

Failure Boundary Estimation For Lateral Collision Avoidance Manoeuvres

James Dunthorne¹, Wen-Hua Chen and Sarah Dunnett

Abstract—This paper proposes a method for predicting the point at which a simple lateral collision avoidance manoeuvre fails. It starts by defining the kinematic failure boundary for a range of conflict geometries and velocities. This relies on the assumption that the ownship aircraft is able to turn instantaneously. The dynamics of the ownship aircraft are then introduced in the form of a constant rate turn model. With knowledge of the kinematic boundary, two optimisation algorithms are used to estimate the location of the real failure boundary. A higher fidelity simulation environment is used to compare the boundary predictions. The shape of the failure boundary is found to be heavily connected to the kinematic boundary prediction. Some encounters where the ownship aircraft is travelling slower than the intruder were found to have large failure boundaries. The optimisation method is shown to perform well, and with alterations to the turn model, its accuracy can be improved. The paper finishes by demonstrating how the failure boundary is used to determine accurate collision avoidance logic. This is expected to significantly reduce the size and complexity of the verification problem.

Keywords: UAVs, Collision Avoidance, Sense & Avoid, Failure Boundary Estimation, Clearance, Safety, Verification

I. INTRODUCTION

For years the military has been using Unmanned Aerial Vehicles (UAVs) to help protect ground troops. Typically they have been operated within military controlled airspace, which has allowed the incorporation of greater functionality, without all of the regulatory hurdles that are present within the civilian world [1]. It is not beneficial to constrain commercial use of UAVs to segregated airspace as this is very costly and time consuming. In order to fly UAVs in non-segregated airspace, one of the challenges is to develop and certify a collision avoidance system which is capable of replicating the ability of a human pilot to “See & Avoid” other airspace users, when other forms of separation have failed.

Collision avoidance systems are safety critical in the sense that failure could result in a catastrophic accident. Because UAVs are operated remotely, there is no guarantee that the pilot will be able to take responsibility for resolving the collision. In some circumstances, such as loss of data link, full control must be handed to the UAV to avoid a conflict, and so these systems need to be verified to a very high degree of confidence.

¹James Dunthorne, Wen-Hua Chen and Sarah Dunnett are from the Aeronautical and Automotive Engineering Department at Loughborough University, Loughborough, United Kingdom, LE11 3TU
j.dunthorne@lboro.ac.uk w.chen@lboro.ac.uk
s.j.dunnett@lboro.ac.uk www.lucasresearch.co.uk

Recent research on the clearance of flight control and collision avoidance systems has focused on optimisation based methods [2], [3]. Computational load increases exponentially with the number of problem dimensions, and so their use is currently restricted to smaller problems. In order to verify lateral collision avoidance systems, we must first try to decompose the problem. Predicting when a system is likely to fail has historically been used to help certify aircraft systems. A stall warning system is a good example of this [4].

This paper proposes a method to predict the point at which a simple lateral collision avoidance manoeuvre fails to separate conflicting aircraft adequately. This is referred to in this paper as Failure Boundary Estimation (FBE) and has already been identified as an important tool in conflict scenarios [5]. In being able to predict the failure boundary, accurate collision avoidance logic and safe operating regions can be determined. This drastically reduces the size and complexity of the verification problem.

II. COLLISION AVOIDANCE ALGORITHM SELECTION

Collision avoidance algorithms were reviewed so that the most promising solutions could be identified [6]. These included procedural resolutions [7], potential fields [8], optimisation based methods [9], intelligent learning algorithms [10] and geometric/ collision-cone based methods [11]. The geometric methods were simple, intuitive and mature, and so this group was analysed further. It was found that many of these methods collapsed to the same solution when an equal number of problem dimensions were considered.

The approach by Goss [11] was found to express the geometric method solution attractively. It calculates the heading command needed to transform the relative velocity vector V_R outside of the intruders cone of collision, as described in [12]. This expression can be analysed to find when the mathematics breaks down in order to give an insight into the problem boundaries. But first is a brief explanation of how the collision avoidance algorithm works.

A. Algorithm Overview

The algorithm in [11] calculates 4 heading commands, one of which corresponds to a manoeuvre which moves the relative velocity vector to the right hand edge of the intruder’s safety bubble (clockwise rotation), as shown in Figure 1. This shall be referred to as a right hand edge (RHE) manoeuvre. The other three solutions are not considered in this paper, as they represent either symmetrical or sub-optimal resolutions.

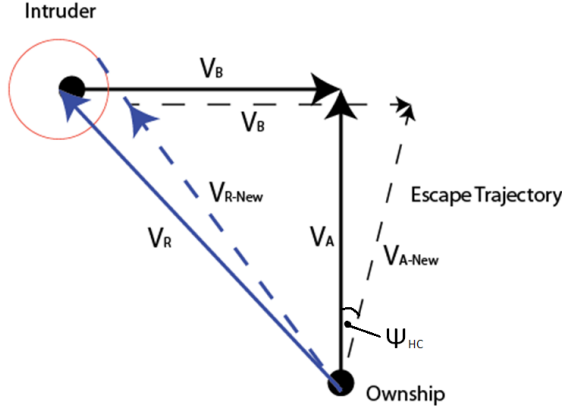


Fig. 1. Geometric algorithm description

Most regulatory guidelines specify the shape of the intruders safety bubble to be cylindrical and so the problem can be simplified to just two spacial dimensions, x and y . The heading command for a RHE manoeuvre is simplified and shown in Equation (1).

$$\psi_{HC} = \beta + \gamma + \sin^{-1} \left(\frac{V_B}{V_A} \sin(\psi_{LAB} - \beta - \gamma) \right) \quad (1)$$

where,

$$\gamma = \sin^{-1} \left(\frac{R}{d} \right) \quad (2)$$

V_A and V_B are the magnitude of the ownship and intruder aircraft's velocities. R is the horizontal safety bubble radius as defined by local air regulation (normally 500ft), and d is the distance between the two aircraft. β is defined in Figure 2 and is measured between the limits $-\pi$ to π .

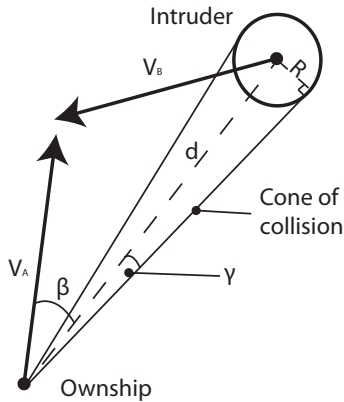


Fig. 2. Definition of encounter variables

ψ_{LAB} is the difference between the intruder's heading, ψ_B and ownship aircraft's heading, ψ_A , and is measured between 0 and 2π .

$$\psi_{LAB} = \psi_B - \psi_A \quad (3)$$

III. KINEMATIC FAILURE BOUNDARY

By inspecting Equation (1) we can find when the algorithm gives an imaginary solution. These cases define the point in an encounter where it becomes impossible to resolve the collision using just the RHE manoeuvre, even if the ownship aircraft were able to turn instantaneously. There are two cases when this happens:

A. Case 1

The first case arises when:

$$d < R \quad (4)$$

The implication of this is not very significant, but means that if the ownship aircraft is already within the intruders safety bubble, a safe resolution trajectory is not available.

B. Case 2 - Kinematic Boundary

For the case where $d \geq R$, the end term of Equation (1) causes a trajectory to give an imaginary solution when:

$$\frac{V_B}{V_A} \sin(\psi_{LAB} - \beta - \gamma) > 1 \quad (5)$$

The maximum value a sine function can take is +1. Therefore the kinematic boundary can only exist for cases where:

$$V_A < V_B \quad (6)$$

C. Limits

Since we are only considering a RHE manoeuvre, γ must be positive. Therefore γ is constrained as follows:

$$0 \leq \gamma \leq \frac{\pi}{2} \quad (7)$$

By re-arranging Equation (5), and introducing the limits above, we obtain the limits of the kinematic boundary.

$$\pi - \sin^{-1} \left(\frac{V_A}{V_B} \right) \leq (\psi_{LAB} - \beta) \leq \frac{3\pi}{2} - \sin^{-1} \left(\frac{V_A}{V_B} \right) \quad (8)$$

D. Kinematic Boundary Distance

To calculate the distance, d_{KIN} , at which the kinematic boundary is breached, we re-arrange Equation (5). When performing the inverse sine operation, two solutions are produced, θ and $\pi - \theta$.

$$\theta = \psi_{LAB} - \beta - \sin^{-1} \left(\frac{R}{d} \right) \quad (9)$$

When within the limits as described in the previous section, the $\pi - \theta$ solution must be used.

$$d_{KIN} = \frac{R}{\sin \left(\sin^{-1} \left(\frac{V_A}{V_B} \right) + (\psi_{LAB} - \beta - \pi) \right)} \quad (10)$$

If the lower limit is breached, the θ solution is used.

$$d_{KIN} = \frac{R}{\sin\left(\sin^{-1}\left(\frac{V_A}{V_B}\right) + (\psi_{\angle AB} - \beta)\right)} \quad (11)$$

If the upper limit is breached, only the boundary as described in Case 1 applies (i.e. $d = R$).

E. Kinematic Boundary Heading

When Case 2 is satisfied, the end term of Equation (1) reduces to $\frac{\pi}{2}$. Substituting in the boundary distance, d_{KIN} , we are able to find the kinematic boundary heading, ψ_{KIN} .

$$\psi_{KIN} = \psi_{\angle AB} + \sin^{-1}\left(\frac{V_A}{V_B}\right) - \frac{\pi}{2} \quad (12)$$

This angle corresponds to a trajectory which maximises the separation between the two aircraft, and it sits perpendicular to the new relative velocity vector. Turning beyond this angle reduces separation and so it is used to constrain the turn.

F. Analysis

Figure 3 shows a plot of the kinematic boundary distance, d_{KIN} , against the encounter geometry, $\psi_{\angle AB}$, for three different velocity ratios ($\frac{V_A}{V_B} = 0.2, 0.6$ and 1.0). It is assumed that the aircraft are involved in a direct collision so that β can be determined from velocity vectors. Limits, as described in the previous sections, have been applied.

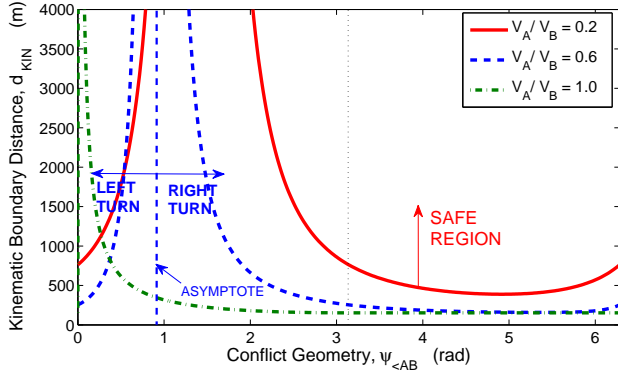


Fig. 3. Kinematic Boundary Distance Variation for Different Velocity Ratios

The left hand side of the graph corresponds to resolutions which would be commonly referred to as overtaking manoeuvres, where the two aircraft's trajectories tend to converge on similar headings. These types of resolution tend to have a much larger kinematic boundary distance, so are best avoided when at low speeds.

At the point where $\psi_{KIN} = 0$, the ownship aircraft is already travelling in the direction of the kinematic boundary heading, and perpendicular to the direction of the relative velocity vector. In this situation it is impossible for a RHE manoeuvre to increase the separation between the two aircraft without changing speed. This results in an asymptote at the point:

$$\psi_{\angle AB} = \frac{\pi}{2} - \sin^{-1}\left(\frac{V_A}{V_B}\right) \quad (13)$$

For kinematically constrained resolutions (where $V_A < V_B$), those encounter geometries to the left of the asymptote correspond to conflicts where the intruder is approaching from behind. In these cases the RHE manoeuvre causes the ownship aircraft to avoid the intruder by altering its heading to the left (i.e. $\psi_{KIN} < 0$). For all encounters to the right of the asymptote, a right hand turn is instigated (i.e. $\psi_{KIN} > 0$) as shown in Figure 3. If the ownship aircraft is travelling faster than the intruder, a RHE manoeuvre always results in a right hand turn.

IV. TRAJECTORY MODELLING

The kinematic boundary is based on the assumption that an instantaneous turn can be made. In order to produce a more realistic estimate of the real failure boundary, a turn model is required. The ownship aircraft's current position and velocity vectors are taken as a reference for these projection models. The position of the ownship aircraft through a manoeuvre can be split into three stages as shown in Figure 4:

- 1) Decision Making
- 2) Manoeuvring
- 3) Kinematic Trajectory

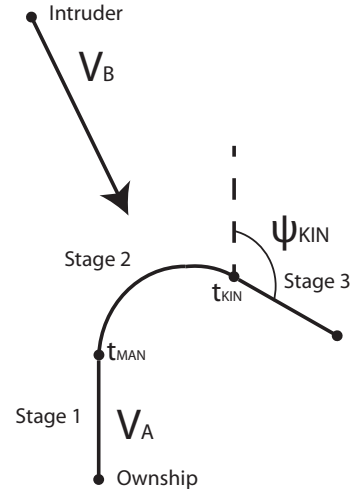


Fig. 4. Kinematically constrained failure boundary resolution

A. Stage 1 - Decision Making

The first stage models the ownship aircraft's position with time up until the manoeuvre is instigated. This can be estimated using a linear trajectory.

For

$$0 < t \leq t_{MAN} \quad (14)$$

$$P_A(t) = \begin{pmatrix} 0 \\ V_A t \end{pmatrix} \quad (15)$$

where P_A is the ownship aircraft's position with projected time, t .

B. Stage 2 - Manoeuvring

For the second stage, a simple constant turn rate model is used to test the technique, however a more comprehensive model can be introduced at a later stage to improve the trajectory estimation. The stage operates between:

$$t_{MAN} < t \leq t_{KIN} \quad (16)$$

The time taken to turn to the kinematic heading can be calculated as follows:

$$t_{TURN} = \frac{\psi_{KIN}}{\dot{\psi}_A} \quad (17)$$

where,

$$\dot{\psi}_A = \frac{g \tan(\phi_{A,MAX})}{V_A} \quad (18)$$

$\phi_{A,MAX}$ is the ownship aircraft's maximum bank angle. A nominal value of 60° is chosen for this work. g is the acceleration due to gravity.

The time at which the manoeuvring phase ends is therefore:

$$t_{KIN} = t_{MAN} + t_{TURN} \quad (19)$$

To calculate the ownship aircraft's position through the turn, we can use the radius of the turn.

$$R_{turn} = \frac{V_A^2}{g \tan(\phi_{A,MAX})} \quad (20)$$

The ownship aircraft's position through a turn, as a function of time, t , can be calculated.

$$P_A(t) = \begin{pmatrix} R_{turn} - R_{turn} \cos(\dot{\psi}_A(t - t_{MAN})) \\ R_{turn} \sin(\dot{\psi}_A(t - t_{MAN})) + V_A t_{MAN} \end{pmatrix} \quad (21)$$

C. Stage 3 - Kinematic Trajectory

The final stage requires a linear trajectory to be added to the position at the end of Stage 2, $P_A(t_{KIN})$. This linear trajectory is directed along the kinematic boundary heading.

For,

$$t > t_{KIN} \quad (22)$$

$$P_A(t) = P_A(t_{KIN}) + \begin{pmatrix} V_A(t - t_{KIN}) \sin(\psi_{KIN}) \\ V_A(t - t_{KIN}) \cos(\psi_{KIN}) \end{pmatrix} \quad (23)$$

D. Intruder's Trajectory Model

An assumption is made that the intruder aircraft maintains its current speed and heading. The intruder's position can then be modelled by a simple linear trajectory model similar to Stage 3 of the ownship model.

The intruder's initial position, $P_{B,0}$ is calculated using the intruder's bearing from the ownship aircraft, β_0 , and the distance between the vehicles, d_0 .

$$P_{B,0} = \begin{pmatrix} d_0 \sin(\beta_0) \\ d_0 \cos(\beta_0) \end{pmatrix} \quad (24)$$

The future position of the intruder is therefore:

$$P_B(t) = P_{B,0} + \begin{pmatrix} V_B t \sin(\psi_{\angle AB}) \\ V_B t \cos(\psi_{\angle AB}) \end{pmatrix} \quad (25)$$

V. FAILURE BOUNDARY ESTIMATION

The FBE method involves two key stages:

- 1) Minimum Distance Estimation - Finding the minimum distance between the two aircraft given that the ownship aircraft manoeuvres at a specific time.
- 2) Failure Boundary Search - Finding the manoeuvring time which causes the two aircraft to miss each other by a distance equal to the safety bubble radius, R .

Both of these searches are done using simple optimisation algorithms as shown in Figure 5.

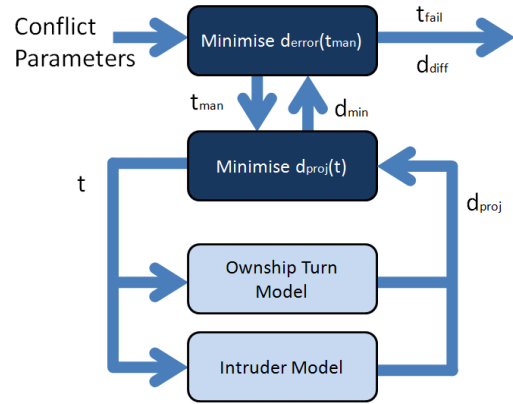


Fig. 5. Failure Boundary Estimation Process

A. Minimum Distance Estimation

In order to search for the minimum distance, d_{min} , between the two aircraft if the ownship manoeuvres at time t_{MAN} , we can use an optimisation function in Matlab such as FMINCON. This allows you to find a local minimum of a constrained non-linear multi-variable function.

$$\min f(t) \text{ such that } |A \cdot t \leq b \quad (26)$$

Our objective function $f(t)$ in this case is the projected distance between the two aircraft, d_{proj} . This can be calculated using trigonometry.

$$d_{proj}(t) = \sqrt{(P_{By}(t) - P_{Ay}(t))^2 + (P_{Bx}(t) - P_{Ax}(t))^2} \quad (27)$$

subject to the following time constraint:

$$t \geq 0 \quad (28)$$

The SQP algorithm was chosen as the solver as it converges quickly and reliably. It is started at the point $t = 0$.

The optimisation function outputs 2 values, the time at the point of closest approach, t_{min} , and the distance at the point of closest approach, d_{min} , which is used to search for the failure boundary.

B. Failure Boundary Search

The second component searches for the failure boundary by finding the manoeuvring time, t_{MAN} , which gives a minimum distance, d_{min} , equal to the safety bubble radius, R . We can also set this up as an optimisation problem, with the objective function:

$$d_{error}(t_{MAN}) = |R - d_{min}(t_{MAN})| \quad (29)$$

subject to the following constraints:

$$t_{MAN} \geq 0 \quad (30)$$

The SQP algorithm is once again chosen as the solver, and the function is started at the point $t_{MAN} = 0$.

This function outputs 2 values, the amount of time prior to the failure boundary being reached, t_{fail} , and the value of d_{error} at that point.

VI. SIMULATION RESULTS

A higher fidelity simulation environment is used to assess the effectiveness of the kinematic and FBE method. An encounter model generates collisions with varying geometries and velocities. Two simple PID controllers are implemented to control roll angle, ϕ_A , and heading, ψ_A . An F2B Bristol Fighter's [13] dynamic model is used for the roll dynamics. The turn dynamics are modelled separately:

$$\psi_A = \int_0^t \frac{g \tan(\phi_A)}{V_A} dt \quad (31)$$

Equation (1) is used to directly command the aircraft from the beginning of the simulation. The simulated failure boundary is found using a simple search algorithm which minimises the time to collision, whilst ensuring resolution success (i.e. $d_{min} \geq R$). All failure boundary times are displayed relative to the point of collision to make their interpretation easier:

$$t_B = t_{TC} - t_{fail} \quad (32)$$

t_B is the failure boundary time as shown in Figures 6 7 and 8, and t_{TC} is the known time to collision.

Results were conducted on a Dell laptop with an i5 processor, and each boundary point took approximately 0.5 seconds to compute.

A. High Speed Scenario

For the high speed scenario, the aircraft's velocities are set as follows: $V_A = 125m/s$ and $V_B = 25m/s$. The kinematic boundary, and FBE methods are compared to the boundary found from the simulation environment. See Figure 6.

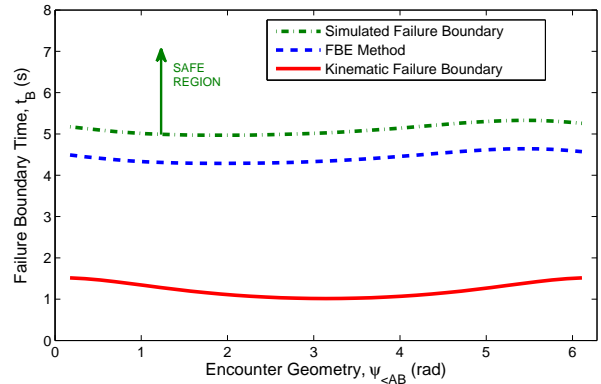


Fig. 6. Constant turn rate model, high speed scenario results

In high speed conflicts the boundary distance tends to be fairly flat across all conflict geometries. This is because the ownship aircraft's dynamics dominate the problem. It can be seen that the FBE method provides a good prediction of the boundary found from simulation. In a high speed scenario, the intruder is always in front of the ownship aircraft. Both LHE and RHE manoeuvres are available for most of the encounter.

B. Low Speed Scenario

For the low speed scenario we set $V_A = 25m/s$ and $V_B = 125m/s$. The results are presented in Figure 7.

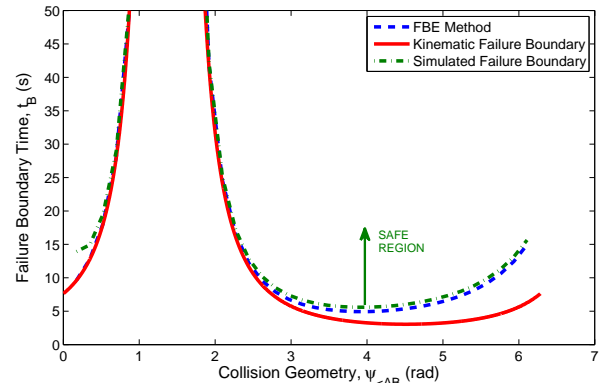


Fig. 7. Constant turn rate model, low speed scenario results

At low speeds the failure boundary is much larger due to the growth of the kinematic boundary. The location of the asymptote is predicted very well and the FBE method provides a good prediction of the true boundary when compared with the simulation results. When the intruder is on the right hand side of the ownship aircraft ($\psi_{\angle AB} > \pi$) the RHE manoeuvre is much more effective.

C. Matching Speed Scenario

For this scenario, both aircraft are set to a speed of 75 m/s. The results are presented in Figure 8.

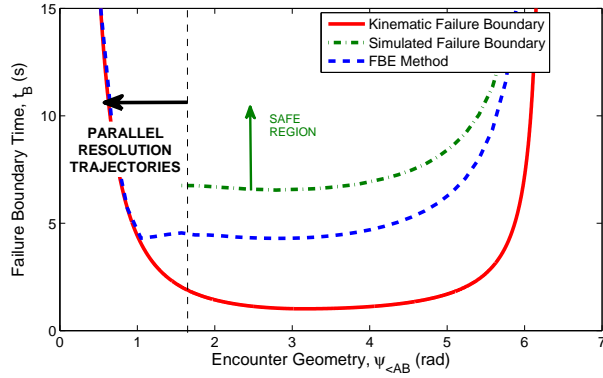


Fig. 8. Constant turn rate model, same speed scenario results

The FBE method is less accurate for these types of encounter due to the inaccuracies that come from the constant turn rate model. Improving the model will improve the prediction. At small values of ψ_{LAB} the aircraft get stuck and end up flying along parallel trajectories. This is because the kinematic boundary heading, $\psi_{KIN} = \psi_{LAB}$. To overcome this problem the heading command would need to be constrained such that a minimum relative velocity is maintained during the resolution. This boundary re-calculation will be covered in future work.

D. Collision Avoidance Logic

Knowledge of the failure boundaries can be used to determine the correct collision avoidance logic. The boundary of the manoeuvre which moves the relative velocity vector to the left hand edge of the safety bubble (a LHE manoeuvre), is found by reflecting the graph through the point $\psi_{LAB} = \pi$. Both of the boundary times are converted to distances by dividing the times by the relative velocity. The failure boundaries for both manoeuvres are plotted in Figure 9 for the Low Speed Scenario.

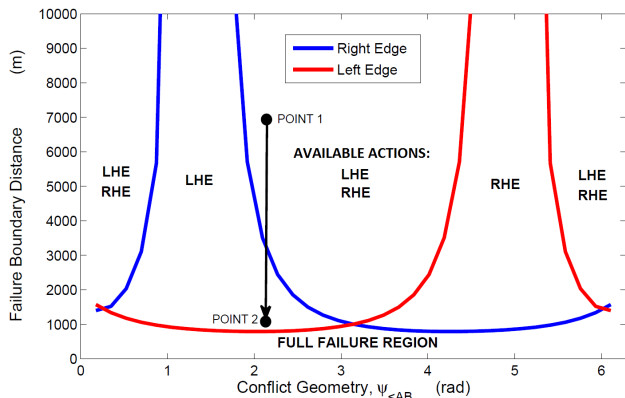


Fig. 9. Application of Failure Boundary Estimation to Determine Collision Avoidance Logic

Each area of the graph is assigned different logic states depending upon the boundary predictions.

If a collision were to be detected at Point 1 ($d = 7km$), both manoeuvres would be available, and the UAV should

maintain its heading according to ROA. At Point 2 ($d = 1.2km$) the RHE manoeuvre would no longer be available, and it would no longer be safe to follow ROA, so the UAV would be instructed to perform a LHE manoeuvre before the full failure region is entered.

VII. CONCLUSIONS & FUTURE WORK

In cases where $V_A < V_B$, the failure boundary was shown to grow much larger due to expansion of the kinematic boundary. In these scenarios a RHE manoeuvre always fails to resolve the encounter when the intruder is approaching directly from the ownship aircraft's left hand side. Encounters where $V_A \gg V_B$, were found to offer a fairly flat failure boundary. Some conflicts where the aircraft were travelling at similar speeds were shown to result in parallel resolution trajectories. A FBE method is developed using optimisation and turn models to predict the failure boundary and is shown to perform well for simple encounters. Knowledge of the failure boundary is then used to correctly formulate collision avoidance logic.

Future work will involve increasing the fidelity of the turn model in order to improve the accuracy of the FBE method. Where the velocities of the aircraft are similar, turns will be constrained by a relative velocity limit and the failure boundary will be re-calculated to ensure aircraft do not fly on parallel trajectories. Finally the FBE method will be validated in X-Plane and realistic uncertainty shall be introduced.

REFERENCES

- [1] Civil Aviation Authority, CAP722 - Unmanned Aircraft System Operations in UK Airspace, Guidance
- [2] A. Varga and A. Hansson, Optimization Based Clearance Of Flight Control Laws. Lecture Notes In Control And Information Sciences, Springer 2012
- [3] S. Srikanthakumar, C. Liu and W. H. Chen, Optimization-Based Safety Analysis of Obstacle Avoidance Systems for Unmanned Aerial Vehicles. Journal of Intelligent & Robotic Systems, January 2012, Volume 65, Issue 1-4, p. 219-231
- [4] E. L. Wiener, Beyond the Sterile Cockpit. Human Factors: The Journal of the Human Factors and Ergonomics Society, February 1985, vol. 27, no. 1 p. 75-90
- [5] M. Perez-Batlle, E. Pastor, X. Prats, P. Royo, and R. Cuadrado, Maintaining separation between airliners and RPAS in non-segregated airspace. ICARUS Research Group, Technical University of Catalonia
- [6] J. R. Dunthorne, W. H. Chen and S. Dunnett, Collision Avoidance Failure Boundary Identification For The Clearance of Civil Unmanned Aircraft. 2nd Year Internal Report, Loughborough University 2013
- [7] S Degen, Reactive image-Based Collision Avoidance System for Unmanned Aircraft Systems. Queensland University of Technology, May 2011
- [8] K. Sigurd and J. How, UAV trajectory Design Using Total Field Collision Avoidance. MIT 2003
- [9] S. L. Waslander, G. Inalhan and C. J. Tomlin, Decentralized Optimisation Via Nash Bargaining. 2004 Stanford
- [10] N. Durand, J. Alliot and J. Noailles, Collision avoidance using neural networks learned by genetic algorithms. IEA-AEI 96
- [11] J. Goss and R. Rajvanshi and K. Subbarao, Aircraft Conflict Detection and Resolution using Mixed Geometric and Collision Cone Approaches. American Institute of Aeronautics and Astronautics
- [12] A. Chakravarthy and D. Ghose, Obstacle Avoidance in a Dynamic Environment: A Collision Cone Approach. Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, Sep. 1998, V.28, Num 5, p. 562-574
- [13] T. I. Fossen, Mathematical models for control of aircraft and satellites. Department of Engineering Cybernetics Norwegian University of Science and Technology, 2011.