

Aircraft Collision Warning with State Estimation Filters

Erik R. Holmgren

December 2, 2024

1 Introduction

It is often said that air travel is safer mile-per-mile than travel by car. This may be true when it comes to commercial aircraft, but the consequences of a mishap involving an aircraft are likely to be more severe than those resulting from a road vehicle collision. And while rigorous training and regulations ensure safe operation of commercial aircraft, those highly trained pilots are not the only authorized users of regulated airspace. Privately owned aircraft flown by students and hobbyists do not share the same "safer than driving" statistic as do their commercial counterparts, but they do share the same airspace and airports [USA23]. While aviation accidents are rare, they can have drastic consequences whether occurring in air or on the ground. Modern technology allows for pilots and air traffic controllers to have more situational awareness than ever before. But without adequate automation and processing, data from airspace monitoring sensors could be more of a distraction than an asset. In pursuit of the requirements for a Masters in Applied Mathematics from the University of Colorado at Boulder, I have completed my Cumulating Experience project in the form of a software package for automated aircraft collision warning utilizing ADS-B aircraft-tracking data.

2 Data

2.1 ADS-B System

All aircraft operating in US airspace after 2020 are required to transmit a set of flight parameters while in the air [FAA10]. The most common system by which this is accomplished is Automatic Dependent Surveillance Broadcast, known as ADS-B. An ADS-B equipped aircraft broadcasts its location, groundspeed, heading, and various other attributes as they are reported by an onboard Global Positioning System (GPS) receiver. These broadcasts are received by air traffic control centers where they are combined with other air surveillance sensors such as radar. Transmissions are also received by other aircraft which can display the surrounding traffic to the pilot on an in-cockpit display.

Fields available from processed ADS-B messages are listed in table 1.

2.2 Variations of ADS-B

As air traffic surveillance has been a need throughout the modern age of flight, historical standards and revisions have led to several airspace monitoring broadcast standards currently existing in use to support air safety. The Mode-S standard provides support for short messages announcing an aircraft's presence, but limited information as to its location. Mode-S Extended Squitter (Mode-S ES) supports longer messages and is still used as the transport mechanism for ADS-B packages on most aircraft. Mode-S and Mode-S ES broadcast on 1090MHz [Sun21]. This is a relatively high frequency band with a wavelength of 27.5 cm, and is not well suited for receiving broadcasts through physical barriers such as walls or trees. As such, the range of a receiver will be greatest if the antenna is mounted in an elevated location clear of obstructions. The Universal Access Transceiver (UAT) standard is emerging within the US as an alternative ADS-B transport standard on 978Mhz. UAT is utilized by smaller private aircraft and is only authorized to replace 1090MHz broadcasts on flights remaining below 18,000 ft [FAA10]. Like Mode-S broadcasts on 1090Mhz, the 30.5 cm wavelength of UAT broadcasts require good line of sight between the transmitting and receiving antennas. Figure 1 describes the overlap of Mode-S, UAT, ADS-B and several historically relevant standards.

Name	Description
adsb icao	ICAO issued identifier for the aircraft
flight	registration string of the aircraft
alt baro	altitude as computed by barometric pressure
alt geom	altitude as reported by GPS
gs	ground speed in knots
ias	indicated air speed in knots
track	heading as reported by GPS
baro rate	change in barometric altitude in ft/min
geom rate	change in GPS altitude in ft/min
lat, lon	position of the aircraft as reported by GPS
rc	radius of containment - reported position uncertainty
nac p	position uncertainty
nac v	velocity uncertainty
rssi	received signal strength in dB

Table 1: Information contained with an ADS-B message.[[ADS24](#)]

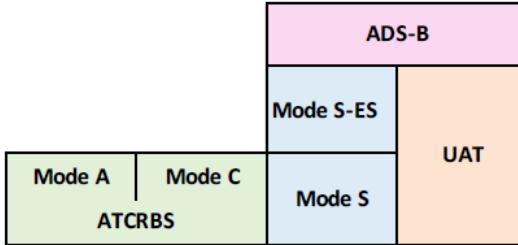


Figure 1: Overlap between various aircraft surveillance standards [[Bur20](#)].

In cases where an aircraft is broadcasting an identifying call sign but not its location, as in the case with the original Mode-S standard or with military aircraft with location transponders disabled, the technique of multilateration can be used to find an approximate solution for the location of the aircraft. Multiple receivers timestamp their receipt of the signal, and send that information along with their locations to a central processing system. The time difference on arrival (TDOA) for this set of messages is computed, from which a solution for location can be found based on the propagation speed of the radio signal through the atmosphere[[ICA07](#)].

2.3 Data Collection

ADS-B data is available through a number of online brokers such as [adsbexchange.com](#), [opensky-network.org](#), [flightaware.com](#) and others. These brokers rely on receivers run by volunteer individuals and institutions to supply them with data, often incentivized by enhanced access to the data exchange in return for "feeding" the service. Such receivers may be commercial products, or they can be assembled from hardware readily available on the hobbyist market along with open source software.

To collect live data from local air traffic the author constructed a receiver following the instructions described in [[ADS24](#)]. The hardware used was a RaspberryPi 4 single board computer paired with two software definer radio (SDR) receivers. One receiver has filtering and amplification circuitry specialized for receiving the primary ADS-B broadcast at 1090MHz. This SDR was paired with an ADS-B specific antenna. A second, general purpose, SDR was paired with a dipole antenna tuned for UAT broadcast on 978Mhz. As the author lives within several miles of the Rocky Mountain Metropolitan Airport, general aviation (GA) traffic utilizing the UAT broadcast is common in the area. As ADS-B is vertically polarized, antennas should be mounted vertically to achieve the best reception performance.

[Adsbsexchange.com](#) provides a ready made operating system image to facilitate configuring the SDRs and for forwarding received messages to their central processing system. This image based on Debian Linux contains the `readsb` package for decoding received messages into several formats including an easily processable json-based ascii format. It also contains the `tar1090` package to visualize detected

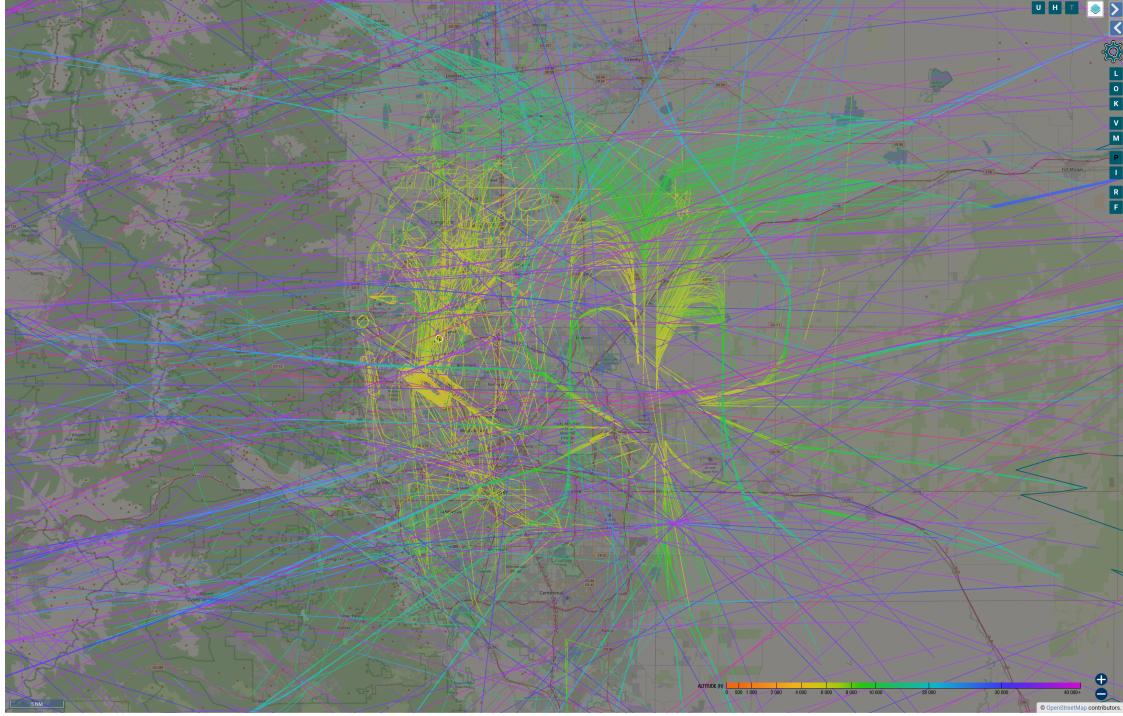


Figure 2: Eight hours of recorded ADS-B traffic from the author’s receiver.

aircraft as seen in Figure 2.

3 Methods

3.1 Kalman Filter

The Kalman Filter family of techniques are commonplace in situations where a dynamical system is measured in the presence of error. Such filters utilize a known dynamics model of the system to estimate the target’s future state, then correct for deviations from this prediction by utilizing the ratio of uncertainty due to measurement noise over the total uncertainty in the system (including that due to measurements and due to deviations from the dynamics model) [KO60]. Note that the term ”target” is used in tracking & state estimation literature to refer to the object we are interested in tracking.

To set up our Kalman Filter we must define a state-vector $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ containing the terms required to compute the dynamics of our system (and whose value we wish to estimate). The subscript \mathbf{x}_k is used to denote the state at discrete time steps.

3.1.1 Kalman Filter Prediction

Kalman Filters and their derivatives operate in two steps. In the prediction step, the state from the last update time is propagated forward to a desired (most often future) time. The previous update state vector is denoted $\hat{\mathbf{x}}_{k-1|k-1}$, with the subscript indicating that this is the posterior estimate for step $k - 1$ after having considered the measurement from step $k - 1$. The prediction is obtained as follows

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} \quad (1)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^T + \mathbf{Q} \quad (2)$$

where \mathbf{F} is the discrete time state transition matrix which advances the system by the time delta from the last update to the desired time, and \mathbf{Q} is the process noise covariance which represents uncertainties in the dynamics that may affect the system during the same time frame. Process noise is discussed in further detail below.

3.1.2 Kalman Filter Measurements

It is assumed that the process obtaining the measurement is a function of the systems state, which has evolved since the previous time-step to be

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_k \quad (3)$$

where $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ is randomly distributed process noise with covariance \mathbf{Q} which has affected the system in a way not captured by the dynamics model \mathbf{F} . As such the measurement vector $\mathbf{y} = [y_1, \dots, y_p]^T$ is taken to be

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \quad (4)$$

where \mathbf{H} is the measurement model and $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ is the measurement error at time-step k with measurement covariance \mathbf{R} .

3.1.3 Kalman Filter Update

To utilize the information contained in a new measurement the update step of the filter is performed. We wish to estimate the true state of the system despite the uncertainty introduced by process and measurement noise. To do so we use the predicted state estimate $\hat{\mathbf{x}}_{k|k-1}$ to compute the predicted measurement

$$\hat{\mathbf{y}}_{k|k-1} = \mathbf{H}\hat{\mathbf{x}}_{k|k-1} \quad (5)$$

from which we subtract the obtained measurement to compute the innovation term

$$\mathbf{z}_k = \mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1} \quad (6)$$

The innovation informs us of the disagreement between the filter's prediction and the received measurement. In the case where we do not trust the filter's prediction (if the filter is still converging \mathbf{P} may be high, or if we have high process noise) we will wish to rely more on the received measurement. However in the case where the filter's estimate is to be trusted (the filter has converged and \mathbf{P} has reached steady-state, and the process noise is lower than the measurement noise), we will wish to rely more on the filter's prediction. In the case of $\alpha - \beta$ or $\alpha - \beta - \Gamma$ filters we must chose this ratio with preconfigured parameters [Kal84]. But the Kalman Filter decides how much of the innovation to use in the update by means of the Kalman gain

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}^T (\mathbf{H} \mathbf{P}_{k|k-1} \mathbf{H}^T + \mathbf{R})^{-1} \quad (7)$$

which is the quantity that minimizes the uncertainty in the system as found by taking the matrix derivative of the cost function defined as the total uncertainty in the system. Intuitively, we can see this quantity as a ratio with the portion of predicted filter covariance \mathbf{P} represented in measurement space on the top, with measurement covariance plus the same filter covariance fully-mapped into measurement space on the bottom.

We now update our filter as such

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \mathbf{z}_k \quad (8)$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{P}_{k|k-1} \quad (9)$$

to obtain the posterior estimate for $\hat{\mathbf{x}}$ at time-step k .

3.2 Process Noise Models

Any physical system will be subject to random disturbances over time. In the case of an aircraft, such a disturbance could be due to turbulence or changes in air pressure. Additionally, while much of a flight may be undertook with constant speed and heading, aircraft must maneuver to align themselves with runways, airways, and to avoid other traffic. Kalman filters provide a means of accounting for control inputs via a control vector \mathbf{u}_k which represents input to the system at time-step k , and control matrix \mathbf{B} which maps the control vector into the system state space. While an Kalman Filter running onboard the aircraft will be able to incorporate maneuvers into its estimate, as outside observers we cannot utilize this mechanism in the surveillance problem. As such, we may account for deliberate deviations to the known dynamics within our process noise model.

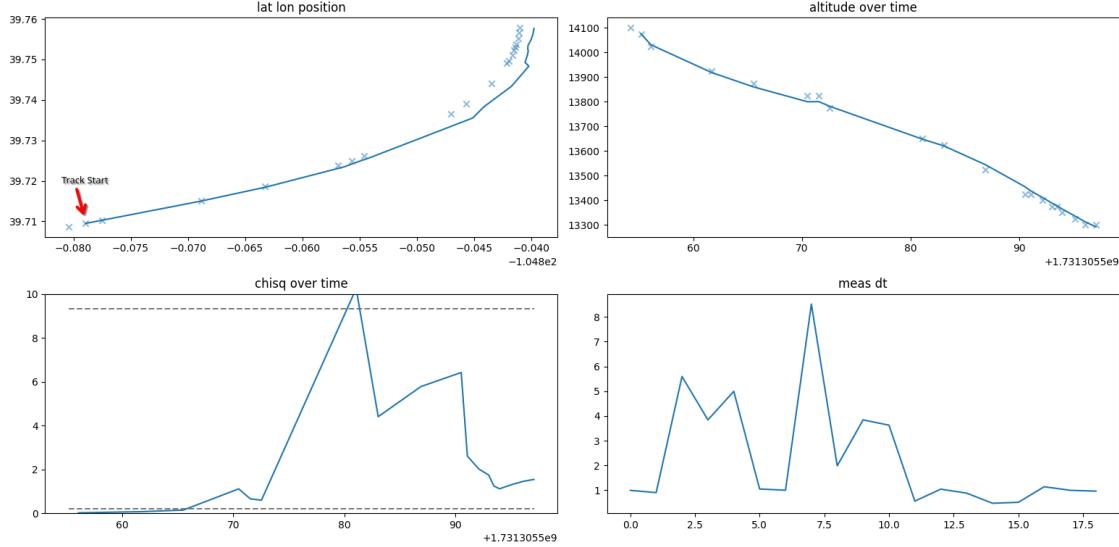


Figure 3: Six-State (x , y , z position & velocity) filter performance with maneuvering aircraft.

3.2.1 Normalized Estimation Error Squared and Normalized Innovation Squared

As we tune the balance between process noise and measurement noise in our filter, it is desirable to have a metric to track our filter's performance. One such metric is the Normalized Estimation Error Squared (NEES), which is the sum of squared errors between the true state and our estimate $\mathcal{N}\mathcal{E}\mathcal{S} = (\mathbf{x}_k - \hat{\mathbf{x}}_k)^T(\mathbf{x}_k - \hat{\mathbf{x}}_k)$. Because we assume errors in our system to be Gaussian, and because we are summing the squares of Gaussians, it is taken that $\mathcal{N}\mathcal{E}\mathcal{S} \sim \chi_n^2$. While the NEES value may assist us while tuning against simulated or otherwise deeply modeled targets, it cannot readily be used on real data or to assess filter performance in real time.

When we do not have access to truth, we can perform a similar calculation on our filter's innovation term. As the innovation can be thought of as the prediction to measurement error, we define $\mathcal{N}\mathcal{I}\mathcal{S} = \mathbf{z}^T \mathbf{z}$ and take $\mathcal{N}\mathcal{I}\mathcal{S} \sim \chi_p^2$. Cumulative computations of NEES and NIS may be computed by combining values across the time-steps of the filter and multiplying the χ^2 degrees of freedom by the number of time-steps[Sim06].

3.3 Limitations of Kalman Filter

By increasing the process noise to account for the possibility of maneuvers we may be able to keep our state estimate close to the true state of the target while it is maneuvering. However, doing so hampers the Kalman Filter's ability to prevent noise from measurements from entering the estimate.

For an example we consider recorded data from an aircraft engaging in a turn. The data was fed through a six-state filter (state vector $\mathbf{x} = [x, \dot{x}, y, \dot{y}, z, \dot{z}]^T$) with the state transition matrix assuming constant motion per standard kinematic equations

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{cv} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{cv} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{cv} \end{bmatrix} \quad (10)$$

$$\mathbf{F}_{cv} = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \quad (11)$$

and process noise covariance matrix

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{cv} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{cv} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_{cv} \end{bmatrix} \quad (12)$$

$$\mathbf{Q}_{cv} = q \begin{bmatrix} dt^2/3 & dt/2 \\ dt/2 & 1 \end{bmatrix} \quad (13)$$

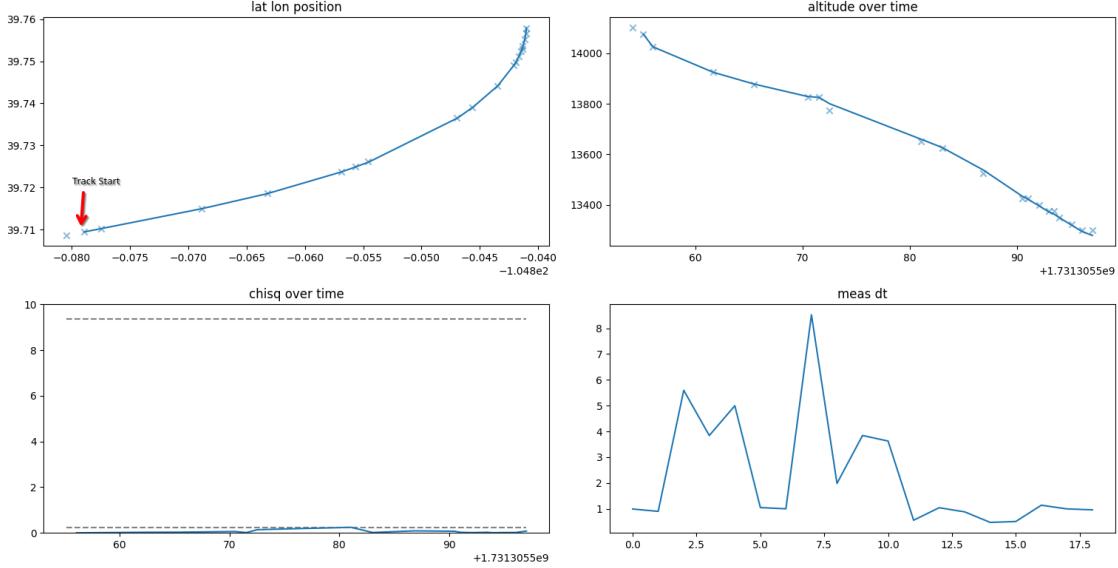


Figure 4: Nine-State (x , y , z position, velocity & acceleration) filter performance with maneuvering aircraft.

with tuning parameter q set to 10. These models come from [Gen01] and will be revisited further on.

Figure 3 shows the estimation results from the six-state filter. We see that our assumption that no higher order dynamics are at play breaks down as the aircraft begins its turn, and our state estimate drifts away from the measurement sequence preferring to stay along the original path. The NIS score is below the $\chi^2_{3,95\%}$ bounds before the maneuver, then drifts above during the maneuver.

Figure 4 shows the result of the same data in a nine-state (state vector $\mathbf{x} = [x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}]^T$) with parameters

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{ca} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{ca} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{ca} \end{bmatrix} \quad (14)$$

$$\mathbf{F}_{ca} = \begin{bmatrix} 1 & dt & dt^2/2 \\ 0 & 1 & dt \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

and process noise covariance matrix

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{ca} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{ca} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_{ca} \end{bmatrix} \quad (16)$$

$$\mathbf{Q}_{ca} = q \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

with tuning parameter q set to 0.1.

With the nine-state filter we see much better tracking of the target to the measurements during the maneuver, but the NIS is below the $\chi^2_{3,95\%}$ bounds for all but the beginning of the maneuver. Further tuning could improve performance, but it will remain a trade off between the components of the track which include changes to acceleration and those which do not.

3.4 Interacting Mixture Model

As aircraft are maneuverable, a filter which restrains us to a single dynamics model for the entire flight will not be able to optimally estimate the target's state. As such, we need a mechanism to switch between models at appropriate portions of the flight. The Interacting Multiple Model (IMM)

filter performs this task by combining multiple Kalman Filters with a Markov process describing the transition between modes of flight. The mode probabilities are combined with the filter estimates to achieve an optimal mixing between the results of the different dynamics models.

The IMM described in [Gen01] was implemented for this task. It consists of 3 linear Kalman Filters, the six-state and nine-state filters described in the previous section and additionally a maneuver filter with the following dynamics

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{\text{ma}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{\text{ma}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{\text{ma}} \end{bmatrix} \quad (18)$$

$$\mathbf{F}_{\text{ma}} = \begin{bmatrix} 1 & \omega^{-1} \sin(\omega dt) & \omega^{-2}(1 - \cos(\omega dt)) \\ 0 & \cos(\omega dt) & \omega^{-1} \sin(\omega dt) \\ 0 & -\omega \sin(\omega dt) & \cos(\omega dt) \end{bmatrix} \quad (19)$$

where ω is the turning range

$$\omega = \frac{\sqrt{\ddot{x}^2 + \ddot{y}^2 + \ddot{z}^2}}{\sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}} \quad (20)$$

The same process noise covariance matrix \mathbf{Q} as used in the nine-state constant acceleration filter described above is used for the maneuvering filter.

The current mode m_k is represented as a probability vector $\mu = [P(m_k = cv), P(m_k = ca), P(m_k = ma)]$ which sums to 1. Mode switching is represented by the Markov system matrix

$$\boldsymbol{\Pi} = \begin{bmatrix} P(m_k = cv|m_{k-1} = cv) & P(m_k = ca|m_{k-1} = cv) & P(m_k = ma|m_{k-1} = cv) \\ P(m_k = cv|m_{k-1} = ca) & P(m_k = ca|m_{k-1} = ca) & P(m_k = ma|m_{k-1} = ca) \\ P(m_k = cv|m_{k-1} = ma) & P(m_k = ca|m_{k-1} = ma) & P(m_k = ma|m_{k-1} = ma) \end{bmatrix} \quad (21)$$

As a corollary of the law of total probability, the rows of $\boldsymbol{\Pi}$ must sum to 1.

3.4.1 IMM Prediction

When predicting the IMM to time-step k , we must account for both the possible modes we were in at time step $k-1$ as well as the N possible modes we may switch into. A predictive mixing coefficient is computed for each possibility, with subscript i denoting the previous mode and j the future mode

$$\hat{\mathbf{x}}_{k|k-1}^{0,j} = \sum_{i=1}^N \hat{\mathbf{x}}_{k|k-1}^i \Omega_{k|k-1}^{i,j} \quad (22)$$

where

$$\Omega_{k|k-1}^{i,j} = \frac{\boldsymbol{\Pi}_{i,j} \mu_{k-1|k-1}^i}{\sum_{i=1}^N \boldsymbol{\Pi}_{i,j} \mu_{k-1|k-1}^i} \quad (23)$$

is the normalizing coefficient for transitioning into the future mode, and $\hat{\mathbf{x}}_{k|k-1}^i$ is the standard Kalman Filter prediction for filter i . Mode covariances are computed as

$$\mathbf{P}_{k|k-1}^{0,j} = \sum_{i=1}^N \Omega_{k|k-1}^{i,j} [\mathbf{P}_{k|k-1}^i + (\hat{\mathbf{x}}_{k|k-1}^i - \hat{\mathbf{x}}_{k|k-1}^{0,j})(\hat{\mathbf{x}}_{k|k-1}^i - \hat{\mathbf{x}}_{k|k-1}^{0,j})^T] \quad (24)$$

where $\mathbf{P}_{k|k-1}^i$ is the prediction covariance from Kalman filter i .

The total IMM prediction is then computed as

$$\hat{\mathbf{x}}_{k|k-1}^0 = \sum_{j=1}^N \hat{\mathbf{x}}_{k|k-1}^{0,j} \mu_{k|k-1}^j \quad (25)$$

$$\mathbf{P}_{k|k-1} = \sum_{j=1}^N \mathbf{P}_{k|k-1}^{0,j} \mu_{k|k-1}^j \quad (26)$$

$$(27)$$

where $\mu_{k|k-1}^j$ is the j -th entry from predicted mode probability vector

$$\mu_{\mathbf{k}|k-1} = \Pi \mu_{\mathbf{k-1}|k-1} \quad (28)$$

3.4.2 IMM Update

To update the IMM with a new measurement \mathbf{y}_k requires updating the individual filters and mixing the resulting states. However, we must use the individual mixed predictions $\hat{\mathbf{x}}_{k|k-1}^{0,j}$ to find our predicted measurements $\hat{\mathbf{y}}_{k|k-1}^j$. With this modification, the j -th individual Kalman Filter is updated as described in the previous section to obtain posterior estimates $\hat{\mathbf{x}}_{k|k}^j$ and covariances $\mathbf{P}_{k|k}^j$. The individual states must now be mixed to compute the total IMM estimate. First, we must update the mode probability vector in light of the agreement of each filter with the newly assigned measurement. Before updating the individual filters, the measurement likelihoods are computed as

$$\mathcal{L}^j = \frac{1}{\sqrt{|2\pi\mathbf{S}^j|}} \exp\left[\frac{-1}{2}(\mathbf{z}^j)^T(\mathbf{S}^j)^{-1}(\mathbf{z}^j)\right] \quad (29)$$

where

$$\mathbf{S}^j = \mathbf{H}^T \mathbf{P}_{k|k-1}^{0,j} \mathbf{H} + \mathbf{R} \quad (30)$$

is the innovation covariance from the j -th filter. Note that this likelihood is from a Gaussian PDF centered at the expected innovation value (\mathbf{o}) and utilizing the innovation covariance for its uncertainty. This likelihood vector is combined with the Markov probability update to find the posterior mode probability vector for time-step k

$$\mu_{k|k}^j = \frac{\mu_{k|k-1}^j \mathcal{L}^j}{\sum_{i=1}^N \mu_{k|k-1}^i \mathcal{L}^i} \quad (31)$$

The total IMM update is then

$$\hat{\mathbf{x}}_{k|k} = \sum_{i=1}^N \mu_{k|k}^i \hat{\mathbf{x}}_{k|k}^i \quad (32)$$

$$\mathbf{P}_{k|k} = \sum_{i=1}^N \mu_{k|k}^i [(\hat{\mathbf{x}}_{k|k}^i - \hat{\mathbf{x}}_{k|k}) (\hat{\mathbf{x}}_{k|k}^i - \hat{\mathbf{x}}_{k|k})^T + \mathbf{P}^i | k | k] \quad (33)$$

3.5 Conflict Detection

Because the result of both the prediction and update steps of an IMM (or Kalman Filter) is a Gaussian distribution centered around our believed location of the target, we can combine predicted distributions from multiple targets to get their joint probabilities of occupying a location at a future time. Such fact is used in the conflict detection algorithm:

When the distance between the positions of two aircraft falls below a configurable threshold, state predictions are made at several future time intervals resulting in a set of position distributions for the two vehicles. A search grid is then constructed around the midpoint between the predictions, and each point in the discrete grid is evaluated for probability density of each aircraft. The resulting grids are normalized to sum to 1 to account for the discrete nature of the grid based method. The normalized grids are then multiplied together pairwise resulting in the joint distribution that each aircraft will occupy a grid point at the predicted time. If this probability is above a configurable threshold, a conflict warning is generated for the point with maximum joint probability.

3.6 Overall Solution

The above methods were implemented in a python software package by the author [Hol]. The package is able to process real time data from the receiver, or historical data which has been recorded. ADS-B

fields used are lat, lon, geom altitude, and nac p. Measurement uncertainty is set based on a lookup table of circular uncertainty in meters based on nac p value.

A track with an IMM filter is spawned for each aircraft entering the covered area, and the track is stopped and recorded once no measurements have been detected for a configurable amount of time. In this case the drop age was set to one minute. Filters are initialized with the position obtained from the initial measurement, and high initial covariance terms are used to allow the higher-order state components (velocity and acceleration) to converge quickly from their bogus initial values of 0. The initial IMM mode probability vector is set to be $\mu_0 = [0.7, 0.2, 0.1]$, favoring initial track being established during a period of constant velocity flight. The mode transition matrix is set to be

$$\Pi = \begin{bmatrix} 0.6 & 0.2 & 0.2 \\ 0.2 & 0.6 & 0.3 \\ 0.1 & 0.3 & 0.6 \end{bmatrix} \quad (34)$$

such that each mode favors returning to itself next step, but allows for the possibility to move to another mode.

While ADS-B utilizes latitude, longitude and altitude (LLA) with respect to the standard WGS-84 earth ellipsoid model as its primary geographic datum, the spherical nature of LLA coordinates presents difficulties for use in linear filters. An Extended Kalman Filter (EKF) is included in the software package for tracking in spherical coordinates. However, use of a different reference frame simplifies the conflict detection logic. LLA measurements are converted to the Earth Centric Earth Fixed (ECEF) coordinate frame, which is a Cartesian frame that rotates with the Earth. Target state vectors are expressed in ECEF coordinates, and translated back to LLA when reported to the user.

When the difference between the position estimates of two aircraft falls below 5km, the conflict detection routine described above is run. A 3D grid of 192 points with side lengths of 5 km is constructed around the midpoint between the two aircraft, and the predicted joint probability of each aircraft occupying a point at future times of 10, 30, 60, 90, and 120 seconds are computed. If the joint probability of any point exceeds $P(A \cup B) \geq 0.0005$, the geographic location corresponding to the highest probability point is reported to the user as a potential conflict at the predicted time.

4 Results

4.1 Baseline Performance

Utilizing the IMM filter allows for better tracking of maneuvering aircraft than could be achieved by an individual Kalman. It also accounts for the possibility of future maneuvers when predicting the behavior of the aircraft at a future time.

Figure 5 shows the IMM performance of the same maneuvering aircraft discussed previously. In addition to the estimate given by the IMM, the individual component filters are provided showing each Kalman Filter's believed state of the target. Figure 6 shows a blown up view of the maneuvering portion of the trajectory. Note that the jagged trajectories in the individual filters are an artifact due to the filters having their individual predictions replaced with the mixed predictions in order to perform each update. We see the smooth red line representing the IMM to prefer the tighter turns allowed by the nine-state constant acceleration filter and maneuvering filter, but the constant-velocity filter retains some influence.

Figure 7 gives the model probabilities over time for the same aircraft as well as the maneuvering filter's maneuver coefficient ω . Following the trajectory of the aircraft in 5, we see the six-state constant velocity filter having the highest probability until roughly half way through the track when the turn starts. From that point forward the probability of constant acceleration or maneuver overtake that of constant velocity. Additional tuning of the maneuver filter may lend to better detection of turns.

Figure 8 shows latitude and longitude trajectories and measurements for all aircraft contained within the baseline scenario. Figure 9 shows the altitudes over time, indicating that the aircraft do exist during the same time-steps and are flying at comparable altitudes, but have adequate horizontal spacing. No conflicts were detected during this engagement, indicating good false alarm performance from the collision detection algorithm during a scenario with no close calls.

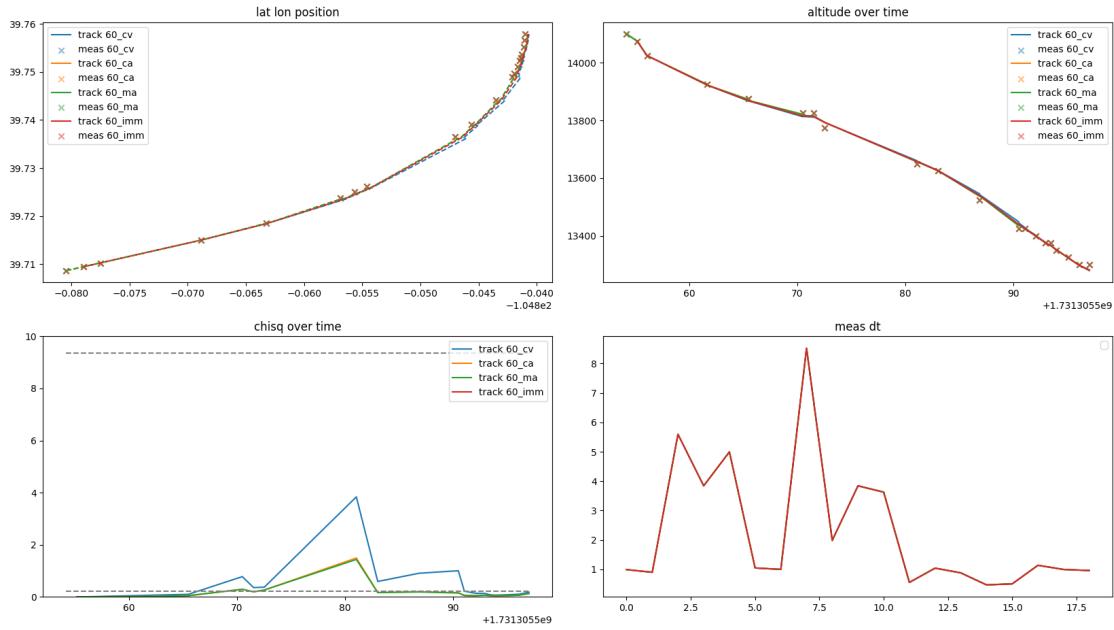


Figure 5: IMM filter performance with maneuvering aircraft.

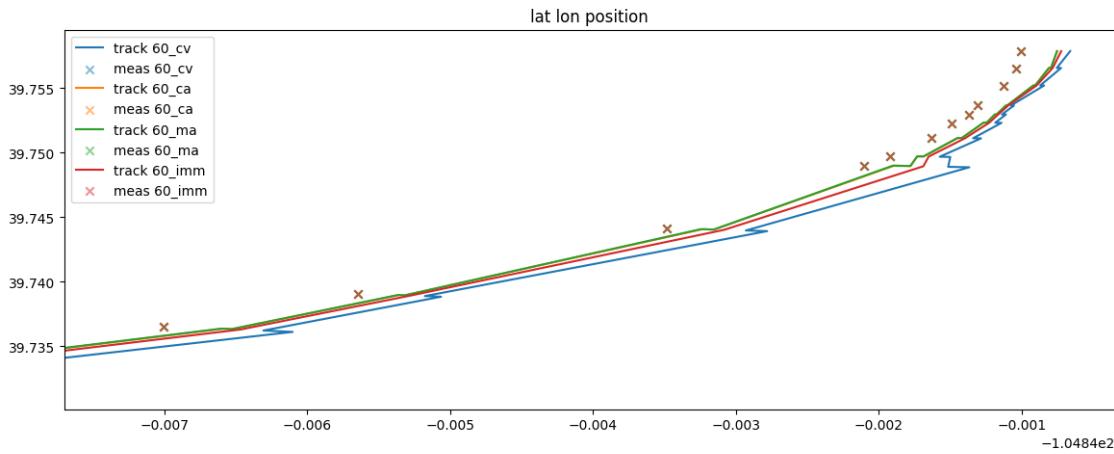


Figure 6: Enhanced view of IMM maneuver.

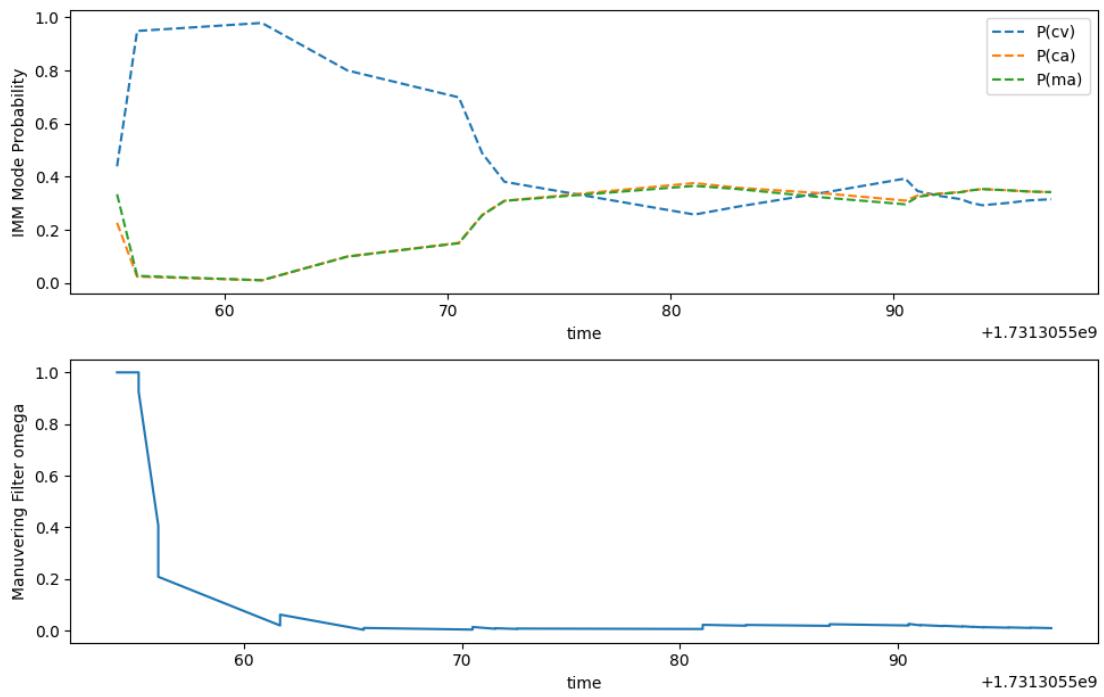


Figure 7: IMM filter model probability and maneuvering filter maneuver coefficient.

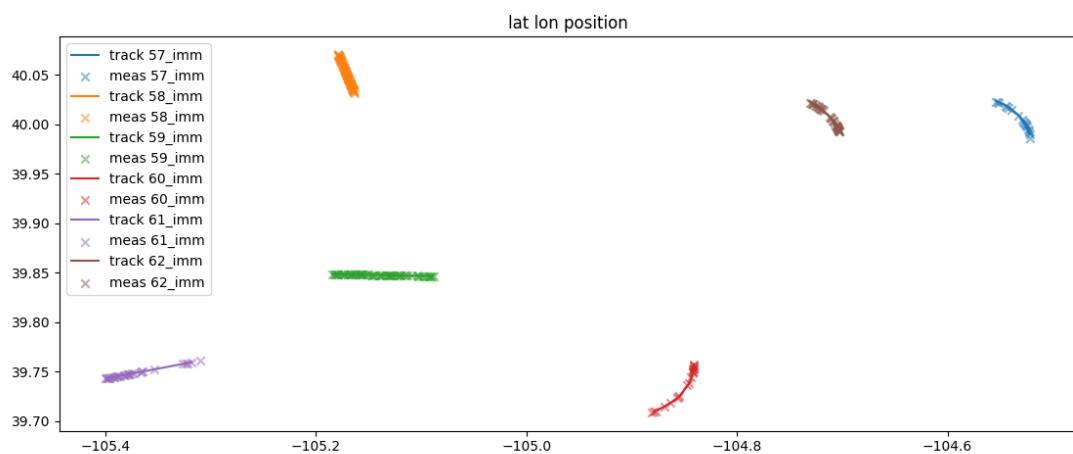


Figure 8: Trajectories of all aircraft processed during the baseline scenario.

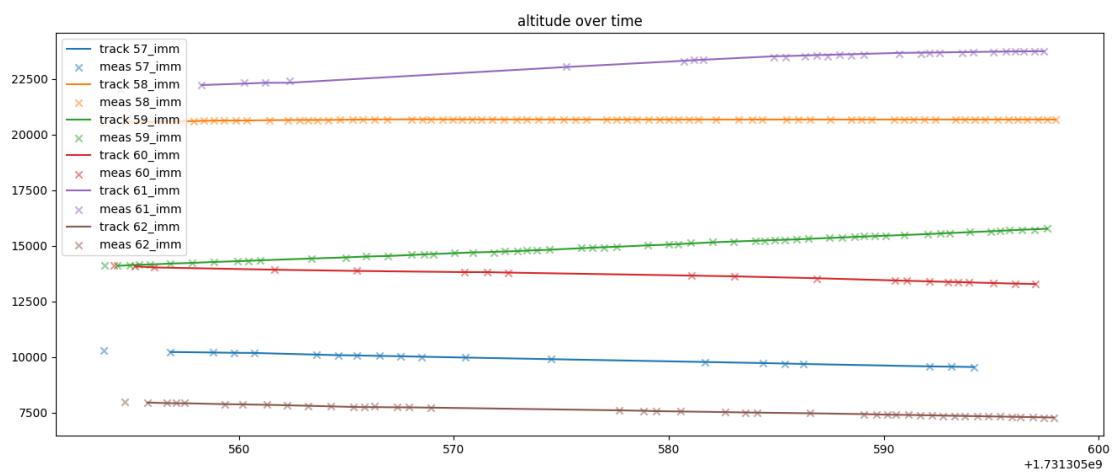


Figure 9: Trajectories of all aircraft processed during the baseline scenario.

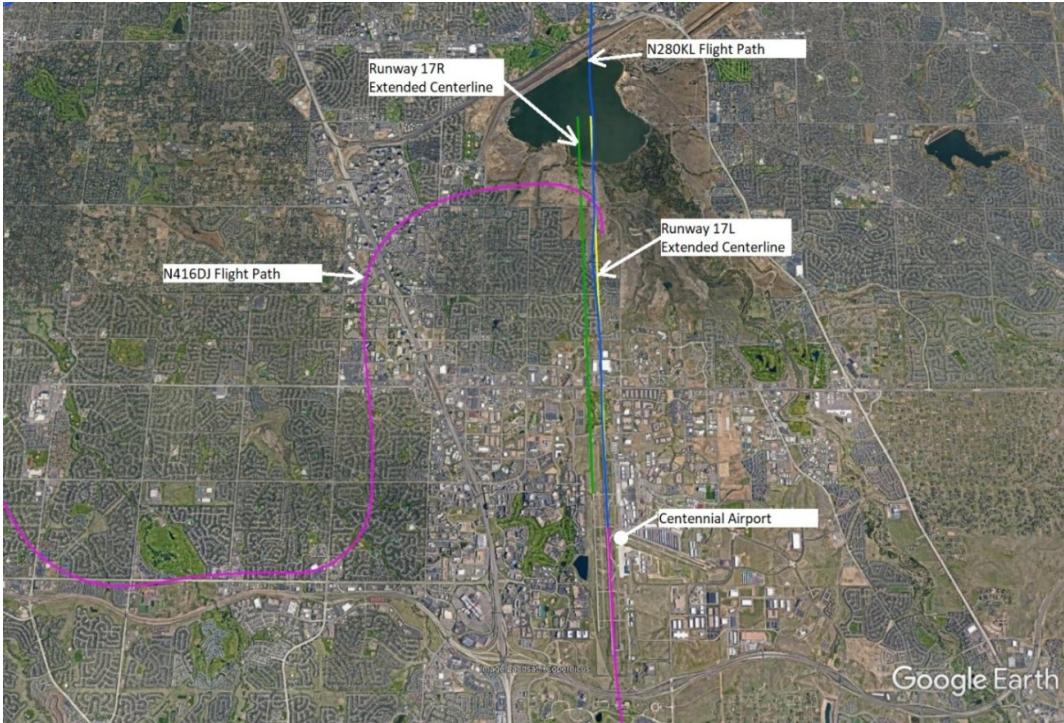


Figure 10: Flight trajectory of colliding aircraft [NTS].

4.2 Conflict Scenarios

4.3 Case Study: N280KL and N416DJ

On the 12th of May 2021, two aircraft collided mid-air while on landing approach to Colorado's Centennial airport. A small commercial cargo aircraft (registration N280KL) was struck from above by a two seat general aviation aircraft (registration N416DJ) which descended onto the cargo plane while approaching from the rear and right 10. Miraculously, no one onboard either aircraft was seriously hurt or killed. The damaged cargo plane continued its decent and landed at Centennial 11a, and while the GA craft began an uncontrolled decent it was able to deploy a small parachute resulting in a successful emergency landing in Cherry Creek State Park 11b [NTS].



(a) Damage to cargo plane [NTS].



(b) Wreck of GA plane [NTS].

Figure 11: Photos of Damaged aircraft.

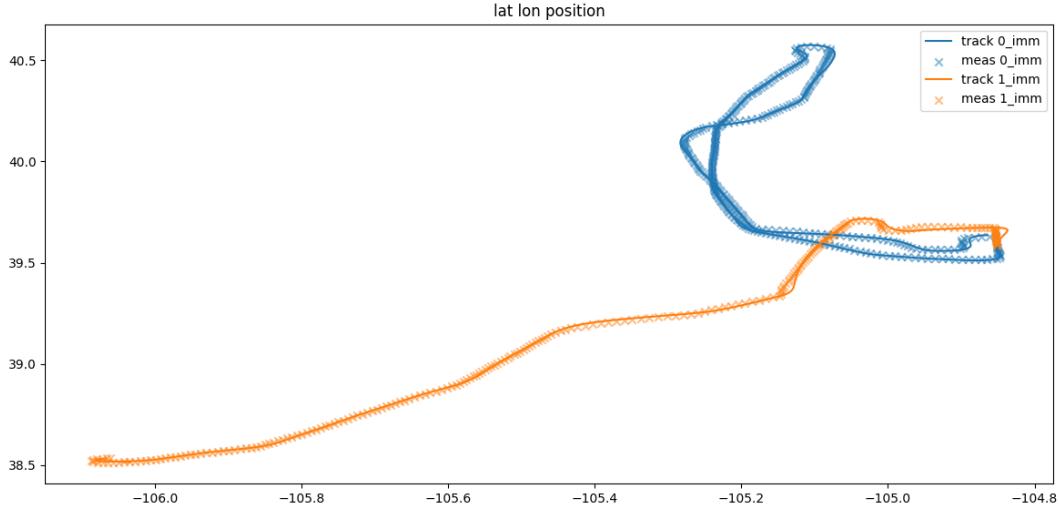


Figure 12: IMM filtered trajectory from May 12 scenario.

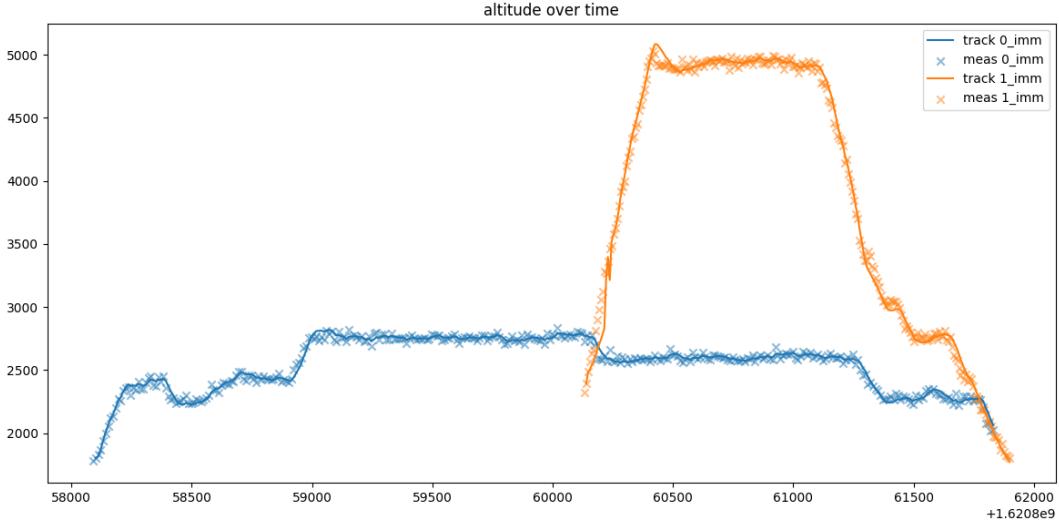


Figure 13: IMM filtered altitudes from May 12 scenario.

4.4 Collision Detection Performance

ADS-B measurements for the May 12th scenario were reconstructed from a KML file describing the recorded trajectories of the aircraft. Realistic measurement error was included in the measurement generation. The resulting ADS-B data was fed into the software to assess the performance against a real collision situation. Position trajectories are shown in Figure 12 and altitude in 13.

Looking in detail at a maneuver section in Figure 14a, we see the IMM is not perfect at following the sequence of measurements, but it is incorporating the estimates from the constant acceleration and maneuver filters to estimate closer to what appears to be the true trajectory. Mixing probabilities for flight N416DJ are shown in Figure 14b.

The collision detection algorithm identified two groups of potential collision points (Figure 15). Note that there was a time preceding the collision when the aircraft also became closely separated. Figure 16 shows aircraft altitude over time alongside the predicted altitudes and a collision score which has been scaled to appear on the same axis. The times and altitudes are offset, potentially indicating an issue in the current implementation of the software. However the correlation of the predicted collision predictions with that of both a close call and the actual prediction indicates there is soundness to this approach.

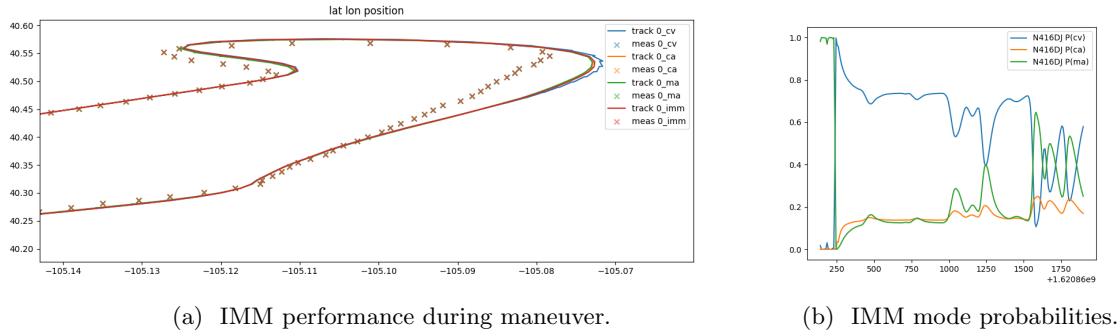


Figure 14: IMM Performance.

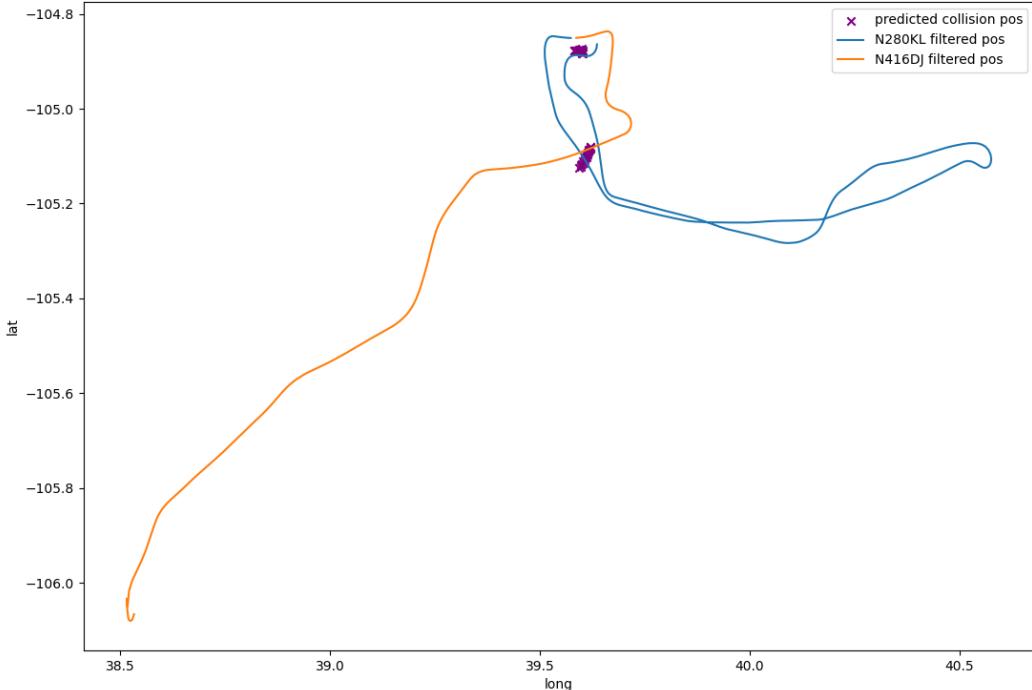


Figure 15: Predicted collision points.

5 Conclusion

This project has shown the potential for state estimation techniques to be applied to air traffic control problems. Namely, the inherent ability of the Kalman Filter to predict a future state distribution lends itself to problems where decisions must be made before an data is available - or in this case before a collision occurs. Existing techniques in this area are more focused on geometry based solutions, and that may be appropriate when an air traffic control processing system is providing pre-filtered trajectories to the conflict detector [KY00]. However, the combination of the filtering and conflict-detection problems by use of estimation filters has hereby been shown to work for self contained systems.

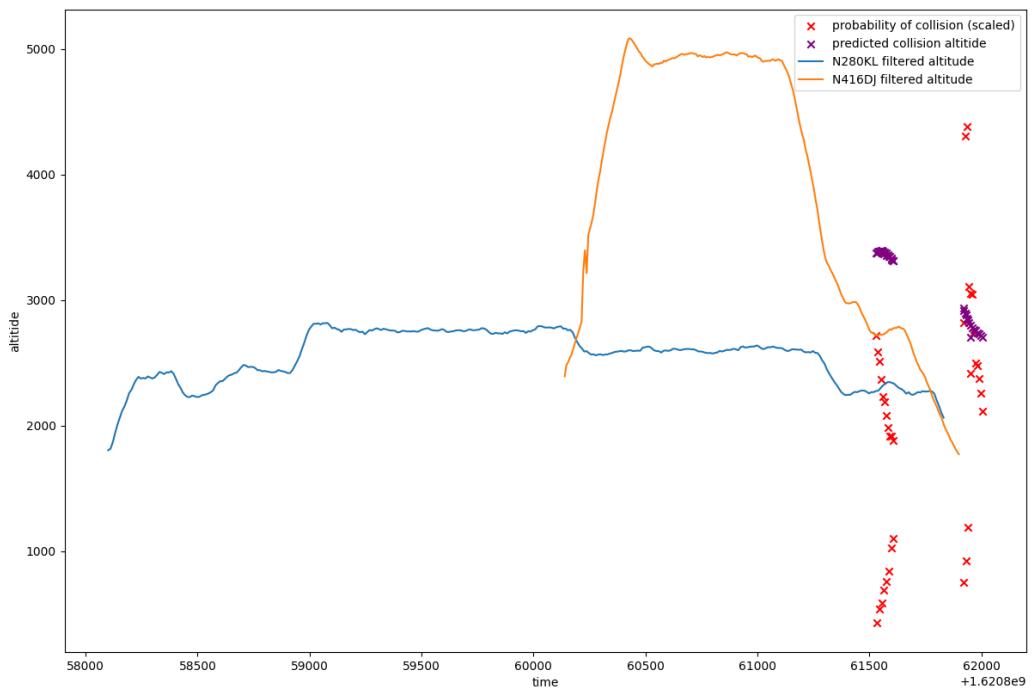


Figure 16: Predicted collision times, altitudes, and scaled probabilities.

References

- [ADS24] Build your own ads-b receiver, Oct 2024.
- [Bur20] Brandon C. Burfeind. "interoperable ads-b confidentiality". Master's thesis, Air Force Institute of Technology, 2020.
- [FAA10] FAA. *Automatic Dependent Surveillance— Broadcast (ADS-B) Out Performance Requirements To Support Air Traffic Control (ATC) Service*. Department of Transportation, May 2010.
- [Gen01] Anthony F. Genovese. The interacting multiple model algorithm for accurate state estimation of maneuvering targets. *Johns Hopkins Apl Technical Digest*, 22:614–623, 2001.
- [Hol] Erik Holmgren. Tda - tracking and data assoceation. <https://github.com/eholmgre/tda/tree/adsb-tracker>.
- [ICA07] Multilateration (mlat) concept of use, 2007.
- [Kal84] Paul R. Kalata. The tracking index: A generalized parameter for - and -- target trackers. *IEEE Transactions on Aerospace and Electronic Systems*, AES-20(2):174–182, 1984.
- [KO60] Rudolph Emil Kalman and Others. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.
- [KY00] J.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, 2000.
- [NTS] May 12, 2021 mid-air collision.
- [Sim06] Dan Simon. *Additional topics in Kalman filtering*, chapter 10, pages 297–329. John Wiley Sons, Ltd, 2006.
- [Sun21] Junzi Sun. *The 1090 Megahertz Riddle: A Guide to Decoding Mode S and ADS-B Signals*. TU Delft OPEN Publishing, 2 edition, 2021.
- [USA23] USAFacts. Is flying safer than driving?, Dec 2023.