Detect and Avoid Algorithm for UAS with 3D-Maneuvers

Richard Alligier, Cyril Allignol, Nicolas Barnier, Nicolas Durand, Ruixin Wang

Detect and Avoid Algorithm for UAS with 3D-Maneuvers

Richard Alligier, Cyril Allignol, Nicolas Barnier, Nicolas Durand, Ruixin Wang
ENAC Lab
7, av. Édouard Belin
31 055 Toulouse, France
firstname.surname@enac.fr

***Abstract*—In this article, we extend the 2D-framework introduced in 2016 to implement a horizontal *detect and avoid* algorithm for UASs flying in Terminal Control Areas. First, we introduce a model able to detect conflicting trajectories in 3D and select combined horizontal and vertical maneuvers, taking the urgency of the conflict into account.**

We use a large data set of recorded real commercial traffic trajectories to evaluate the ability of our improved algorithm to avoid any loss of separation with commercial airliners. We test two different types of UASs, flying at $80\,\mathrm{kt}$ or $160\,\mathrm{kt}$, with six different missions: constant heading or turning and leveled, climbing or descending. We consider both heading change and vertical maneuver so that UASs have more freedom to avoid conflicts, but keep the speed constant as they mostly have poor speed-up performance.

The article investigates the influence of the various parameters on the separation achieved and the amount of maneuvers required, especially the strategy used to select the best maneuver among the allowed headings. The analysis of our results shows that, amid two "basic" and "extreme" strategies that respectively favor minimal heading and vertical changes or the expectation to escape the conflict, the combination of both, switching from the first one to the second whenever the time before the conflict falls under a given threshold, gives the best results with very few remaining close encounters, while keeping low the amount and amplitude of maneuvers.

***Index Terms*—UAS, self-separation, conflict resolution, geometrical algorithm.**

I. Introduction

Civilian UASs are used for diverse missions (ranging from fire detection and river bed surveillance to small items delivery, agriculture, etc). The demand for opening spaces to them is very high [1]. Most of Civilian UASs will fly in lower airspace (under FL180) and sometimes close to airports where commercial aviation traffic is dense. Developing new algorithms to help separating UASs from the rest of the traffic is critical for safety reasons. In this context, different approaches are investigated.

In a centralized approach, separation could be entirely managed by Air Navigation Service Providers. Experimental results [2] have shown that mixing UASs with conventional traffic creates difficult situations for Air Traffic Controllers because their performances and time responses are different.

In an autonomous approach, the separation task could be shared by *both* commercial aircraft and UASs. However, in such a context, air traffic controllers would have to validate the maneuvers of commercial aircraft before they are executed.

It seems more reasonable to let UASs deal with their separation with the commercial traffic. This requires that the positions and speeds of surrounding aircraft are available (through ADS-B for example) and that the performances of UASs enable them to react early enough to handle the whole separation task.

In a previous article [3], we showed that a self-separation algorithm used in robotics in the horizontal plane could be adapted to our problem and we checked different scenarios on thousands of real commercial traffic samples to validate our model. In a small percentage of cases, our approach failed, despite a mixed strategy that focused either on keeping the maximal distance with the traffic (*Safest*) or tried to minimize the separation (*Closest*).

We then made our approach more realistic by adding noise to the detection signal recorded by the UASs [4]. We were able to show that our approach is generally robust to uncertainties.

In this article, we generalize our approach by adding vertical maneuvers to UASs. Our approach is close to Snape et al.'s [5] who first introduced a geometrical approach for self-separation for aircraft conflict resolution. We model the forbidden zones with cylinders instead of spheres or ellipsoids in order to comply with various separation standards. We also adapt the hybrid strategy presented in our horizontal separation model that switches between the *Safest* and *Closest* modes. We also use a fallback strategy to minimize the constraint violation in case the separation standards are not satisfied.

A first detect and avoid algorithm was designed by van den Berg et al. [6] and tested with different speed constraint hypotheses by Durand et al. [7] in the context of autonomous air conflict resolution. In [3] we tailored van den Berg's geometrical approach to model the performances of UASs and consider specific fallback strategies to handle cases for which the first approach failed to maintain separation.

As in our previous approach:

- The UAS assumes the whole avoidance maneuver.
- UASs considered fly with low speeds compared to surrounding commercial aircraft. The ratio used in this article can go from 1.5 to 5. We focus on the lower airspace where the aircraft speed is theoretically limited at $250\,\mathrm{kt}$, but recorded data show that in practice some aircraft fly much faster (up to $400\,\mathrm{kt}$). More specifically, we consider as in [3] two types of UASs: Fast UASs flying at $160\,\mathrm{kt}$ and Slow UASs flying at $80\,\mathrm{kt}$. However, we now also

allow UASs to climb at a rate up to $2200\,\mathrm{ft\,min^{-1}}$ and descend at a rate up to $4200\,\mathrm{ft\,min^{-1}}$.

- Because most civilian UASs have very poor speed-up performances compared to conventional aircraft, we only consider maneuvers at constant speed for UASs. Our approach could easily be generalized with variable speed, but relaxing this constraint would have little effect on the resolution process with realistic traffic.
- Commercial aircraft flying in the lower airspace are generally climbing or descending and their speeds are constantly changing, either increasing when climbing or decreasing when descending, and changing direction as well. This factor has a great influence on the detect and avoid strategy in order to ensure that a reasonable distance to the encountered traffic can be maintained. Using real traffic data is therefore essential to validate a resolution algorithm for such evolving and intricate traffic.

1) Related Works: The Free-flight concept emerged in the early 90s. In Europe, the main idea was then to equip every aircraft with a detect and avoid algorithm able to ensure separation with the rest of the traffic.

Approaches inspired by physical laws were tested. Zeghal used sliding forces to coordinate maneuvers between aircraft [8]. Potential or vortex fields [9], as well as a model based on an analogy with electrical particle repulsion [10], were also used. In 2001, we proposed a token allocation strategy combined with an A^* algorithm to solve conflicts with realistic maneuvers [11], [12]. Even if some maneuvers could be simultaneously decided, a complete ranking of aircraft was necessary and finding an optimal ranking has been shown to be problem-dependent [13]. We also tried artificial Neural Networks on the two-aircraft problem [14] but they could not be generalized to handle more aircraft. All these approaches have been tested on en-route traffic, mainly with leveled aircraft.

Geometrical algorithms have also been widely studied in robotics [5], [6], [15], [16]. ORCA, the powerful technique developed by van den Berg et al. [6] to coordinate maneuvers in a distributed fashion, can handle thousands of agents in a small space. It was applied to aircraft by Snape et al. [5], but the hypotheses of the algorithm require simultaneous vertical and horizontal speed changes. We also tested them [7] in the horizontal plane with speed constraints and showed that this algorithm is unable to deal with high densities of traffic when the speed norm cannot be changed.

More generally, conflict resolution has been proven to be a highly combinatorial optimization problem [17]. Most centralized approaches that have been proposed to solve conflicts can be broadly divided into two main categories. The first ones [18]–[20] use greedy sequential algorithms to optimize trajectories one by one after ranking the aircraft (ordering aircraft is however very challenging [13]). The others try to find the global optimum without the need to prioritize aircraft. Among this second category, many models define aircraft trajectories through simple analytic expressions that introduce strong limits on the type of situations that can be dealt with, as the ones described in [21]–[26]. In [27], [28], we proposed a model to solve multiple aircraft conflicts based on Metaheuristics (Genetic Algorithm and Tabu Search) using trajectory simulation with uncertainties. However, these works mainly targeted en-route traffic control and used simulated traffic only with the BADA model on real flight plans.

2) Outline: In this article, we focus on a realistic traffic environment. We consider UASs flying in the lower airspace (under FL180) and design various conflict scenarios with real recorded commercial aircraft trajectories in TCAs (Terminal Control Area). The aim of the study is to assess the performances of a "3D - detect and avoid" strategy for UASs to maintain a reasonable horizontal and vertical separation with commercial traffic. We first consider conflicts involving one UAS and one commercial aircraft only, then we introduce multiple aircraft encountering the UAS.

In section II of this article, we detail our geometrical approach inspired by the concept of *Velocity Obstacles* [29] in the case of a single UAS avoiding non cooperative aircraft. We show how our previous 2D algorithm can be adapted to handle arbitrary separation volume in 3D by discretizing the maneuvers space. In section III-A, we describe how the conflict scenarios were built from real traffic data and the hypotheses that were chosen for the UAS. In section III-B, we give some results obtained on the different scenarios and show the influence of the various parameters on the quality of the results in terms of separations achieved and amounts of maneuvers imposed on the UAS. The last section draws some conclusions about the results obtained with the simulations and highlights directions for future work.

II. Detect and Avoid Model in 3D

Building upon the 2D horizontal model of conflict resolution between a UAS and intruding aircraft described in [30], we present a somewhat simpler geometric view which can be naturally extended in 3D. This new model allows to efficiently discriminate conflicting maneuvers among a discrete set of alternatives and find the best one according to the current strategy.

We present as well a new maneuver selection strategy improving on the distance-based one described in [30] to take into account the time before conflict to switch between the so-called *Closest* and *Safest* modes.

As the former one, our new model can be easily generalized to handle multiple aircraft.

A. Single Aircraft

In [30], we presented an adaptation of the ORCA algorithm [6] tailored to the context of the integration of a UAS with limited speed change ability in a TCA with evolving airliners. Instead of equally dividing among the two aircraft an arbitrary maneuver that avoids the conflict, we constrained the scheme such that only the UAS is assigned a maneuver and the norm of its speed is kept constant. In the following, we resort to the more simple concept of *Velocity Obstacles* introduced by [29] and at the core of the ORCA algorithm [6].

As depicted in figure 1, let d_h be the standard separation in the horizontal plane between UAS A and intruding aircraft B with respective velocities $\overrightarrow{v_A}$ and $\overrightarrow{v_B}$, $\overrightarrow{v_r} = \overrightarrow{v_A} - \overrightarrow{v_B}$ their

relative speed and τ the look-ahead time. Whereas ORCA represents the future conflicting zone at time τ by a homothetic transformation of center A and ratio $\frac{1}{\tau}$ of the B-centered protection disc, we simply scale the relative velocity $\overrightarrow{v_r}$ by τ.

In the former, a conflict would occur within the τ look-ahead time-window whenever the endpoint of $\overrightarrow{v_r}$ lies in the rounded cone formed by the protection disc and its transform. However, in our equivalent version, we only need to check if the scaled vector $\tau\overrightarrow{v_r}$ crosses the protection disc around B, avoiding the need to check intersections with a composite shape. Indeed, as can be seen on the top of figure 1, the position of a point p along the line defined by A and $\overrightarrow{v_r}$ can be parameterized by the time t:

$$p = A + t\overrightarrow{v_r}$$

The time-window $[t_h^1, t_h^2]$ of the horizontal conflict, if any, can therefore be computed by the values of t at the intersection between this line and the protection circle. Thus, a conflict will occur in the horizontal plane within τ iff:

$$[0, \tau] \cap [t_h^1, t_h^2] \neq \varnothing \tag{1}$$

To extend our approach to the third dimension, the same reasoning is applied in the vertical plane where the protection volume is defined as a minimal vertical separation d_v (usually 1000 ft). The 3D protection volume is therefore a cylinder of radius d_h and height d_v, but any shape for which an intersection function can be provided could be used instead. As shown on figure 1, the intersection time-window $[t_v^1, t_v^2]$ in the vertical plane is computed, if any.

In 4D, a conflict will occur iff $[t_h^1, t_h^2]$ and $[t_v^1, t_v^2]$ intersects within $[0, \tau]$:

$$[0, \tau] \cap [t_h^1, t_h^2] \cap [t_v^1, t_v^2] \neq \varnothing \tag{2}$$

To avoid the conflict within τ, a maneuver must then be chosen such that this intersection is empty.

In the following, we note $cf(A, B, \overrightarrow{v_A}, \overrightarrow{v_B})$ a boolean function corresponding to the negation of inequality (2), i.e. indicating that the UAS A with a speed $\overrightarrow{v_A}$ is free of conflict until τ with the aircraft B at the speed $\overrightarrow{v_B}$.

B. Maneuvers Range

In our former 2D approach [30], the whole set of constant norm maneuvers could be easily computed as a union of allowed angle ranges for the new velocity $\overrightarrow{v_A}'$. With the complex geometry of dynamic cylinders, the non-convex surface of all conflict-free combinations of heading and path angle is much more difficult to compute. Therefore, we chose to discretize the set of maneuvers to check, trading complex geometric computations with brute force intersection checks (which could easily be done in parallel with several cores or a GPU).

To assess the performance of this new 3D model over the previous one, the norm of the UAS speed is maintained constant – even if this constraint could easily be relaxed at the price of multiplying the intersection checks described in the previous section. The maneuvers are therefore described

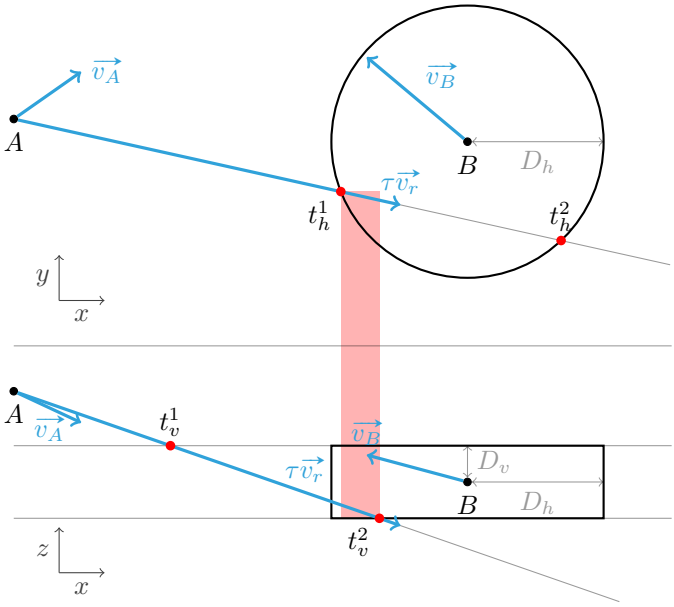

Fig. 1. Horizontal (above) and vertical (below) projections of the geometric model in 4D: a conflict will occur within time τ if and only if the relative speed $\overrightarrow{v_r}$ scaled by τ crosses the protection volume.

by two parameters, the heading θ and the path angle ϕ. Horizontally, the heading deviation is symmetrically constrained by $\theta_{\max}$ on both side:

$$\theta' \in [\theta - \theta_{\max}, \theta + \theta_{\max}] = \Theta$$

(with $\theta_{\max} = 30°$ typically for a 10 s time step) whereas the vertical angle must be constrained both absolutely and relatively around the current path angle to roughly model the capacities of typical medium-size UAS:

$$\phi' \in [\phi_{\min}^{\text{abs}}, \phi_{\max}^{\text{abs}}] \cap [\phi - \phi_{\min}^{\text{rel}}, \phi + \phi_{\max}^{\text{rel}}] = \Phi$$

In the end, the possible maneuvers is a set PM containing all the speeds obtained after a deviation has been applied to $\overrightarrow{v_A}$:

$$PM = \left\{ \overrightarrow{v_A}' \;\middle|\; \overrightarrow{v_A}'_{heading} \in \Theta, \overrightarrow{v_A}'_{path} \in \Phi, \|\overrightarrow{v_A}'\| = \|\overrightarrow{v_A}\| \right\}$$

These maneuver alternatives are then approximated by discretizing the aforementioned ranges with angular steps η_h and η_v, leading to a worst-case $O(\frac{\Theta}{\eta_h} \times \frac{\Phi}{\eta_v})$ intersections computation (e.g. around 1000 with typical 60° and 20° of horizontal and vertical angular openings respectively, and $\eta_h = \eta_v = 1°$). Note that once an angular deviation is fixed on one of the parameters ϕ' or θ', if the intersection with the look-ahead time window $[0, \tau]$ is empty (e.g. violating inequation 1 for the horizontal plane), then the intersection processing in the other dimension can be spared (as inequation 2 will be necessarily violated as well).

C. Heading Change Strategies

With the previous sections, we can easily compute the set of conflict-free maneuvers noted CF. More specifically, this set can be defined by using the boolean function cf defined

in Section II-A. This function indicates that the aircraft B and the UAS A are free of conflict within τ minutes. With this function, the mathematical definition of CF is easy:

$$CF = \{\overrightarrow{v_A}' \in PM \ \mid \ cf\,(A, B, \overrightarrow{v_A}', \overrightarrow{v_B})\}$$

We need to chose one maneuver in CF. To do so, there are a number of strategies. As observed in our previous study ([30]), one strategy alone does not offer a good compromise between conflicts avoidance and maneuver quantity (cf. section III-B). But a hybrid strategy able to switch between minimal maneuvering and fastest escaping when the conflict becomes dire seems to be the most promising. To determine the current strategy, we present a new criterion that takes the dynamic of the conflict into account instead of the distance alone.

If the current speed does not generate a conflict within τ minutes (i.e. $\overrightarrow{v_A} \in CF$) then the UAS can keep its current speed. If the current speed generates a conflict at a date $t_{conflict} \leqslant \tau$ (i.e. $\overrightarrow{v_A} \notin CF$), then we have to choose a maneuver. If we can escape the conflict in one maneuver (i.e. $CF \neq \varnothing$) then we may have enough time to escape the conflict with a minimal deviation if $t_{conflict}$ is large (Closest). If $t_{conflict}$ is small then the actual conflict is imminent and we chose the fastest escaping maneuver among the conflict-free maneuvers (Safest). If the conflict cannot be escaped in one maneuver (i.e. $CF = \varnothing$) then the situation is dire and we opt for the fastest escaping maneuver (Fallback).

The difficulty in the conception of this hybrid strategy is to build a balanced criteria used to switch from strategy to one another. As a consequence this hybrid strategy is parameterized by a parameter γ. The value of $\gamma\tau$ is the threshold that decides if the conflict is imminent (i.e. $t_{conflict} < \gamma\tau$). Therefore, this parameter allows to continuously slide from a pure *Closest* strategy for $\gamma = 0$ to a pure *Safest* strategy for $\gamma = 1$. The influence of parameter γ is discussed in section III-B.

To summarize, the hybrid strategy choosing $\overrightarrow{v_A}'$ is parameterized by $\gamma \in \mathbb{R}$ which varies from 0 to 1:

$$\overrightarrow{v_A}' = \begin{cases} \overrightarrow{v_A} & \text{if } \overrightarrow{v_A} \in CF \\ Closest & \text{if } \gamma\tau \leqslant t_{conflict} \leqslant \tau \text{ and } CF \neq \varnothing \\ Safest & \text{if } t_{conflict} < \gamma\tau \text{ and } CF \neq \varnothing \\ Fallback & \text{otherwise (i.e. } CF = \varnothing) \end{cases}$$

This hybrid strategy uses different strategies that are described in the following sub-sections:

1) Closest: In order to minimize the maneuver burden assigned to the UAS when the threat is low, the conflict-free maneuver closest to the UAS current velocity is chosen. In 2D, it was sufficient to minimize the heading deviation, but in 3D, the euclidean *distance* between current and new speed must be minimized:

$$Closest: \qquad \underset{\overrightarrow{v_A}' \in CF}{\arg\min} \|\overrightarrow{v_A} - \overrightarrow{v_A}'\| \tag{3}$$

where $\overrightarrow{v_A}'$ is defined by a feasible conflict-free maneuver in CF, i.e. for which inequation 2 is false.

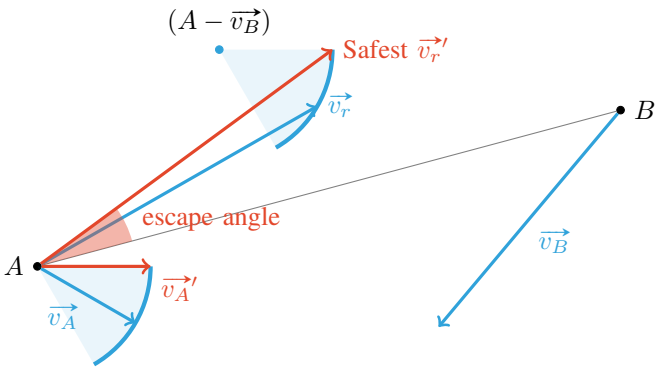

Fig. 2. *Safest* maneuver and underlying escape angle.

As the only global minimum of this distance trivially is the current speed $\overrightarrow{v_A}$ (or any reference velocity), the search for the best conflict-free maneuver could consider them by increasing distance and select the first admissible one encountered to speed up the sampling process.

Note that this criterion can be used as well if the speed norm is allowed to change.

Other criterion, for example biased towards horizontal or vertical maneuvers to emphasize the least costly change, or taking into account the flight plan of the UAS with a reference velocity, could be used in the same fashion.

2) Safest: When the conflict is imminent or when unexpected changes of the intruding aircraft trajectory are worsening the threat, the *Closest* strategy is not always able to solve the conflict. More radical maneuvers should then be selected to maximize the expectation of escaping the conflict. In [30], we proposed to choose the most "robust" maneuver defined as the maneuver authorized by the turn rate of the UAS that is the closest to the middle angle of the largest conflict-free angle range (without taking maneuverability into account).

With discretized 3D maneuvers, we could define a similar criterion based on the conflict-free maneuvers *graph* regardless of the UAS maneuverability, where nodes are maneuvers defined by $(\theta', \phi') \in [-\pi, \pi] \times [-\pi/2, \pi/2]$ and edges connect nodes that exactly differ of one discretization step on one component, i.e. the spherical grid graph that discretizes (θ', ϕ') space. We could then select the center of its largest connected component as the analog of the most robust maneuver in the horizontal plane.

However, it may prove too costly to compute the set of all conflict-free maneuver over $[-\pi, \pi] \times [-\pi/2, \pi/2]$ and we resort to a much more simple criterion to avoid the worst threats: the *escape angle* between $\overrightarrow{v_r}'$ and $\overrightarrow{AB}$, which should be chosen to avoid the conflict. Therefore, the *Safest* maneuver can be selected as the one that minimizes the cosine of the escape angle, which can be computed by:

$$Safest: \qquad \underset{\overrightarrow{v_A}' \in CF}{\arg\min} \frac{\overrightarrow{v_r}' \cdot \overrightarrow{AB}}{\|\overrightarrow{v_r}'\| \left\|\overrightarrow{AB}\right\|} \tag{4}$$

This is illustrated in figure 2 with an UAS at position A and an intruding aircraft at position B.

Similarly to the *Closest* strategy based on a distance from a reference vector, the search for the optimal maneuver w.r.t. the *Safest* one could be sped up by starting with potential minima, regardless of the conflict constraints. However, as the 3D dot product involves two cosine functions, there can be up to four local minima from which the search should start to find the global optimum.

Moreover, these schemes can easily be generalized for a UAS with variable speed and the graph-based one naturally extends to the multiple intruding aircraft case.

3) Fallback: If the conflict-free maneuvers set CF is empty, it means that no turning angle nor vertical move can guarantee the horizontal and vertical separation distance d_h and d_v for the next τ minutes.

In such a case, the hybrid strategy resorts to a fallback strategy consisting in choosing the maneuver among PM which gives the biggest *escape angle* between $\overrightarrow{v_r}'$ and $\overrightarrow{AB}$. Thus, the maximal *escape angle* among the possible maneuvers PM is chosen:

$$\textit{Fallback:} \qquad \underset{\overrightarrow{v_A}' \in PM}{\arg\min} \frac{\overrightarrow{v_r}' \cdot \overrightarrow{AB}}{\|\overrightarrow{v_r}'\| \left\|\overrightarrow{AB}\right\|} \tag{5}$$

D. Multiple Aircraft

In real traffic situations, conflicts involving more than two aircraft often occur and cannot always be solved pairwise in sequence, but should be globally handled. Fortunately, our geometrical model can be easily extended to take into account several aircraft. For n intruding aircraft B_i, we just need to compute the intersection of all individual $CF_i = \{\overrightarrow{v_A}' \in PM \mid cf\,(A, B_i, \overrightarrow{v_A}', \overrightarrow{v_{B_i}})\}$ to obtain the global conflict-free maneuvers set ***CF***:

$$CF = \bigcap_{i=1}^{n} CF_i$$

For the *Closest* strategy, the selection of the best maneuver is then identical to the single aircraft case with this new definition of CF. It is not so straightforward for the *Safest* strategy as the criterion depends on $\overrightarrow{v_r}'$ and $\overrightarrow{AB}$, so will change for each intruding aircraft B_i considered.

Various techniques could be used to solve this "multi-objective" optimization problem. For example the best conflict-free maneuver for the aircraft involved in the most dire conflict could be selected. The "direness" of a conflict could be estimated by the time before collision for closing trajectories:

$$\frac{\left\|\overrightarrow{AB_i}\right\|^2}{\overrightarrow{v_r} \cdot \overrightarrow{AB_i}}$$

or by the size of the conflict-free maneuvers set CF for each intruding aircraft B_i.

III. Experiments

Our geometrical model has been implemented and thoroughly tested for the single aircraft case on recorded real TCA traffic enhanced by injecting various conflicting UASs scenarios. The following section describes the data and scenarios generation, and the next one reports the results and analysis of our experiments.

UAS	Speed	Max. climb rate	Max. descent rate	Turn rate
UAS1	80 kt	700 ft min^{-1}	2100 ft min^{-1}	3 ° s^{-1}
UAS2	160 kt	2200 ft min^{-1}	4200 ft min^{-1}	3 ° s^{-1}

TABLE I
UAS SPECIFICATIONS FOR SIMULATION.

A. Description of simulations

Simulations have been carried out on recorded traffic data from a terminal maneuvering area in the south-west of the French airspace on 2013/09/14. The use of recorded tracks in TCA provides a most realistic picture of the trajectories that UASs might be confronted to in operations, as many UAS missions are flown in the lower airspace. Besides, such scenarios put the detect and avoid algorithm to stress because the trajectories of surrounding aircraft are convoluted and exhibit more complex 3D profiles than en-route traffic. The traffic sample contains 478 flights, evenly shared between departures and arrivals.

The simulated trajectories for UASs are derived from recorded tracks of aircraft with similar performances. Two types of UAS have been tested in our experiments, representing a small UAS with relatively poor flying performance and a more maneuverable UAS respectively. The specifications used in the simulations are described in table I. For each UAS model, six missions are implemented, corresponding to all possible combinations of two horizontal profiles (constant heading or circle around a fixed point) and three vertical profiles (constant altitude, climb or descent). Thus, for a given set of parameters and a given UAS, 2868 simulation scenarios are simulated. Each of these scenarios is built in such a way that, if no maneuver is issued at all, a collision would occur between aircraft and UAS.

A fast-time simulator enables us to play the trajectories (both recorded and built) and to modify them by sending maneuver messages consisting of a heading change and a turn rate for horizontal maneuvers, and a flight level change and a climb (or descent) rate for vertical maneuvers. Those messages are sent to UASs only, other aircraft are left unchanged.

To evaluate the outcomes of our experiments, we use several separation volumes around the aircraft corresponding to different objectives of the detect and avoid function, as illustrated in figure 3. The smallest one (in red) is the *collision volume*, in which the UAS should never enter. It is defined by ICAO [31] as a cylinder of radius 500 ft and height 200 ft centered on the aircraft position. The second volume (in green) around the collision volume corresponds to the *remain well clear* function as defined in [32]. This volume is also an aircraft-centered cylinder, with radius 4000 ft and height 900 ft. Finally, we use a third cylindrical volume (in blue) corresponding to the target separation of our algorithm. The dimensions of this target volume vary, in this study, from the dimensions of the well

	ST1	ST2	ST3	ST4
Radius	4000 ft	1 NM	2 NM	3 NM
Height	900 ft	1000 ft	1400 ft	2000 ft

TABLE II
SEPARATION TARGETS (ST) VOLUMES USED IN SIMULATION. ST1 CORRESPONDS TO THE WELL CLEAR VOLUME AND ST4 CORRESPONDS TO ATC SEPARATION IN TCA. ST2 AND ST3 ARE INTERMEDIATE VALUES.

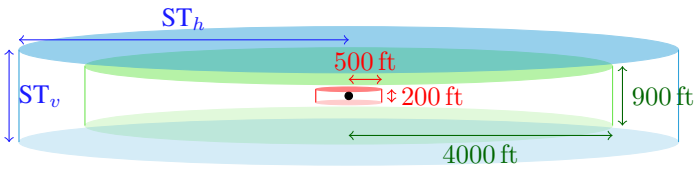

Fig. 3. Description of the aircraft-centered separation volumes: in red the collision volume, in green the well clear volume and in blue the target separation volume. ST$_h$ and ST$_v$ are the horizontal and vertical separation targets respectively.

clear volume up to 3 NM and 2000 ft (which corresponds to ATC separation in TCA in France). The various values used for the target separation are provided in table II. For each scenario where the algorithm could not avoid it, we keep track the fact that the UAS entered the well clear volume or the collision volume.

We also record the number of maneuvers given by the algorithm, as well as the cumulated quantity of deviation from original flight direction (both in horizontal and vertical planes).

For each set of parameters (i.e. γ and the separation target) and for each UAS and mission, we perform three different simulations, allowing only horizontal maneuvers, only vertical maneuvers or both horizontal and vertical maneuvers (these are referred to as H, V and 3D respectively in the following results).

B. Results and analysis

In this section, we analyze the outcomes of many simulations: two UASs, eight values for γ and four target separation volumes have been tested. For each set of parameters, all 2868 traffic scenarios are played, leading to a total of more than 180 000 single simulations.

1) Impact of γ on the maneuvers: In order to quantify the impact of γ on the maneuvers, we have summed all the deviations (both vertical and horizontal) given to the UAS. This sum is then divided by the number of scenarios giving us the average maneuver quantity. This value is depicted in figure 4, which also depicts the average number of maneuvers. γ is increased from 0 to 1, which corresponds to a shift from a Closest-only strategy to a Safest-only strategy.

We see that the average maneuver quantity increases slowly with γ, with a higher rate low γ values and for the 3D maneuvers. This is expected, as the Safest maneuvers tend to produce be more intense than Closest maneuvers.

Interestingly, the average number of maneuvers decreases with γ. This means that, even if the Safest strategy gives larger deviations than the Closest strategy, the maneuvers tend to be more efficient at maintaining separation, thus needing less further corrections in the following time steps. With the closest strategy, however, minimal deviations are applied that will more likely need to be adapted.

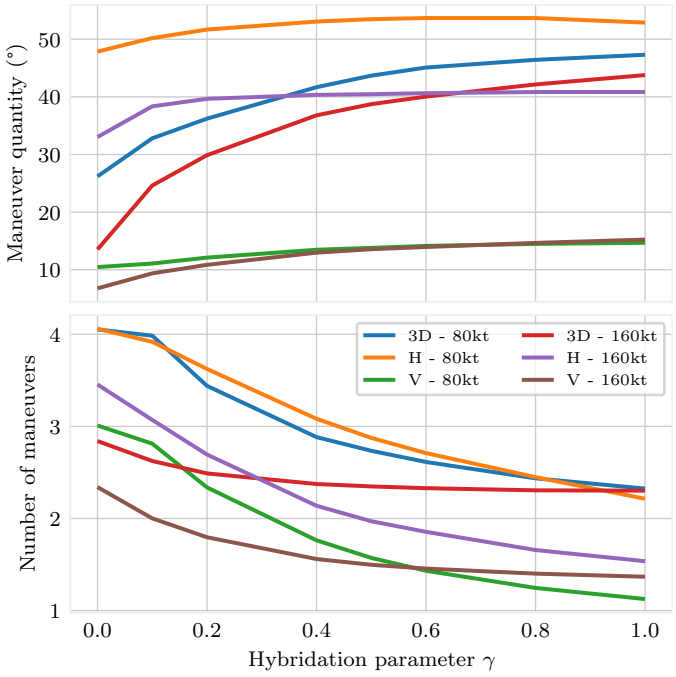

Fig. 4. Average number of maneuvers and average maneuvering quantity w.r.t. hybridization parameter γ.

2) Separation and Deviation: The main objective of the detect and avoid algorithm is to maintain a given separation distance between the UAS and surrounding aircraft. Besides, the deviation of the UAS from its mission also matters significantly for the sake of efficiency. However, these two criterion are often antagonistic.

Our hybrid method is parameterized by γ and the separation volume. These parameters must be tuned to find the best combination. As two opposed criterion must be taken into account, it is not possible to find a parameter combination that is best for both criterion. Actually, we observe that several sets of parameters lead to solutions that are not dominated for both criteria.

These non dominated criteria constitute a Pareto frontier, which is depicted in figures 5 and 6. More specifically, figure 5 depicts the percentage of success of "remain well clear" with respect to the quantity of maneuver that is necessary to achieve this result.

On figure 5, we can see that we can remain well clear in 95 % of the scenarios if the parameters are well chosen. This can be achieved with a maneuver quantity inferior to 40° except for UAS1 when only heading changes are authorized.

On figure 6, we can see that we can avoid the collision volume in 99.8 % of the scenarios if the parameters are well chosen. This can be achieved with a maneuver quantity inferior to 40° except, again, for UAS1 when only heading changes are authorized. This corresponds to only a handful of situations (one or even zero depending on the parameters with 3D maneuvers, compared to the 2868 scenarios), that need further analyze to adapt our algorithm.

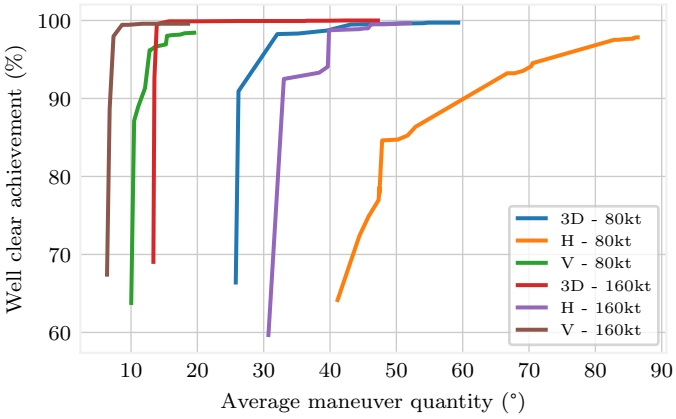

Fig. 5. Percentage of success of "remain well clear" w.r.t. the necessary quantity of maneuver.

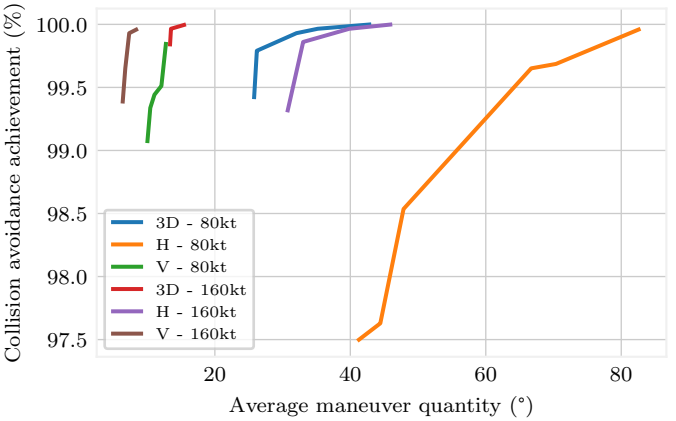

Fig. 6. Percentage of success of collision avoidance w.r.t. the necessary quantity of maneuver.

For both figures, the best results were obtained with UAS2, the fastest UAS. This can be explained by the fact that a deviation on the heading or the path have a different impact on the new relative speed depending on the speed of the UAS. The higher the UAS speed, the higher is the impact on the relative speed.

Also, the vertical maneuvers are more efficient to solve the conflicts than the horizontal maneuvers. This is especially true for UAS2. This can be explained by the fact that the vertical norm is smaller than the horizontal one. In terms of separation success percentage, the 3D maneuvers are the most efficient. In that respect, the use of vertical maneuvers seems mandatory for UAS1, the slowest UAS.

After analyzing our results with respect to both separation and efficiency objectives, we found that targeting a separation volume $50\,\%$ larger than the well clear volume and setting γ to 0.2 was a particularly interesting setting. This enables for an achievement of well clear separation of $95\,\%$ for UAS1 and almost $99\,\%$ for UAS2, while keeping the deviation to a very low level. Also, $\gamma = 0.2$ corresponds, in our settings, to triggering a maneuver only one minute before the potential loss of separation.

IV. Conclusion and Future Work

In this study, we proposed a three-dimensional detect and avoid algorithm for UAS integration in air traffic, based on previous work on a 2D approach. This algorithm aims at providing separation between a UAS and commercial traffic at departure or arrival, but proved to be efficient for short-term collision avoidance as well.

The possible maneuvers for the UAS, either horizontal, vertical or both, are sampled and each one of them is geometrically confronted to the constraint induced by the intruder aircraft, in order to obtain a set of feasible trajectories. Then, two strategies have been defined and then combined to choose the best maneuver. The first strategy (Closest) follows the mission as close as possible, while the second one (Safest) tries to escape from the conflicting area in the fastest possible way. The algorithm is set up to switch between strategies depending on the seriousness of the situation.

This approach has been intensively tested in simulations involving real traffic recorded in a French TCA. With the proper set of parameters, the required separation was provided with a very high success rate, and collision avoidance was ensured at almost $100\,\%$. The underlying trajectory deviations are kept at an operationally acceptable level, and the algorithm can be set up so that the maneuvers are triggered only one minute before the predicted loss of separation.

In further works, we plan to increase the possibilities of the UAS by authorizing speed changes, which might help in finding solutions with less deviations from the mission.

One of the pitfalls of our method is that it only takes into account the current state, so that any further change in the aircraft state could break the resolution. In order to improve the robustness of the maneuvers, we plan to try and anticipate better, both on the aircraft intentions and the UAS capabilities.

Knowing the past positions of the aircraft, it is possible to build a short-term predicted trajectory, based on the analysis of the derivatives of its speed and turn angle. For example, the beginning or the end of a turn, a climb or a descent could be inferred. Particular care would have to be taken during the calibration phase, especially when choosing the number of past states to consider: too much states would create some latency in the prediction, whereas too few states would yield unreliable ones.

If the aircraft trajectory could be predicted this way, then it becomes particularly interesting to anticipate several maneuvers for the UAS. This could be planned optimally with an A* or Dijkstra algorithm, using the geometrical algorithm at each step to prune the search tree or validate the existence of partial solutions, at the cost of a significantly longer computation. It could also be performed geometrically by an approximation of a few maneuvers aggregated into a single one.

References

[1] ABIresearch, "Small unmanned aerial systems market exceeds US$8.4 billion by 2019, dominated by the commercial sector and driven by commercial applications," January 2015.

[2] DCF, Sagem, DSNA, and ENAC, "ODREA demonstration report," tech. rep., SESAR Joint Undertaking, 2015.

[3] C. Allignol, N. Barnier, N. Durand, and É. Blond, "Detect & Avoid, UAV Integration in the Lower Airspace Traffic," in *ICRAT 2016, 7th International Conference on Research in Air Transportation*, ICRAT 2016 Proceedings, (Philadelphia, United States), June 2016.

[4] C. Allignol, N. Barnier, N. Durand, G. Manfredi, and É. Blond, "Assessing the Robustness of a UAS Detect & Avoid Algorithm," in *12th USA/Europe Air Traffic Management Research and Development Seminar*, (Seattle, United States), June 2017.

[5] J. Snape and D. Manocha, "Navigating multiple simple-airplanes in 3D workspace," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3974–3980, 2010.

[6] J. van den Berg, S. J. Guy, M. C. Lin, and D. Manocha, "Reciprocal n-body Collision Avoidance," in *14th International Symposium on Robotics Research*, pp. 3–19, 2011.

[7] N. Durand and N. Barnier, "Does ATM need centralized coordination? Autonomous conflict resolution analysis in a constrained speed environment," in *11th ATM R and D Seminar*, 2015.

[8] K. Zeghal, "A comparison of different approaches based on force fields for coordination among multiple mobiles," in *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, vol. 1, pp. 273–278 vol.1, Oct 1998.

[9] J. Kosecka, C. Tomlin, G. Pappas, and S. Sastry, "2-1/2 D Conflict Resolution Maneuvers for ATMS," in *in Proc. 37th IEEE Conf. Decision Control*, pp. 2650–2655, 1998.

[10] M. Eby and I. Kelly, W.E., "Free flight separation assurance using distributed algorithms," in *Aerospace Conference, 1999. Proceedings. 1999 IEEE*, vol. 2, pp. 429–441 vol.2, 1999.

[11] G. Granger, N. Durand, and J. Alliot, "Token allocation strategy for free-flight conflict solving," in *IJCAI'01*, 2001.

[12] G. Granger, N. Durand, and J. Alliot, "Optimal resolution of en route conflicts," in *4th ATM R and D Seminar*, 2001.

[13] N. Archambault and N. Durand, "Scheduling heuristics for on-board sequential air conflict solving," in *Digital Avionics Systems Conference, 2004. DASC 04. The 23rd*, vol. 1, pp. –3.1–9 Vol.1, Oct 2004.

[14] N. Durand, J. Alliot, and F. Medioni, "Neural nets trained by genetic algorithms for collision avoidance," *Applied Artificial Intelligence*, 2000.

[15] I. Hwang, J.Kim, and C. Tomlin, "Protocol-based conflict resolution for air traffic control," *ATC Quarterly*, vol. 15, pp. 1–34, 2007.

[16] J. Le Ny and G. Pappas, "Geometric programming and mechanism design for air traffic conflict resolution," in *American Control Conference (ACC), 2010*, pp. 3069–3074, June 2010.

[17] N. Durand, J.-M. Alliot, and J. Noailles, "Automatic aircraft conflict resolution using genetic algorithms," in *Proceedings of the Symposium on Applied Computing, Philadelphia*, ACM, 1996.

[18] F. Krella *et al.*, "Arc 2000 scenario (version 4.3)," tech. rep., Eurocontrol, April 1989.

[19] Y.-J. Chiang, J. Klosowski, C. Lee, and J. Mitchell, "Geometric algorithms for conflict detection/resolution in air traffic management," in *Decision and Control, 1997., Proceedings of the 36th IEEE Conference on*, vol. 2, pp. 1835–1840 vol.2, Dec 1997.

[20] J. Hu, M. Prandini, A. Nilim, and S. Sastry, "Optimal coordinated maneuvers for three dimensional aircraft conflict resolution," *AIAA Journal of Guidance, Control and Dynamics*, vol. 25, p. 2002, 2002.

[21] J.-H. Oh, J. Shewchun, and E. Feron, "Design and analysis of conflict resolution algorithms via positive semidefinite programming [aircraft conflict resolution]," in *Decision and Control, 1997., Proceedings of the 36th IEEE Conference on*, vol. 5, pp. 4179–4185 vol.5, Dec 1997.

[22] E. Frazzoli, Z.-H. Mao, J.-H. Oh, and E. Feron, "Resolution of conflicts involving many aircraft via semidefinite programming," *AIAA Journal of Guidance, Control and Dynamics*, vol. 24, Jan-Feb 2001.

[23] L. Pallottino, A. Bicchi, and E. Feron, "Mixed integer programming for aircraft conflict resolution," in *AIAA Guidance Navigation and Control Conference and Exhibit*, 2001.

[24] L. Pallottino, E. Feron, and A. Bicchi, "Conflict resolution problems for air traffic management systems solved with mixed integer programming," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 3, pp. 3–11, Mar 2002.

[25] M. A. Christodoulou and C. Kontogeorgou, "Collision avoidance in commercial aircraft free flight via neural networks and non-linear programming," *Int. J. Neural Syst.*, vol. 18, no. 5, pp. 371–387, 2008.

[26] M. Gariel and E. Feron, "3D conflict avoidance under uncertainties," in *Digital Avionics Systems Conference, 2009. DASC '09. IEEE/AIAA 28th*, pp. 4.E.3–1–4.E.3–8, Oct 2009.

[27] N. Durand and J.-M. Alliot, "Optimal resolution of en route conflicts," in *First ATM Seminar Europe/USA, Saclay*, 1997.

[28] C. Allignol, N. Barnier, N. Durand, and J.-M. Alliot, "A New Framework for Solving En-Route Conflicts," in *10th USA/Europe Air Traffic Management Research and Developpment Seminar*, 2013.

[29] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[30] C. Allignol, N. Barnier, N. Durand, G. Manfredi, and É. Blond, "Integration of UAS in Terminal Control Area," in *DASC 2016 IEEE/AIAA 35th Digital Avionics Systems Conference*, Digital Avionics Systems Conference (DASC), 2016 IEEE/AIAA 35th, (Sacramento, United States), Sept. 2016.

[31] EUROCONTROL, *ACAS Guide – Airborne Collision Avoidance*, 3.0 ed., Dec. 2017.

[32] S. P. Cook, D. Brooks, R. Cole, D. Hackenberg, and V. Raska, "Defining well clear for unmanned aircraft systems," in *AIAA Infotech @ Aerospace*, (Kissimmee, FL), American Institute of Aeronautics and Astronautics, Jan. 2015.