Detect & Avoid, UAV Integration in the Lower Airspace Traffic

Cyril Allignol, Nicolas Barnier, Nicolas Durand, Éric Blond

Detect & Avoid, UAV Integration in the Lower Airspace Traffic

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

Éric Blond
DSNA/DTI
1, av. du Dr Maurice Grynfogel
31 035 Toulouse, France
firstname.surname@aviation-civile.gouv.fr

***Abstract*—In this article, we test a horizontal *detect and avoid* algorithm for UAVs flying in the lower airspace (under FL180). We use recorded commercial traffic trajectories and randomly build 3000 conflicts scenarios with UAVs to check the ability of such an algorithm to ensure the separation with commercial aviation. We consider two different types of UAVs, the first type flying at $80\,\mathrm{kn}$ and the second type flying at $160\,\mathrm{kn}$ with six different missions: flying straight or turning and leveled, climbing or descending. We only focus on horizontal maneuvers in order not to interfere with aircraft TCAS. The article investigates the influence of the various parameters on the separation achieved. The analysis of results from over 200 000 simulations provides minimum requirements on the frequency and anticipation time of the resolution process for an efficient detect and avoid strategy and brings up remaining issues in scenarios where the UAV has a low maneuverability and encounters fast airliners with changing speed and heading.**

Keywords: conflict resolution, detect and avoid, self-separation, UAV, geometrical algorithm

I. Introduction

In the next years, the demand to operate civilian UAVs for very diverse missions (ranging from fire detection and river bed surveillance to small items delivery and archaeological exploration...) has rocketed [1]. Most of these UAVs operate within the lower airspace (under FL180) and their trajectories may interfere with commercial aviation traffic in Terminal Maneuvering Areas (TMA). Developing new algorithms to help separating UAVs from the rest of the traffic is becoming critical for safety reasons. In this context, several approaches could be investigated.

First, separation can be entirely managed by Air Navigation Service Providers. Experiments [2] have shown that Air Traffic Controllers resolution process is hindered by UAVs mixed with conventional traffic because they have unusual performance specifications and interact with different time responses.

Otherwise, separation could be delegated to *both* commercial aircraft and UAVs which could autonomously maneuver to resolve potential conflicts. However, complex processes of coordination should be considered in such a context to keep Air Traffic Controllers aware of the resolution process and able to interfere in it.

Finally, conflict resolution could be taken care of by UAVs only such that they do not disrupt the commercial traffic. This seems a more realistic approach, provided that the positions and speeds of surrounding aircraft are available (through ADS-B for example) and that the performances of UAVs and the resolution anticipation are sufficient to solve all traffic situations. In this article, we investigate the latter by adapting a self-separation algorithm used in robotics to our context and experiment with various parameters and strategies on real traffic samples.

1) Detect & Avoid: This algorithm was designed by van den Berg et al. [3] and tested with different speed constraint hypotheses by Durand et al. [4] in the context of autonomous air conflict resolution. Compared to these first experiments, we here tailor the geometrical approach of van den Berg further to model the performances of UAVs and consider specific fallback strategies to handle cases for which the first approach fails to maintain separation.

More specifically, adapting the algorithm to integrate UAVs in commercial traffic leads to the following major differences from the preliminary work presented in [4]:

- The UAV is here supposed to avoid other aircraft, which do not avoid the UAV. This means that the whole avoidance maneuver is endorsed by the UAV.
- UAVs used in a civilian context generally fly with low speeds compared to commercial aircraft. The ratio we 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 $250\,\mathrm{kn}$, but recorded data show that in practice some aircraft fly much faster (up to $400\,\mathrm{kn}$). In this study, we consider two types of UAVs: Fast UAVs flying at $160\,\mathrm{kn}$ and Slow UAVs flying at $80\,\mathrm{kn}$.
- Moreover, most civilian UAVs have very poor speed up performances compared to conventional aircraft. We will therefore only consider maneuvers at constant speed for UAVs, as this degree of freedom would have almost no 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.
- Air traffic trajectory prediction, which is one of the main component of a conflict solver, is always tainted with uncertainties that must be taken into account to assess the efficiency of an algorithm. We show how our approach could be extended to handle uncertainties and provide robust resolution maneuvers.

2) Related Works: When the concept of Free-flight emerged in the 90s, one of the ideas was then to equip every aircraft with a detect and avoid algorithm able to ensure separation with the rest of the traffic.

The first effective approach used sliding forces to coordinate maneuvers between aircraft [5]. Potential or vortex fields [6] as well as a model based on an analogy with electrical particle repulsion [7] were also used. In 2001, we proposed a token allocation strategy combined with an A^* algorithm to solve conflicts with realistic maneuvers [8], [9]. 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 [10]. We also tried artificial Neural Networks on the two-aircraft problem [11] 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 [3], [12]–[14]. The powerful technique developed by Van den Berg et al. [3] can handle thousands of agents in a small space. It was applied to aircraft by Snape et al. [14], but the hypotheses of the algorithm require simultaneous vertical and horizontal speed changes. We also tested them [4] 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 [15]. Most centralized approaches that have been proposed to solve conflicts can be broadly divided into two main categories. The first ones [16]–[18] use greedy sequential algorithms to optimize trajectories one by one after ranking the aircraft (ordering aircraft is however very challenging [10]). 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 [19]–[24]. In [25], [26], 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.

3) Outline: In this article, we come back to a simpler problem in a more realistic environment. We consider UAVs flying in the lower airspace (under FL180) and design various conflict scenarios with real recorded commercial aircraft trajectories in TMAs. The aim of the study is to assess the performances of a "detect and avoid" strategy for UAVs to maintain a reasonable horizontal separation with commercial traffic. We first consider conflicts involving one UAV and one commercial aircraft only, then we introduce multiple aircraft encountering the UAV.

In section II of this article, we detail the geometrical approach developed by Van den Berg et al. [3] in the case of a single UAV avoiding non cooperative aircraft. We show how the algorithm can be adapted to take into account the constant speed constraint and propose several strategies to choose the maneuver when a future conflict is detected. We also introduce a fallback scheme that reduces the targeted separation distance when no solution is available to maintain separation. Moreover, we describe how our tailored model can be easily extended to take more than two aircraft into account. Finally, we show how some uncertainties on trajectory prediction can be simply taken into account in our model. In section III-A, we describe how we built the conflict scenarios from real traffic data and the hypotheses that were chosen for the UAVs. 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 maneuvers imposed on the UAVs. The last section draws some conclusions about the results obtained with the simulations and highlights directions for future work.

II. Detect and Avoid Model

This section describes the algorithm developed in [3] and its adaptation to the case where only one aircraft (the UAV) maneuvers. We first detail the model with a single aircraft and a minimizing strategy that requires to change the speed of the UAV, then show how the speed of the maneuvering UAV can be constrained to have a constant norm at the cost of greater heading deviations. We also describe a fallback strategy to minimize the separation violation when our model has no solution within the allowed turning range. Our model is then further extended to simultaneously take into account several aircraft while keeping the UAV speed constant. Finally, we describe how uncertainties on aircraft headings could be simply handled.

A. Single Aircraft

Let d be the standard separation between aircraft and τ be a look ahead time. In figure 1, let us consider UAV A and aircraft B. We can represent the position of aircraft B in the referential of UAV A. If we draw a circle of radius d centered at aircraft B, the two lines issued from position A, tangent to the circle of radius d form a cone. If the relative speed $\overrightarrow{v_r} = \overrightarrow{v_A} - \overrightarrow{v_B}$ lies in this cone, a conflict will occur in the future. If we draw a circle of size $\frac{d}{\tau}$ tangent to the two previous lines, we obtain a new zone (in light red) bounded by the bold line in figure 1. It is then straightforward to understand that a conflict will occur within time τ if and only if $\overrightarrow{v_r}$ lies in this zone.

When a potential conflict occurs, i.e. the endpoint of $\overrightarrow{v_r}$ lies in the forbidden zone, $\overrightarrow{v_r}$ must be changed to escape it. If the speed change is not constrained (by the performances of

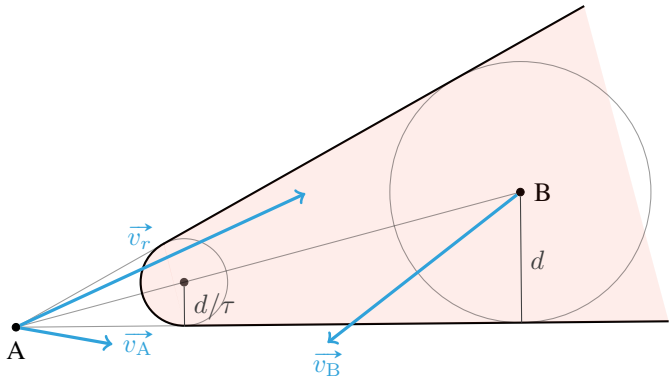

Fig. 1. Conflicting aircraft model: a conflict will occur within time τ if and only if the relative speed $\vec{v_r}$ lies in the forbidden zone in red.

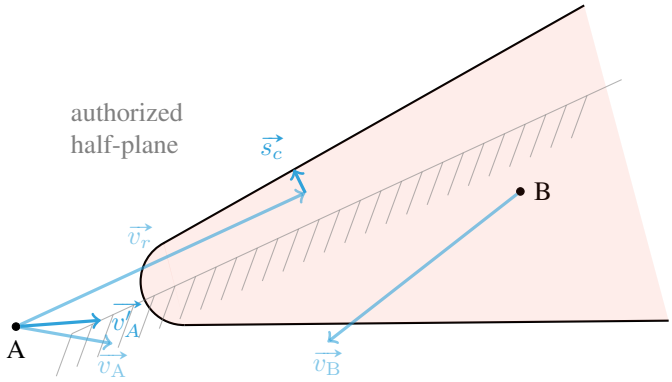

Fig. 2. Conflict resolution model: the necessary speed change to move $\vec{v_r}$ out of the forbidden zone is endorsed by the UAV.

the aircraft for example), $\vec{v_r}$ can be projected on the nearest bold line (see figure 2) to solve the conflict pairwise between UAV A and aircraft B while minimizing the speed change. In [3], this change is shared in half by both aircraft. Here, we constrain the necessary speed change $\vec{s_c}$ that moves $\vec{v_r}$ out of the forbidden zone to be entirely attributed to UAV A.

We can then define a half-plane bounded by the line going through point $A + \vec{v_A} + \vec{s_c}$ parallel to the corresponding edge of the cone on the side determined by $\vec{s_c}$: if the new speed of the UAV $\vec{v'_A}$ is chosen such that its endpoint lies in this half-plane, then the new relative speed $\vec{v_r}$ is out of the forbidden speed zone, as illustrated on figure 2.

However, this minimizing strategy generally results in changing the norm of the speed of the UAV, which is not realistic considering the average performances of commercial UAVs. We show in section II-B how to take this additional constraint into account in our geometrical model.

B. Constant Speed

We consider here that a UAV can only change its heading and not its speed norm. In reality, a UAV can change its speed norm, but the change rate is very limited and almost negligible with respect to the speed of surrounding aircraft (typically airliners).

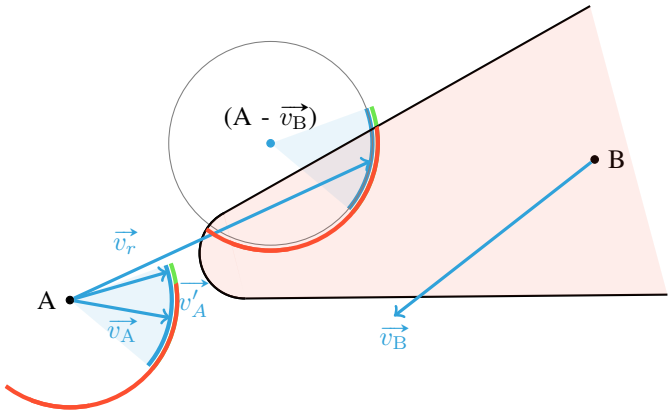

Fig. 3. Model with constant speed norm for the UAV: the end of $\vec{v'_r}$ must lie on the circle but outside of the forbidden zone (in light red) and in the allowed turn angle range of 30° around $\vec{v_A}$, which leaves the tiny green safe zone as possible new heading for the UAV.

To ensure that the norm of the speed of the UAV remains constant throughout the conflict resolution process, we must have:

$$\left\|\vec{v'_A}\right\| = \|\vec{v_A}\|$$

as $\vec{v'_r} = \vec{v'_A} - \vec{v_B}$, we also have:

$$(x_{v'_r} + x_{v_B})^2 + (y_{v'_r} + y_{v_B})^2 = \|\vec{v_A}\|^2$$

which means that the possible endpoints of $\vec{v'_r}$ belongs to a circle of radius $\|\vec{v_A}\|$ centered at $A - \vec{v_B}$ as shown on figure 3. But $\vec{v'_r}$ must also lies outside the forbidden zone defined in the previous section, which removes all angle ranges of the circle included in it (in red on the figure). The remaining angles must be further filtered by intersecting the allowed turn angle range θ corresponding to the performance of the UAV, i.e. an arc of $\pm\theta$ around the current speed shown in blue on figure 3 (where $\theta = 30°$).

The conflict-free heading change thus ranges over an arc of a circle (pictured in green on figure 3) which is the difference of the allowed (in blue) and forbidden (in red) arcs. Note that the resulting allowed headings may comprise two disjoint angle ranges, which is generally the case for facing aircraft for example (as shown on figure 4) and corresponds to the combinatorial decision to turn right or left to solve the future conflict.

The new speed for the UAV can then be aggressively chosen as the closest available angle to minimize the heading change, which will result in optimizing the "maneuvers quantity" (a measure defined in section III-B). As it is expected that the relative performances of airliners and UAVs will incur many unsolvable scenarios, for which losses of separation will occur, it may be wiser to choose an angle that leaves some leeway for the next resolution steps, i.e. which does not saturate the separation constraint.

To maximize the expectation of escaping a future conflict in case of further maneuvering of aircraft B (without any

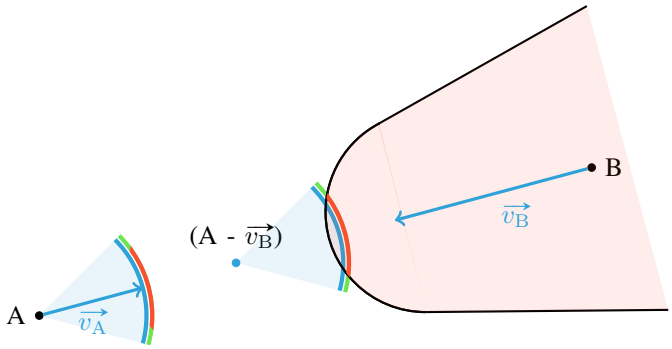

Fig. 4. Disjoint angle ranges (in green) for facing aircraft: the conflict can be solved either by turning left or right.

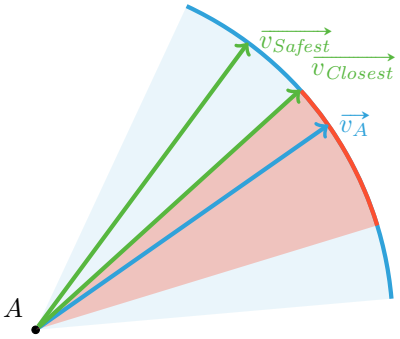

Fig. 5. *Closest* and *Safest* speeds, computed from $\overrightarrow{v_A}$, given the constraints in red.

knowledge of its intended trajectory), the new heading could instead be chosen as the median value of the largest allowed angle range (in case of several disjoint angle ranges), which will optimize the robustness of the maneuver. In the remaining sections, the former strategy will be referred to as *Closest* and the latter as *Safest*. Both strategies are pictured in figure 5.

C. Fallback Strategy

If the permitted heading range is empty, it means that no turning angle can guarantee the separation distance d for the next τ minutes. However, even if a conflict occurs, a brief one at a distance just below the target separation distance is preferable to a lasting close encounter.

To take into account this criterion while choosing a maneuver whenever a loss of separation is inevitable, our algorithm resorts to a fallback strategy consisting in reducing the separation distance until a solution can be found anew. Our current implementation uses a linear scheme with a constant step (0.1 NM in our experiments) and returns the corresponding maneuver as soon as a solution is found – though a dichotomous scheme could be used to speed up the search if more precision is deemed relevant.

The resulting maneuver therefore attempts to minimize the conflict, i.e. to maximize the minimal distance between the aircraft and the UAV, hopefully achieving a minimization of the conflict duration as well.

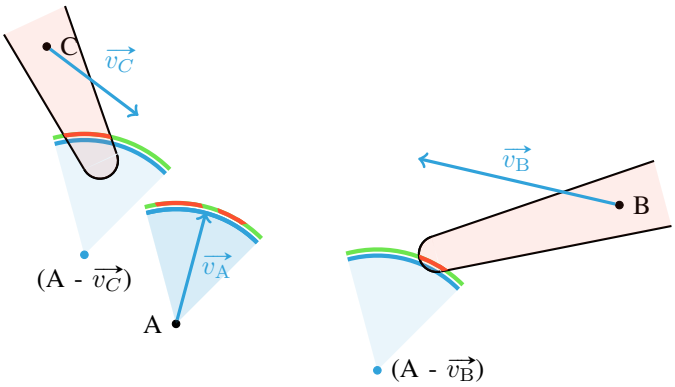

Fig. 6. Several disjoint allowed heading ranges (in green) for UAV A in presence of multiple aircraft (B and C).

D. Multiple Aircraft

In real traffic situations, conflicts involving more than two aircraft often occur and cannot be solved pairwise in sequence, but must be globally handled. Fortunately, our geometrical model can be easily extended to take into account any number of aircraft.

For each aircraft, a forbidden angle range is determined exactly as in the pairwise case detailed in section II-B and removed from the possible turning range of the UAV. The resulting allowed headings may then comprise several (possibly more than two) disjoint angle ranges, because each new constraint may remove a new hole in the remaining angle ranges (as illustrated in figure 6).

E. Uncertainties

In order to make the algorithm robust to uncertainties on trajectories, our multiple aircraft model could be easily extended by adding slightly noised duplicates of the adverse aircraft with different headings and speeds, then solving the overall problem including the fictitious aircraft.

Moreover, instead of generating symmetric (i.e. left *and* right, slower *and* faster) or random duplicates, the speed vector shifting could be biased by a trajectory prediction component (such as mentioned in the further works section) that records the past trajectory and infers short-term intentions: if the aircraft seems in a state of a stable right turn or speed decrease, right-shifted or slower duplicates should only be considered.

III. Experiments

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

A. Test Description

Experiments were conducted on real traffic data, recorded on 2013/09/14 in the south-west of the French airspace. This gives a much more realistic picture of the types of trajectories

that UAVs may encounter when flying in real traffic. This is all the more necessary because UAVs are essentially meant to fly in the lower airspace where most of the commercial traffic is climbing or descending.

UAV trajectories are built from recorded tracks of aircraft with similar performances for two UAV types (flying at 80 kn and 160 kn respectively), three vertical profiles (leveled flight, climb and descent) and two horizontal profiles (constant heading and circle around a fixed point), for a total of 6 different patterns for each UAV type.

We filtered trajectories that had at least six minutes of flight under FL195, leaving us with a set of 475 aircraft for a total of 2850 simulation scenarios. For each aircraft and each UAV pattern defining a scenario, we then isolate a plot $(x_{\mathrm{AC}}, y_{\mathrm{AC}}, \mathrm{FL}_{\mathrm{AC}})$ (respectively $(x_{\mathrm{UAV}}, y_{\mathrm{UAV}}, \mathrm{FL}_{\mathrm{UAV}})$) in the trajectory of the aircraft (respectively the UAV) that has at least 3 minutes of past positions. These points are used to create a conflict for this scenario, i.e. we adjust the UAV trajectory in space and time dimensions so that $x_{\mathrm{AC}} = x_{\mathrm{UAV}}$, $y_{\mathrm{AC}} = y_{\mathrm{UAV}}$ and $\mathrm{FL}_{\mathrm{AC}} = \mathrm{FL}_{\mathrm{UAV}}$. If no maneuver were issued, there would be a collision between the aircraft and the UAV.

A fast-time simulator enables 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. Those messages are send to UAV only, other aircraft are left unchanged.

In the experiments, we chose $\tau = 5$ minutes and tried to achieve 3 NM of separation when possible. We do not take the vertical separation into account because we want to test the efficiency of the detect and avoid process in the horizontal plane only. Further research will be conducted in the vertical dimension. We chose to detect and avoid conflicts every *step* seconds ($step = 10$ s when not specified otherwise).

B. Results with Various Speed and Turn Rate Ranges

Simulations were run on every scenario for 82 different sets of parameters, totaling up 233 700 individual runs[1]. We measured various indicators of the efficiency of the conflict avoidance maneuvers and count the occurrences of close distance events, that we set up to be the simulations were the distance between UAV and aircraft went under 1 NM. In the following, we will call an *airprox* (aircraft proximity) this kind of event.

1) UAV maneuverability: Figure 7 depicts the number of airprox events for various combinations of speed, turn rate and resolution method. As expected, escaping a conflict is easier with a greater speed or a bigger turn rate, and the *Safest* strategy provides a better separation. A speed increase weighs more than a better turn rate, such that the value of $\theta.\|\overrightarrow{v_A}\|^2$ seems to give a good indication of the maneuverability of the UAV in this context, as pictured in figure 8.

The histograms on figure 9 show the distribution of scenarios with respect to the closest distance of approach during the simulation for both *Closest* and *Safest* strategies, with a UAV flying at 160 kn and maneuvering at 3 °/s. We observe a peak at 3 NM (i.e. the target separation distance) for the *Closest* strategy , which attests the correctness of the algorithm. Some conflicts remain as noticeable in the left part of the curve. The *Safest* strategy (red curve) helps solve more of those conflicts, although it yields huge separation distances, so that the UAV loses more time and fuel during the maneuver.

The number of airprox events is the primary indicator for the efficiency of the detect and avoid strategy. Yet, the deviation of the UAV from its trajectory and the number of maneuvers matter as well. Figure 10 shows the mean maneuver quantity (i.e. the sum of all heading changes given to the UAV) for our two strategies and for various values of the UAV maneuverability. The *Safest* strategy (in red) provides a maneuver quantity that increases with the turn rate, as the UAV is constantly adapting its heading, even for subtle changes in the constraint. With the *Closest* strategy however (in blue), the maneuver quantity slightly decreases when the turn rate increases. With a high maneuverability, the UAV is able to perform a more efficient maneuver, which solves the conflict sooner, and then needing fewer adjustments, whereas a less

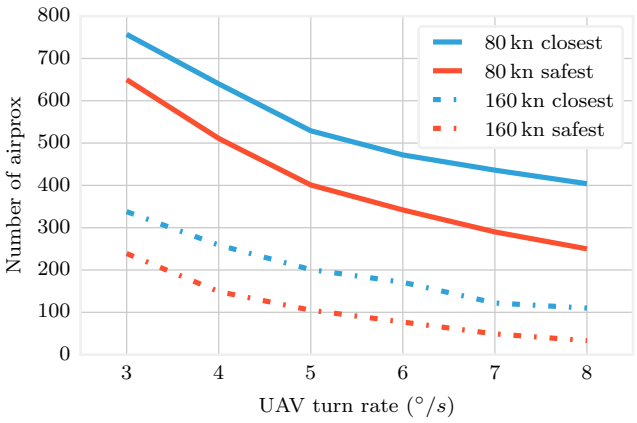

Fig. 7. Number of airprox events w.r.t. UAV turn rate.

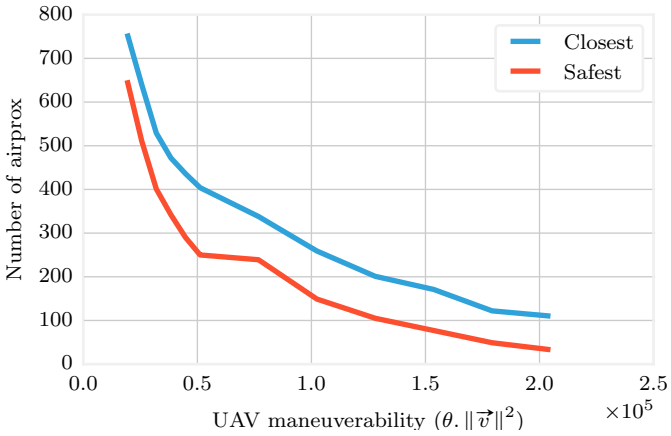

Fig. 8. Number of airprox events w.r.t. the maneuverability of the UAV.

[1]Each run represents a 6 minutes flight scenario and is executed within one tenth of a second on an Intel® Xeon® CPU with a clock rate of 3.4 GHz.

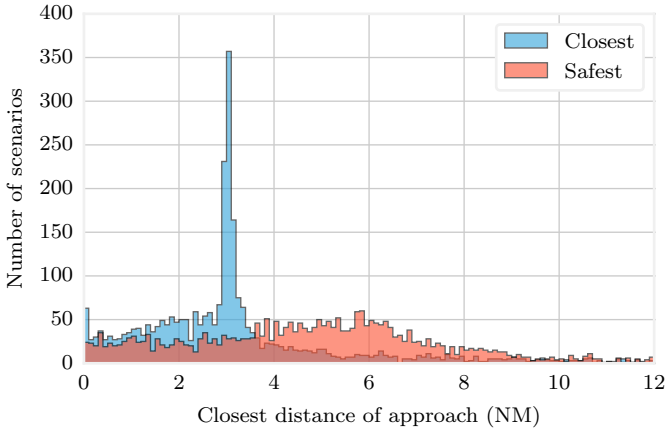

Fig. 9. Comparison of *Closest* (blue) and *Safest* (red) strategies: distribution of scenarios w.r.t. closest distance of approach.

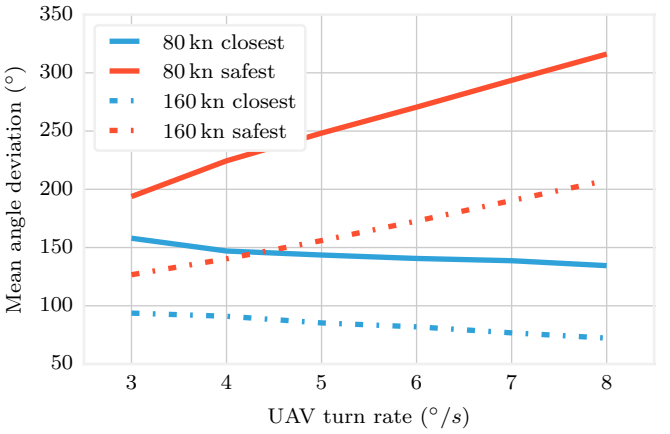

Fig. 10. Maneuver quantity w.r.t. UAV turn rate.

maneuverable UAV will have to adjust its maneuver more often, each correction being less effective.

2) Algorithm parameters: In order to validate our choices for parameters, we tried different values for the anticipation time τ and the target separation distance d. Figures 11 and 13 illustrate the influence of these parameters on the number of airprox events.

A small value for τ reduces drastically the performance of the detect and avoid algorithm, which can be explained by the fact that the UAV cannot see what is to happen after the next τ minutes and is thus unable to perform any efficient move once the constraining aircraft is unveiled. Figure 11 seems to indicate that τ has less influence beyond 3 minutes, but increasing its value from 3 to 5 minutes still enabled a drop of almost 200 airprox events for the less maneuverable UAV, as pictured in figure 12. We chose to stick to 5 minutes for our experiments.

The target separation distance also has a strong influence on the collision avoidance efficiency, as seen on figure 13. We observe that, no matter the target, the algorithm cannot ensure it will be strictly enforced, since we assume that the

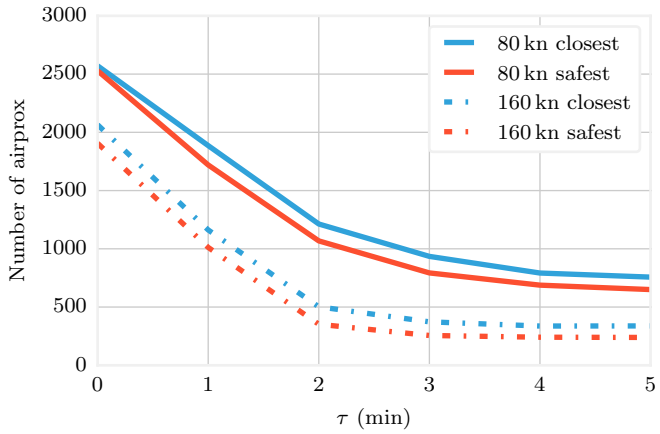

Fig. 11. Number of airprox events w.r.t. the anticipation time τ.

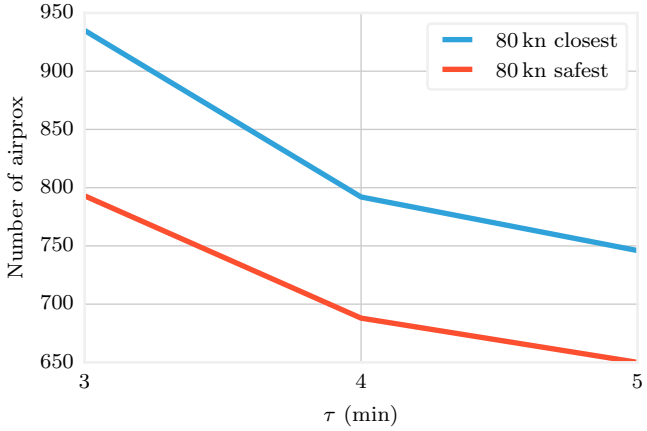

Fig. 12. Number of airprox events w.r.t. the anticipation time τ (zoom on the right part of figure 11).

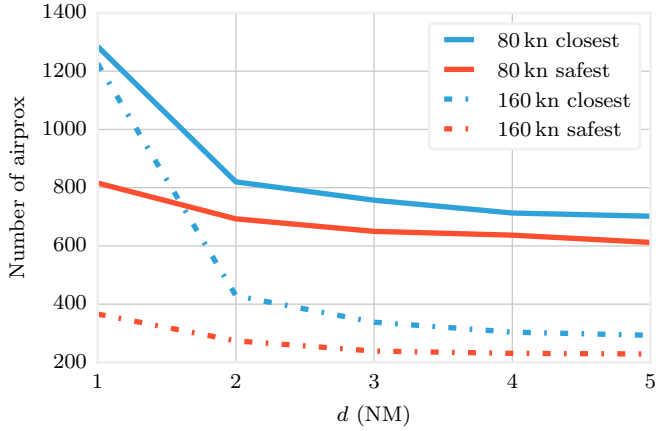

Fig. 13. Number of airprox events w.r.t. target separation distance d.

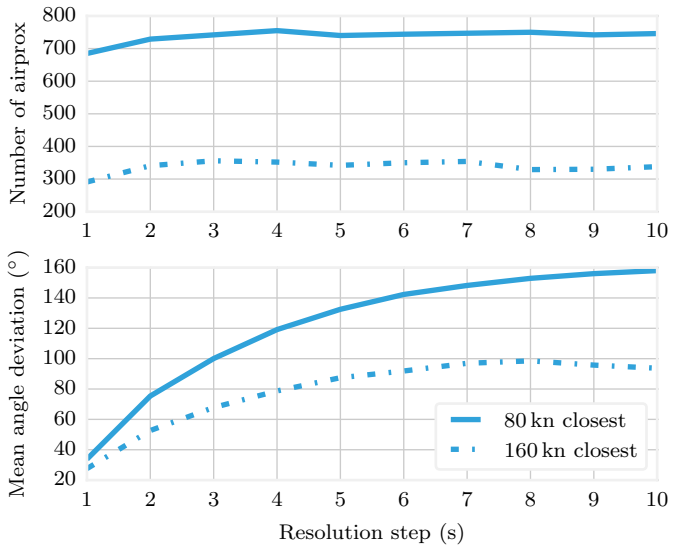

Fig. 14. Influence of the resolution step on the efficiency of collision avoidance.

UAV can turn immediately by $\theta.step$ (30° in most of our experiments), whereas in the simulation, it can only turn at the rate of $\theta°/\mathrm{s}$. This phenomenon leads to the drastic increase of the number of airprox events when we reduce the target distance to 1 NM. Increasing this target beyond 3 NM does not improve the resolution of airprox events.

The geometrical algorithm execution time is extremely short (in the order of 1 ms), therefore it is possible, even with an on-board CPU, to increase the frequency of the resolutions, in order to be more reactive. Figure 14 shows the influence of the time step between two successive resolutions on the behavior of the geometrical algorithm with the *Closest* strategy. The number of airprox events is not much impacted, although a slight improvement is noticeable for the lowest values. The maneuver quantity, however, can be drastically reduced with a higher resolution rate. This indicates that, in most cases, being more reactive enables to have more efficient maneuvers. Nevertheless, in the worst cases, this high reactivity produces an instability in the computed heading, and consequently a closest distance during the airprox.

Depending on the parameters, many airprox still remain despite the detect and avoid maneuvers. Most of these events occur when the computed heading leads to the (yet unknown) path of the aircraft. Then, when the aircraft finally turns towards the UAV, the speed difference makes it impossible for the UAV to escape, no matter how much it can turn. These scenarios will require particular attention for our future work, and might need the use of a vertical maneuver strategy, combined with the geometrical algorithm, to be taken care of.

IV. Conclusion and Future Work

In the context of the integration of UAVs in the lower airspace traffic, we have implemented a geometrical "detect and avoid" algorithm based on data collected from ADS-B-equipped surrounding aircraft. This algorithm attempts to solve air traffic conflicts between a UAV and airliners in initial climb and approach phase by changing the UAV heading only. The geometrical resolution is based upon the model described in [3], modified with additional constraints enabling the UAV to take the responsibility of the entire maneuver at constant speed.

Once the constraints induced by the aircraft computed, we have proposed two alternative strategies, one of them (*Closest*) leading to fewer deviations from the initial trajectory, the other one (*Safest*) providing more robust maneuvers. This method was intensively tested on realistic traffic issued from data recorded in the French lower airspace.

In most situations, the aircraft was avoided as planned, with a separation greater than the target. Nevertheless, in the most tricky circumstances where the aircraft turn out to change its heading towards the escape trajectory computed during the previous time steps, airprox events could not be avoided. Those situations will be further studied.

The analysis of results from over 200 000 simulations over 82 sets of parameters has established that a reasonable configuration of the algorithm would require a 3 NM target separation distance and 3 to 5 minutes of anticipation. A measure of the maneuverability of the UAV, defined as its turn rate multiplied by the square of its speed, also emerged from this analysis.

The *Safest* strategy is better than the *Closest* strategy for conflict avoidance, but at the cost of a significant increase in the quantity of maneuvers given to the UAV. It might be interesting to try and combine the two strategies, for example by switching from *Closest* to *Safest* under a given condition on the relative positions and speeds.

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, especially with the *Closest* strategy. In order to improve the robustness of the maneuvers, we plan to try and anticipate better both on the aircraft intentions and the UAV 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 UAV. 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 into a single aggregated maneuver.

For the hardest scenarios, we plan to check the interaction

of the current algorithm with the airliner's TCAS and, if necessary, to implement a vertical strategy for the UAV that could comply with airliner TCAS maneuvers.

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] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

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

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

[10] 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.

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

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

[13] 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.

[14] 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.

[15] 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.

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

[17] 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.

[18] 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.

[19] 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.

[20] 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.

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

[22] 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.

[23] 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.

[24] 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.

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

[26] 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.

Cyril Allignol is an assistant professor at École Nationale de l'Aviation Civile (ENAC). He graduated from ENAC as an engineer in 2006, and received a Ph.D. (2011) in computer science from the University of Toulouse. His main research topics include combinatorial optimization, metaheuristics and their application to Air Traffic Management problems.

Nicolas Barnier is a lecturer and research assistant at ENAC. He graduated from ENAC as an engineer in 1997, and received a Ph.D. in computer science in 2002 from the University of Toulouse. He is one of the authors of FaCiLe, an open source Constraint Programming library for the functional language OCaml. His research interests focus on Constraint Programming, Local Search and Combinatorial Optimization in general.

Éric Blond graduated from ENAC as an engineer in 1975. He is currently (since 1999) Head of Human-Machine Interaction applied research Division at the Centre d'Études de la Navigation Aérienne (now DSNA/DTI/EEI). He is the author of Rejeu, the traffic simulator used for this study.

Nicolas Durand graduated from the École Polytechnique de Paris in 1990 and the École Nationale de l'Aviation Civile (ENAC) in 1992. He has been a design engineer at the Centre d'Études de la Navigation Aérienne (now DSNA/DTI R&D) since 1992, holds a Ph.D. in Computer Science (1996) and got his HDR (french equivalent of tenure) in 2004. He is currently professor at the ENAC/MAIAA (Applied Maths, Computer Science, Automatic applied to Aviation) laboratory.