Proceedings of the EuroGNC 2013, 2nd CEAS Specialist Conference
on Guidance, Navigation & Control, Delft University of Technology,
Delft, The Netherlands, April 10-12, 2013

Cooperative Autonomous Collision Avoidance System for Unmanned Aerial Vehicle

Yazdi Ibrahim Jenie, Erik-Jan van Kampen, Bart Remes

Abstract Autonomous collision avoidance system (ACAS) was defined and investigated in this paper to support UAVs integration to the national airspace system. This includes not only UAVs on-board system, but also the definition of requirements, collision avoidance structure, and the avoidance rules. This paper focuses on the cooperative avoidance, where UAVs (or any aircraft) involved avoid each other using rules previously agreed by involved parties. A novel algorithm of avoidance was developed, named as Selective Velocity Obstacle (SVO) method. Several simulations were conducted and show satisfying result on how well the algorithm work to avoid separation violations. In the end of the paper, using Monte Carlo simulation, violation probabilities were derived for three setups. These simulations shows the performance of the developed algorithm for cooperative ACAS, and suggesting the need to derive a new parameter the minimum required turning rate of avoidance.

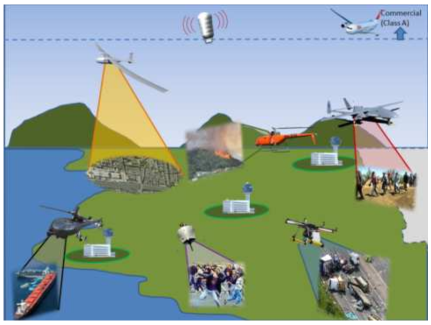
1 Introduction

Like other technologies which were first started at a military base, UAVs will start affecting civilian live in just a couple years from now. Several industries even has been erected and commercially provides low end UAVs technologies for various non-military purpose, most of them are recreational and remote-controlled toys and fly in a secluded area with minimum impact on the airspace. However, with the fast advancement of technology, Civilian-UAVs are not just toys anymore. The variation of mission that a UAV could handle became large, that government department like Police and Fire Brigade began to count the possibility of deploying UAVs more often, in a non-secluded area. DeGarmo and Nelson [4] give several predictions on what will become of UAVs in the future that affects civilians life, each of them will exposed a certain level of danger. The discussion

of UAVs (or UAS in wider term) being used by non-military purposes become a topic of integrating UAS into the National Airspace System (NAS).

In order to be used widely in the National Airspace System, Unmanned Aircraft System required to be able to demonstrate an equivalent level of safety. This includes a solid definition of its Collision Avoidance System, which should be applicable not only between UAVs, but also take into account the already-settled manned aircraft traffic. To simplify the problem, the system could be divided into two parts based on how the UAVs reacts when there's possibility of collisions, which are cooperative and non-cooperative collision avoidance system.

Fig. 1 DeGarmo and Nelson [4] predictions on what will become of UAVs in the future that affects civilians' life.



Thus, the research presented in this paper aims **to define and investigate the collision avoidance system for UAVs, in context of integrating UAVs into the National Airspace System**. This includes not only the UAVs on-board system, but also the definition of requirements, collision avoidance structure, and the avoidance rules. A mathematical model also being developed to simulate the capability of the defined system, along with several parameter derivations that described the systems level of safety. It will become clear in chapter 2 that there will be two main part of the collision avoidance structure, the cooperative and non-cooperative avoidance. This paper, however, only focussed on global structure and the cooperative part of the system. The other part will be included in the continuation of this research.

This paper presents the research as follows. After this introduction, the second chapter discuss the derivation of collision avoidance structure designed for UAVs to integrate with the national airspace system. In order to accommodate the cooperative avoidance, chapter three would define the rules of avoidance, based on the right-of-way rules that applied in the manned-flight. The On Board Collision System for UAVs would be proposed in chapter four, along with the algorithms that define the avoidance criteria. Then, chapter five presents the simulation on avoidance using the structure, rules and on-board system defined in the previous three chapters. A mathematical model was developed for this purpose and explained briefly also in chapter five. Using Monte Carlo method, safety

parameters are investigated in chapter six, and then the paper ends with some con-conclusions and suggestions in chapter seven.

2 Defining UAVs Collision Avoidance System Structure

Manned flight established its collision avoidance system in several layers of safety. Dalamagkidis, et al., [3] described the six layers of safety that are available in manned civil flight, shown in Figure 2. The grey area highlights the techniques to ensure separation, rather than to avoid possible collisions. The remaining three layers are system available at the current time to avoid collision between aircraft which lost it separation.

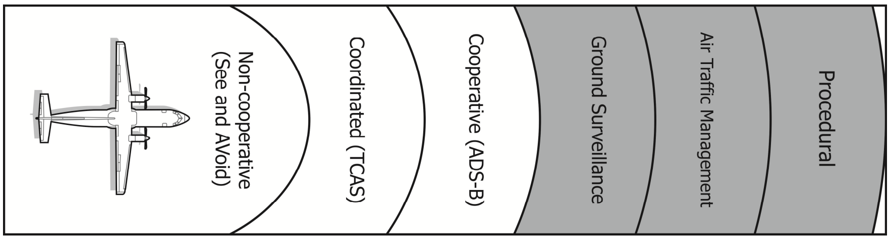


Fig. 2 Collision Avoidance System Structure for manned-flight.

On Cooperative and Coordinated layer, the avoidance system designed to handle collision-probable scenario where all aircraft involved follows a same consensuses rule. On manned flight, the Right-of-Way rules were commonly applied [10]. This rule state that when an aircraft, based on its condition, get the right-of-way, it have privilege to continue its course, while other that do not, have to conduct necessary avoidance manoeuvre. Pilots in manned flight were directed by ADSB or TCAS to follow those rules. Since the ADS-B will dominated the navigations in the near future [11], TCAS layer is merged into the Cooperative layer.

For non-cooperative layer, the avoidance system required to handle more complex scenarios. These include static obstacle, aircraft that follows different rules; aircraft that does not follow any rules at all (rogue); and moreover, objects with violent intentions (aim to collide). On manned flight, there is still no specific system to provide avoidance in this layer, except to use their own pilots' judgements.

In the context of integrating UAVs flight into the National Airspace system, UAVs required also to avoid collision with the already established manned-flight, besides avoiding collision between each other, to adopt those manned-flight layers of safety directly. However, due to many different characteristics in UAVs, compares to manned flight, several adjustments are required.

Unlike manned aircraft, which have limited manufacturers and operators, UAVs could be produced anywhere from a small scale company, and operated by almost anyone. Handling all those UAVs traffic using area-based ATM system appears to be unpractical. It is more reasonable to focus the design of UAVs collision avoidance system in the last two layers on the safety layer shown in Figure 2, the See and Avoid.

Barfield [1] designed a comprehensive structure for as requirements for an autonomous collision avoidance system (ACAS). The structure divided the avoidance into two sphere, named de-confliction and avoidance sphere. In the de-confliction sphere, an aircraft could avoid an obstacle while still maintaining its original path, while in the avoidance sphere; Aircraft should solely escape as fast as possible.

Barfield's de-confliction and avoidance sphere could be treated as cooperative and non-cooperative layer of safety, respectively. This will imply the followings:

1. The cooperative avoidance will be conducted inside the de-confliction sphere. The non-cooperative avoidance is conducted inside the avoidance sphere.
2. The cooperative avoidance will incorporate the common data of neighbouring vehicle in the area (from broadcaster i.e. ground surveillance or GPS) and apply the Right-of Way rule (also adjusted for UAVs later in chapter 3). The non-cooperative avoidance should also use any on-board sensor available and avoid the non-cooperative vehicle using somewhat more loose rules.
3. The cooperative avoidance is a de-confliction manoeuvre that still takes into account the original flight path, with the point to start the manoeuvre could take place anywhere in the de-confliction sphere. The non-cooperative avoidance is an aggressive manoeuvre aims solely to escape as fast/soon as possible and neglects its original flight path
4. The cooperative avoidance manoeuvre should in any case avoid the violation of the avoidance sphere. The non-cooperative avoidance should in any case avoid collision with obstacle. Turn rate requirements for avoidance could be set base on this.

Although is not explicitly described, Barfield choice of 25 second for de-confliction sphere might derived from manned flights TCAS. Therefore, as shown in Figure 3, the Traffic Warning sphere is introduced, which span until the 40 second distance. This sphere is where the Collision Avoidance system should begin to give warning to operators about the traffic ahead.

These (1) Traffic Warning, (2) De-confliction, and (3) Avoidance -Sphere define a novel structure of Collision Avoidance System for UAVs, in the context of integration to the National Airspace System. This structure should work seamlessly with the manned-flight, since it uses the same parameters they have already established.

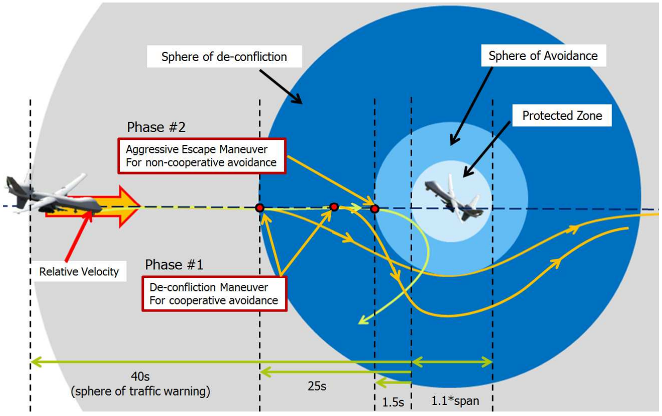


Fig. 3 Concept of Collision Avoidance System Structure on UAVs

3 UAVs Cooperative Avoidance Rules

As explained in the introduction, this paper will present the cooperative part off UAVs avoidance system, where the avoidance manoeuvres are based on a common consensuses rule. Similar with the avoidance structure described in the previous chapter, it is best to start defining the rule from the already established Right-of-way rule in the manned flight, stated in [10]. The following subchapter will describe the suggested rule of cooperative avoidance in two parts, the category priorities and the situational priorities.

3.2 Category Priorities for UAVs

With the large variation of UAVs, it's only logical to set some category priorities for them. Many documents have presented classifications of UAVs, especially based on its dimension (size) or weight, e.g. CAP 722 [9].

Quite different, on the category priorities, manned flight use the performance of aircraft category; aircraft that have slower or lower performance in manoeuvring will get the right of way [10]. Based on this, UAVs need to be categorised based on performance. Furthermore, since the Collision Avoidance structure defines in the last chapter is based on time-described distances, velocity would be a good parameter for the categorization.

Spreading out the CAP 722 classification that based on weights, it appears that UAVs could easily be categorized by its cruise velocity. The new classification that based on velocity is listed in Table 1. The class on the upper row will always have right of way (priority) to the lower rows.

Using the velocities limits, the structure of collision avoidance system could be easily defined for each categories, or for between categories. Derivation of the spheres radius (when each category meets an assumed static object) could be observed also in Table 1.

Table 1. UAVs classifications, base on Velocity

CAP 722 Classifications	Velocity Classifications	Velocity [km/h]	Velocity [m/s]	Sphere Radius [m] (in collision with Static Object)		
				1.5s	25s	40s
Small UAVs	Small Slow UAVs	<50	<13.89	20.83	347.22	555.56
	Small Fast UAVs	<100	<27.78	41.67	694.44	1111.11
Light UAVs	Light UAVs	<250	<69.44	104.17	1736.11	2777.78
Large UAVs	Large Slow UAVs	<500	<138.89	208.33	3472.22	5555.56
	Large Fast UAVs	>500	>138.89	416.67	6944.44	11111.11

In scenario when a UAV from one category meets another category UAVs (i.e. A Small-Slow UAVs face a Light UAVs), the spheres radius will change according to the relative velocity limit of both UAVs. Table 2 shows calculation result for the avoidance sphere radius, in case where each category meets one another. Some shaded column indicates the unlikely-to-happen scenario due to difference on operation altitudes. On the continuity of this research, analysis will be extensively focused on the Small Slow UAVs, especially to plan the real-world experiments on the avoidance concepts.

Another priority that needs to be defined is the interaction with manned aircraft. Barfield proposed UAVs to follows Asimov's three robotic laws [1]. In short, UAVs should always give the right of way to manned aircraft, regardless their velocity or weight.

Table 2. Avoidance sphere radius for each categories encounter

	Avoidance Sphere Radius [m]					
	Static Object	Small Slow UAVs	Small Fast UAVs	Light UAVs	Large Slow UAVs	Large Fast UAVs
Small Slow UAVs	20.83	41.67	62.50	125.0	229.17	437.5
Small Fast UAVs	41.67	62.5	83.33	145.83	250.0	458.33
Light UAVs	104.17	125.00	145.83	208.33	312.5	520.83
Large Slow UAVs	208.33	229.17	250.00	312.5	416.67	625.00
Large Fast UAVs	416.67	437.5	458.33	520.83	625.00	833.33

3.3 Situational Priorities for UAVs

The situational priorities in UAVs flight could easily be adopted from the manned-flight. This is true especially for the cooperative collision avoidance. The summary of these rule listed as follows:

1. On converging encounter, the one on the right hand have the right of way
2. On head-on encounter, both aircraft should move to the right side
3. The one that are about to be taken over have the right of way
4. Avoidance should not go over or under, or in front of other aircraft that have right of way, except when it is clear

For UAVs system, the converging, head-on, and taking over encounter need to be defined quantitatively. One way to define those is to use the definition of crossing, opposite and same flight path for manned flight Air Traffic control, stated in [12] which are described in the Figure 4.

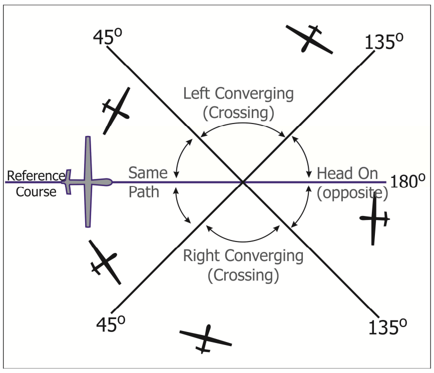


Fig. 4 Flight path definitions for manned air traffic control, adapted from [12].

The adaptation of those definitions in the UAVs ACAS is described thoroughly in the next chapter.

For non-cooperative avoidance, on the other hand, definitions of its situational priorities will not be discussed further in this paper; instead it will be investigate on the continuation of this research.

4 Defining the On-board Collision Avoidance System for UAVs

Based on the collision avoidance structure and rules, an on-board collision avoidance system functional concept is derived in this chapter. The design where influence by the twelve requirements set by Barfield [1].

4.1 System Functional Concept

An autonomous system for collision avoidance (ACAS) was highly suggested for UAVs applications, including in [1], since the task of avoidance in UAVs could not be handled only by pilot/operators. This is due the fact that the UAVs operator will only manage the UAVs flight to finish it mission autonomously, and even if there are such ground pilots controlling the UAVs, they do not have the required awareness of the surroundings.

Nuisance free is another requirement that need to be fulfilled by the UAVs ACAS. This means that the ACAS should be separated from the normal control system that is operating the UAVs, and only interferes when it's needed. Interrupt and restore criteria should be defined for this purpose. In accordance to this, warning cues to the pilot when the system detects traffics are also required. In Figure 5, these concept where compactly drawn, with also highligthed the used of ADS-B.

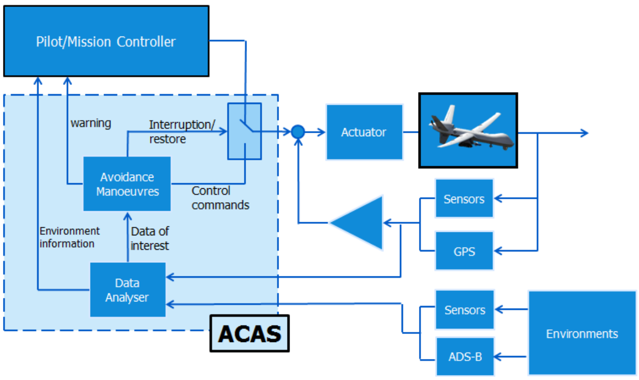


Fig. 5 Cooperative ACAS system concept, integrated with the normal mission controller of a UAVs

4.2 Avoidance Algorithms: Selective Velocity Obstacle Method

A method called the Velocity Obstacle (VO) Method [2,6,7,8], or sometime the Forbidden Zone Beam Method [5], is used to define avoidance criteria. The VO-method was chosen due to its simple implementation and geometrically understandable. A complete explanation of the original VO-method could be found in [6]. To be suitable for the implementation in UAVs ACAS, including adopting the rules described in previous chapter, several modifications were made, producing a new branch of the Velocity Obstacle Method, which from this point, will be referred to the Selective Velocity Obstacle Method (SVO).

4.2.1 Velocity Obstacles (Original) in UAVs collision avoidance system

This section presented the explanation of the original VO [6] in context for UAVs ACAS applications, explained in previous chapters. Since the focus is to set an algorithm in each UAVs separately, the own UAVs (should-avoid) and obstacle are treat differently. This original VO will be referred as OVO.

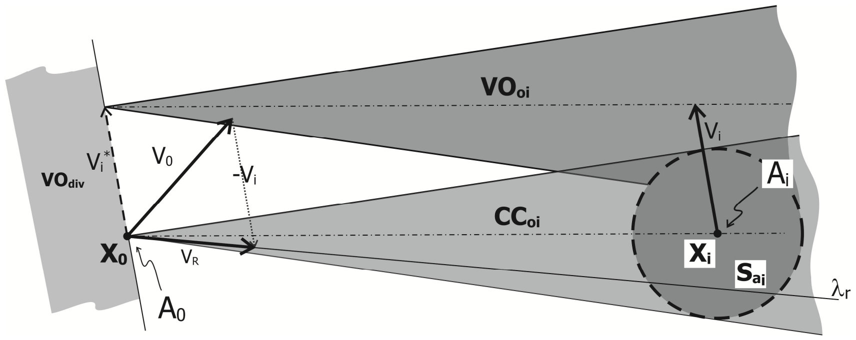


Fig. 6 VO cone definition in the original concept, adapted from [6]

First we designated A_o and A_i to symbolized the should-avoid agent and the obstacle agent, respectively. Let S_{a_i} be the avoidance sphere, with centred by the A_i position X_i , and moving with constant velocity V_i . Let X_o be the position of A_o , moving with constant velocity V_o . According to OVO method, to decide if these two agents are on a collision course, it is sufficient to consider their current positions together with their relative velocity $V_R = V_o - V_i$. If one elongate the V_R from X_o by a sufficient positive scaling (symbolized as $\lambda_R = \{X_o + \mu V_R \mid \mu \geq 0\}$), it is clear that the two agent are on a collision course, if and only if λ_R cuts the area S_{a_i} or formally, $S_{a_i} \cap \lambda_R \neq 0$. The set of λ_R that cuts S_{a_i} is called collision cone CC_{oi} .

To be able to decide directly whether V_o will collide or not, it was suggested to define the so called velocity obstacle set/cone of S_{a_i} from X_o , as:

$$VO_{oi} = \{V_o \mid (V_o - V_i) \in CC_{oi}\} \tag{1}$$

Or,

$$VO_{oi} = V_i + CC_{io} \tag{2}$$

Thus, for A_o, any velocity $V_o \in VO_{oi}$ from X_o, will lead to a violation on S_{a_i}, and any velocity $V_o \notin VO_{oi}$ will avoid those violations, in scenario where the initial position where X_o and X_i respectively.

In reality, it might happen that A_o was confronted with more than one obstacle. In this general cases, Let $i = 1,2,3,....,n$, the number of obstacle under consideration. The velocity obstacle that A_o need to defined for all the obstacle is simply the union of each velocity obstacle,

$$VO = \bigcup_i VO_{oi}, \quad i = 1,2,3,...,n \tag{3}$$

For any velocity $V_o \notin VO$ from X_o, A_o will not violate any S_{a_i}, where $i = 1,2,3,...,n$.

Figure 6 also shows another area named the VO diverging area, VO_{div}. This area defined as one of two areas separated by the infinite elongation of vector V_i through X_o, that does not contain any set of VO. Fiorinni [6] already define this area as a set of vector that A_o could chose to diverge completely from the obstacle. However, this area has not been employed in any of VO previous research. It will become clear that VO_{div} could set a handy definition on the avoidance manoeuvre.

On the OVO, a simple navigation scheme based on which velocity could be chosen to ensure no collision is used. The position and velocity of each agent were continuously tracked, and all information was used to update V_o. The velocity is chosen based on the goals of the agents, for example to avoid while still in the same path, or taking its maximum velocity to avoid each other.

4.2.2 Selective Velocity Obstacle (SVO)

SVO was designed to accommodate rules and requirements of the UAVs ACAS system. The idea is to selectively use any VO area developed around the velocity vector V_o, based on the position of each VO position from X_o. Using this, the algorithm will select which VO should be avoided, and which VO could be ignored. Several area definitions around X_o are added to the OVO, and extend the criteria on which A_o should take a manoeuvre. These area, different from VO, relate to the obstacle velocity shadow V_i^* from X_o, or, the origin of each VO. The additional areas explained here were meant to represent the rules described in chapter 3, however, could easily be modified for other rule schemes.

First we define two circle centred by X_o, S_{V_o}, and S_{cat1}, with radius of V_o, and V_{cat1} a respectively. V_{cat1} is the velocity limit of a UAV category explained in

section 3.2, which for the Slow-Small UAVs, is 13.89 m/s. Next, using A_o motion axis (or wind-axis) as the frame of reference (where V_o is pointing up), we divide S_{cat1} into four equal set of velocity vector coming from X_o, named S_{r1}, S_{r2}, S_{r3} and S_{r4}, as shown in Figure 7. Notice that this represents the flight path definitions explained in section 3.3.

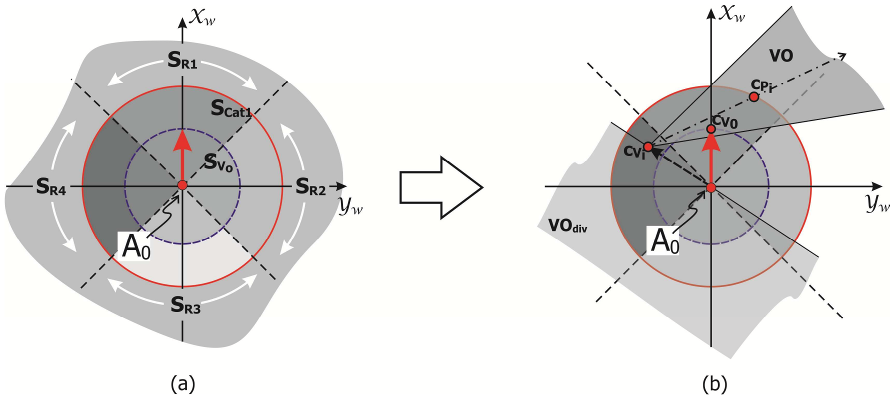


Fig. 7 Selection Circle on SVO (a) Area definition; (b) VO (and VO_{div}) implemented

Lastly, we define three points that will set the criteria, c_{V_o}, c_{V_i} and c_{P_i}. c_{V_o} is simply the end of V_o vector from X_o and c_{V_i} is the end point of the shadow of the obstacle velocity V_i from X_o, or simply, the origin of the Velocity Obstacle VO. c_{P_i} is the intersection point of VO axis with the edge of S_{cat1}. This last point not really necessary, and could be replace by the real position of the obstacle, X_i. However, it is added for a compact figure and explanations.

Next section will describe how the additional areas were used to selectively treat the Velocity Obstacles.

4.2.3 Algorithm for the Selective Velocity Obstacle

With those setups, we could finally define the algorithm required, to accommodate all rules into the UAVs ACAS via this Selective Velocity Obstacle Method. As mentioned before, the algorithm is designed to still give UAVs freedom to choose their own avoidance manoeuvre, as long as they follows the rules, explained in chapter 3. Generally, there will be three main manoeuvre type that UAVs ACAS need to handle, which are (1) Avoid, (2) Maintain, and (3) Restore, denoted as q_1, q_2, and q_3, respectively. Restore here means that the ACAS give back the control to the original controller/pilot so the UAVs could

continue its mission. The ACAS itself need only to define what manoeuvre it should take for the Maintain and Avoid.

Thus, the avoidance rules for the SVO for cooperative avoidance of UAVs, for category one (Slow-Small UAVs) are mathematically modelled as follows:

$$\begin{cases} q_1, & \text{if } \; c_{V_o} \in VO_{oi} \wedge c_{V_i} \in S_{cat1} \cap \left(S_{r3} \cup S_{r4} \cup \left(S_{r1} \cap S_{V_0}\right)\right) \\ q_2, & \text{if } \; c_{V_o} \notin VO_{oi} \cup VO_{div} \wedge c_{V_i} \in S_{cat1} \\ q_3, & \text{otherwise} \end{cases} \tag{4}$$

Here, the velocity obstacle only need to be avoided when the origin of any VO (c_{V_i}) lies inside S_{r3}, representing head-on encounter, inside S_{r4}, representing right-encounter, or inside $S_{r1} \cap S_{V_0}$, which simply represent a take-over manoeuvre of a slower vehicle in the same path. Notice that these algorithms only activated when c_{V_i} is inside S_{cat1}, interrupting the normal controller. In case of c_{V_0} already escapes VO_{oi} but still not inside the VO_{div}, the system treat it as not safe enough to give back the control to the original controller, instead it maintain its course and wait for any event that still could happen, including being back again inside VO_{oi}. Only when c_{V_0} is inside VO_{div}, should the restoration maneuver happen.

As it might have been notice, SVO also discard the set of reachable velocities that originaly used in the OVO [6]. The main reason of this is the fact that UAVs commonly use rotation as the control input for manoeuvring, instead of arbitrary velocity vectors. Thus, SVO describe a minimum turning rates (ω_{req}) required for avoidance manoeuvre, which will depend on velocities, distances and positions. This turning rate will be derived on the continuation of this research.

5 Implementations

Using the defined collision avoidance structure in chapter two, the cooperative avoidance rule in chapter three, and the on-board ACAS system and algorithms in chapter four, several computer simulation were conducted. A MATLAB program was developed and designed to be highly customable that it could accommodate any initial positions and velocities, avoidance rules and algorithms used, the UAVs involved dynamics, normal control systems, and many more. This MATLAB program is still on-going development and will also be used in the continuation of the research.

5.1 Mathematical Model and Simulation Setup

Since it will be applied in a relatively large area, we could treat vehicles involved as a point mass, eliminating the need to model each aircraft dynamics. The mathematical model of each aircraft motion was linear, discrete and single phased, focusing more on the development of the right algorithm to accommodate avoidance. Position and velocity data of each aircraft were broadcasted between each other in same time step, simulating the use of ADSB that support this cooperative avoidance.

Depends on how many agents involved in a scenario, the MATLAB program first generate them as an object that embedded these linear discrete equation that describe each agent propagations through the simulation.

$$x(k+1) = Ax(k)$$

$$x = \begin{bmatrix} x \\ y \\ V_x \\ V_y \end{bmatrix}; \quad A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & -\omega\Delta t \\ 0 & 0 & \omega\Delta t & 1 \end{bmatrix}; \quad \omega = \begin{cases} \omega_{avo}; & \text{if } q_1 \\ 0; & \text{if } q_2 \\ \omega_{Goal}; & \text{if } q_3 \text{ or } q_{init} \end{cases} \tag{5}$$

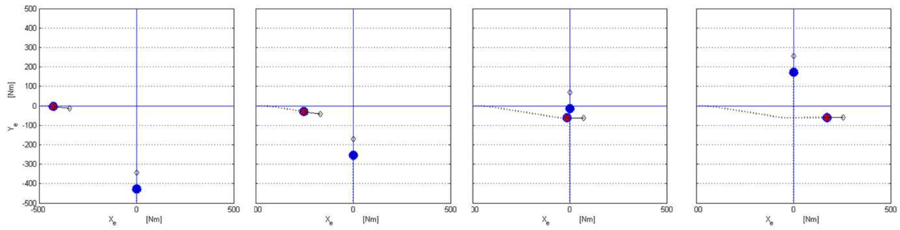
Inputs for the equation were highly depends on the result from SVO algorithm, explained before (q_1, q_2, or q_3). q_{init} is simply the initial setup before any detection of obstacles. In the conducted simulation, these values are simply the direction to each agent original end point. ω denotes the modes turning rates, where it is the ω_{avo}, 0, ω_{Goal} on mode q_1, q_2, and q_3, respectively. ω_{avo} was assumed to be 5 deg/s (0.0873 rad/s) for every agent. ω_{Goal} obtained from any normal controller that is used, that guides the UAV back to its original mission. In this research, ω_{Goal} simply direct each UAV to its original waypoints.

Unit time step (Δt = 1 second) was used for every simulation, in assumption it also match the ADSB update rates. For simplification on these preliminary simulations, all avoidance happens on the edge of de-confliction sphere. Lastly, all agent considered is a Category 1 UAVs, the Slow-Small UAVs (see section 3.2).

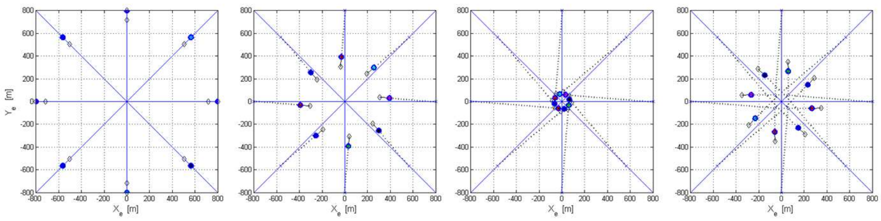
5.2 Simulation results

There are unlimited collision scenarios which could be tested, even though only working on one UAV category. A few important scenarios were presented in this paper, selected according to the converging, head-on and same path areas described before in chapter 3. The entire results are presented using agent position time-captures from above (top view) on four important positions. The arrow on

each agent represents the velocity vector. Notice that the entire rules described in chapter 3 were fulfilled for each avoidance.

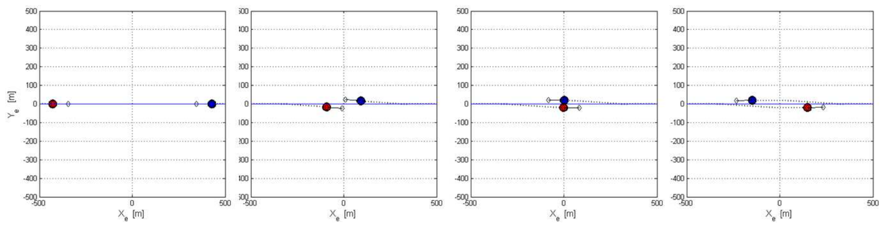


(a)

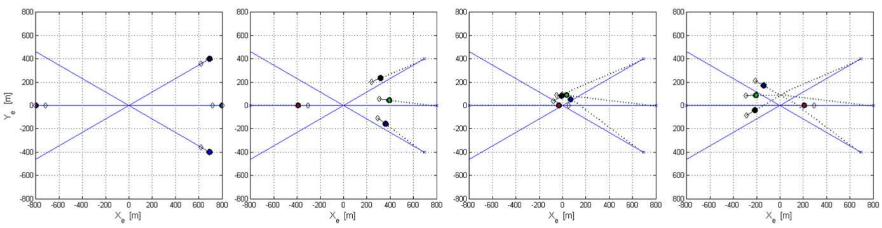


(b)

Fig. 8 Simulation on converging encounter scenario; **(a)** two-agents, 90^o encounter from the right, **(b)** eight-agents, symetrical circle encounters.

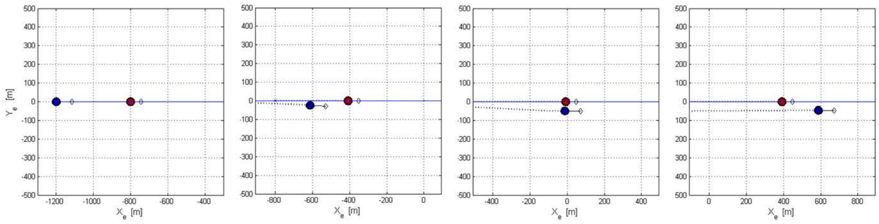


(a)

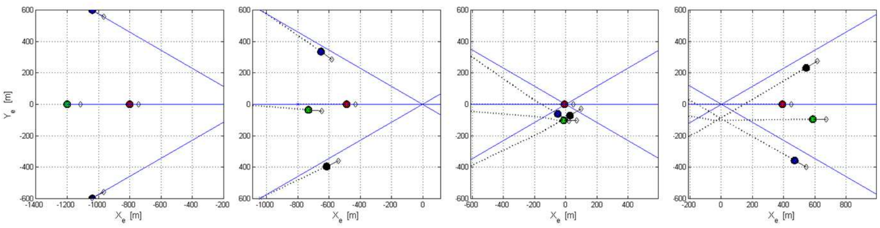


(b)

Fig. 9 Simulation on Head-On encounter scenario; **(a)** two-agents, directly Head-On **(b)** four-agents, 30^o, 0^o and -30^o encounter forward



(a)



(b)

Fig. 10 Simulation Take-Over scenario; **(a)** two-agents, same path **(b)** four-agents, 30^o, 0^o and -30^o encounter from behind

On Head-On encounter (Figure 9), since both agents are avoiding each other to the right, the course deviation is not as large as the converging case (Figure 8). Interesting to observe in Figure 10-b that the agent heading to the right did not conduct any avoidance; instead, it goes straight as its original course. Analysis revealed that this happens because the other three agents on the opposite are closer to each other, and start avoiding each other sooner. Those manoeuvres create a situation where the one agent heading to the right will not collide at all, and hence it keeps it original flight path.

On Figure 10, to be able to simulate a taking over encounter, a different velocity is required. Therefore, one agent, which will be taken over, has 8 m/s velocity, as opposed to other agents behind it that use 12 m/s. In the end, all taking-over where successfully conducted, even when there are more than one agent are taking over.

6 Violation Probability (using Monte Carlo Simulations)

The entire simulations in chapter 5 were conducted smoothly without any avoidance sphere violations. However, these results not necessarily mean the avoidance system and algorithm guaranteed to works for every scenario. Therefore, this chapter will present a Monte Carlo simulation where a large number of random scenarios were tested, in order to find the violation probability

of the avoidance. The derivations were conducted for two, three, four and five UAVs (agents). Similar with the simulations in chapter 5, this violation probability derivation in this paper will only discusses the first category of UAVs.

The derivation of violation probability shows how well the performance of the algorithms developed, and even, act as a tool to find any scenario that make the algorithm fail. In accordance to the Equivalent Level of Safety (ELOS), this violation probability needs to be zero. ELOS are based on the failure of the system due to time. The algorithm itself should be guaranteed to solve any scenario possible.

6.1 Monte Carlo Simulation Setup

To assess the performance of both system architecture and the proposed algorithm (and rules), several parameters were introduced. The cooperative ACAS performance were measured using the probability of separation violantions, P_{vio}, formulated as:

$$P_{vio} = \frac{N_{vio}}{N_{MC}} \tag{6}$$

Where N_{vio} and N_{MC} denotes number of scenario that collision happen and number of Monte Carlo samples, respectively. The value of P_{vio} will fluctuated with N_{MC}, and as N_{MC} become larger, it should converge to a certain value, which define as the final value.

Other parameter to set up the Monte Carlo simulation are the selected area of interest, A_{int}, the area of separations, A_{sep}, and the area density, $\rho_{A_{\text{int}}}$, formulated as:

$$\rho_{A_{\text{int}}} = \frac{NA_{sep}}{A_{\text{int}}} \tag{7}$$

Where N denotes the number of agents involved. Notice that A_{sep} is a circle area with radius of half of the de-confliction sphere, conserving the total de-confliction distance.

The position (x_n, y_n) of each agent is randomized on the X-Y planes, while keeping no violation in the beginning of simulation. The x_n, y_n position is assumed to be spread randomly in a square, instead of a circle area, for simplifications. As can be observed in Table 3, the position range is set according to the number of agents, and the radius of Traffic Sphere used (the 40s sphere, see section 2), which have radius r_{tra}. Consequently, A_{int} and $\rho_{A_{\text{int}}}$ also depends on this sphere, where $\rho_{A_{\text{int}}}$ becomes constant for every number of agents involved, set at

0.3. Two other initial parameters were randomized as well, the headings (ψ_i) and the velocity magnitudes (V_i), detailed in Table 3 Using these setup, it is possible to have a scenario where the agents are not bound to violate each other, and thus make it possible also to derived the violation probability where no ACAS is implemented.

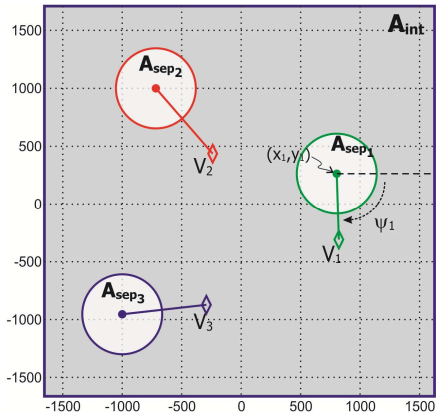


Fig. 11 Random scenario of encounters (e.g. 3 agents involved) for Monte Carlo simulation setup

All avoidance manoeuvre used the same turning rate of avoidance (ω_{avo}) of 5 deg/s ~ 0.0873 rad/s. Furthermore, all avoidance starting point take place on the edge of the de-confliction sphere (25s sphere) and using the same turning rate.

Table 3. Ranges of randomized parameters for Monte Carlo simulations

Variable	Range	
Positions (x_n, y_n)	$\left[-\frac{1}{2}N\times r_{tra}, \frac{1}{2}N\times r_{tra}\right]$	[m]
Velocity Magnitude (V_i)	$[8,13]$	[m/s]
Heading (ψ_i)	$[-\pi,\pi]$	[rad]
Avoidance Point (D_{avo})	$(1)[1,1]$; $(2)[0,1]$	[-]
Monte-Carlo Samples	10^6	samples

Results of this Monte-Carlo simulation (coded 'MC01') are presented in the next section (Figure 12 and 13). Those results, however, neglect the freedom that each cooperative agent should have, to choose their own avoidance manoeuvre. Consequently, another Monte-Carlo simulation was conducted (coded 'MC02'), with one more randomized variable, D_{avo}, which denote the ratio of avoidance starting point with the de-confliction sphere radius. The turning rate of avoidance (ω_{avo}), however, was still assumed to be 5 degree/s.

6.2 Results and Analysis

Figure 12 shows the results of P_{vio} versus the number of Monte-Carlo samples. The figures compiles results from both Monte-Carlo simulations (MC01 and MC02), with addition of the calculation result where no ACAS is used (coded MC00). It could be observed that the Monte Carlo simulation produced convergent results on number of samples of 10^6, for each agent number configurations. Figure 13 shows the final violations probability for each number of involved agents, for MC00, MC01 and MC02.

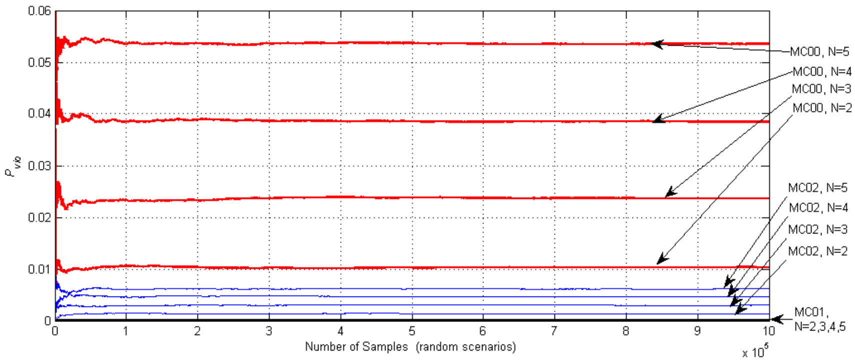


Fig. 12 Monte Carlo simulation convergence results on Probabilities of Separation Violation for two-, three-, four- and five- agents, on MC00, MC01, and MC02.

Several analyses were made based on the Monte Carlo simulation. First one is regarding the Area Density parameter $\rho_{A_{\text{int}}}$. Evidently, this parameter is less dominant than the number of agents involved; even though the area of interest (A_{int}) enlarged as more agents involved, violation probability (P_{vio}) still become larger. This may be caused by the enconters combinations between agents in the area.

MC01 results was satisfying, resulting zero violations for every number of agents scenario. MC02, however, only shows violations reductions that is stilll unacceptable. On observations on those failed cases, it was concluded that distance might be the problem, since every failure happen at Avoidance Point (D_{avo}) lower than 0.5. This also explain why MC01 results zero violation; MC01 only use D_{avo} = 1. Observation on those failed scenario also reveals that agents are indeed avoiding, however, the distances were too close, and the avoidance is not fast enough. This suggest the need to adjust the avoidance turning rate (ω_{avo}) according to D_{avo} . If the adjustment of ω_{avo} could be derived, it coud be set as a requirements for thes cooperative avoidance between UAVs, the minimum required turning rate, $\omega_{a.req}$. This derivation, however, will not be discussed in this paper.

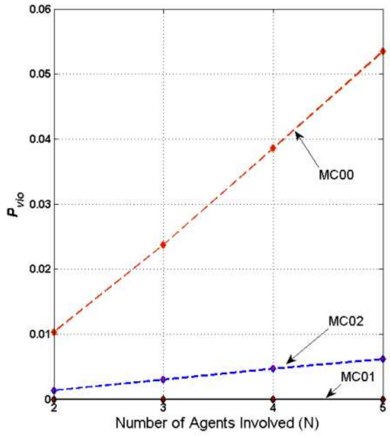


Fig. 13. Collision Probabilities comparison with the use of cooperative ACAS (MC01 and MC02) and without (MC00).

7 Conclusions

Several concluding remarks could be summarized form this research, including as follows:

1. In order for UAVs to be integrated to the national airspace system, a complete collision avoidance system was investigated. This include not only the system on-board a UAV, but also the structure and rules that could define a common guideline for UAVs avoidance.
2. The structure of the collision avoidance system for UAVs is then divided into two main parts, the cooperative part, which was in accordance to a de-conflicting manoeuvre, and the non-cooperative part, which will use an aggressive avoidance manoeuvre. This paper, however, only continue to focus only on the cooperative collision avoidance system.
3. To support the Cooperative Collision Avoidance, several ground rules were defined based on the rules of the air in manned-flight.
4. Finally, a functional concept for the onboard system was defined, incorporating several requirements. A Novel algorithm for cooperative ACAS for UAVs, named Selective Velocity Obstacle (SVO) method, was introduced.
5. A MATLAB program was created as a tool to simulate various scenario of collision. All simulation of the selected scenarios were conducted smoothly and the use of designed cooperative ACAS evidently could prevent separation violations.
6. To quantitize the probability of violations, and then state the performance of the designed cooperative ACAS, a Monte Carlo simulations were

conducted. The results suggest the need to derive a minimum requirements for avoidance turning rate, $\omega_{a.req}$, base on distances of avoidance.

References

1. Barfield F. (2000) Autonomous collision avoidance. The technical requirements. In: National Aerospace and Electronics Conference (NAECON). - Dayton : IEEE. - Vol. 1. - pp. 808-813.
2. Berg, J vd, Lin M, Manocha, D. (2008) Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation. In IEEE International Conference on Robotics and Automation, Pasadena, CA, USA.
3. Dalamagkidis K., Valavanis K.P. and Piegl L.A. (2009) On Integrating Unmanned Aircraft Systems into the National Airspace System: Springer Science+Business Media, B.V. - Vols. International Series on Intelligent Systems, Control, and Automation: Science and Engineering, Volume 36.
4. DeGarmo M.T. and Nelson G.M. (2004) Prospective unmanned aerial vehicle operations in the future national airspace system. In: Collection of Technical Papers - AIAA 4th Aviation Technology, Integration, and Operations Forum, ATIO. - Chicago : Scopus - Vol. 1. - pp. 172-179.
5. Ellerbroekffi, J., Visser, M., van Dam., S.B.J., Mulder, M., van Paassen, M.M., (2009) Towards an Ecological Design of a Four-Dimensional Separation Assistance Interface. In: AIAA Guidance, Navigation, and Control Conference and Exhibit; Chicago, IL FBZ paper
6. Fiorini P., Shiller Z.(1998) Motion planning in dynamic environments using velocity obstacles. In: International Journal on Robotics Research 17(7): 760-772.
7. Kluge B. (2003) Recursive Probabilistic Velocity Obstacles for Reflective Navigation. In: Proc. of 1st Int. Workshop on Advances in Service Robots, Bardolino.
8. Snape, J.,Berg, J vd., Guy, S.J., Manocha, D., (2011) The Hybrid Reciprocal Velocity Obstacle. In IEEE Transactions on Robotics, 27 (4) , art. no. 5746538 , pp. 696-706
9. Civil Aviation Authority, (n.d), CAP 722 Unmanned Aircraft System Operations in UK Airspace – Guidance. Retrieved January 10, 2013 from http://www.caa.co.uk/ application.aspx?catid=33&pagetype=65&appid=11&mode=detail&id=415
10. Federal Aviation Administration, (n.d), Federal Aviation Regulations (FAR) Chapter I, subchapter F – Air Traffic and General Operating Rules, 91.113 Right-of-way rules: Except water operations. Retrieved January 10, 2013 from http://www.ecfr.gov/cgi-bin/text-idx?c=ecfr&tpl=/ecfrbrowse/Title14/14tab_02.tpl
11. Federal Aviation Administration, (2008-2012) FAA Flight Plan—Charting the Path for the Next Generation, Washington, DC.
12. Federal Aviation Administration, (n.d) JO 7110.65U, Air Traffic Control, Chapter 1: General. Retrieved January 13, 2013 from http://www.faa.gov/regulations_policies/ orders_notices/index.cfm/go/document.information/documentID/1019803