Future Technologies Conference (FTC) 2017
29-30 November 2017| Vancouver, Canada

An Automated Avoidance Approach for Multiple General-Aviation Conflicts

Yousra Almathami
Department of Computer Science and Engineering
University of Connecticut
Storrs, Connecticut 06269
Email: yousra.almathami@uconn.edu

Reda Ammar
Department of Computer Science and Engineering
University of Connecticut
Storrs, Connecticut 06269
Email: reda@engr.uconn.edu

***Abstract*—Air traffic controllers (ATC) face daily major concerns which are controlled Airspace (CAS) infringements. An infringement is when a general aviation (GA) aircraft penetrates CASs without an advanced clearance from the ATC. These infringements could cause a mid air collision with authorized aircraft inside CAS which their conflict was not resolved ahead of time. It also disrupts ATCs operations by creating additional workload and revising new manoeuvre tactics. In the last two papers, they were focused on predicting future locations and find their probability of infringements to alert ATC in advance. So far, they have been dealing when one aircraft approaching CAS, in this paper however, focuses on the scenario when multiple aircraft infringe CAS, in case the ATC did not react quickly enough. As of 2020, it is mandatory for all GA to be equipped with a transponder which sends information such as flight ID, exact location and altitude. Therefore, using this assumption we are investigating a possible model which alerts and directs multiple GA out of CAS without interfering with commercial traffic. Kinetic triangulation method will be used as an automated manoeuvring tactic, leaving the ATC focusing on only to direct commercial flights.**

Keywords*—*Switching Kalman filters; controlled airspace; aircraft infringements; ground based safety system; polygon triangulation

I. Introduction

Generally most airports are surrounded by controlled airspace which can be divided into five classes A to E. These classes are being monitored by air traffic controllers (ATC) seven days a way 24 hours a day. Each of these classes differs in terms of their volume shape and size (lower bound and upper bound in feet). Any general aviation (GA) aircraft can fly within uncontrolled airspace class “G” without ATC authorization. If the GA pilots decides to enter a CAS zone; he or she must initiate communicate with ATC and get an advance clearance to avoid possible conflicts inside CAS. A conflict is defined as the lose of minimum required separation between two aircraft as shown in Fig. 1:

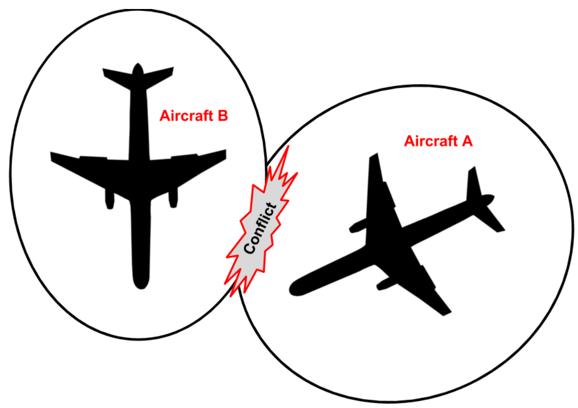

Fig. 1. Losing minimum separation between two aircraft causes a conflict.

The minimum separation varies around airport area from country to country depends on the type of airport, but generally for aircraft en-route it has to have at least 5 nautical miles horizontally and 1000ft vertically of minimum separation. In addition to the possibility of conflicts; infringements causes disruption to ATCs operations by creating additional workload and revising new manoeuvre tactics. So far, GA aircraft are not equipped with the advanced transponder required to provide accurate data to the ATC. For the ATC to detect GA aircraft, he or she must rely on two things: either using the primary surveillance radar which only detects the estimated location of the aircraft with imprecise altitude; or the pilot initial communication. Since most aircraft fly under visual flight rules, they do not have specific flight path. A survey was conducted by Eurocontrol [1] to find out why these infringements occur more frequently every year. These are the common reasons:

- The pilot is unaware with the airspace location or its boundaries.
- The pilot is trying to avoid bad weather on the way such as heavy clouds.
- The pilot is lost due to bad GPS system.
- The pilot is not experienced.
- The lack of new published VFR routes.
- Outdated GA pilot maps or the aircraft GPS database.

Due to these issues and reasons a research was conducted in [2] and [3] where they examined the following:

1) Implementing a switching Kalman filters (SKF) that predicts future aircraft locations (20 seconds ahead) instead of 4 seconds.
2) Implementing a modified version for the SKF online learning of its errors.
3) Extended available probability of infringement method by proposing a Monte carlo sampling in the event the aircraft is around a CAS vertex (corner).
4) Developed a classifier to enhance our probability of infringement to reduce false alerts.

In this paper, it examines the scenario when multiple aircraft infringe CAS zones and the ATC can not resolve them in timely manner.

Therefore, we would like to develop a routing plan, which generate an automated exit routes for all unauthorized GA aircraft while not disturbing commercial aircraft original routes. In this part of the research, it will implement a kinetic polygon triangulation of the CAS zone as the polygon and the GA aircraft as indices. This only works given the assumption that by 2019, the General Avitation Administration FAA requires all GA must be equipped with a Automatic Dependent Surveillance transponder (ADS-B) that communicates automatically with the ATC. It will send the flight ID, exact location and altitude once every 1 second to the ATC. According to FAA cite [4], ADS-B uses satellites instead of radars, since radars are primitive in tracking GA where it relies on estimated radio signals and antennas to determine an aircraft's location. Whereas ADS-B uses satellite signals to track aircraft movements. However, the missing output of this transponder is that it still can not determine GA future locations and way-points. We would like to use the first part of the research which is the aircraft future prediction and find exit routes from CAS that would be sent to the GA aircraft automatically. This paper is presented as the following: in Section II is the literature review in the airspace conflicts; in Section III general aviation aircraft tracking and prediction; in Section IV is the brief introduction to polygon triangulation. Section V shows the proposed resolution to multiple infringements. In Section VI is infringement resolution analysis. The conclusion and the proposed future work are presented in Section VII.

II. Literature Review on Conflict Resolution

Several researches were introduced to solve the issue of a conflict between two commercial aircraft. Fig. 2 breaks the related work in this field, where our focus is highlighted in the red rectangle.

The majority of proposed research work falls in the left side of the proposed work in the field of conflict detection. They all assume the availability of both advance transponders on-board and way-points (the direction of their trip). In [5] proposed an warning system that uses Monte Carlo simulations (MC) to estimate the probability of a possible conflict of air traffic encounters in the course of time. They assumed that there is a line of communication

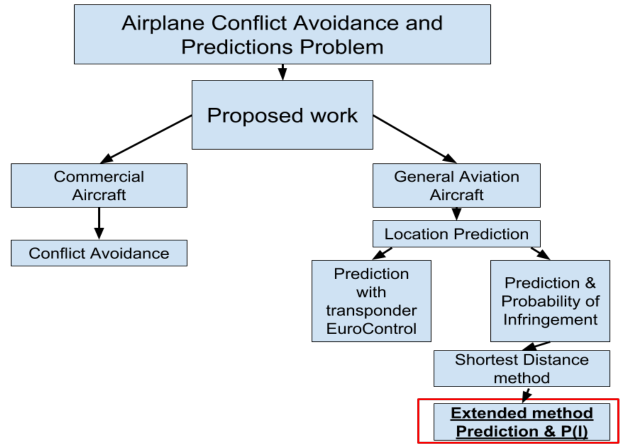

Fig. 2. Researches involve aircraft conflict detection.

between aircraft in the air. The idea of the in air communication line is to collect other aircraft information such as current location and its future trajectories (way-points), speed, heading and altitude. These information will be input to the MC machine as the initial state. Then with time the MC runs a trajectories of paths for both aircraft and predicts if there is a cross over (conflict). The MC machine then issues an alert when the host aircraft protected zone is violated by one of these trajectories. The research also proposed a protected zone around the aircraft which defines the levels warning. They divided it into four stages where 1 means far away aircraft and 4 means very close by aircraft. Here Air Traffic Controllers should take control and provide them with save manoeuvring route. They defined the size of the protected zone as a trade off between the true positives and false positives which was done by the System Operative Characteristic (SOC). Because running the MC simulation online is considered computationally expensive, in [6] they decided to reduce the amount of this computation. Here they assume since they have all the future waypoint of aircraft they can create a series of trajectory lines between these waypoints, where each endpoint can be a change of heading or speed. Then check if the one aircraft trajectory line intersects the other aircraft trajectory line. The MC simulation engine has an input of the intent information (future trajectories), current state (speed and altitude), protected zone level and their uncertainties such as tracking errors and manoeuvring characteristics. It then outputs a probability of a conflict $P(conflict)$. Another study was proposed in [7] to predict a conflict in free flight mode. Free flight mode is when the pilot and aircraft do not need the constant monitoring of the ATC and rely on the transponder to make changes in mid-air trajectories. In their paper, their method is to be applied on two aircraft travelling along a straight line with the assumption that they fly with constant errors. First, they modelled the trajectory errors as randomly distributed using live air traffic information. Then they combined both error covariances into a single error covariance relative to the their position to cancel common errors. Finally, they estimated the probability of conflict given the prediction is the area under the combined error ellipse within the extended conflict zone. Moving to the right side of the related work in conflict detection, a recent research in [8] under the SESAR WP-E project. They objective is similar to our research which is to examine a model that can predict future location of a GA aircraft using past flight data paths as an input to their model and produce future paths and how they deliver that information to ATC. Their focus was meant to familiarize the ATC with the amount of future density of flights in CAS and trigger an alert if the aircraft is approaching controlled Airspace. Their method however, assumes that the aircraft should be equipped with transponders communicating with a ground based system. This system gathers the information broadcast by the transponders and predicts the flight path ahead. Most of an unauthorized entry to CAS zones are caused by human related errors, and since the last part of the research suggests a mandatory transponder, this research method would reduce human error by introducing an automated method for CAS zone exit route sent directly to the GA aircraft transponder. In the next section, a brief introduction to prediction models implemented in the previous work is presented.

III. General Aviation Tracking and Prediction

A. *Kalman Filter*

The Kalman Filter was introduced by Rudolf Kalman back in the 60s [9]. It observes a series of measurements over time which contains random statistical noise and other inaccuracies (ex. radar errors), and estimates an unknown states that tend to be be closer to the true ones. It is considered as recursive state estimator of any linear dynamical system as shown in Fig. 3.

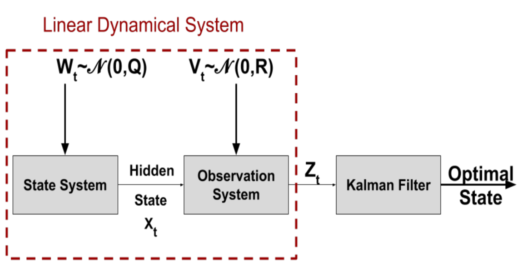

Fig. 3. Kalman Filter estimation of linear dynamical system.

It alternates between two steps: prediction and correction. The standard Kalman filter assumes the observations and the hidden states are linear and normally distributed.

where $w_{t+1} \backsim \mathcal{N}(0, Q)$; A and H are the state and observation transition matrices, $\widehat{P}$ and $\overline{P}$ are the predicted and estimated error covariances, Q and R are the system state and observation error covariances; r, S and K are residual, residual covariance and optimal Kalman Gain respectively. Finally, I is the identity matrix. In prediction, the filter predicts the location of the next state $\widehat{x}_{t+1}$ and its uncertainty $\widehat{P}_{t+1}$ given the current location state estimate $\overline{x}_t$. When the time increments $t+1$, the filter will receive a

Algorithm 1 Kalman Filter

$\widehat{x}_1 = \overline{x}_1 = z_{t=1}$, $\overline{P}_1 = \widehat{P} = Q_{t=1}$
procedure Forward Pass(z_t) ▷ observation z at time t
  for $t \leq T$ **do**
    $\widehat{x}_{t+1} = A\overline{x}_t + w_{t+1}$ ▷ Prediction
    $\widehat{P}_{t+1} = A\overline{P}A' + Q$
    $r = z_{t+1} - H\widehat{x}_{t+1}$
    $S = H\widehat{P}_{t+1} + R$
    $K = \widehat{P}_{t+1}H/S$
    $\overline{x}_{t+1} = \widehat{x}_{t+1} + Kr$ ▷ Correction
    $\overline{P}_{t+1} = (I - KH)\widehat{P}_{t+1}$
  end for
end procedure

new observation z_{t+1} (with added noise), the predicted state estimate $\widehat{x}_{t+1}$ will be updated using a weighted average. The higher the weight added to a prediction the higher certainty about its location. The KF output is a new state estimate $\overline{x}_{t+1}$ that lies between the predicted and measured state, with higher uncertainty. Because the algorithm is recursive, it runs in real time. As a result, it is useful to track and predict a moving object in time such as aircraft. Fig. 4 shows a sample track and the KF process.

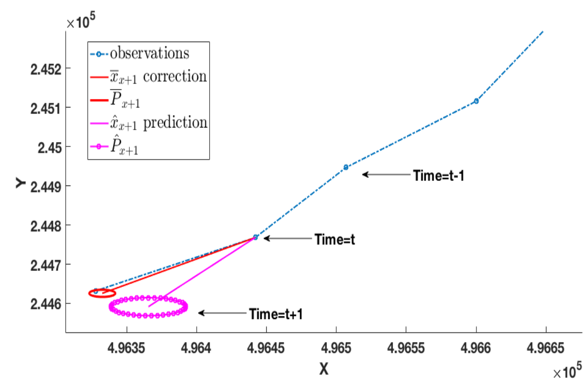

Fig. 4. Kalman Filter prediction and correction process on a sample aircraft track.

B. *Switching Kalman Filter Model*

Aircraft tend to fly into several patterns such as in a straight line, turning, acceding and descending. In these patterns, the aircraft is either in constant velocity (CV) or constant acceleration (CA). Using one static kalman filter to track and predict aircraft locations will not be accurate. It will cause the kalman filter to generate large uncertainties "errors" making the future prediction imprecise specially if the prediction is more than one step ahead. As a result, a switching Kalman filters and smoothers are implemented for each mode. The "*switching*" part is based on choosing which of the KF models has the highest probability of observations given the predictions. Where both KFs receive the same observation z_t from the same primary radar. The linear equation for the observation system is defined as the following:

$$z_t = H\overline{x}_t + v_t,\ \textit{where } v_t \backsim \mathcal{N}(0,R) \textit{ and}$$
$$H = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Both Kalman filters have the following different parameters $\lambda = (A, H, Q, R)$ where both A and H will be fixed the entire prediction process.

1) Constant Velocity KF: In this Kalman filter, the aircraft is moving in a constant velocity where there is no acceleration applied while flying. The linear equation for the CV KF state transition system is defined as the following:

$$\hat{x}_{t+1} = A_{CV}\overline{x}_t = \begin{pmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} \overline{x}_{loc} \\ \overline{x}_{Vel} \\ \overline{x}_{Acc} \\ \overline{y}_{Loc} \\ \overline{y}_{Vel} \\ \overline{y}_{Acc} \\ \overline{z}_{Loc} \\ \overline{z}_{Vel} \\ \overline{z}_{Acc} \end{pmatrix}$$

$$\hat{x}_{t+1} = \begin{pmatrix} \overline{x}_{loc} + \overline{x}_{Vel} \times T \\ \overline{x}_{Vel} \\ 0 \\ \overline{y}_{Loc} + \overline{y}_{Vel} \times T \\ \overline{y}_{Vel} \\ 0 \\ \overline{z}_{Loc} + \overline{y}_{Vel} \times T \\ \overline{z}_{Vel} \\ 0 \end{pmatrix}$$

where $T = 4$ *sec*, the time interval to observe the next location from the radar $(1RPM = 4$ *sec*$)$. However, now that the aircraft will be tracked via satellites instead of radars the next time step location will be every $T = 1$ *sec*.

2) Constant Acceleration KF: When the aircraft applies an acceleration the SKF will switch the CA KF and the linear equation for its state transition system is defined as the following:

$$\hat{x}_{t+1} = A_{CV}\overline{x}_t = \begin{pmatrix} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \overline{x}_{loc} \\ \overline{x}_{Vel} \\ \overline{x}_{Acc} \\ \overline{y}_{Loc} \\ \overline{y}_{Vel} \\ \overline{y}_{Acc} \\ \overline{z}_{Loc} \\ \overline{z}_{Vel} \\ \overline{z}_{Acc} \end{pmatrix}$$

$$\hat{x}_{t+1} = \begin{pmatrix} \overline{x}_{loc} + \overline{x}_{Vel} \times T + \overline{x}_{Acc} \times \frac{T^2}{2} \\ \overline{x}_{Vel} + \overline{x}_{Acc} \times T \\ \overline{x}_{Acc} \\ \overline{y}_{Loc} + \overline{y}_{Vel} \times T + \overline{y}_{Acc} \times \frac{T^2}{2} \\ \overline{y}_{Vel}\overline{y}_{Acc} \times T \\ \overline{y}_{Acc} \\ \overline{z}_{Loc} + \overline{z}_{Vel}T + \overline{z}_{Acc} \times \frac{T^2}{2} \\ \overline{z}_{Vel} + \overline{z}_{Acc}T \\ \overline{z}_{Acc} \end{pmatrix}$$

C. Prediction with Online Learning

Because the aircraft changes flight modes during its flight time, having a fixed SKF error covariances the entire time is not efficient. It is crucial for the SKF model to learn and update the state, prediction and observation error covariances while predicting (online). By doing so, it will minimize the errors in the state estimation,prediction while increasing the likelihood of the observation. To maximize the likelihood of our observations given the new parameters of the model, this research implemented a version of the expectation maximization algorithm (EM) method [10]. Its a recursive learning method which runs into two steps: the E-step which estimates the log-likelihood of the observations given the current parameters and the M-step which maximizes the log-likelihood estimated in the E-step by re-estimating the parameters $\theta = (R, Q)$. In this research it is the following:

1) E-Step:

- Kalman filter (forward pass)
- Kalman Smoother (backward pass)

2) M-step:

- Re-estimate both KFs (CV,CA) error parameters $\theta^* = (R, Q)$

In this section, we focus on the M-step which re-estimates the errors of the models. The derivation for parameters re-estimation in [11] is not within our research scope. However, the final equations are as follows:

$$R^*_m = \sum_{t}^{\tau} \frac{(z_t - H\tilde{\text{x}}_t)(z_t - H\tilde{\text{x}}_t)^{'} + (H\tilde{\text{P}}_t H^{'})}{\tau}$$
$$D = \sum_{2}^{\tau} (\tilde{\text{x}}_{t-1}\tilde{\text{x}}^{'}_{t-1} + \tilde{\text{P}}_{t-1})$$
$$E = \sum_{2}^{\tau} (\tilde{\text{x}}_t\tilde{\text{x}}^{'}_{t-1} + \tilde{\text{P}}\text{P}_t)$$
$$F = \sum_{2}^{\tau} (\tilde{\text{x}}_t\tilde{\text{x}}^{'}_t + \tilde{\text{P}}_t)$$
$$Q^*_m = \frac{F - EA^{'}_m - E^{'}A_m - A_m D A^{'}_m}{\tau - 1}$$

where $\tilde{\text{x}}$ and $\tilde{\text{P}}$ are the smoothed state and error covariances; (R^*_m, Q^*_m) are the new observation and state error covariances for the *m*-KF; $\tilde{\text{P}}\text{P}$ is the smoothed state error covariance between times $t-1$ and $t-2$. Finally, T is the total number of observation sequence. In this research however, the equations have been modified by adding weights defined as the loglikihood of the observations. These weights can reduce or increase error covariance faster than the original equations. In addition to the weights, we defined τ to be the last 15 steps of way-points instead of the whole track. Algorithm 2 shows the algorithm after modification where $\text{W} = Loglik(observation|stateestimate,)$.

Algorithm 2 SKF online learning

$\tau = 15, D = E = F = 0$
procedure RE-ESTIMATING($W_{T-\tau}, z_{T-\tau:T}, \widetilde{PP}_{T-\tau:T}, \tilde{x}_{T-\tau:T}, \overline{x}_{T-\tau:T}, \overline{P}_{T-\tau:T}$)

  if $T \geq 17$ **then** ▷ Check to see if there is at least a history of 17
    $t = T - \tau$
    $k = \tau$
  else
    $t = 2$
    $k = T$
  end if
  $R^*_m = \sum_t^T W_m \frac{(z_t - H\tilde{x}_t)(z_t - H\tilde{x}_t)' + (H\tilde{P}_t H')}{T}$ ▷ re-estimating R for KF m=CA,CV
  $D = \sum_t^k W_m(\tilde{x}_{t-1}\tilde{x}_{t-1}' + \tilde{P}_{t-1})$
  $E = \sum_t^k W_m(\tilde{x}_t\tilde{x}_{t-1}' + \tilde{P}P_t)$
  $F = \sum_t^k W_m(\tilde{x}_t\tilde{x}_t' + \tilde{P}_t)$
  $Q^*_m = \frac{F - EA_m' - E'A_m - A_m D A_m'}{\sum_t^k W}$ ▷ re-estimating Q for KF m=(CA,CV)
end procedure

D. Probability of Infringement and Results

Most of the researches done in the field of airspace safety involved conflicts and resolution between several commercial aircraft with advanced equipment and way-point trajectories. However, in this research it is purely focused on GA aircraft conflicts inside CAS zones that lacks these assumptions (ex: known way-point trajectories, accurate locations ... etc). A model was proposed in [12] which finds the probability of infringement of GA aircraft. For their model to work accurately they assumed the following:

1) Finding the probability of infringement *P(I)* using one step prediction based on the current heading.
2) If the GA aircraft approaches CAS is flying in constant velocity (straight line).
3) The CAS boundary its approaching is straight (not approaching a CAS corner).

The research method in [2] however, extended that method to incorporate the following:

1) Finding the probability of infringement *P(I)* using one and five steps predictions (20 seconds ahead).
2) The GA aircraft is flying in either constant velocity or constant acceleration.
3) The CAS boundary its approaching is either straight or a corner.

Also, their method used the time of day forecast to either increase or decrease the probability of infringement. In [3], it uses weather, time and day of the week to enhance the probability of infringement. The research methods of prediction and probability of infringement performance and results are presented in [2] and [3].

IV. Polygon Triangulation

Here in this section, it will provide a brief overview of the computational geometrical method called “Kinetic Polygon Triangulation” is presented and apply it to the current research.

A. Kinetic Polygon Triangulation KPT

A polygon is a shape which has line segments as “edges” connected via end points called “vertices”. Fig. 5 below shows different simple polygon shapes:

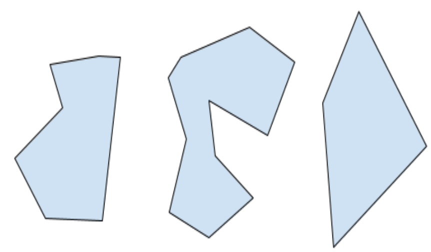

Fig. 5. Different shapes of simple polygons.

A polygon triangulation (PT) is the process of partitioning a polygon P into non-overlapping triangles. Here are the basic properties for simple polygons:

1) A simple polygon is a closed polygonal curve without self-intersection.
2) Every simple polygon admits a triangulation.
3) There is exactly $n - 2$ triangles in $n - gon$

Fig. 6 shows a triangulation of a simple polygon (real CAS zone), it has 24-gon, therefore, has 23 triangulates:

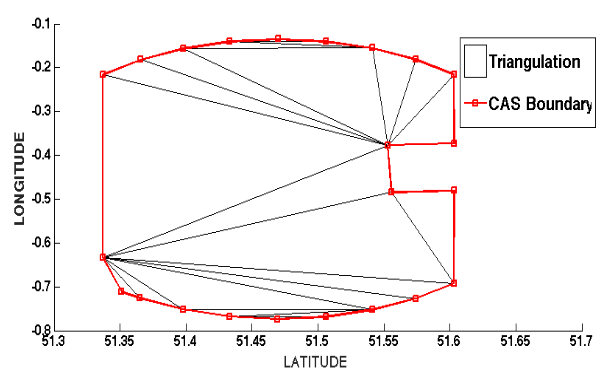

Fig. 6. Different shapes of simple polygons.

The kinetic aspect in the triangulation involves real time moving of the vertices and therefore, the triangulation changes with time to maintain the three basic properties presented above.

B. CAS Triangulation

We will apply KPT presented in IV-A to CAS zones, since CAS zones maintain different simple polygon shapes and volume sizes which have all three basic properties. Fig. 7 shows real 105 different CAS zones around London and south east coast of UK.

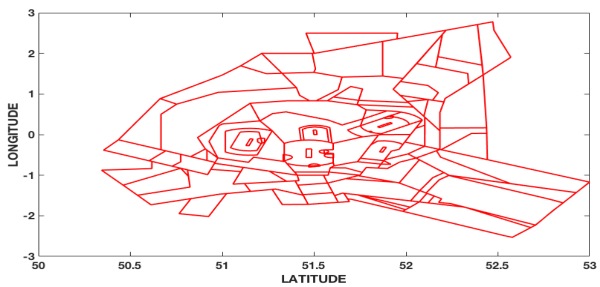

Fig. 7. Different CAS polygon shapes over UK southeast side.

V. Multiple Infringements and Resolution

In this section, we will show a real scenario where multiple GA aircraft had got inside CAS zone where there is a commercial aircraft in route for landing. The KPT will be applied on the infringed CAS zone P and the GA aircraft as vertices V_{GA} inside P At specific time t. Fig. 8 show a KPT of V_{GA} inside P .

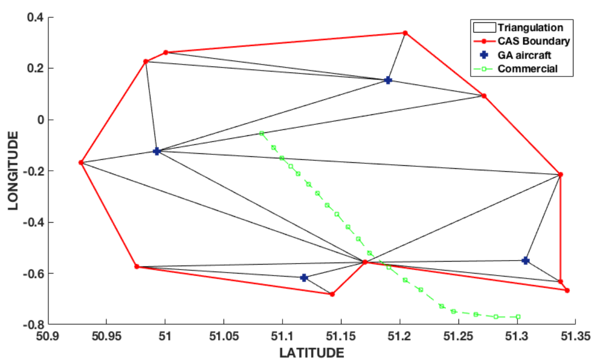

Fig. 8. KPT of CAS zone with multiple GA aircraft inside at time t.

The method proposed in this paper is to take advantage of the KPT and use the edges of the triangles as waypoints to exit CAS. To choose the exit route edge after triangulating $P = CAS$ it should follow two simple rules:

- No aircraft share the same exit route (edge).
- Each edge that intersect commercial aircraft way-point will not be removed from the exit route (not usable edge).

A. Conflict Routes

Before determining which edges GA aircraft should take as way points, we first need to eliminate edges that intersect commercial aircraft way-points segments "edges" inside P.

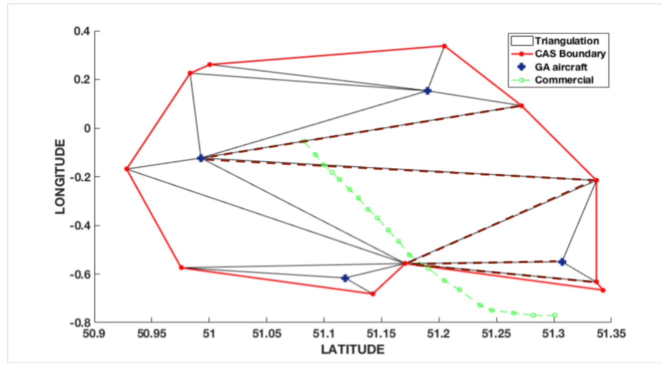

Fig. 9. KPT of CAS zone with edges marked in "- -" as not suitable for exit route.

Fig. 9 shows which edges are flagged "Not an exit route" or "conflict route":

To find "conflict edges" we will use the line segment intersection test between these edges and all other edges. We first need to assume that the line segments to be tested are in general position, meaning the following:

- No three endpoints are collinear (three points lie on same straight line).
- No two endpoints have the same x-coordinate. Specifically, no segment should be absolute vertical.
- No segment with no two points "just a point", and no two segments share an endpoint.

The naive way to find intersections is brute force where the algorithm checks every segments against all of them which takes $O(n^2)$ where n is the total number of line segments in polygon P. The algorithm used here for the line segment intersection test is called *sweep line algorithm* which takes $O((n + k)logn)$, where k is the number of intersections (output) which in this case is conflict edges. This algorithm is the faster way to find intersections between multiple line segments instead of only comparing two segments [13]. The line sweep algorithm begins by sorting all segment endpoints 2n (2 endpoints for each segment) along x-axis. It then passes a sweeping line from left to right, checking at each endpoint (event) and compares above and bellow the current even.It takes O(n log n) in time. Fig. 10 shows the sweep line process on a polygon (real CAS zone):

This algorithm uses Binary search tree and a queue to store segment labels. That what made it attractive since the running time will depend not only on the number of vertices but also on the output (number of intersections).

B. Automated Re-Routing

After identifying the conflict routes, the proposed model in this paper should transmit exit routes to the GA aircraft inside CAS zone. Since each vertex in P has least two edges connected to it. The decision rules on which edge the aircraft should follow depend on the following:

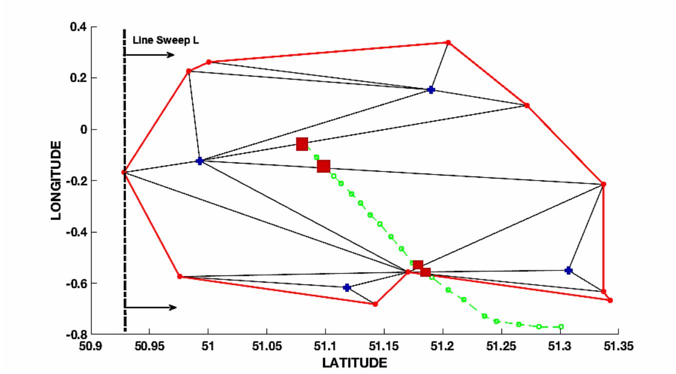

Fig. 10. Sweep line l passes through the plane to find intersections marked in ■.

1) The chosen exit route will be the shortest distance path to CAS boundary vertex.
2) The exit route "edge" should be used only by one GA aircraft (no two or more aircraft share the exit route).

Fig. 11 and 12 shows the direction of the chosen exit routes represented by the arrows "$\longrightarrow$".

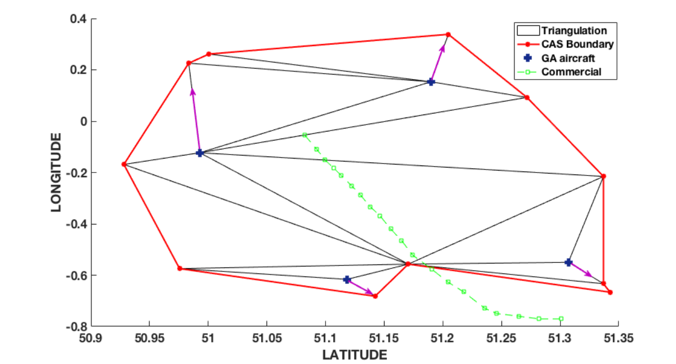

Fig. 11. Exit routes for 4 GA aircraft inside CAS.

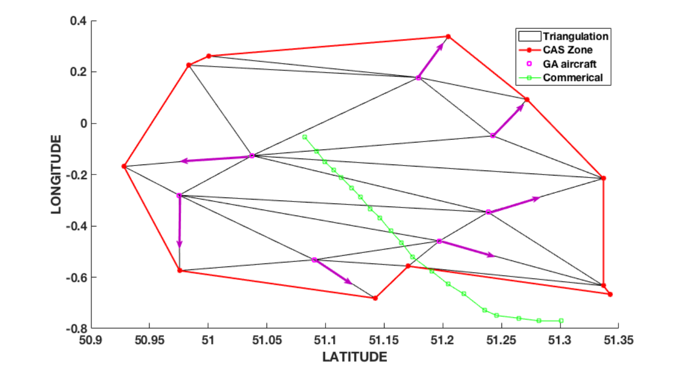

Fig. 12. Exit routes for 7 GA aircraft inside CAS.

If there are multiple commercial and GA aviation in one CAS zones, we then need to be flexible with the first decision rule and choose the one that is not conflict with other GA aircraft.

VI. Analysis

The method will work if the three basic properties in Section IV-A hold and no three GA aircraft are collinear in the space (all three in the same line segment). Fortunately, we are dealing with large geographical map locations it would be rare to find all three aircraft would overlap each other *exactly* in either (latitude or longitude). Furthermore, with the use of satellites instead of radars for tracking, the prediction accuracy in the model would increase, therefore, we use the GA aircraft prediction location at $t = 4$ to 20 sec ahead as vertices in P and not the current location at time t. This will reduce the frequency of triangulation to every 4 to 20 seconds instead of every 1 second. And by doing so it will give more time for the model to estimate and generate routes accordingly.

VII. Conclusion

The objective of this research is to investigate and build a possible hybrid model that can predict future infringements accurately. Therefore, a hybrid model was implemented to track aircraft while learning their error covariances. In the first two parts of this research [2], [3] they have shown promising results so far. They were dealing with with one GA aircraft infringing CAS zones. In reality however; there could be multiple infringements occur simultaneously. The ATC will not be able to resolve them on time, therefore, in this paper we studied this scenario and proposed a method were it can simplify the ATC manoeuvring tasks by finding an automated way to re-route GA aircraft towards the UCAS or to another CAS zone with less aircraft density within it. This method was proposed under the assumption that the FAA mandating every GA aircraft must be equipped with an ADS-B transponder. In this paper, it took advantage of the geographical map of CAS zones and computational geometry algorithms such as polygon triangulations and line intersection tests using Line segment sweep algorithm. This algorithm takes $O((n+k)logn)$ in running time which gave the model time to implement the exit routes. The method in this paper "polygon triangulation" and finding routes in space can be applied on drones. For instance, Amazon inc is looking to switch their delivery system from commercial air and ground shipping to drones, however, they are looking to implement a grid in the space dedicated for drones to travel with no conflicts with other flying objects such as GA aircraft, commercial aircraft and other drones.

References

[1] "General aviation airspace infringement survey." [Online]. Available: http://www.eurocontrol.int

[2] Y. S. Almathami and A. Reda, "Probabilistic controlled airspace infringement tool," in *in Signal Processing and Information Technology (ISSPIT) 2015*, 2015.

[3] Y. Almathami and R. Ammar, "Controlled airspace infringements and warning system," 2016, pp. 45–50.

[4] F. A. Administration, "Nextgen equip ads-b," 2015. [Online]. Available: https://www.faa.gov/nextgen/equipadsb/

[5] L. C. Yang, J. K. Kuchar, L. C. Yang, and J. K. Kucharf, "Prototype conflict alerting system for free flight," *Journal of Guidance, Control, and Dynamics*, pp. 768–773, 1997.

[6] L. C. Yang and J. K. Kucharf, "Using intent information in probabilistic conflict analysis," in *in AIAA Guidance, Navigation, and Control Conf*, 1998.

[7] R. A. Paielli and H. Erzberger, "Conflict probability estimation for free flight," *AIAA JOURNAL OF GUIDANCE CONTROL AND DYNAMICS*, vol. 20, pp. 588–596, 1997.

[8] C. L. Tallec, D. Taurino, C. Lancia, and J. Verstraeten, "Predicting the future location of a general aviation aircraft," NLR Air Transport Safety Institute and Onera and Deep Blue, Nederlands, Tech. Rep., 2014.

[9] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, vol. 39, no. 1, pp. 1–38, 1977.

[11] R. H. Shumway and D. S. Stoffer, "An approach to time series smoothing and forecasting," *Journal of Time Series Analysis*, vol. 3, no. 4, pp. 253–264, 1982.

[12] K. Mcdonald-Wallis, "An approach into the probabilistic prediction of the movement of uncontrolled aircraft to improve uk aviation safety," Master's thesis, University of Southampton, UK, 2009.

[13] M. d. Berg, O. Cheong, M. v. Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed. Santa Clara, CA, USA: Springer-Verlag TELOS, 2008.