JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 30, No. 2, March–April 2007

New Algorithms for Aircraft Intent Inference and Trajectory Prediction

Javier Lovera Yepes,* Inseok Hwang,† and Mario Rotea‡
Purdue University, West Lafayette, Indiana 47907

DOI: 10.2514/1.26750

Aircraft tracking, intent inference, and trajectory predictions are important tools for enhanced capacity in air traffic control operations. In this paper, we propose an algorithm that performs these three tasks accurately. The algorithm uses a hybrid estimation algorithm to estimate the aircraft's state and flight mode. These estimates are combined with knowledge about air traffic control regulations, the aircraft's flight plan, and the environment to infer the pilot's intent. Trajectory predictions are computed as a function of the aircraft's motion (state and mode estimates) and the inferred intent. The result is an algorithm that provides, in real-time, accurate intent and trajectory predictions for aircraft. We analyze and test the performance of the proposed algorithm with various scenarios representative of current and future aircraft operations within the National Airspace System.

Nomenclature

A_j, C	=	mode-matched state transition matrix and the mode-independent output matrix
$\hat{I}$	=	aircraft's estimated intent
k	=	discrete time index
l, S_l	=	prediction look-ahead time and the propagation time of the estimated states
m, $\hat{m}$	=	the flight mode and its estimate
Q_j, R	=	mode-dependent process noise covariance matrix and mode-independent measurement noise covariance matrix
T	=	sampling time
$\mathrm{TTG}(w_I)$	=	time to go to w_I
V_I, V_{ac}, $\hat{V}_{\mathrm{ac}}$	=	intent model ground speed, that of the aircraft, and its estimate
w_I, $\boldsymbol{e}_I$	=	waypoint and unit vector associated with intent model I
$\boldsymbol{w}_j$, $\boldsymbol{v}$	=	mode-dependent process noise and mode-independent measurement noise
x, $\dot{x}$, $\ddot{x}$, y, $\dot{y}$, $\ddot{y}$	=	position, velocity, and acceleration in the east and north directions
z, $\dot{z}$	=	altitude and altitude rate
$\delta\lambda_I$	=	rate of change of the intent model likelihood for intent model I
$\Lambda_j^{\mathrm{RMIMM}}$	=	jth mode-matched Kalman filter likelihood function as computed by residual-mean interacting multiple model
λ_I, Λ_I	=	intent model likelihood and the time-averaged likelihood of intent model I
μ_{ij}, μ	=	mixing probability and the mode probability vector
$\boldsymbol{\xi}$, $\hat{\boldsymbol{\xi}}$, $\boldsymbol{\zeta}$	=	aircraft state vector, its estimate, and the measured position and velocity
$\hat{\boldsymbol{\xi}}_j$, P_j	=	state estimate and the state estimation error covariance matrix computed by mode-matched Kalman filter j
π_{ij}	=	mode transition probability from mode j to mode i
ψ, γ, ψ_I, γ_I	=	heading and flight path angles for the aircraft and the intent model unit vector

Presented as Paper 5824 at the AIAA Guidance, Navigation, and Control Conference and Exhibit, San Francisco, California, 15–18 August 2005; received 26 July 2006; revision received 13 November 2006; accepted for publication 28 November 2006.

*Graduate Research Assistant, School of Aeronautics and Astronautics; jlovera1@purdue.edu. AIAA Student Member.

†Assistant Professor, School of Aeronautics and Astronautics, 315 North Grant St.; ihwang@purdue.edu. AIAA Member.

‡Professor, School of Aeronautics and Astronautics; rotea@purdue.edu. AIAA Member.

I. Introduction

THE goal of air traffic control (ATC) is to safely and efficiently manage the flow of aircraft operating within the National Airspace System (NAS). Currently, a centralized system is being used in which pilots and ground controllers interact to decide upon the future trajectory of the aircraft. Once an intent is decided, the aircraft must comply with this intent [1].

To sustain safe and efficient aircraft flow management, decision support tools that aid controllers have been developed [2–9]. These decision support tools are based on trajectory prediction algorithms because traffic advisories are dependent on the aircraft's future trajectory. Trajectory prediction algorithms are also important to develop recent concepts for future air traffic management, such as free flight [1,10–17]. In this type of system, aircraft will be able to make decisions independently, except near the terminal area. To do so without sacrificing safety, there is a need for accurate information about the future behavior of the aircraft.

The research scope of this paper is the development of tracking and prediction algorithms to accurately track maneuvering aircraft, infer the pilot's intent, and compute trajectory predictions.

Tracking of maneuvering aircraft is a challenging problem. Difficulties arise when the aircraft changes flight modes in an unanticipated manner (such as from straight flight to coordinated turns). For aircraft tracking, algorithms such as the $\alpha - \beta$ and $\alpha - \beta - \gamma$ algorithms have been used in ATC [18,19]. To improve the tracking performance of maneuvering aircraft, adaptive state estimation algorithms have been proposed [18,20–24].

Trajectory prediction methodologies can be divided into nominal, worst case, and probabilistic techniques [10,25,26]. Nominal trajectory prediction gives the aircraft position by propagating estimated states into the future along a single trajectory [2,4,8]. This methodology does not account for uncertainties in the aircraft's future behavior and therefore, its accuracy could degrade as the look-ahead time, or the time for which a prediction is made, increases. Worst-case techniques assume that an aircraft will perform any of a set of maneuvers and the worst case, defined by the application, is considered for aircraft trajectory prediction. Probabilistic methods can also be used by modeling uncertainties to describe potential changes in the future trajectory of an aircraft [27,28]. This has been

done in at least two ways: by adding a position error to a nominal trajectory [29,30], or by developing a complete set of possible future trajectories weighted by their probability of occurrence. Uncertainty about the future behavior of an aircraft can be reduced by using intent information which determines the goal a pilot is pursuing and using this in trajectory prediction. The problem of extracting such information is referred to as intent inference and has received a great deal of attention [10–17].

An intent inference algorithm (IIA) [11,12,31] models the aircraft's intent as a discrete set of intent models and uses path correlation measures to identify which intent model best describes the actual aircraft's intent. The IIA is a sophisticated algorithm whose functionality is briefly explained as follows. The finite set of intent models are ranked based on their correlation with measured aircraft motion, in the horizontal, vertical, and speed dimensions, independently. The intent model with the highest correlation is declared as the inferred intent of the aircraft at a specific point in time. The algorithm computes nominal trajectory predictions as a function of the aircraft's inferred intent in each dimension (i.e., horizontal, vertical, and speed) by joining the current position of the aircraft with the waypoint associated with the inferred intent.

The IIA has successfully increased the amount of intent information that can be obtained from a pilot's actions, flight plan, and environment information. The IIA is a remarkable algorithm, which has motivated our work. Through extensive testing, we have noticed that this algorithm may present inference delays. Also, because the IIA does not track the current mode of the aircraft, its trajectory prediction methodology may result in degraded accuracy, especially on the short term (7 min or less of look-ahead time).

We propose the intent-based trajectory prediction (IBTP) algorithm to address the limitations encountered in the trajectory predictions of the IIA and the propagation of the current estimated states. We are motivated by the possibility of creating an algorithm that exhibits the strengths of each technique. For state estimation, the IBTP algorithm uses the residual-mean interacting multiple model (RMIMM) algorithm [23,24] which provides accurate state estimates and flight-mode estimates that are used to infer the pilot's intent. Intent information is obtained by using an intent inference algorithm which ranks the intent models according to their likelihood of representing the actual pilot's intent. This likelihood is a function of the estimated states and the time to go to the waypoints associated with each of the intent models. The intent inference algorithm uses flight-mode change information provided by the RMIMM algorithm to reduce inference delays. The IBTP algorithm exploits accurate state and mode estimates, and intent information to compute nominal trajectory predictions.

We demonstrate through simulations for various test scenarios and analysis that the IBTP algorithm increases the accuracy of the intent inference and reduces inference delays observed when using the IIA. We also show that the IBTP improves on the IIA and the RMIMM algorithms in trajectory prediction over a wide range of look-ahead times.

This paper is organized as follows: Sec. II explains in detail the IBTP algorithm, Sec. III presents simulations results, and finally, Sec. IV states our conclusions.

II. Intent-Based Trajectory Prediction Algorithm

In this section, we describe the IBTP algorithm. The algorithm assumes that an aircraft trajectory can be described by flight segments or modes, such as straight lines and arcs. The aircraft position and velocity change continuously (continuous dynamics), whereas the flight segment changes discretely (discrete dynamics). This can be modeled as a hybrid dynamical system. To track the trajectory of the hybrid system, a hybrid estimation algorithm is used which can estimate aircraft states and modes simultaneously. These estimates are combined with information on the aircraft's environment, such as ATC regulations and flight plan to estimate the aircraft's intent. The IBTP algorithm assumes a set of intent models which describes the behaviors of aircraft operating within the NAS. To estimate the aircraft intent, the IBTP algorithm identifies a single intent model which is most likely to represent the actual intent of the aircraft. The IBTP algorithm combines the state and mode estimates of the aircraft and the inferred intent to accurately compute trajectory predictions.

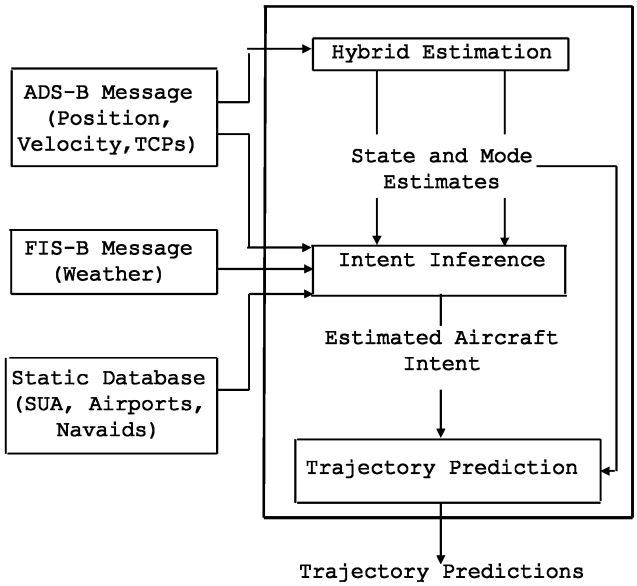


Fig. 1 Block diagram of the intent-based trajectory prediction algorithm.

The information required to execute the IBTP algorithm is assumed to be provided by the flight information services-broadcast (FIS-B) message, static databases, and the automatic surveillance-broadcast (ADS-B) message. FIS-B is a ground based data-link technology which broadcasts weather and other flight information. The static databases contain information such as special use airspace, airport locations, navigational aids, and fix locations. ADS-B is a data-link technology being deployed to communicate among aircraft their measured states (i.e., position and velocity), and flight plan information [i.e., trajectory change points (TCPs)] [11,32].

Figure 1 shows a block diagram of the IBTP algorithm. The inputs and outputs of the algorithm are displayed with arrows going into and out of the thick-lined box, respectively. The hybrid estimation block uses the ADS-B measurements to compute the state and mode estimates, i.e., the aircraft's current position, velocity, and flight mode. These are used in the intent inference block, together with environment information (FIS-B message and static database) and the flight plan information (ADS-B message) to identify an intent model. The trajectory prediction block uses the state and mode estimates together with the inferred intent to perform trajectory predictions. Note that the IBTP algorithm can be also used for the current ATC system which uses radar information instead of ADS-B because the IBTP algorithm needs only aircraft position and/or velocity measurements for intent inference and trajectory prediction.

In the remainder of this section, we describe in detail each of the blocks that compose the IBTP algorithm following the sequence displayed in Fig. 1.

A. Aircraft Dynamic Model

The dynamic models described in this section are used by the hybrid estimation block to compute the current state and mode estimates of the aircraft. The dynamic models are matched to the different flight modes of an aircraft. The modes chosen in this paper describe the flight segments commonly performed by aircraft operating in the NAS. They are as follows.

1) CV: constant velocity mode; straight flight for cruising.

2) CT: coordinated turn mode; maneuvers for heading changes with constant speed, yaw rate, and altitude.

3) CA: constant acceleration/deceleration mode without heading change; maneuvers for speed changes with constant heading and altitude.

4) CH: ascending/descending mode with constant altitude change rate and constant ground speed; maneuvers for altitude changes.

5) CHCA: ascending/descending mode with constant altitude change rate, and with constant acceleration/deceleration in the horizontal dimension; maneuvers for takeoff and landing.

The dynamics of the aircraft are characterized by the following state vector

$$\boldsymbol{\xi} = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} & \ddot{x} & \ddot{y} \end{bmatrix}^T \tag{1}$$

In each mode, the aircraft dynamics are represented by a discrete-time linear dynamic system of the form

$$\boldsymbol{\xi}(k+1) = A_j \boldsymbol{\xi}(k) + B_j \boldsymbol{w}_j(k) \tag{2a}$$

$$\boldsymbol{\zeta}(\boldsymbol{k}) = C\boldsymbol{\xi}(k) + \boldsymbol{v}(k) \tag{2b}$$

where $j \in \{\text{CV}, \text{CT}, \text{CA}, \text{CH}, \text{CHCA}\}$ denotes the mode. The mode-dependent process noise is assumed to be a Gaussian random process with the following statistics:

$$E[\boldsymbol{w}_j(k)] = 0, \qquad E[\boldsymbol{w}_j(k)\boldsymbol{w}_j(k)^T] = Q_j \tag{3}$$

The mode-independent measurement noise term is also assumed to be a Gaussian random process with the following statistics:

$$E[\boldsymbol{v}(k)] = 0, \qquad E[\boldsymbol{v}(k)\boldsymbol{v}(k)^T] = R \tag{4}$$

Now we define the mode-dependent system parameters for each of the modes previously mentioned. Let the sampling interval be set to 1 s to match the ADS-B message update rate. Let us define the following matrices useful for notational simplicity:

$$\mathcal{A}_1 = \begin{bmatrix} T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \mathcal{A}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$
$$\mathcal{A}_3 = \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ 0 & 0 \end{bmatrix}, \qquad \mathcal{A}_4 = \begin{bmatrix} T & 0 \\ 0 & T \\ 0 & 0 \end{bmatrix} \tag{5a}$$

$$\mathcal{A}_5 = \begin{bmatrix} T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix}, \qquad \mathcal{A}_6 = \begin{bmatrix} \frac{T^2}{2} & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 \\ 0 & 0 & T \end{bmatrix}$$
$$\mathcal{A}_7 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad \mathcal{A}_8 = \begin{bmatrix} T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5b}$$

Also, let $\mathcal{I}_r$ denote the identity matrix with r rows and $\mathcal{O}_{r\times c}$ denote a zero matrix with r rows and c columns. Then, for the CV mode, the system matrices are given by

$$A_{\text{CV}} = \begin{bmatrix} \mathcal{I}_3 & \mathcal{A}_1 & \mathcal{O}_{3\times 2} \\ \mathcal{O}_{3\times 3} & \mathcal{A}_2 & \mathcal{O}_{3\times 2} \\ \mathcal{O}_{2\times 3} & \mathcal{O}_{2\times 3} & \mathcal{O}_{2\times 2} \end{bmatrix}, \qquad B_{\text{CV}} = \begin{bmatrix} \mathcal{A}_6 \\ \mathcal{A}_1 \\ \mathcal{O}_{2\times 3} \end{bmatrix}$$
$$Q_{\text{CV}} = \begin{bmatrix} 0.002 & 0 & 0 \\ 0 & 0.002 & 0 \\ 0 & 0 & 0.002 \end{bmatrix} \tag{6}$$

For the CT mode, we use the Wiener-sequence acceleration model [18,19,33], that is

$$A_{\text{CT}} = \begin{bmatrix} \mathcal{I}_3 & \mathcal{A}_1 & \mathcal{A}_3 \\ \mathcal{O}_{3\times 3} & \mathcal{A}_2 & \mathcal{A}_4 \\ \mathcal{O}_{2\times 3} & \mathcal{O}_{2\times 3} & \mathcal{I}_2 \end{bmatrix}, \qquad B_{\text{CT}} = \begin{bmatrix} \mathcal{A}_6 \\ \mathcal{A}_1 \\ \mathcal{A}_7 \end{bmatrix}$$
$$Q_{\text{CT}} = \begin{bmatrix} 0.7 & 0 & 0 \\ 0 & 0.7 & 0 \\ 0 & 0 & 0.002 \end{bmatrix} \tag{7}$$

The dynamic model for the CA mode also corresponds to the Wiener-sequence acceleration model. The difference between the models for CT and CA comes from the magnitude of the process noise covariance in the components of the horizontal dimension.

$$A_{\text{CA}} = A_{\text{CT}}, \qquad B_{\text{CA}} = B_{\text{CT}}, \qquad Q_{\text{CA}} = \begin{bmatrix} 0.07 & 0 & 0 \\ 0 & 0.07 & 0 \\ 0 & 0 & 0.002 \end{bmatrix} \tag{8}$$

Note that the covariance in Eq. (8) is one order of magnitude smaller than the one in Eq. (7), except for the vertical velocity component. This is because both modes are representative of motion in the horizontal plane, without altitude changes.

For the CH mode, the model matrices are given by

$$A_{\text{CH}} = \begin{bmatrix} \mathcal{I}_3 & \mathcal{A}_5 & \mathcal{O}_{3\times 2} \\ \mathcal{O}_{3\times 3} & \mathcal{I}_3 & \mathcal{O}_{3\times 2} \\ \mathcal{O}_{2\times 3} & \mathcal{O}_{2\times 3} & \mathcal{O}_{2\times 2} \end{bmatrix}, \qquad B_{\text{CH}} = \begin{bmatrix} \mathcal{A}_6 \\ \mathcal{A}_8 \\ \mathcal{O}_{2\times 3} \end{bmatrix}$$
$$Q_{\text{CH}} = \begin{bmatrix} 0.002 & 0 & 0 \\ 0 & 0.002 & 0 \\ 0 & 0 & 0.5 \end{bmatrix} \tag{9}$$

Finally, for the CHCA mode, the model matrices are as follows:

$$A_{\text{CHCA}} = \begin{bmatrix} \mathcal{I}_3 & \mathcal{A}_5 & \mathcal{A}_3 \\ \mathcal{O}_{3\times 3} & \mathcal{I}_3 & \mathcal{A}_4 \\ \mathcal{O}_{2\times 3} & \mathcal{O}_{2\times 3} & \mathcal{I}_2 \end{bmatrix}, \qquad B_{\text{CHCA}} = \begin{bmatrix} \mathcal{A}_6 \\ \mathcal{A}_8 \\ \mathcal{A}_7 \end{bmatrix}$$
$$Q_{\text{CHCA}} = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.5 \end{bmatrix} \tag{10}$$

The ADS-B message contains measurements for position and speed in the horizontal and vertical dimensions. Therefore, the mode-independent output matrix in Eq. (2) is given by

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{11}$$

The measurement noise covariance matrix in Eq. (4) is obtained from the ADS-B system performance parameters [32]. This matrix is defined as

$$R = \begin{bmatrix} 4225\ \text{ft}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4225\ \text{ft}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 900\ \text{ft}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.67\ \frac{\text{ft}^2}{\text{s}^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.67\ \frac{\text{ft}^2}{\text{s}^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1\ \frac{\text{ft}^2}{\text{s}^2} \end{bmatrix} \tag{12}$$

Let $m(k)$ denote the flight mode at time k. The mode dynamics are modeled as a finite Markov chain with known mode-transitions probabilities [18,23]

$$\pi_{ij} \triangleq \text{Prob}\{m(k+1) = i | m(k) = j\} \tag{13}$$

from mode j at time k to mode i at time $k+1$. This Markov chain description of the aircraft's modes is used to model an unknown pilot's inputs. The Markov transition matrix Π is chosen as

$$\Pi = \begin{bmatrix} 0.9 & 0.025 & 0.025 & 0.025 & 0.025 \\ 0.1 & 0.9 & 0 & 0 & 0 \\ 0.1 & 0 & 0.9 & 0 & 0 \\ 0.1 & 0 & 0 & 0.9 & 0 \\ 0.1 & 0 & 0 & 0 & 0.9 \end{bmatrix} \tag{14}$$

The values for the entries of Π are chosen following concepts developed in [18,31]. The first row represents the mode-transition probabilities from the modes $j \in \{\text{CV}, \text{CT}, \text{CA}, \text{CH}, \text{CHCA}\}$ to the mode $i = \text{CV}$. The second row represents the mode-transition probabilities from the modes $j \in \{\text{CV}, \text{CT}, \text{CA}, \text{CH}, \text{CHCA}\}$ to the mode $i = \text{CT}$, and so on. The elements in the diagonal represent the probability of the aircraft remaining in the same mode from time k to time $k+1$. The off-diagonal entries equal to zero represent mode transitions that are not allowed in our model. For example, because the element in the second row and third column is zero ($\pi_{23} = 0$), the mode transition from the CA mode to the CT mode is not allowed. In this way, we only allow mode transitions from any mode to CV, and from CV to any mode, because this is representative of mode transitions commonly performed by aircraft. The system described by Eqs. (2) and (13) is called a stochastic linear hybrid system [34].

B. Hybrid Estimation Algorithm

The hybrid system in Eqs. (2) and (13) contains the continuous state dynamics and mode transitions. To track the trajectories of such a system, a hybrid estimation algorithm is required. This type of algorithm can estimate both the continuous state and the aircraft mode. For hybrid estimation we use the residual-mean interacting multiple model algorithm [23,24]. This algorithm is chosen among other hybrid estimation algorithms because it uses the mean of the residual of each Kalman filter to improve the differentiation of the likelihood of each filter being the correct one. As a result, false mode estimation is reduced. This is done by exploiting the fact that if the dynamic model in a given Kalman filter is not representative of the actual flight mode of the aircraft, the residual produced by this filter is not zero. A brief explanation is given subsequently.

The outputs of RMIMM are the continuous state estimate, its covariance, and the mode estimate, which are computed as [18,23,24]

$$\hat{\boldsymbol{\xi}}(k+1) = \sum_j \hat{\boldsymbol{\xi}}_j(k+1)\mu_j(k+1) \tag{15a}$$

$$P(k+1) = \sum_j \{P_j(k+1) + [\hat{\boldsymbol{\xi}}_j(k+1) - \hat{\boldsymbol{\xi}}(k+1)][\hat{\boldsymbol{\xi}}_j(k+1) - \hat{\boldsymbol{\xi}}(k+1)]^T\}\mu_j(k+1) \tag{15b}$$

$$\hat{m}(k+1) = \arg\max_j \mu_j(k+1) \tag{15c}$$

where $\hat{\boldsymbol{\xi}}_j(k+1)$ and $P_j(k+1)$ are the state estimate, and its covariance, computed by the Kalman filter matched to mode j and $\mu_j(k+1)$ is the probability of mode j being the correct one at time $k+1$. Thus, the state estimate is a weighted sum of the mode-matched state estimates, and the mode estimate is the mode that has the maximum mode probability.

RMIMM has a bank of Kalman filters matched to each mode. The Kalman filter j, matched to mode j, computes its own state estimate and its covariance as

$$\hat{\boldsymbol{\xi}}_j(k+1) = A_j\hat{\boldsymbol{\xi}}_{0j}(k) + K_j(k+1)[\boldsymbol{\zeta}(k+1) - \text{CA}_j\hat{\boldsymbol{\xi}}_{0j}(k)] \tag{16a}$$

$$P_j(k+1) = A_jP_{0j}(k)A_j^T + Q_j - K_j(k+1)[C(A_jP_{0j}(k)A_j^T + Q_j)C^T + R]K_j(k+1)^T \tag{16b}$$

where $\boldsymbol{\zeta}(k+1)$ is the measurement, $K_j(k+1)$ is the gain of Kalman filter j, and $\hat{\boldsymbol{\xi}}_{0j}(k)$ and $P_{0j}(k)$ are inputs to the Kalman filter j at time $k+1$. These inputs are computed as

$$\hat{\boldsymbol{\xi}}_{0j}(k) = \sum_i \hat{\boldsymbol{\xi}}_i(k)\mu_{ij}(k|k) \tag{17a}$$

$$P_{0j}(k) = \sum_i \{P_i(k) + [\hat{\boldsymbol{\xi}}_i(k) - \hat{\boldsymbol{\xi}}_{0j}(k)][\hat{\boldsymbol{\xi}}_i(k) - \hat{\boldsymbol{\xi}}_{0j}(k)]^T\}\mu_{ij}(k|k) \tag{17b}$$

where $\mu_{ij}(k|k)$ is the probability that mode i was in effect at time k given that mode j is in effect at time $k+1$, and given the measurement up to time k. This probability is called the mixing probability and is calculated as

$$\mu_{ij}(k|k) = \frac{1}{c_j}\pi_{ji}\mu_i(k), \quad \text{where } c_j = \sum_i \pi_{ji}\mu_i(k) \tag{18}$$

where $\mu_i(k)$ is the mode probability at time k, and $i, j \in \{\text{CV}, \text{CT}, \text{CA}, \text{CH}, \text{CHCA}\}$. The mode probability at time $k+1$, $\mu_j(k+1)$, is computed as

$$\mu_j(k+1) = \frac{c_j}{c}\Lambda_j^{\text{RMIMM}}(k+1), \quad \text{where } c = \sum_j \Lambda_j^{\text{RMIMM}}c_j \tag{19}$$

where $\Lambda_j^{\text{RMIMM}}(k+1)$ and can be computed using Eqs. (26)–(29) in Hwang et al. [23]. The initial condition $\mu_i(0)$ can be obtained from the properties of the system.

C. Intent Inference

As seen in Fig. 1, the intent inference algorithm uses the state and mode estimates, $\hat{\boldsymbol{\xi}}(k)$ and $\hat{m}(k)$, respectively, together with environment information (FIS-B message and static database) and flight plan information (ADS-B message) to estimate the aircraft's intent at time k, denoted as $I(k)$.

The set of possible intents of an aircraft at any time could have infinitely many elements. However, in ATC these intents are constrained by ATC procedures and regulations. Thus, in this paper, we create a finite set of intent models based on the idea in [11,12]. These intent models are chosen to represent most of the possible pilot's actions in the current and future NAS. For simplicity, Tables 1 and 2 present a subset of intent models developed in [12,35] corresponding to the ones used in the test scenarios in Sec. III.

The intent models are grouped into regulation related intents and flight plan related intents, denoted as Θ_{reg} and Θ_{fp}, respectively. Furthermore, they are subdivided into the different dimensions: horizontal, vertical, and speed, denoted as $\mathcal{M}_h$, $\mathcal{M}_v$, and $\mathcal{M}_s$, respectively.

Each intent model $I \in \mathcal{M}_h \cup \mathcal{M}_v$ has an associated waypoint similar to the spatial location of the TCPs and an associated unit vector in the direction of the waypoint. For example, for intent model *go-to-TCP* in Fig. 2, the associated waypoint is $w_{\text{go-to-TCP}} = \text{TCP}$ and the corresponding unit vector $\boldsymbol{e}_{\text{go-to-TCP}} = \boldsymbol{e}_{\text{TCP}}$. In the speed dimension, each intent model $I \in \mathcal{M}_s$ has an associated ground speed that the aircraft should follow to achieve the intent model's goal.

Table 1 Intent models related to ATC regulations (RNP: required navigation performance)

Regulation related intents, Θ_{reg}		
Horizontal intents, $\mathcal{M}_h$	Vertical intents, $\mathcal{M}_v$	Speed intents $\mathcal{M}_s$
Avoid aircraft	avoid aircraft	avoid aircraft
Hold course outside RNP	hold course outside RNP	——
Blundering aircraft	blundering aircraft	blundering aircraft

Table 2 Intent models related to flight plan actions (MIT: miles in trail)

Flight plan related intents, Θ_{fp}		
Horizontal intents, $\mathcal{M}_h$	Vertical intents, $\mathcal{M}_v$	Speed intents, $\mathcal{M}_s$
Go-to-TCP	hold TCP pressure altitude	speed to meet TTG TCP
Go-to-TCP + 1	hold TCP pressure altitude + 1	speed to meet TTG TCP + 1
Return to flight plan	climb/descend to TCP altitude	speed to meet MIT restriction
Blundering aircraft	blundering aircraft	blundering aircraft

The intent inference problem is posed as a maximum likelihood inference problem in which the inferred intent is the one that maximizes a likelihood-like function Ω. The likelihood-like function is composed of two factors, κ_1 and κ_2. The inferred intent is computed as

$$\hat{I} = \arg \max_{I \in \Theta_{\text{reg}} \cup \Theta_{\text{fp}}} \Omega = \arg \max_{I \in \Theta_{\text{reg}} \cup \Theta_{\text{fp}}} \kappa_1 \cdot \kappa_2 \tag{20}$$

Note that $\Theta_{\text{reg}} \cup \Theta_{\text{fp}} = \mathcal{M}_h \cup \mathcal{M}_v \cup \mathcal{M}_s$. In the following two subsections, we will discuss how to construct the factors κ_1 and κ_2. These factors are referred to as intent model likelihoods; κ_1 is based on aircraft state only whereas κ_2 is based on the time required to reach the waypoint associated with a particular intent model.

1. Intent Model Likelihood Based on Aircraft State

The intent model likelihood κ_1 is a measure of how likely an intent model is of representing the actual pilot's intent based on the current estimated states. Consider Figs. 2 and 3, which contain a simple yet illustrative problem in the horizontal dimension. The algorithm computes the intent model likelihood in the horizontal dimension by comparing the estimate of the aircraft's current heading angle, and the estimate of the heading angle of the intent models' unit vector, denoted in this case as $\hat{\psi}_{\text{tcp}}$ and $\hat{\psi}_{\text{tcp+1}}$, respectively. These angles can be computed from a unit vector pointing in the direction of the aircraft's heading, denoted as $\boldsymbol{e}_\Psi$, and from the intent model unit vectors, $\boldsymbol{e}_{\text{tcp}}$ and $\boldsymbol{e}_{\text{tcp+1}}$.

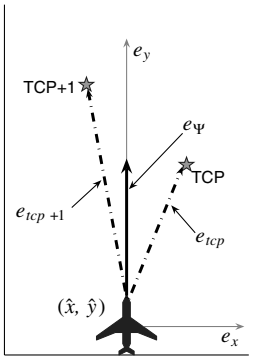


Fig. 2 Waypoints and unit vectors associated with intent models go-to-TCP and go − to − TCP + 1.

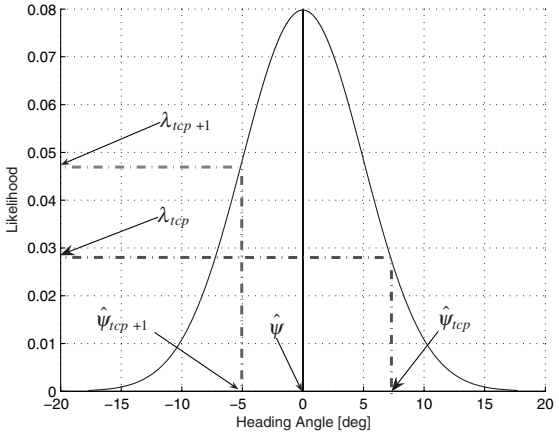


Fig. 3 Intent model likelihoods for the heading angles shown in Fig. 2.

The intent model likelihood is a function of the difference between $\hat{\psi}_I$ and $\hat{\psi}$, and the maximum intent model likelihood corresponds to the case when the aircraft's heading is perfectly aligned with waypoint w_I. As seen in Fig. 3, this can be achieved by using a normal distribution whose mean is the aircraft's heading angle, and calculating the likelihood for a given value of $\hat{\psi}_I$. In the horizontal dimension, the intent model likelihood for intent model I at time k is defined as

$$\lambda_I(k) \triangleq \mathcal{N}[\hat{\psi}_I(k); \hat{\psi}(k), \sigma_h^2], \quad \forall\ I \in \mathcal{M}_h \tag{21}$$

In Eq. (21), σ_h is a design parameter which is set to $\sigma_h = 5$ deg, guaranteeing that only those intent models whose heading angles are within ± 15 deg from the aircraft's heading will have a significant intent model likelihood. Note that 15 deg corresponds to the 3σ bound of the distribution, which contains 99% of the heading angle values with significant likelihood.

Similarly, in the vertical dimension, the algorithm compares the differences between the flight path angles of the intent models and the flight path angle of the aircraft. Thus, in the vertical dimension, the intent model likelihood is defined as

$$\lambda_I(k) \triangleq \mathcal{N}[\hat{\gamma}_I(k); \hat{\gamma}(k), \sigma_v^2], \quad \forall\ I \in \mathcal{M}_v \tag{22}$$

where $\sigma_v = 5$ deg.

In the speed dimension, the intent model likelihood is computed by comparing the differences between the estimate of the aircraft's ground speed and the ground speed associated with each of the intent models in this dimension. Then, in the speed dimension, the intent model likelihood is defined as

$$\lambda_I(k) \triangleq \mathcal{N}\left(\frac{\hat{V}_{\text{ac}}(k)}{V_I}; 1, \sigma_s^2\right), \quad \forall\ I \in \mathcal{M}_s \tag{23}$$

where $\sigma_s = 0.033$. This value guarantees the aircraft speed is at least 90% of the intent model speed, and at most 110% of it. This corresponds to the 3σ bound of the distribution, which contains 99% of the speed ratios with significant likelihood.

Because the intent model likelihoods in Eqs. (21–23) are functions of the current estimated states, they are likely to be subject to measurement noise. In the horizontal case, if the aircraft is in a nonmaneuvering mode, such as CV, its intent is fixed. However, the noisy characteristics of the intent model likelihoods may produce false intent estimates. Thus, we propose to filter the noise by computing a time-averaged likelihood:

$$\Lambda_I(k) \triangleq \frac{1}{G_k} \sum_{n=k_o}^{k} f^{k-n} \lambda_I(n), \quad \text{with} \quad G_k = \sum_{n=k_o}^{k} f^{k-n}, \qquad G_{k_o} = 1 \tag{24}$$

where f is the fading memory factor ($0 < f < 1$, $f = 1$ for no fading) used to exponentially reduce weighting on past data points, k_o is the initiation time of the current flight mode, and $\lambda_I(k)$ is defined in Eq. (21). If the whole time history of the intent model likelihood is considered, i.e., $k_o = 0$ s is the time of initiation of the whole flight, this could cause intent inference delays because $\Lambda_I(k)$ in Eq. (24) includes old intent model likelihood information. These delays could

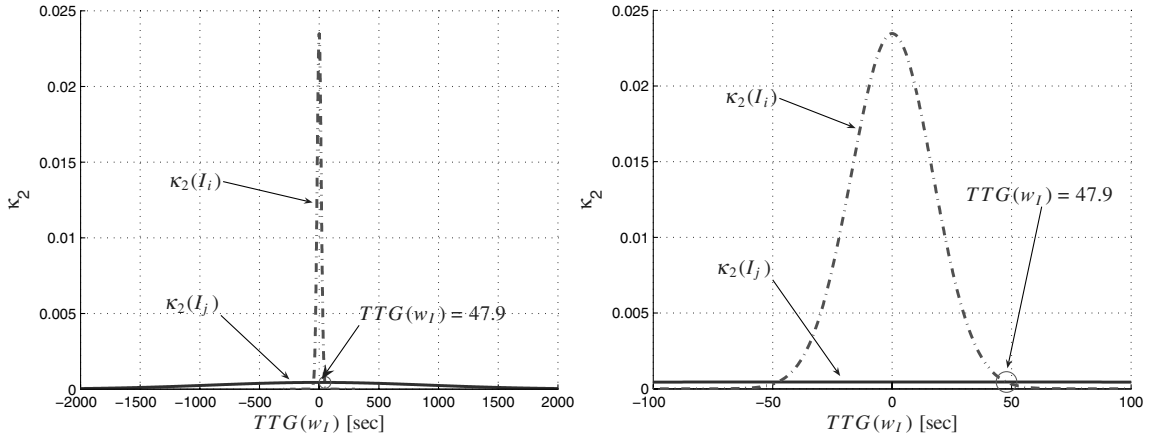


Fig. 4 Regulations and flight plan intent models, a) $\kappa_2(I)$, and b) zoom-in version of 4a.

arise particularly when the aircraft has maneuvered, and thus changed its intent. To reduce these delays, we divide the time history in segments corresponding to the duration of flight in a specific mode, and reinitialize the filter in Eq. (24) when a new flight segment is initiated. For example, if at time $k = 500$ s the aircraft changes modes from CV to CT, then $k_o = 500$ s.

We have observed through simulations that when the aircraft initiates a maneuver (e.g., CT flight) the time-averaged likelihood in Eq. (24) is not a good measure to infer the aircraft's intent in an agile fashion. Maneuvers in the horizontal dimension are detected by the RMIMM algorithm through the detection of mode transitions from CV to CT mode. That is, if $\hat{m}(k) \neq \hat{m}(k-1)$ and $\hat{m}(k) = \text{CT}$, a mode transition has occurred and the aircraft is in the maneuvering mode in the horizontal dimension. To rapidly identify the correct intent model when a maneuver is detected, we investigate the rate of change of the intent model likelihood, defined as

$$\delta\lambda_I(k) \triangleq \lambda_I(k) - \lambda_I(k-1), \quad \forall\ I \in \mathcal{M}_h \tag{25}$$

which is a measure of how fast each intent model is increasing its likelihood of being correct. If the rate of change of the intent model likelihood is negative for all intent models, the aircraft is not changing its heading toward any of the waypoints associated with the intent models in Tables 1 and 2. Then, the aircraft is declared as *blundering*. This intent model allows us to identify an aircraft that does not follow its flight plan and thus could be used for aircraft conformance monitoring.

Depending on whether the aircraft is maneuvering or not, we create a function $\kappa_1(I)$ for all intent models which is directly related to either their time-averaged likelihood for the nonmaneuvering cases, or the rate of change of the intent model likelihood for the maneuvering cases. This function is defined as

$$\kappa_1(I) \triangleq \begin{cases} \delta\lambda_I(k) & \forall\ I \in \mathcal{M}_h \cup \mathcal{M}_v \cup \mathcal{M}_s, \quad \text{if } \hat{m}(k) \neq \hat{m}(k-1) \\ \Lambda_I(k) & \forall\ I \in \mathcal{M}_h \cup \mathcal{M}_v \cup \mathcal{M}_s, \quad \text{otherwise} \end{cases} \tag{26}$$

and is a function of the estimated states of the aircraft only.

An intent inference algorithm that considers only the quantity κ_1 may fail if two intent models have their corresponding waypoints w_{I_i} and w_{I_j} closely collinear with the aircraft's position. This scenario can happen fairly often, especially when the aircraft is closely following its flight plan, and the next two TCPs are closely aligned with the aircraft heading. To prevent the algorithm from failing, we introduce another likelihood measure which is a function of temporal information.

2. Intent Model Likelihood Based on Temporal Information

The IBTP algorithm could discern between intent models whose time-averaged likelihoods or the rate of change of their intent model likelihoods are very similar in value. To do so, it ranks the intent models in the time dimension according to their value of a second weight, $\kappa_2(I)$, which contains information about how far into the future the goal related to each intent model resides, i.e., the time-to-go to w_I.

Because pilots should follow ATC regulations, it is reasonable to give more weight to intent models related to ATC regulations than to those related to the flight plan. To do so, for all $I_i \in \Theta_{\text{reg}}$ and for all $I_j \in \Theta_{\text{fp}}$, we have $\kappa_2(I_i) \geq \kappa_2(I_j)$ $(i \neq j)$ for a significant time interval. This is achieved by defining $\kappa_2(I)$ as

$$\kappa_2(I) \triangleq \begin{cases} \mathcal{N}[\text{TTG}(w_I); 0, (\sigma_{\text{reg}})^2] & \text{if } I \in \Theta_{\text{reg}} \\ \mathcal{N}[\text{TTG}(w_I); 0, (\sigma_{\text{fp}})^2] & \text{if } I \in \Theta_{\text{fp}} \end{cases} \tag{27}$$

where $\sigma_{\text{reg}} > \sigma_{\text{fp}}$.[§] We set $\sigma_{\text{reg}} = 900$ s, which corresponds to the maximum time for which a potential conflict is investigated [11,12]. The time it would take an aircraft flying at a ground speed of 673 ft/s (e.g., the Boeing 747-200) to travel 5 n mile is 48 s, approximately. This distance corresponds to the minimum en route separation requirement for safety. Outside the 5 n mile range to a hazard, more weight is given to intent models related to ATC regulations than to those related to the flight plan. To achieve this, it is necessary that for all $I_j \in \Theta_{\text{reg}}$ and for all $I_i \in \Theta_{\text{fp}}$, $\kappa_2(I_j) \geq \kappa_2(I_i)$ for all times between 47.9 s $<$ TTG$(w_{I_j}) <$ 900 s. To satisfy this condition $\sigma_{\text{fp}} = 17$ s. As seen in Fig. 4, the intersection between $\kappa_2(I_j)$ and $\kappa_2(I_i)$ occurs at 47.9 s.

3. Intent Inference Cost Function and Optimization Problem

Given $\kappa_1(I)$ in Eq. (26) and $\kappa_2(I)$ in Eq. (27), the intent models are ranked according to their value of the cost function

$$\Omega(I) = \kappa_1(I) \cdot \kappa_2(I) \tag{28}$$

Note that $\Omega(I)$ incorporates spatial information about the likelihood of an intent as a function of the aircraft's states and the environment information, through $\kappa_1(I)$. It also includes temporal information and the type of the intent, through $\kappa_2(I)$. The intent model with the maximum $\Omega(I)$ will be declared as the inferred intent at time k. That is,

$$\hat{I}(k) = \arg \max_{I \in \mathcal{M}_h \cup \mathcal{M}_v \cup \mathcal{M}_s} \Omega(I) \tag{29}$$

We now analyze the performance of the intent inference algorithm for the nonmaneuvering case in the horizontal dimension, i.e., CV

[§]Because TTG cannot be negative, the Gaussian distribution in Eq. (27) can be considered as a truncated Gaussian, i.e., $\mathcal{N}(x; 0, \sigma^2) = 0$ if $x < 0$.

flight. This analysis presents results on the performance of the intent inference algorithm as a function of the estimation errors.

4. Performance Analysis of the Intent Inference Algorithm in the Horizontal Dimension

Let k_{cv} denote the time instant at which the aircraft initiated the current CV flight segment, then $k_o = k_{\text{cv}}$ in Eq. (24). Suppose that intent model i correctly describes the aircraft's current intent, that is $I(k) = I_i$. Then, the intent inference algorithm using the likelihood function in Eq. (26) (i.e., spatial information only) can correctly identify the actual aircraft's intent at time k if

$$\Lambda_{I_i}(k) > \Lambda_{I_j}(k), \quad \forall\ I_j \in \mathcal{M}_h - \{I_i\} \tag{30}$$

For simplicity in the analysis, we use the log-likelihood function because it contains the same properties and information as the likelihood function itself [35]. Thus, we define the log-likelihood for the intent model at time k and a new time-averaged likelihood measure as

$$\ell_n\, \lambda_I(k) \triangleq \ell_n \frac{1}{\sigma_h\sqrt{2\pi}} + \left(\frac{-[\hat{\psi}_I(k) - \hat{\psi}(k)]^2}{2\sigma_h^2}\right) \tag{31a}$$

$$\Gamma_I(k) \triangleq \frac{1}{G_k} \sum_{n=k_{\text{cv}}}^{k} f^{k-n}\, \ell_n\, \lambda_I(n) \tag{31b}$$

An equivalent expression to Eq. (30) using $\Gamma_I(k)$ is

$$\Gamma_{I_i}(k) > \Gamma_{I_j}(k), \quad \forall\ I_j \in \mathcal{M}_h - \{I_i\} \tag{32}$$

A sufficient condition for Eq. (32) to be satisfied is that

$$|\hat{\psi}_{I_i}(n) - \hat{\psi}(n)| < |\hat{\psi}_{I_j}(n) - \hat{\psi}(n)|, \quad \forall\ n \in \{k_{\text{cv}}, k_{\text{cv}}+1, \ldots, k\} \tag{33}$$

Note that condition (33) is a function of the estimated quantities, which are dependent on the estimated position and velocity and are subject to estimation errors.

Let $\Delta\psi(k)$, $\Delta\psi_{I_i}(k)$, and $\Delta\psi_{I_j}(k)$ denote the estimation errors for the angles ψ, ψ_{I_i}, and ψ_{I_j} respectively. Then,

$$\Delta\psi(k) \triangleq \psi(k) - \hat{\psi}(k), \qquad \Delta\psi_I(k) \triangleq \psi_I(k) - \hat{\psi}_I(k) \tag{34}$$

Let the aircraft's actual position and velocity vectors and their estimates be denoted as

$$\boldsymbol{v}(\boldsymbol{k}) = \begin{bmatrix} \dot{x}(k) & \dot{y}(k) \end{bmatrix}^T, \qquad \boldsymbol{p}(\boldsymbol{k}) = \begin{bmatrix} x(k) & y(k) \end{bmatrix}^T \tag{35a}$$

$$\hat{\boldsymbol{v}}(\boldsymbol{k}) = \begin{bmatrix} \hat{\dot{x}}(k) & \hat{\dot{y}}(k) \end{bmatrix}^T, \qquad \hat{\boldsymbol{p}}(\boldsymbol{k}) = \begin{bmatrix} \hat{x}(k) & \hat{y}(k) \end{bmatrix}^T \tag{35b}$$

Figure 5a shows a geometrical description for the relationship between $\Delta\psi$ and the estimation error vector for velocity denoted as $\boldsymbol{\epsilon_v(k)}$. Figure 5b shows a relationship between the estimation error vector for position, denoted as $\boldsymbol{\epsilon_p(k)}$, and $\Delta\psi_{I_j}$. Expressions for $\Delta\psi$ and $\Delta\psi_{I_j}$ can be found as a function of $\boldsymbol{\epsilon_v(k)}$ and $\boldsymbol{\epsilon_p(k)}$ using the laws of sines and cosines:

$$\begin{aligned} \Delta\psi(k) &= \sin^{-1}\left(\frac{\| \boldsymbol{\epsilon_v(k)} \| \sin\beta}{\| \boldsymbol{v(k)} \|}\right) \\ \Delta\psi_I(k) &= \sin^{-1}\left(\frac{\| \boldsymbol{\epsilon_p(k)} \| \sin\varphi_I}{\| \phi_I(k) \|}\right) \end{aligned} \tag{36}$$

Using the estimated heading angles and their estimation errors in Eq. (36), condition (33) can be written as

$$\begin{aligned} &|\psi_{I_i}(n) - \psi_{I_j}(n)| > \Delta\psi_{I_i}(n) + \Delta\psi_{I_j}(n) + 2\Delta\psi(n), \\ &\quad \forall\ n \in \{k_{\text{cv}}, k_{\text{cv}}+1, \ldots, k\} \end{aligned} \tag{37}$$

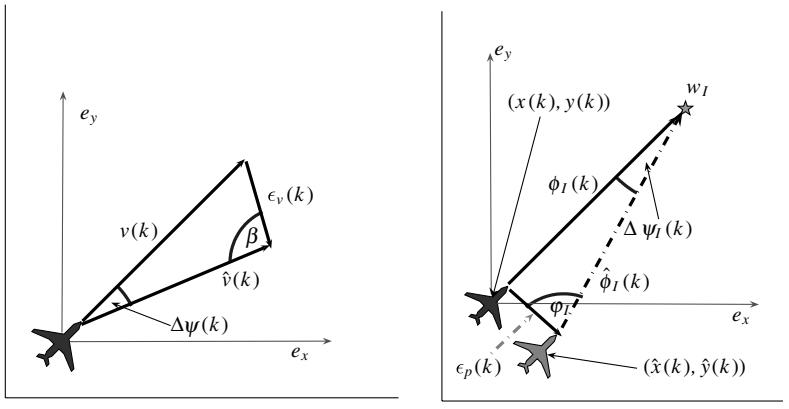
Let us consider the worst case in which the right-hand side of Eq. (37) attains the maximum value. To this end, we seek expressions for the maximum values of $\Delta\psi_{I_i}$, $\Delta\psi_{I_j}$, and $\Delta\psi$, and examine the properties of the estimation errors in position and velocity. If the estimated mode is correct, then the mean of the estimation error converges exponentially to zero [23]. Let $\hat{\boldsymbol{\xi}}(k)$ denote the estimated state vector at time k and $\boldsymbol{\epsilon(k)}$ denote the estimation error vector. Then the error ellipsoid, defined by

$$\boldsymbol{\epsilon}(\boldsymbol{k})^T P^{-1}(k) \boldsymbol{\epsilon(k)} = 1 \tag{38}$$

will be centered at the actual aircraft's position [36–38], where $P(k)$ is the estimation error covariance matrix, defined in Eq. (15). Assuming that the estimation error for position is uncorrelated with the estimation error for velocity, $P(k)$ becomes a block diagonal matrix of the form

$$P(k) = \begin{bmatrix} P_{\text{pos}}(k) & 0 \\ 0 & P_{\text{vel}}(k) \end{bmatrix} \tag{39}$$

where $P_{\text{pos}}(k)$ and $P_{\text{vel}}(k)$ denote the estimation error covariance matrices for position and velocity, respectively. If we assume that the position estimation error in the vertical dimension is uncorrelated with those of the horizontal dimension, the major and middle axes of the error ellipsoid will be located in the horizontal plane. This follows because in CV flight, the standard deviation for the estimation error in the vertical dimension is smaller than those for the errors in the horizontal dimension. Paielli and Erzberger [36] have assumed that this ellipsoid has its major and middle axes aligned with the along-track and cross-track directions, respectively. We do not make this assumption, and therefore allow a more general description of the error ellipsoid.



a) $\Delta\psi$ **b) $\Delta\psi_I$**

Fig. 5 Heading estimation error for a) the aircraft, and b) the intent model.

Consider the error ellipsoid

$$\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k})^T(3P_{\text{pos}})^{-1}(k)\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k}) = 1 \tag{40}$$

which corresponds to the 3σ bound and thus contains 99% of the estimation realizations at time k [35]. The length of its semimajor axis, denoted as $\|\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k})\|_{\max}$, can be defined as [37,38]

$$\|\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k})\|_{\max} \triangleq \sqrt{\bar{\tau}[3P_{\text{pos}}(k)]} \tag{41}$$

where $\bar{\tau}[3P_{\text{pos}}(k)]$ denotes the maximum eigenvalue of the estimation error covariance matrix $3P_{\text{pos}}(k)$. The maximum estimation error for position in the horizontal dimension is defined to be $\|\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k})\|_{\max}$. Therefore, the maximum estimation errors for the heading angles of the aircraft and of the intent unit vector corresponding to the intent model, denoted as $\Delta\bar{\psi}$ and $\Delta\bar{\psi}_I$, respectively, are computed using Eq. (36) as

$$\begin{aligned}\Delta\bar{\psi}(k) &= \sin^{-1}\left(\frac{\|\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k})\|_{\max}\sin\beta}{\|v(k)\|}\right)\\ \Delta\bar{\psi}_I(k) &= \sin^{-1}\left(\frac{\|\boldsymbol{\epsilon}_{\boldsymbol{v}}(\boldsymbol{k})\|_{\max}\sin\varphi_I}{\|\phi_I(k)\|}\right)\end{aligned} \tag{42}$$

Proposition 1: Suppose the aircraft is flying in CV mode at time k and that the mode estimate is correct. That is $\hat{m}(k) = m(k) = \text{CV}$. Let the true aircraft's intent be represented by intent model i, $I(k) = I_i$. Then, with a 99% confidence level, an intent inference algorithm that considers the value of the time-averaged likelihood $\Lambda_I(k)$ only, can identify the true intent of the aircraft in the horizontal dimension if

$$\begin{aligned}&|\psi_{I_i}(n) - \psi_{I_j}(n)| > \sin^{-1}\left(\frac{\|\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k})\|_{\max}\sin\varphi_{I_j}}{\|\phi_{I_j}(n)\|}\right)\\ &\quad+\sin^{-1}\left(\frac{\|\boldsymbol{\epsilon}_{\boldsymbol{p}}(\boldsymbol{k})\|_{\max}\sin\varphi_{I_i}}{\|\phi_{I_i}(n)\|}\right)+2\sin^{-1}\left(\frac{\|\boldsymbol{\epsilon}_{\boldsymbol{v}}(\boldsymbol{k})\|_{\max}\sin\beta}{\|\boldsymbol{v}(n)\|}\right)\\ &\quad=\Delta\bar{\psi}_{I_i}(n)+\Delta\bar{\psi}_{I_j}(n)+2\Delta\bar{\psi}(n)\\ &\quad\forall\ I_j\in\mathcal{M}_h-\{I_i\};\quad\forall\ n\in\{k_{\text{cv}},k_{\text{cv}}+1,\ldots,k\}\end{aligned} \tag{43}$$

Proposition 1 gives an important insight to the intent inference problem. Specifically, an intent inference algorithm that considers only the value of Λ_I will fail if w_{I_i} and w_{I_j} are closely collinear with the aircraft's position. In this case, the heading angles associated with their corresponding intent unit vector will be very close in value, making the left-hand side of Eq. (43) equal or very close to zero. Hence, the inequality is hard to be satisfied, thus justifying the use of $\kappa_2(I)$. Until now we have presented analytical results for the nonmaneuvering case. We now shift our attention to the maneuvering case which in the horizontal dimension corresponds to CT mode.

Recall that when the aircraft is maneuvering, the algorithm infers the aircraft's intent as a function of the rate of change of the intent model likelihood, denoted as $\delta\lambda_I$ for all $I \in \mathcal{M}_h$. Let us assume that $m(k) = m(k-1) = \text{CT}$, i.e., if there is no mode transition at time k. Suppose that intent model i correctly describes the actual intent of the aircraft at time k, that is $I(k) = I_i$. The intent inference algorithm will correctly identify the actual intent based on the rate of change of the intent model likelihoods if

$$\begin{aligned}&\delta\lambda_{I_i}(k) > \delta\lambda_{I_j}(k) \Rightarrow \lambda_{I_i}(k) - \lambda_{I_i}(k-1) > \lambda_{I_j}(k) - \lambda_{I_j}(k-1),\\ &\quad\forall\ I_j\in\mathcal{M}_h-\{I_i\}\end{aligned} \tag{44}$$

Inequality (44) is a function of the estimated quantities and their associated estimation errors. To define the worst-case scenario, let inequality (44) be expressed as

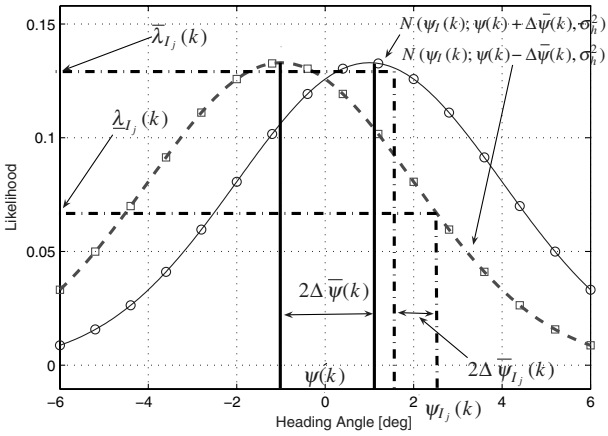


Fig. 6 Maximum and minimum intent model likelihoods.

$$\underline{\lambda}_{I_i}(k) - \bar{\lambda}_{I_i}(k-1) > \bar{\lambda}_{I_j}(k) - \underline{\lambda}_{I_j}(k-1),\quad\forall\ I_j\in\mathcal{M}_h-\{I_i\} \tag{45}$$

where $\underline{\lambda}_I(k)$ and $\bar{\lambda}_I(k)$ are defined using Eq. (42) as

$$\underline{\lambda}_I(k) = \mathcal{N}\{|\psi_I(k)| + [\Delta\bar{\psi}_I(k) + \Delta\bar{\psi}(k)]; |\psi(k)|, \sigma_h^2\} \tag{46a}$$

$$\bar{\lambda}_I(k) = \mathcal{N}\{|\psi_I(k)| - [\Delta\bar{\psi}_I(k) + \Delta\bar{\psi}(k)]; |\psi(k)|, \sigma_h^2\} \tag{46b}$$

as shown in Fig. 6. For notational simplicity, let

$$a := |\psi_{I_i}(k) - \psi(k)|, \qquad b := \Delta\bar{\psi}_{I_i}(k) + \Delta\bar{\psi}(k) \tag{47a}$$

$$\begin{aligned}c &:= |\psi_{I_i}(k-1) - \psi(k-1)|\\ d &:= \Delta\bar{\psi}_{I_i}(k-1) + \Delta\bar{\psi}(k-1)\end{aligned} \tag{47b}$$

$$e := |\psi_{I_j}(k) - \psi(k)|, \qquad f := \Delta\bar{\psi}_{I_j}(k) + \Delta\bar{\psi}(k) \tag{47c}$$

$$\begin{aligned}g &:= |\psi_{I_j}(k-1) - \psi(k-1)|\\ h &:= \Delta\bar{\psi}_{I_j}(k-1) + \Delta\bar{\psi}(k-1)\end{aligned} \tag{47d}$$

We perform the small perturbation analysis. Let

$$\begin{aligned}\Delta\bar{\psi}(k-1) = \Delta\bar{\psi}_{I_i}(k-1) = \Delta\bar{\psi}_{I_j}(k-1) = 0\\ \Delta\bar{\psi}(k) \approx \Delta\bar{\psi}_{I_i}(k) \approx \Delta\bar{\psi}_{I_j}(k) \ll 1\end{aligned}$$

which corresponds to estimation errors at time $k-1$ equal to zero, and those at time k small. This results in $d = h = 0$, $b \ll 1$, and $f \ll 1$ according to Eq. (47). Then any quadratic terms that are functions of $\Delta\bar{\psi}(k)$, $\Delta\bar{\psi}_{I_i}(k)$, or $\Delta\bar{\psi}_{I_j}(k)$ will be close to zero. This yields the following Proposition.

Proposition 2: Suppose that the aircraft is flying in CT mode at time k, and that the mode estimate is correct, that is $\hat{m}(k) = m(k) = \text{CT}$. Assume that there is no mode transition at time k, and that the mode estimate is correct at time $k-1$. Suppose that intent model i correctly describes the actual intent of the aircraft at time k, that is $I(k) = I_i$. Assume small estimation errors, that is

$$\begin{aligned}\Delta\bar{\psi}(k-1) = \Delta\bar{\psi}_{I_i}(k-1) = \Delta\bar{\psi}_{I_j}(k-1) = 0\\ \Delta\bar{\psi}(k) \approx \Delta\bar{\psi}_{I_i}(k) \approx \Delta\bar{\psi}_{I_j}(k) \ll 1\end{aligned}$$

Then, the intent inference algorithm can identify the actual intent of the aircraft in the horizontal dimension based on the rate of change of

the intent model likelihoods if

$$\frac{\exp(g^2+e^2)}{\exp(a^2+c^2)} > \exp(2ab)\left[\frac{\exp(g^2+2ef)-\exp(e^2)}{\exp(c^2)-\exp(a^2+2ab)}\right] \quad (48)$$

where b and f are functions of the estimation errors, $\Delta\bar{\psi}$, $\Delta\bar{\psi}_{I_i}$, and $\Delta\bar{\psi}_{I_j}$ at time k, and a, c, g, and e are defined in Eq. (47).

Because it is assumed that $\delta\lambda_{I_i}(k) > 0$ for nonblundering aircraft, it follows that $|c| > |a|$ because $|\psi_{I_i}(k-1) - \psi(k-1)| > |\psi_{I_i}(k) - \psi(k)|$. This will not hold as the term $2ab$ increases its value, which is a linear function of b, i.e., as the estimation error for the heading angles at time k increases in value. In the asymptotic case, the right-hand side of Eq. (48) will tend to infinity as $c^2 \to a^2 + 2ab$. This corresponds to an aircraft with a very small turn rate. In this case, the algorithm tends to fail. Because it is assumed that $\delta\lambda_{I_j}(k) > 0$ for nonblundering aircraft, it follows that $|g| > |e|$. Then, the term $\exp(g^2+2ef) - \exp(e^2)$ grows exponentially as a function of f, which represents the estimation errors for the heading angles at time k. Hence, the algorithm tends to fail as the estimation errors for the heading angles at time k increase, which are dependent on the 2-norm of the estimation error covariance matrices as seen from Eq. (36).

In the horizontal dimension, we assume that the mode estimate at time k is correct and corresponds to CV or CT, that is $\hat{m}(k) \in \{\text{CV}, \text{CT}\}$. Then, for the nonmaneuvering case in steady-state conditions, the mode probability for CV, $\mu_{\text{CV}} \approx 1$, and $\mu_m \approx 0$ for $m \neq \text{CV}$. The residual for the CV mode-matched Kalman filter is close to zero, that is $\hat{\boldsymbol{\xi}}_{\text{CV}}(k) - \hat{\boldsymbol{\xi}}(k) \approx 0$. Therefore, according to Eq. (15), $P(k)$ is approximately equal to the estimation error covariance of the CV mode-matched Kalman filter, $P_{\text{CV}}(k)$. For a single Kalman filter in steady-state conditions, the estimation error converges to a function of the process noise covariance [38]. Thus, $P(k)$ will converge to a function of the process noise covariance for CV mode, denoted as Q_{CV}. The RMIMM algorithm allows the use of smaller process noise covariance for each mode than the one necessary for the implementation of a single Kalman filter [38]. Because we reduce the value of this norm, we guarantee smaller estimation errors in the heading angles, favoring the intent inference algorithm performance in the nonmaneuvering case. This analysis is also valid for the maneuvering case. The performance analysis of the intent inference algorithm in the vertical and speed dimensions can be done in a similar way [39].

D. Trajectory Prediction

The inputs to the trajectory prediction block in Fig. 1 are the state and mode estimates, computed in the hybrid estimation block, and the inferred intent, computed in the intent inference block. The trajectory prediction block also uses the FIS-B data, the static database, and the ADS-B message.

The state and mode estimates represent the current motion of the aircraft and the inferred intent represents the goal the aircraft is pursuing. The combination of these two sets of information could represent the future trajectory of the aircraft until the goal associated with the inferred intent is achieved, that is, until the predicted position of the aircraft reaches $w_{\hat{I}}$. This can be seen in Fig. 7. In this figure, the aircraft is flying in CT mode toward $w_{\hat{I}}$. It can be seen that trajectory predictions calculated using only the current aircraft state and mode estimates result in degraded prediction accuracy as the look-ahead time increases. This is the result of ignoring the aircraft's intent information, which characterizes the long-term trajectory of the aircraft. On the other hand, predicting the aircraft trajectory using intent information only, results in degraded accuracy for small look-ahead times. This is the result of ignoring the aircraft's current motion.

The IBTP algorithm exploits the strengths of each methodology: it combines the state and mode estimates with the inferred intent information to compute trajectory predictions, which are composed of two segments. The first segment is calculated by propagating the state estimates obtained from the hybrid estimation block until a junction point, which is located at a position S_l s into the future. The second segment, corresponding to look-ahead times $l \geq S_l$, is computed by projecting the predicted position of the aircraft at S_l with a straight line up to the waypoint associated with the inferred intent, namely $w_{\hat{I}}$. Calculating an appropriate value for S_l is not a trivial problem. This value is a function of the estimated mode of operation and the location of $w_{\hat{I}}$ with respect to the aircraft. The maximum value for S_l is a design parameter. In this paper, we set this value to be

$$S_l^{\max} = \begin{cases} 500 \text{ s} & \text{if } \hat{m}(k) \in \{\text{CV}, \text{CA}, \text{CH}, \text{CHCA}\} \\ 60 \text{ s} & \text{if } \hat{m}(k) = \text{CT} \end{cases} \quad (49)$$

These values correspond to steady-state and transient conditions, respectively.

If the estimated mode of the aircraft is $\hat{m}(k) \in \{\text{CV}, \text{CA}, \text{CH}, \text{CHCA}\}$, then the aircraft's heading and flight path angle remains constant. If $w_{\hat{I}}$ is aligned with the aircraft heading and flight path angle, then there is no information that indicates that the pilot will change its flight mode. In this situation, it is appropriate to propagate the estimated states for a maximum look-ahead time of 8 min (500 s), which corresponds to the first segment of the prediction. Note that for CT mode, because the aircraft is assumed to perform coordinated turns in cruise at a yaw rate of $\dot{\psi} = \pm 1.5 \frac{\text{deg}}{\text{s}}$, after $S_l^{\max} = 60$ s the total heading change will be ± 90 deg. We assume that heading change maneuvers greater than ± 90 deg in cruise are highly unlikely because that would result in a complete course change for the aircraft, which is only performed in emergency situations, for airborne holding, or around the terminal area. We now explain the trajectory prediction methodology.

Let $\hat{x}_p(k + S_l)$ denote the predicted position of the aircraft at time $k + S_l$. The trajectory prediction algorithm propagates the estimated state vector until one of the following conditions is met:

1) S_l is the time at which $\hat{x}_p(k + S_l) = w_{\hat{I}}$.

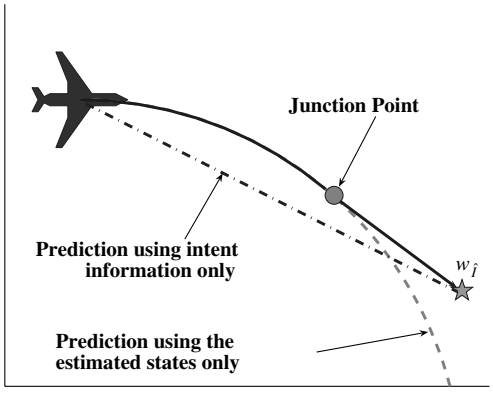


Fig. 7 Aircraft trajectory prediction as a function of the current motion and the intent of the aircraft.

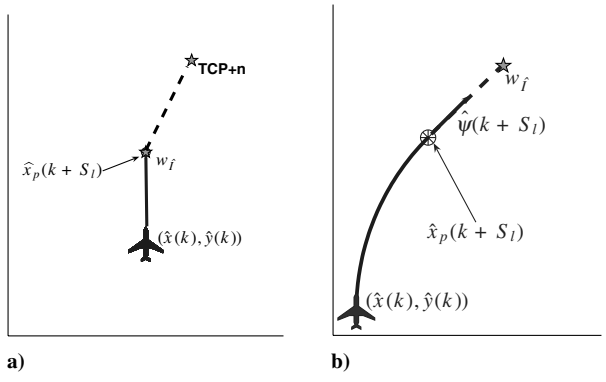
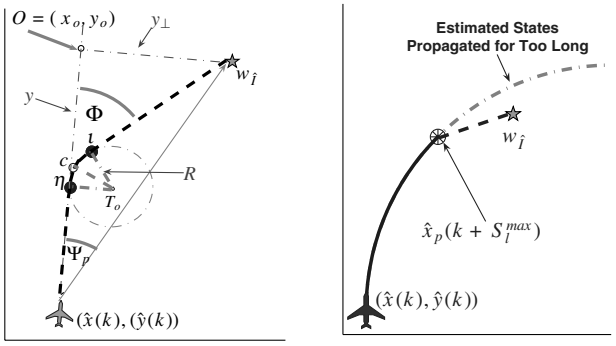


Fig. 8 CV flight with $w_{\hat{I}}$ straight ahead. Comparison of a) condition 1, and b) condition 2.



a) b)

Fig. 9 Comparison of a) condition 3 (modeling the pilot's reaction time), and b) condition 4.

2) S_l is the time at which the aircraft's heading at time $k + S_l$, denoted as $\hat{\psi}(k + S_l)$, is pointing toward $w_{\hat{i}}$.

3) If the aircraft is in CV mode, and $w_{\hat{i}}$ is to the side, $S_l \leq 5$ s.

4) $S_l = S_l^{\max}$; at this point we join $\hat{x}_p(k + S_l^{\max})$ with $w_{\hat{i}}$ using a straight line.

Figure 8a represents condition 1 in the horizontal dimension and CV flight. The solid line represents the propagation of the estimated states, which is done until $\hat{x}_p(k + S_l)$ reaches $w_{\hat{i}}$. To extend the look-ahead time of the prediction, we do the following: if the algorithm has information about future TCPs which are beyond $w_{\hat{i}}$, denoted in Fig. 8 as TCP + n, the prediction is extended until these points using straight lines. This is done unless the aircraft is violating ATC regulations, and is represented by the dashed line in Fig. 8a.

Figure 8b represents condition 2. The IBTP algorithm propagates the estimated states until a point S_l s into the future where the predicted aircraft heading aligns with $w_{\hat{i}}$. The prediction look-ahead time is extended by joining $\hat{x}_p(k + S_l)$ and $w_{\hat{i}}$ using a straight (dashed) line to incorporate the inferred intent information, as mentioned before.

As soon as pilots notice that the aircraft is off-course, they will usually correct their heading and proceed toward their next waypoint. A problem lies on how to model the time it takes a pilot to notice this course deviation. We propose a solution to this problem by using the geometric construction shown in Fig. 9a. In this figure, Φ is an angle that defines the location of a point of interest η, which is located straight ahead of the aircraft, and Ψ_p denotes the angle difference between $\hat{\psi}$ and $\psi_{\hat{i}}$. The trajectory prediction is computed by propagating the estimated states until the predicted position of the aircraft is at η, and then turning the aircraft so that its heading aligns with $w_{\hat{i}}$. In Fig. 9a, the trajectory prediction is represented by the dashed line. Thus, the pilot's reaction time (value of S_l) is modeled as a function of Φ. For example, setting $\Phi = \Psi_p$ is equivalent to a pilot initiating a turn maneuver immediately, because the point η would coincide with the aircraft's current estimated position, denoted as $[\hat{x}(k), \hat{y}(k)]$, such that $S_l = 0$. Currently we use $\Phi = \Psi_p + 5$ deg, which corresponds to a reaction time of less than 5 s. However, this is a design parameter that can be adjusted. To find the location of η, simple geometric functions are used [39]. There might be a case in which the aircraft's heading will not align with $w_{\hat{i}}$ if the estimated yaw rate is maintained throughout the maneuver. In this case, the algorithm cuts the propagation of the estimated states after $S_l^{\max}$ s and then joins the predicted position at that time with $w_{\hat{i}}$ using a straight line. This is represented in Fig. 9b. In this way, we prevent the prediction from diverging and not passing through $w_{\hat{i}}$. With this case, we conclude the explanation of the trajectory prediction block.

III. Simulation Results

In this section, we test the performance of the RMIMM algorithm, the IIA, and the IBTP algorithm using two scenarios representative of current and future air traffic operations within the NAS. In these scenarios, the measurements and flight plan information (TCPs) for each aircraft are assumed to come from the ADS-B message. To generate the accurate ADS-B message and the static database, we use a 6-DOF simulation of a Boeing 747-200 aircraft, combined with various levels of atmospheric turbulence.

Table 3 RMS position and velocity errors for maneuvering aircraft

	Average RMS errors		Largest RMS errors	
	Position, ft	Velocity, ft/s	Position, ft	Velocity, ft/s
IIA	70.0	1.50	71.0	1.50
IBTP	5.5	0.30	8.6	0.60

Figure 10 presents a scenario corresponding to a pairwise noncooperative aircraft avoidance scenario. In this scenario, the maneuvering aircraft, identified as id50, performs a series of CT flights followed by CV flights to avoid aircraft id49. Aircraft id49 maintains its course and speed, following its flight plan exactly.

A. Tracking Performance

We now compare the tracking performance of the IIA [11,12] with that of the IBTP algorithm. The comparison is made using a 50-trial Monte Carlo simulation. This is performed for each aircraft; however, only the results for the maneuvering aircraft are included, because the estimation errors are higher when maneuvers are executed along the trajectory.

Table 3 contains the average and largest rms values of the position and velocity tracking errors for the aircraft avoidance scenario, where a significant reduction of the average and largest rms errors indicates increased performance in state estimation. From Table 3, it is clear that the IBTP algorithm significantly reduces the IIA's total average rms and largest rms errors in position and velocity. Similar results have been obtained for many different scenarios, including ones discussed in this section.

B. Intent Inference Performance Comparison

Now we compare the intent inference performance of the IIA with that of the IBTP algorithm. The comparison is made by using a Monte Carlo simulation: for each scenario, we feed a set of 50 measured trajectories into the IIA and the IBTP algorithm to compute 50 sets of intent model likelihoods. These intent model likelihoods are averaged for each intent model, and the intent inference algorithm is executed based on these averaged intent model likelihoods. It is expected from the tracking performance results and propositions 1 and 2 in Sec. II that the IBTP algorithm will outperform the IIA in the accuracy of the intent inference for both the maneuvering and nonmaneuvering cases.

Figure 11 includes the intent inference results in the horizontal dimension for the maneuvering aircraft in Fig. 10. Three time traces are shown to represent the actual intent of the aircraft (true), the

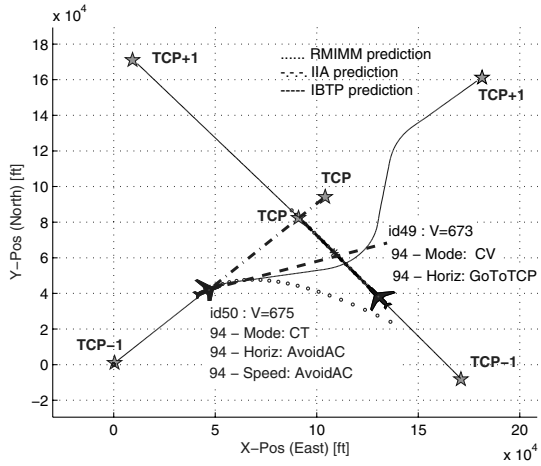
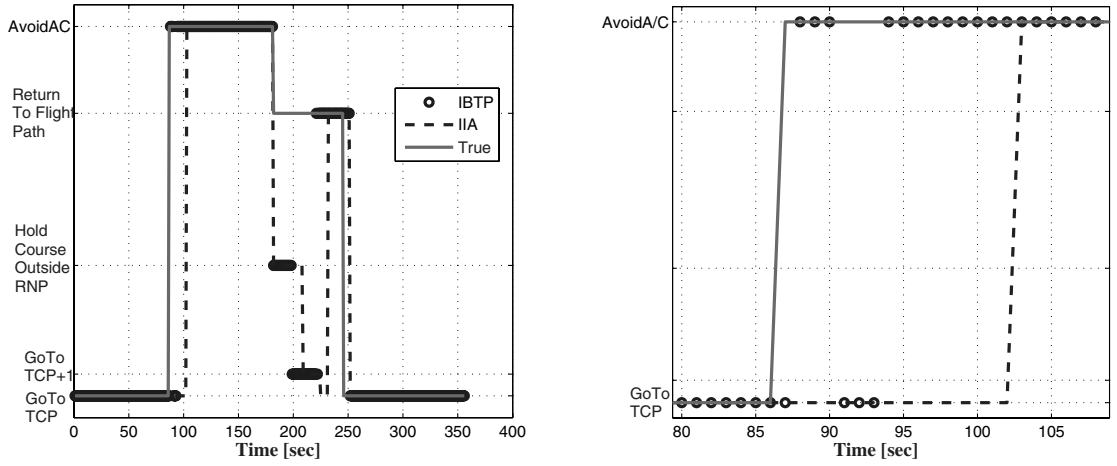


Fig. 10 Horizontal aircraft (AC) avoidance scenario and trajectory prediction comparison at time $k = 94$ s.



a) Inferred intent horizontal **b) Inferred intent horizontal (zoom–in)**

Fig. 11 Intent inference results for aircraft id50 in the aircraft avoidance scenario.

inferred intent by the IIA, and the inferred intent by the IBTP algorithm. It is important to note that the signal jumps simply represent changes of intent. The first CT flight is initiated at time $k = 88$ s. At this time, the actual intent of the aircraft in the horizontal dimension changes to *avoid-aircraft*. It can be seen in Fig. 11b that by using $\delta\lambda_I$ the IBTP algorithm is able to identify this intent with a delay reduction of 15 s over the IIA. However, during this time interval, the short period transitions seen in IBTP's intent inference are due to the sensitivity added by $\delta\lambda_I$. A delay reduction of 12 s (220 s $< k <$ 232 s) can also be seen when the aircraft is turning to return to its flight plan. In this case, the IIA incorrectly infers the aircraft's intent by using only the time-averaged likelihood of the different intent models.

C. Trajectory Prediction Comparison

Now the trajectory prediction results for the two scenarios are discussed. The comparison is made by feeding a set of 50 measured trajectories into the IIA, the RMIMM, and the IBTP algorithms to compute the trajectory predictions. The prediction times correspond roughly to the initiation time of the maneuver to avoid the aircraft.

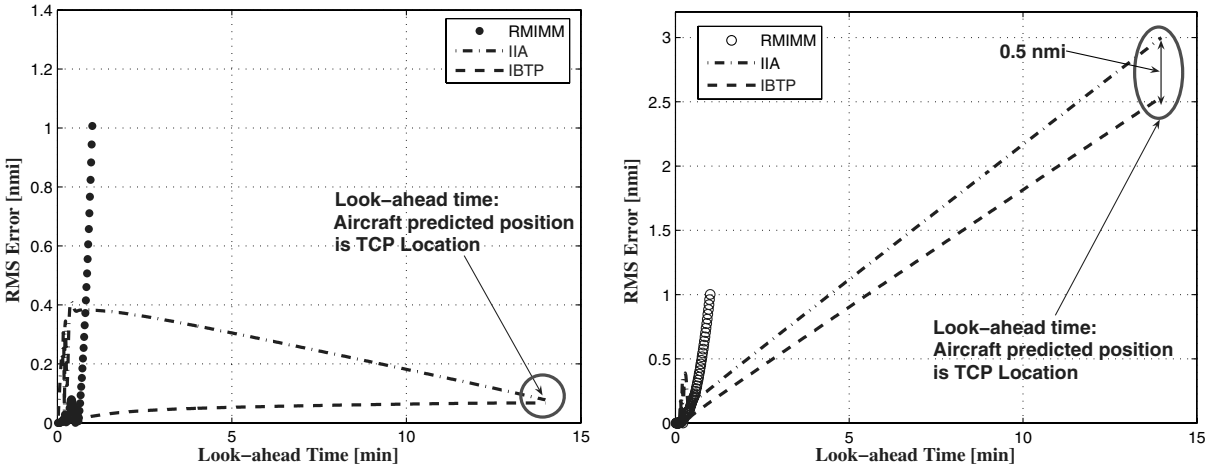
In Fig. 10, the IBTP prediction closely follows the actual trajectory, whereas the RMIMM and IIA predictions diverge quickly from it. These results demonstrate the benefits of using the IBTP algorithm for trajectory prediction in the horizontal and vertical dimensions. Similar results to those presented thus far (tracking, intent inference, and trajectory prediction performance) have been obtained in scenarios including maneuvers in the vertical and speed dimensions.

D. Prediction Error Comparison

This section presents a scenario in which we test the prediction performance of the RMIMM algorithm, the IIA, and the IBTP algorithm. The aircraft follows the flight plan (TCPs), and preforms a series of CV and CT flight segments. At time $k = 1045$ s the aircraft initiates a turning maneuver, and its intent is go-to-TCP. Even though this scenario is fairly simple, it is useful to test the trajectory prediction performance of the algorithms.

To test the trajectory prediction performance, we perform Monte Carlo simulations: 50 sets of trajectory predictions are computed by each algorithm at time $k = 1045$ s. The RMS values for the predicted position errors are computed and transformed into a reference frame aligned with the along-track direction. This results in along-track and cross-track errors for look-ahead times between 1 and 900 s (maximum look-ahead time in this case). Figure 12 contains the cross-track and along-track errors. In this figure, there are three time traces corresponding to the errors for each algorithm.

From Fig. 12a, it is clear that the performance of the RMIMM algorithm is good for short look-ahead times, specifically for look-ahead times less than 1 min, after which the performance degrades as a result of not using intent information. The performance of the IIA is poor for small look-ahead times, because it does not consider the current motion of the aircraft, but the performance improves as the look-ahead time increases. The peak prediction error for this algorithm is approximately 0.4 n mile in this case. It can be seen that the IBTP algorithm outperforms both algorithms for all times by incorporating both the current estimated motion and the intent information. Note that the IIA and the IBTP algorithm prediction



a) Cross–track errors **b) Along–track errors**

Fig. 12 RMS Prediction Errors.

performances converge as the look-ahead time increases. This is because both predictions use the intent information and predict the aircraft flying through TCPs.

Figure 12b contains the along-track errors for this scenario. Again, the RMIMM algorithm displays poor prediction performance as time increases. This is also the case for the IIA and the IBTP algorithm because the lengths of the predicted trajectories for both of these algorithms are shorter than the actual trajectory of the aircraft. Thus, the predicted trajectories of the IIA and the IBTP algorithm reach the TCP in less time than the actual trajectory does, creating increasing along-track errors. It is important to note that after 14 min of look-ahead time, the IBTP along-track error is 20% less than that of the IIA, and it corresponds to 2.5 n mile, which is approximately half of the safety distance between aircraft in ATC.

IV. Conclusions

In this paper, we have proposed the intent-based trajectory prediction algorithm to track maneuvering aircraft, estimate the aircraft's intent, and compute trajectory predictions. The algorithm uses a hybrid estimation algorithm called the residual-mean interacting multiple model algorithm to compute state and flight-mode estimates, and surpasses the state estimation accuracy of the intent inference algorithm in [11,12]. An intent inference methodology has been developed that uses the state and mode estimates, ATC regulations, and environment information to estimate the aircraft's intent in the horizontal, vertical, and speed dimensions. The intent inference problem is posed as a discrete optimization problem whose cost function uses both spatial and temporal information. Intent inference delay reductions of up to 30 s were obtained by using the rate of change of the intent model likelihoods when the aircraft is maneuvering. The trajectory predictions are computed by combining the state and mode estimates and the inferred intent. The contributions of this paper are to develop an algorithm for accurate trajectory predictions and accurate intent inference with reduced delays. These contributions have been demonstrated through analysis and various simulation results, which test the algorithm in different scenarios representative of current and future ATC operations.

Acknowledgments

This work has been supported in part by the Purdue University e-Enterprise Center, West Lafayette, Indiana 47906-4145. The authors would like to thank Jimmy Krozel, Chris Schwalm from Metron Aviation, Inc., and Dominick Andrisani for their invaluable comments and their assistance during the development of the intent-based trajectory prediction algorithm, specifically the area concerning intent inference.

References

[1] Nolan, M. S., *Fundamentals of Air Traffic Control*, 4th ed., Thomson Learning, London, 2004.

[2] Green, S. M., Vivona, R. A., and Grace, M. P., "Field Evaluation of Descent Advisor Trajectory Prediction Accuracy for En-Route Clearance Advisories," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Boston, MA, Aug. 1998*, AIAA, Reston, VA, pp. 1668–1685.

[3] Lambrecht, M., and Slate, G. L., "Departure Trajectory Modeling for Air Traffic Control," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Portland, OR, Aug. 1999*, AIAA, Reston, VA, pp. 1507–1520.

[4] Chatterji, G. B., "Short-Term Trajectory Prediction Methods," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Portland, OR, Aug. 1999*, AIAA, Reston, VA, pp. 1496–1506.

[5] Ebrahimi, Y., and Warren, A. W., "CTAS Performance Model Validation," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Denver, CO, Aug. 2001*, AIAA, Reston, VA, pp. 1693–1703.

[6] Coppenbarger, R. A., "En Route Trajectory Prediction Enhancement using Airline Flight Planning Information," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Portland, OR, Aug. 1999*, AIAA, Reston, VA, pp. 1077–1087.

[7] Chan, W., Bach, R., and Walton, J., "Improving and Validating CTAS Performance Models," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Denver, CO, Aug. 2000*, AIAA, Reston, VA, pp. 1684–1692.

[8] Chatterji, G., Sridhar, B., and Bilimoria, K., "En-route Flight Trajectory Prediction for Conflict Avoidance and Traffic Management," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, San Diego, CA, Aug. 1996*, AIAA, Reston, VA.

[9] Wanke, C. R., Callaham, M., Greenbaum, D. P., and Masolinis, A. J., "Measuring Uncertainty in Airspace Demand Predictions for Traffic Flow Management Applications," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Austin, TX, Aug. 2003*, AIAA, Reston, VA, pp. 2340–2350.

[10] Yang, L. C., and Kuchar, J. K., "Using Intent Information in Probabilistic Conflict Analysis," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Boston, MA, Aug. 1998*, AIAA, Reston, VA, pp. 797–806.

[11] Krozel, J., "Intent Inference for Free Flight Aircraft," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Denver, CO, Aug. 2000*, AIAA, Reston, VA.

[12] Krozel, J., and Andrisani, D., "Intent Inference and Strategic Path Prediction," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 2, March–April 2005, pp. 225–236.

[13] Krishnamurthy, K., and Ward, D., "Intelligent Flight Director for Autonomous Aircraft," *Proceedings of the AIAA 38th Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan. 2000*, AIAA, Reston, VA.

[14] Stengel, R. F., "Toward Intelligent Flight Control," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 6, Nov.–Dec. 1993, pp. 1699–1717.

[15] Zhao, Y., Haissig, C., and Hoffman, M. J., "Analysis of Pilot Intent Parameters in Air Traffic Control," *Proceedings of the American Control Conference, Philadelphia, PA, June 1998*, pp. 1789–1792.

[16] Wing, D. J., Barmore, B. E., and Krishnamurthy, K., "Use of Traffic Intent Information by Autonomous Aircraft in Constrained Operations," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Monterrey, CA, Aug. 2002*, AIAA, Reston, VA, pp. 435–448.

[17] Krishnamurthy, K., and Ward, D., "An Intelligent Inference Engine for Autonomous Aerial Vehicles," *Proceedings of the AIAA Atmospheric Flight Mechanics Conference and Exhibit, Portland, OR, Aug. 1999*, AIAA, Reston, VA, pp. 650–660.

[18] Bar-Shalom, Y., and Rong Li, X., *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Norwood, MA, 1993.

[19] Rong Li, X., and Bar-Shalom, Y., "Design of an Interacting Multiple Model Algorithm for Air Traffic Control Tracking," *IEEE Transactions on Control Systems Technology*, Vol. 1, No. 3, Sept. 1993, pp. 186–194.

[20] Mazor, E., Averbuch, A., Bar-Shalom, Y., and Dayan, J., "Interacting Multiple Model Methods in Tracking: A Survey," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 34, No. 1, 1998, pp. 103–123.

[21] Lainiotis, D. G., "Partitioning: A Unifying Framework for Adaptive Systems 1: Estimation," *Proceedings of the IEEE*, Vol. 64, No. 8, Aug. 1976, pp. 1126–1142.

[22] Sworder, D. D., and Boyd, J. E., *Estimation Problems in Hybrid Systems*, Cambridge Univ. Press, Cambridge, England, U.K., 1999.

[23] Hwang, I., Balakrishnan, H., and Tomlin, C., "State Estimation for Hybrid Systems: Applications to Aircraft Tracking," *IEE Proceedings Control Theory and Applications*, 2006 (to be published).

[24] Hwang, I., Hwang, J., and Tomlin, C., "Flight-Mode-Based Aircraft Conflict Detection using a Residual-Mean Interacting Multiple Model Algorithm," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Austin, TX, Aug. 2003*, AIAA, Reston, VA.

[25] Kuchar, J. K., and Yang, L. C., "Review of Conflict Detection and Resolution Modeling Methods," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 1, No. 4, Dec. 2000, pp. 179–189.

[26] Yang, L. C. and Kuchar, J. K., "Prototype Conflict Alerting System for Free Flight," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 4, July–Aug. 1997, pp. 768–773.

[27] Innocenti, M., and Pollini, L., "Spatial Trajectory Generation for Conflict Avoidance in Air Traffic Management," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Denver, CO, Aug. 2000*, AIAA, Reston, VA, pp. 857–866.

[28] Innocenti, M., Gelosi, P., and Pollini, L., "Air Traffic Management Using Probability Function Fields," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Portland, OR, Aug. 1999*, AIAA, Reston, VA, pp. 1088–1097.

[29] Isaacson, D., and Erzberger, H., "Design of a Conflict Detection Algorithm for the Center/TRACON Automation System," *Proceedings of the 16th Digital Avionics Systems Conference, Irvine, CA, Oct. 1997*, Vol. 2.

[30] Brudnicki, D., Lindsay, K., and McFarland, A., "Assesment of Field Trials, Algorithmic Performance, and Benefits of the User Request Evaluation Tool (URET) Conflict Probe," *Proceedings of the 16th Digital Avionics Systems Conference, Irvine, CA, Oct. 1997*, pp. 35–44.

[31] Krozel, J., and Andrisani, D., "Intelligent Path Prediction for Vehicular Travel," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 25, No. 2, Feb. 1995, pp. 478–487.

[32] RTCA, "Minimum Aviation System Performance Standards for the Automatic Surveillance-Broadcast (ADS-B)," DO-242A, Jan. 1998.

[33] Rong Li, X., and Jilkov, V. P., "Survey of Maneuvering Target Tracking, Part 1: Dynamic Models," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 39, No. 4, Oct. 2003, pp. 1333–1364.

[34] Lunze, J., "What is a hybrid system?" *Modelling, Analysis, and Design of Hybrid Systems*, edited by S. Engell, G. Frehse, and E. Schnieder, Vol. 279, Springer, New York, 2002, pp. 3–14.

[35] Hogg, R. V., and Craig, A. T., *Introduction to Mathematical Statistics*, 5th ed., Prentice–Hall, Upper Saddle River, NJ, 1995.

[36] Paielli, R. A., and Erzberger, H., "Conflict Probability Estimation for Free Flight," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 3, May–June 1997, pp. 588–596.

[37] Strang, G., *Linear Algebra and its Applications*, 3rd ed., Harcourt, San Diego, CA, 1987.

[38] Maybeck, P. S., *Stochastic Models, Estimation, and Control*, Academic Press, New York, 1979.

[39] Lovera Yepes, J., "Algorithms for Aircraft Intent Inference and Trajectory Predictions," Master's Thesis, Aeronautics and Astronautics Dept., Purdue Univ., West Lafayette, IN, Dec. 2005.