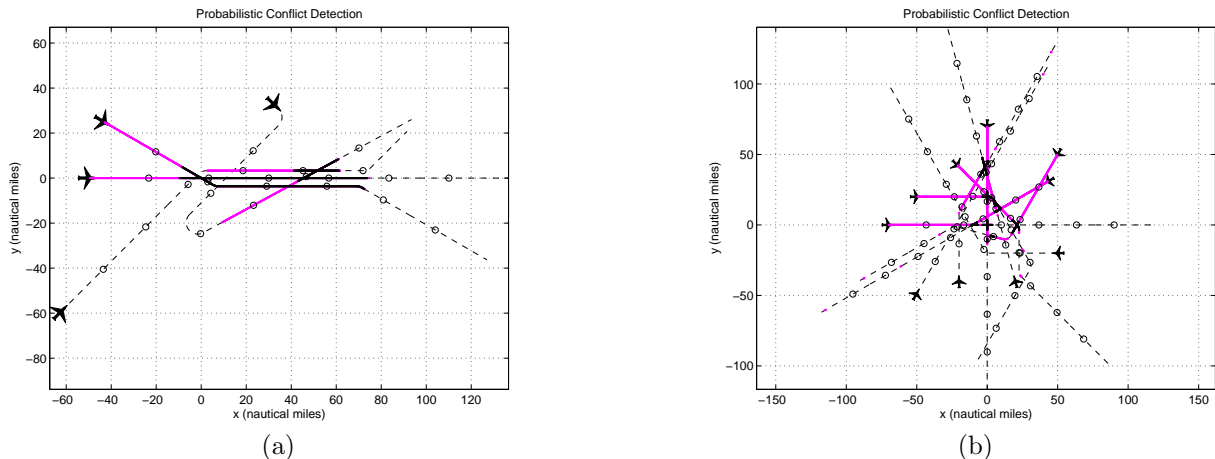


Fig. 9 Simulations for multiple-aircraft conflict detection. (a) Structured airspace scenario (four aircraft). (b) Free Flight scenario (ten aircraft).

($\tau = 20sec$) even if there is no flight mode change. For the aircraft pairs where conflict probability is greater than or equal to $\bar{P} = 0.7$, conflict detection is performed every 5 seconds ($\tau_c = 5sec$).

There are 4 aircraft in the multiple-aircraft conflict scenario for structured airspace in Figure 9-(a). The circles correspond to positions that are 100 seconds away from each other. All the aircraft fly at $8nm/min$ and have turn rates of $1.5^\circ/sec$. The grey regions indicate where the aircraft detect conflict. The black regions are where the aircraft are actually in conflict. The Free Flight scenario in Figure 9-(b) has the same parameters except that there are ten aircraft. Probabilistic methods are used for both scenarios. Here the circles are spaced 200 seconds apart for clarity. We also validate the probabilistic conflict detection algorithm through simulations with real air traffic data, called Enhanced Traffic Management System (ETMS) data, for air traffic in several sectors of Oakland Center airspace. In these simulations, the conflict detection and resolution algorithm is implemented in a centralized way that it performs aircraft tracking, conflict detection, and conflict resolution for all the aircraft. Since computational efficiency is important for real-time air traffic control, we compute average time required for state estimation and conflict probability computation. Average computation times for two aircraft tracking using the IMM and the RMIMM are 0.01 seconds and 0.013 seconds respectively. Average computation time for conflict probability for two aircraft conflict is 0.0041 seconds. These values are obtained using MATLAB on a 500 MHz Pentium III PC.

Finally, we combine the probabilistic conflict detection algorithm with the protocol-based conflict resolution algorithm.^{17,18} For the protocol-based conflict resolution algorithm for multiple aircraft, aircraft are assumed to cruise at the same altitude with varying velocities. Each aircraft's position and velocity are assumed to be available to all aircraft which are involved

in the conflict through an ADS-B data link. A conflict resolution maneuver is composed of a heading change with a possible velocity change, which are based on a closed-form analytic solution. We then construct a finite partition of the airspace around the conflict. Using the analytic solution, we derive a protocol for resolving the worst-case conflict within each partition. Since the protocol is based on the closed-form analytic solution, it can be implemented in airborne system for real-time conflict resolution, as well as in ATC ground systems, with guarantees of safety of the protocol (to within the limits of the models used). We provide guarantees of safety of the protocol. For details of the algorithm, see.^{17,18} We perform simulations for the symmetric and asymmetric conflict scenarios as shown in Figure 10. Here, the look-ahead time is 20 minutes. Figure 10 illustrates that the combined conflict detection and resolution algorithm successfully detects and resolves the conflicts.

Conclusions

In this paper, we proposed conflict detection algorithms which are based on both the aircraft continuous state and flight mode estimates. These algorithms can also be used for blunder detection in which an aircraft (blunderer) makes a sudden maneuver which may cause a conflict. To obtain both the current state and flight mode estimates at the same time, we proposed the Residual-Mean Interacting Multiple Model algorithm (RMIMM), which uses the mean of the residual produced by each Kalman filter to reduce the number of false mode estimates. We validated the conflict detection algorithms through examples. Finally, we combined the probabilistic conflict detection algorithm with a protocol-based conflict resolution algorithm and illustrate it through multiple-aircraft conflict examples. The proposed conflict detection algorithms can be applied to other applications. For example, these can be useful for conflict detection or blunder detection for formation flying of the unmanned aerial vehicles

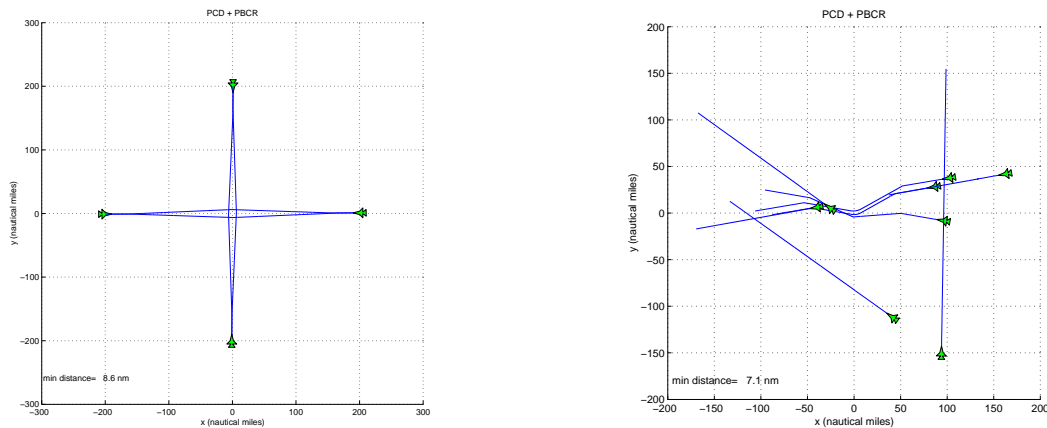


Fig. 10 Simulation for multiple-aircraft conflict detection and resolution using the probabilistic conflict detection and the protocol-based conflict resolution. (left) Symmetric conflict scenario (four aircraft). (right) Asymmetric conflict scenario (eight aircraft).

(UAVs). Since in the formation flying of the UAVs, the distance between the UAVs could be small, it is important to detect sudden maneuvers for safety.

Acknowledgements

The authors would like to thank Hamsa Balakrishnan for her invaluable discussion and comments during the development of the Residual-Mean Interacting Multiple Model algorithm.

References

- ¹J.K. Kuchar and L.C. Yang. A review of conflict detection and resolution modeling methods. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):179–189, December 2000.
- ²L.C. Yang and J.K. Kuchar. Prototype conflict alerting system for free flight. *Journal of Guidance, Control, and Dynamics*, 20(4):768–773, July-August 1997.
- ³R.A. Paielli and H. Erzberger. Conflict probability estimation for free flight. *Journal of Guidance, Control, and Dynamics*, 20(3):588–596, 1997.
- ⁴M. Prandini, J. Hu, J. Lygeros, and S. Sastry. A probabilistic approach to aircraft conflict detection. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):199–219, December 2000.
- ⁵Y. Bar-Shalom and T.E. Fortmann. *Tracking and Data Association*. Academic Press, 1988.
- ⁶H.A.P. Blom, R.A. Hogendoorn, and B.A. van Doorn. Design of a multisensor tracking system for advanced air traffic control. In Y. Bar-Shalom, editor, *Multitarget-Multisensor Tracking: Application and Advances*, volume 2, pages 31–63. Artech House, 1990.
- ⁷E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan. Interacting multiple model methods in tracking: A survey. *IEEE Transactions on Aerospace and Electronic Systems*, 34(1):103–123, 1998.
- ⁸D.G. Lainiotis. Partitioning: A unifying framework for adaptive systems I: Estimation. *Proceedings of the IEEE*, 64:1126–1142, August 1976.
- ⁹Y. Bar-Shalom and X.R. Li. *Estimation and Tracking: Principles, Techniques and Software*. Artech House, Boston, 1993.
- ¹⁰D.D. Sworner and J.E. Boyd. *Estimation Problems in Hybrid Systems*. Cambridge University Press, 1999.
- ¹¹H.A.P. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783, August 1988.
- ¹²X.R. Li and Y. Bar-Shalom. Design of an interacting multiple model algorithm for air traffic control tracking. *IEEE Transactions on Control Systems Technology*, 1(3):186–194, September 1993.
- ¹³A. Balluchi, L. Benvenuti, M.D. Di Benedetto, and A.L. Sangiovanni-Vincentelli. Design of observers for hybrid systems. In C. Tomlin and M.R. Greenstreet, editors, *Hybrid Systems: Computation and Control. LNCS*, volume 2289, pages 76–89. Springer-Verlag, 2002.
- ¹⁴R. Vidal, A. Chiuso, and S. Soatto. Observability and identifiability of jump linear systems. In *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 3614–3619, Las Vegas, NV, December 2002.
- ¹⁵I. Hwang, H. Balakrishnan, and C. Tomlin. Observability criteria and estimator design of stochastic linear hybrid systems. In *Proceedings of European Control Conference 2003*, Cambridge, UK, September 2003.
- ¹⁶P.D. Hanlon and P.S. Maybeck. Characterization of Kalman filter residuals in the presence of mismodeling. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 1254–1259, December 1998.
- ¹⁷I. Hwang and C. Tomlin. Protocol-based conflict resolution for air traffic control. Technical Report SUDAAR-762, Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, 2002.
- ¹⁸I. Hwang and C. Tomlin. Protocol-based conflict resolution for finite information horizon. In *Proceedings of the 2002 American Control Conference*, volume 1, pages 748–753, Anchorage, AL, May 2002.
- ¹⁹P. Vacher, I. Barret, and M. Gauvrit. Design of a tracking algorithm for an advanced ATC system. In Y. Bar-Shalom, editor, *Multitarget-Multisensor Tracking: Application and Advances*, volume 2, pages 1–29. Artech House, 1990.
- ²⁰X.R. Li and V.P. Jilkov. A survey of maneuvering target tracking: Dynamic models. In O.E. Drummond, editor, *Proceedings of SPIE: In Signal and Data Processing of Small Targets*, volume 4048, pages 212–235, 2000.
- ²¹RTCA Special Committee. Minimum aviation system performance standards for automatic dependent surveillance-broadcast (ADS-B). Technical report, January 1998.
- ²²J. Basch, L.J. Guibas, and J. Hershberger. Data structures for mobile data. In *Proceedings of 8th ACM-SIAM Symposium on Discrete Algorithms*, pages 747–756, 1997.
- ²³L.J. Guibas. Kinetic data structures: A state of the art report. In *Proceedings of 3rd Workshop on Algorithmic Foundations of Robotics*, 1998.
- ²⁴R. Teo and C. Tomlin. Computing danger zones for provably safe closely spaced parallel approaches. *Journal of Guidance, Control, and Dynamics*, May-June 2003.