Journal of Open Aviation Science (2023), Vol.**1**
doi:10.59490/joas.2023.7222

JOAS

PROCEEDINGS | *The 11th OpenSky Symposium*

A Catalogue of Deconfliction Actions Extracted from Historical ADS-B Data

Kim Gaume,*[,1,2] Xavier Olive,[1] David Gianazza,[2] Richard Alligier,[2] and Nicolas Durand[2]

[1]ONERA DTIS, Université de Toulouse, France
[2]ENAC, Université de Toulouse, France

*Corresponding author: kim.gaume@onera.fr

(Received: 25 October 2023; Revised: 1 January 2024; Accepted: 17 January 2024; Published: 21 January 2024)
(Editor: Martin Strohmeier; Reviewers: Vishwanath Bulusu and Raúl Sáez)

Abstract

A conflict in Air Traffic Management is defined by a potential future risk of loss of separation. To solve a conflict, air traffic controllers take proactive actions to ensure safe separations between aircraft. They issue specific instructions to pilots for corrective measures, such as lateral manoeuvres, changes in altitude or speed adjustments. To alleviate the workload of controllers, the conflict detection and resolution process can be automated, resulting in recommendations for efficient manoeuvres. Traditional conflict resolution algorithms often neglect factors inherent to controllers' decision-making, leading to seemingly impractical manoeuvre suggestions from a human standpoint, causing reluctance in acceptation among controllers. The aim of our research is to obtain a catalogue of prevalent deconfliction practices, incorporating controllers' uncertainty models derived from actual flight data. In the current contribution, we focus on lateral deconfliction manoeuvres in en-route air traffic, and implement a simple heuristic method to extract a catalogue of resolved conflict situations from historical ADS-B data. This catalogue will provide insights into controllers' decision-making processes. In future works, we intend to use this catalogue to identify the best practices for traffic deconfliction, taking into account human factors and operational uncertainties, and to incorporate them into conflict resolution algorithms.

Keywords: Air Traffic Management; deconfliction; ADS-B

1. Introduction

Air traffic controllers (ATCOs) have two key responsibilities: ensuring safe operations by maintaining a safe separation between aircraft, and optimizing trajectories to align with time and cost constraints. In the upper airspace, flights must remain separated by at least 5 nautical miles laterally and 1000 feet vertically. In this context, a conflict is defined as an anticipated loss of separation between two flights. To detect and solve these conflicts, air traffic controllers estimate the future positions of all aircraft flying within their assigned airspace sector. Should these estimates bring two or more trajectories too close together, controllers intervene by delivering instructions to pilots for the execution of corrective actions, which may involve altering course, altitude, or speed. During the process of conflict resolution, controllers incorporate safety margins that serve to ensure a consistently comfortable level of separation in the presence of various uncertainties such as changing wind conditions, variations in pilots' reaction times, and potential communication misunderstandings.

Many algorithms have been proposed in the scientific literature to automatically solve air traffic conflicts. They aim to find conflict-free manoeuvres while minimizing other criteria such as fuel consumption or extra flying time induced by the manoeuvre. Most rely on mathematical models for the trajectories, their uncertainties, and the conflict detection. However, the air traffic controllers' decision-making isn't solely dictated by algorithmic considerations: it is also influenced by their training and the innate complexities of human perception. Traditional methods often fail to account for the cognitive processes and human factors influencing the controllers' decisions, resulting in unrealistic solutions. This lack of alignment with controllers' real-world practices raises a pertinent acceptability challenge. Solutions that significantly deviate from their established decision-making patterns are less likely to gain their approval. To earn acceptance from human operators, a conflict resolution tool must suggest solutions that are consistent with the approach of human controllers and that accommodate their internal uncertainty model.

As a first step towards this ultimate objective, we would like to extract a catalogue of deconflicted situations from historical ADS-B data coupled with flight plan information. This is a challenging task as ADS-B data consist solely of position sequences, lacking real-time controllers' radio instructions to pilots. In order to detect lateral manoeuvres issued by the controllers, we propose a heuristic method combining the detection of deviating trajectory segments and considerations on the lateral separation with nearby flights. In this paper, we focus on tactical deviations.

In Section 2, we detail a literature review to establish the background and context of our research. The methodology used to address the research questions is presented in Section 3. The results are presented in Section 4. A qualitative validation, through visual inspection, of a subset of our results is presented in Section 5. In Section 6, we discuss our method and results, and outline potential paths for improvement. Finally, we conclude our study in Section 7 with a summary of the main findings and their implications, and some perspectives for further research directions.

2. Literature Review and Previous work

Prior research has explored diverse methodologies for conflict detection, including trajectory prediction using probabilistic models [1, 2], confidence intervals [3], and specialized machine learning models for time-series data [4]. Additionally, some machine learning methods bypass trajectory prediction entirely. For instance in [5], conflicts and critical scenarios are simulated by modifying historical ADS-B data from the OpenSky Network, with machine learning employed to predict conflicts among the extracted trajectories.

Concerning conflict resolution, several methods have been proposed in the literature. For instance, in [6, 7], mixed-integer programming is applied to find optimum conflict resolutions, without considering trajectory prediction uncertainties. In [8], the conflict resolution problem is modelled as a minimum weight maximum clique problem and solved with a mixed-integer linear programming method, without taking uncertainties into account. Other approaches, such as Evolutionary Algorithms (AE) [9, 10] or Ant Colony optimization [11] do not guarantee the optimality of the solutions, but can compute good solutions within a limited time budget. They can also easily accommodate more realistic uncertainty models. In [12], A Constraint Programming approach is compared with an Evolutionary Algorithm, taking speed and lateral uncertainties into account, using pre-computed lateral manoeuvre combinations.

More recently, a specific attention has been given to human factors in conflict resolution. If we want to provide appropriate tools to assist controllers in their work, we first need a solid understanding of the way they detect and solve conflicts. Various aspects of controllers' behaviour have been studied. In [13], for example, the authors conducted experiments to study controllers' decision process and its implications in terms of workload calculation. More data-driven approaches take advantage of the current availability of large real-life flight databases by trying to extract manoeuvres and controllers' actions from historical data and characterize them. This is the case for [14] which uses unsupervised learning, or [15] with an approach based on the use of an auto-encoder architecture and reconstruction error.

As we now know that controllers are more likely to accept solutions that align with their own thought process [16, 17], integrating human perception in conflict detection and resolution models can help increase the acceptability of solutions. Several approaches now try to produce more realistic deconfliction suggestions by including real-life data produced by controllers using machine learning. For example, [18] uses a reinforcement learning method based on deconflictions performed by real ATCOs on simulated conflicts, whereas [19] uses imitation learning with the objective to learn from deconfliction manoeuvres in historical data.

Due to their training and perception, controllers have to build an internal uncertainty model to detect conflicts in incoming traffic, but also to maintain a safe separation during deconfliction while working under variable conditions. An attempt at explicitly extracting uncertainties is presented in [20]. This method is trained on a data set of conflict resolutions and takes the uncertainties as decision variables to try to reconstruct them. On an artificially generated data set with known uncertainty values, it manages to find the lateral and speed uncertainties embedded in the data. Later, this same method showed promising results on a small data set of real-time experiments, but has not yet been experimented on a large real-life conflict resolution data set, due to the unavailability of such data set at present.

In the current paper, we propose a heuristic method to extract a large data set of conflict resolutions from OpenSky ADS-B trajectories and their corresponding flight plans, focusing on the detection of lateral manoeuvres, as a first step. The proposed heuristic method is similar, in its principle, to the approach used in [21] to identify critical scenarios of risks. In [21], the authors use a "what-if" methodology examining what could have happened if the controller had not issued an instruction deviating the trajectory. In our study, we do not have the instructions issued by the controller in our data, but we can detect when a trajectory does not align with its flight plan, and observe what would have happened if it had followed its planned route after the beginning of the deviation. This allows us to determine if the lateral deviation was caused by a controller's instruction, or not.

3. Methodology

In this section, we provide an overview of our data sources and methodology to identify and extract conflict situations arising within a same horizontal plane, and that were solved through lateral track deviations only.

3.1 Data Collection and Preprocessing

The OpenSky Network [22] gives us access to a massive ADS-B database of real traffic. We start experimenting with this base by performing a first extraction on the AIRAC cycle 2207 (July 14 to August 10, 2022) in the Air Control Center of Bordeaux, France (LFBBDX).

We enhance this dataset by incorporating flight plans corresponding to each flight within the ADS-B database. This information was provided by the Air Navigation Services, along with information to the associated navaids. This augmentation resulted in a more complete database, encompassing not only trajectories but also their corresponding flight plan information, associated to a unique identifier. We used the Traffic data structure provided by the Python library *traffic* [23] to structure and manipulate this dataset. The resulting dataset was resampled to one data point per second to ensure uniformity and enable meaningful comparisons across flights. Additionally, we used filters to exclude data points with abnormal values related to altitude, track angle, vertical rate, and ground speed. Given our specific focus in en-route deconfliction, we retained only those trajectory segments that exceeded a predefined altitude threshold, set at 20,000 feet.

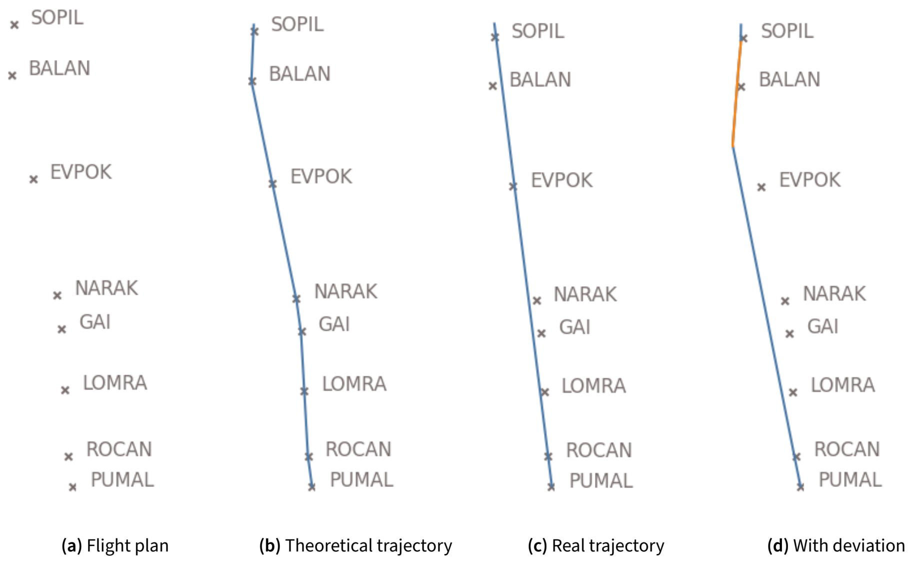
3.2 Deviations from the Flight Plan

One way to detect a deconfliction manoeuvre is by observing deviations from an aircraft's original trajectory. What we call *deviations* in this study is illustrated in Figure 1 with the example of the portion of flight plan `SOPIL BALAN EVPOK NARAK GAI LOMRA ROCAN PUMAL`.

In Figure 1a, we have the original flight plan, and in Figure 1b, we visualize the corresponding trajectory if the flight plan is precisely adhered to. It becomes apparent upon observing real trajectories in conjunction with their associated flight plans that it is not uncommon for aircraft to take shortcuts and bypass several navaids.

In Figure 1c, we present an actual trajectory linked to this flight plan, demonstrating instances where certain navaids are bypassed in favour of a more direct route. While this trajectory may not precisely follow the flight plan, it remains aligned with at least one of the navaids in the flight plan at any given time.

In Figure 1d, we encounter a different situation. The orange segment denotes a portion of the trajectory in which the aircraft's alignment does not correspond to any of the navaids specified in the flight plan. This orange trajectory section either aligns with a navaid that is not in its flight plan, or lacks alignment with any navaid whatsoever. For better precision, we only check alignment with navaids that are within a distance of 200 NM or less.



(a) Flight plan **(b)** Theoretical trajectory **(c)** Real trajectory **(d)** With deviation

Figure 1. Illustration of what is considered as a deviation in our paper

These last occurrences give an indication that something unexpected happened during the flight and led to a manoeuvre order, potentially related to a deconfliction. Our first step is to identify and extract them from our dataset. As we focus on lateral deconflictions only, we also need to eliminate instances that exhibit instability in altitude. We do this by checking that, during the entire time when the trajectory is deviated, its altitude remains stable within a threshold alt_{margin}. Finally, we need to take into consideration that some changes in directions can lead to deconflictions that are unrelated to an avoidance manoeuvre, especially when the alignment simply switches from one navaid to the next. As these turns are perfectly normal, we choose to filter them out using a simple threshold $duration_{min}$, under which we ignore the deviation. This process is summarized in Algorithm 1.

Algorithm 1 Find deviations

```
Require: Angle precision $prec_{angle}$, minimum duration threshold $duration_{min}$.
function FIND_DEVIATIONS($traffic, flightPlan$)
    $\mathcal{D} \leftarrow \{\}$
    for each $flight$ in $traffic$ do
        for each $dev$ in UNALIGNED_PORTIONS($flight, flightPlan$) do
            if $dev.duration > duration_{min}$ and isSTABLE($dev, alt_{margin}$) then
                $\mathcal{D} \leftarrow \mathcal{D} \cup \{(dev, closest)\}$
            end if
        end for
    end for
    return $\mathcal{D}$
end function
```

3.3 Neighbouring Flights Potentially Involved in a Conflict

Once these deviations are clearly identified, we examine them in relation to the surrounding air traffic. For each deviated aircraft, we try to identify which other aircraft caused the deviation, if any.

Given an aircraft A for which a lateral deviation starting at time t_s has been detected, we define a time horizon τ and consider the trajectories of A and the surrounding flights in the time interval $[t_s, t_s + \tau]$. The default value for τ is 20 minutes. If the trajectory ends or leaves Bordeaux ACC airspace before these 20 minutes, the horizon corresponds to the timestamp at the last available position. Additionally, if we detect a significant change in altitude after the conclusion of the deviated portion[1], we only consider the stable trajectory segment and adjust the horizon to match the timestamp of the last position within this segment.

To be considered as potentially problematic to flight A, the trajectory of another aircraft B must intersect with the horizontal plane where the deviation of A occurs, and fly at a close enough distance within a relevant time frame. The aircraft B might not fly at a constant level over the considered time period. Only the trajectory portions within a given altitude interval around the altitude of aircraft A are kept. This means that some trajectories in the set $\mathcal{N}$ of nearby trajectories may not be continuous. In the current study, the chosen altitude interval is $+/-50$ feet around the altitude of aircraft A. We are aware that this value is too restrictive to actually detect all conflicts with climbing/descending aircraft, that should be handled as a specific case. This is left for further work.

Algorithm 2 Choice of relevant neighbours.

```
function SELECT_NEIGHBOURS($dev, traffic$)
    $\mathcal{N} \leftarrow traffic \backslash \{flight\}$
    $\mathcal{N} \leftarrow traffic.between\_timestamps\,(dev.start, min\,(dev.stop, horizon\tau))$
    $\mathcal{N} \leftarrow traffic.between\_altitudes(dev.altmin - margin_{alt}, dev.altmax + margin_{alt})$
    return $\mathcal{N}$
end function
```

A conflict is defined as an anticipated loss of separation between predicted trajectories. Consequently, we make the assumption that a flown trajectory that is deviated due to a resolution remains close to the conflicting trajectory, while respecting the minimum separation of 5 nautical miles. If a

[1] The lateral deviation takes place at a same flight level, but the aircraft might climb or descent after the end of the deviation and before time $t_s + \tau$

deconfliction manoeuvre has occurred, we might identify the other involved aircraft by examining the lateral separations with the flights in set $\mathcal{N}$. For each deviated flight, we study the neighbouring aircraft with the smallest lateral separation.

3.4 Extraction of Deconflicted Traffic Situations

Estimating the next positions in the trajectory without taking the deviation into account helps us evaluate the possibility that a conflict actually occurred and has been solved by the controller. In other words, a loss of separation happening between the predicted trajectory of a deviated aircraft A and its closest neighbour B may be indicative of the fact that the deviation of A is actually a lateral manoeuvre initiated to avoid a conflict with B.

We can predict the deviation-free trajectory using simple methods: a straight-line trajectory can be relevant in that it gives us an idea of what would have happened if the aircraft had continued on its current course. This technique might be interesting when considering short-term trajectory predictions, but it loses its relevance when considering large prediction time horizons and when the planned route beyond the start of the deviation is not straight.

In this paper, we use a prediction following the flight plan. As we know that, before deviating, the aircraft was aligned with one of the navaids in its flight plan, we consider that the trajectory keeps its alignment with this navaid, and then follows exactly the other navaids in its flight plan. The prediction is based on a constant altitude, specifically the aircraft's altitude just before deviating. Speed is an unchanging average derived from the 20 minutes leading up to the deviation. The prediction assumes instantaneous turns and does not consider physical constraints during this process.

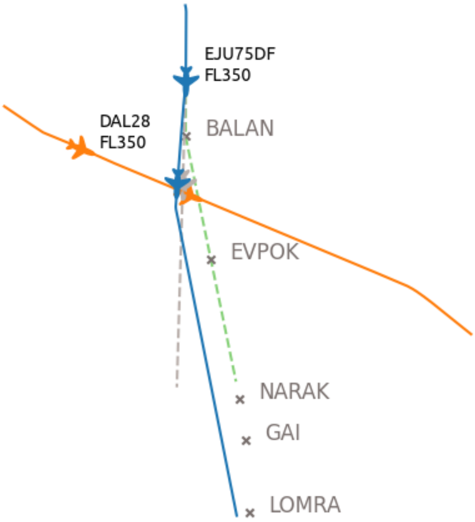


Figure 2. The blue trajectory is deviated. Predictions are represented with dashed lines, the straight-line prediction in grey and the one following the flight plan in green.

Figure 2 shows an example of a detection of a solved conflict. The flight in blue is the deviated one. The solid blue line is the flown trajectory. The dashed grey line is the predicted trajectory when following the current course. The dashed green line is the predicted trajectory when following the planned route. The trajectory in orange is the one closest to the deviated flight. In this example, the closest separation between the straight line prediction and the closest neighbour's trajectory is 5.9 nautical miles (NM). The closest separation when using the planned trajectory is 3.0 NM. As previously mentioned, trajectories routinely skip one or several navaids in their trajectory, therefore using the planned trajectory is not infallible, but it solves part of the bias due to a straight-line prediction.

Let us denote *Min_real_sep* the closest separation between the flown trajectories of the deviated flight *A* and its closest neighbour *B*, and *Min_pred_sep* the closest separation between the planned trajectory of *A* and the flown trajectory of *B*. Considering a required minimum separation threshold *Sep*, we shall label a traffic situation as a solved conflict if and only if:

$$Min\_pred\_sep < Min\_real\_sep \quad \& \quad Min\_pred\_sep \leq Sep$$

With ideal trajectory predictions, one could set the required separation parameter *Sep* at the standard separation $Sep_{std} = 5\,NM$. The real example shown in Figure 2 is ideal in the sense that the predicted separation using the planned route is below the standard separation threshold, whereas the actual separation between the flown trajectories is above Sep_{std}.

For many other real situations, however, and given the uncertainties in the predictions made by the controller and the additional margin that she or he may add to the standard separation, we might want to choose a required minimum separation greater than 5 nautical miles in our heuristic. This rises the question of which value to choose for this threshold, which will be investigated in the next section.

4. Results

4.1 Lateral Deviations

When extracting the flights for which a lateral deviation was observed from the ADS-B dataset (78,316 flights), we obtain 4.950 different situations. A summary of these results is given in Table 1.

Table 1. First application of the heuristic

	Min_real_sep [NM]	*Min_pred_sep* [NM]	*Duration* [s]	*Time_to_CPA* [s]	*Difference* [NM]
Mean	14.83	13.51	417.0	423.4	1.320
Standard deviation	16.04	17.43	233.8	313.2	5.814
Minimum	0.276	0.009	121.0	1.000	-156.3
Q25	7.732	5.129	231.0	174.0	-0.048
Q50	9.495	7.988	367.0	372.0	1.169
Q75	13.92	14.43	552.0	618.0	3.645
Maximum	207.9	208.2	2043	1198	35.05

In this table:

- *Min_real_sep* is the minimum lateral separation between the aircraft executing the deviation and their respective closest neighbour.
- *Min_pred_sep* is the estimation of the smallest lateral separation had the deviation manoeuvre not been initiated, providing a baseline for comparison. This estimation is based on a trajectory prediction that considers that, without a deviation, the trajectory would have followed its original flight plan perfectly.
- *Duration* is the total length of the deviation.

- *Time_to_CPA* is the time between the loss of alignment and the Closest Point of Approach (CPA) in the real data.
- *Difference* is the difference between the real minimum separation and the predicted minimum separation *Min_real_sep* − *Min_pred_sep*.

We observe that the minimum predicted separation can reach very high values, with a maximum of 208.2 NM. It is clear that such high separation values do not correspond to a conflict resolution. Relying only on the detection of lateral deviation to extract deconflicted situations is not enough.

The distributions of the actual (*Min_real_sep*) and predicted (*Min_pred_sep*) separations are shown in Figure 3, with separation limited to 40 NM on the x-axis of the plot to improve legibility.

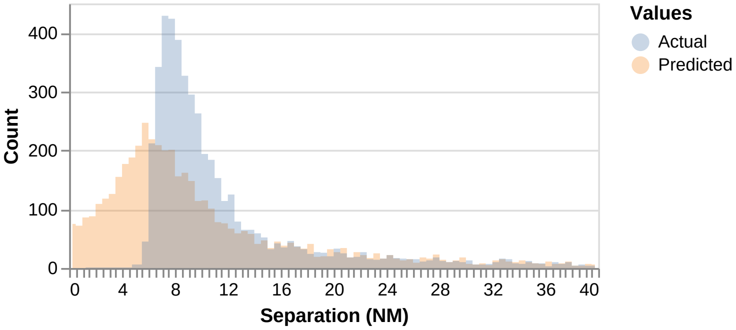


Figure 3. Distribution of actual and predicted lateral separation

As can be expected, the vast majority of the flown trajectories are separated by more than 5 NM, whereas the distribution of the predicted separation is more symmetrical, with a lot of values below 5 NM.

The above results show that we need to filter out instances in which the predicted closest separation exceeds an acceptable threshold, and that the difference between the actual and predicted separation might be a quantity of interest.

4.2 Lateral Deviations with Filtering Threshold $Sep = 15$ NM

As a first guess, let us choose an arbitrary threshold of 15 NM to filter out the lateral deviations that are probably not caused by a conflict resolution. Besides, we expect that a deconfliction manoeuvre would increase the minimum separation between the conflicting trajectories. We account for this factor by adding a criterion for *difference* > 0.

These filtering criteria (*Min_pred_sep* < 15 NM and positive difference) correspond to the heuristic described in Section 3, with a separation threshold $Sep = 15$ NM.

The 3,038 results of this extraction are summarized in Table 2.

With smaller predicted minimum separations, our results are more likely to be related to deconfliction manoeuvres, and we can see that, on average, the deviation seems to significantly increase separation. However, the threshold of 15 NM is arbitrary and there is no guarantee that it effectively allows us to filter out situations unrelated to deconflictions.

4.3 Extracting a Filtering Threshold from the Data

The core idea explored in the following is that, from a statistical point of view, lateral deviations that are not the result of a conflict resolution are just as likely to reduce the separation with the

Table 2. Results with Min_pred_sep < 15 NM and difference > 0

	Min_real_sep [NM]	*Min_pred_sep* [NM]	*Duration* [s]	*Time_to_CPA* [s]	*Difference* [NM]
Mean	9.161	5.917	447.4	460.1	3.244
Standard Deviation	2.553	3.099	225.4	281.5	2.558
Minimum	3.929	0.009	121.0	1.000	0.000
Q25	7.368	3.781	270.2	241.0	1.175
Q50	8.550	5.735	414.5	428.0	2.762
Q75	10.27	7.703	586.7	630.7	4.849
Maximum	32.29	14.97	2043	1198	21.27

closest aircraft as to increase it. On the opposite, lateral deviations that are resulting from a conflict resolution are expected to increase this separation.

In statistical terms, we may expect the median of the difference between the actual and predicted separation to be around zero for lateral deviations unrelated to deconflictions, whereas it should be strictly positive for lateral deviations caused by a conflict resolution.

Figure 4 shows a scatter plot of the difference between the actual and predicted minimum separation *Min_real_sep*−*Min_pred_sep* as a function of the predicted closest separation *Min_pred_sep*. The blue dots represent the raw data. The black line is obtained by applying a Median k-Nearest Neighbours (KNN) Regression technique with 100 neighbours to the raw data. In other words, it represents the median of the difference as a function of the predicted closest separation. The vertical red-dashed line represent a threshold value of 8 NM for *Min_pred_sep*.

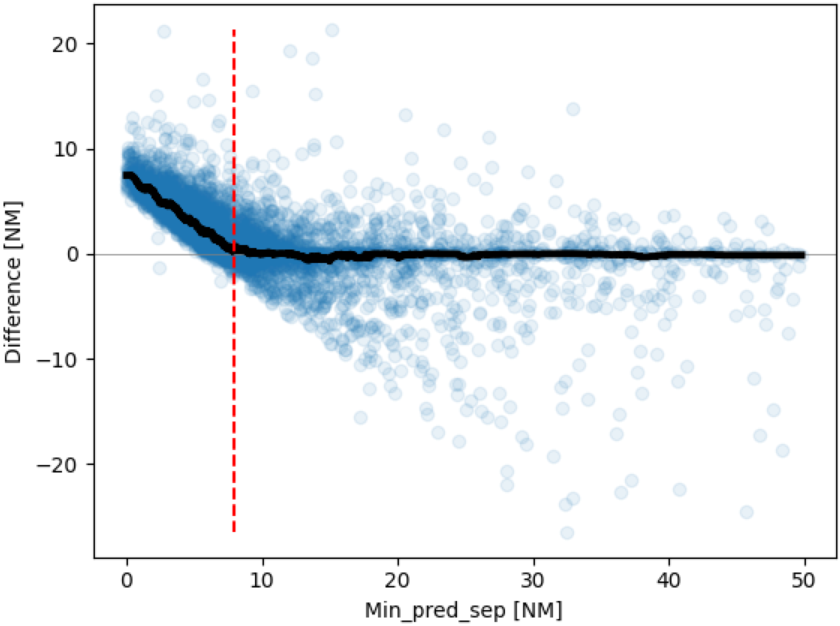


Figure 4. Difference between actual and predicted minimum separation as a function of the predicted closest separation.

We observe that, for predicted separation values below 8 NM, the median value of the difference is strictly positive and seems to follow a linear curve of equation $y = 8\,NM - x$. Beyond this threshold of 8 NM, the median is more or less constant, with value 0.

From these observations, we can conclude that lateral deviations with positive difference and *Min_pred_sep* < 8 NM are more likely to be related to a deconfliction than those beyond this threshold.

4.4 Lateral Deviations with Filtering Threshold *Sep* = 8 NM

Table 3 details the results of filtering out the lateral deviations with positive difference between the actual and predicted separation and for which the predicted minimum separation is beyond 8 NM. This extraction yields a total of 2,352 occurrences, corresponding to 3.00% of the flights in the initial dataset.

Table 3. Results with Min_pred_sep < 8 NM and difference > 0

	Min_real_sep [NM]	*Min_pred_sep* [NM]	*Duration* [s]	*Time_to_CPA* [s]	*Difference* [NM]
Mean	8.357	4.641	477.5	495.1	3.716
Standard Deviation	1.798	2.058	221.3	261.4	2.492
Minimum	3.929	0.009	121.0	1.000	0.000
Q25	7.140	3.133	310.0	296.0	1.703
Q50	7.982	4.958	454.0	472.0	3.414
Q75	9.208	6.276	613.0	653.0	5.338
Maximum	23.98	7.993	2043	1198	21.27

We notice that the minimum value for Min_real_sep is 3.93 NM, which violates the standard separation constraint. Upon further inspection, this violation concerns only one occurrence and is associated with erroneous values in the dataset. Though we do apply a filter to our data, abnormalities can remain and add noise to our results, which we will address in future developments. Excluding this specific instance, the lowest recorded value for Min_real_sep is 5.39 NM.

5. Validation

As a first attempt to validate our heuristic, we visually checked the results obtained from a subset of our dataset, specifically data from July 14, 2022 and extract a total of 100 results, which represents 3.61% of the 2,763 flights in the catalogue of deconflicted situations.

Figure 5 shows two examples extracted from this subset, that actually seem to correspond to deconfliction manoeuvres.

Upon a thorough examination of the results, we have observed that the majority of trajectories align with our expectations regarding the situations we intended to extract. However, six trajectories exhibit an unexpected shape, and these correspond to three pairs, each associated with a distinct situation. In each of these pairs, the trajectory contains two deviations, seemingly in an effort to avoid a conflict with the same aircraft on both occasions. In these specific situations, we have the option to either consider that both deviations were genuinely employed to address the same conflict, or focus solely on the most recent one as being relevant. These exceptional cases will be considered in future developments of this methodology.

6. Discussion and Paths for Improvement

In Section 4.3, we have shown that a good value for the filtering threshold *Sep* might be 8 NM. Note however that by selecting this value, we might filter out a number of actual deconflictions for

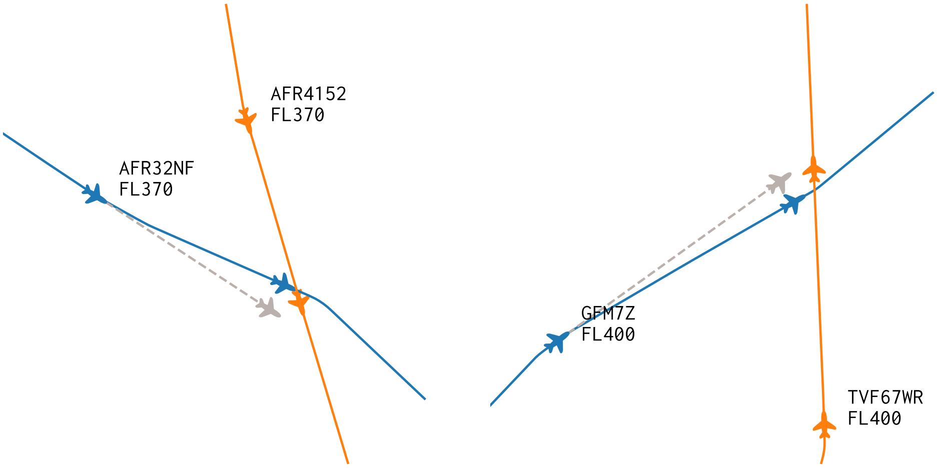


Figure 5. Examples of two detected deconfliction situations, with the deviated trajectory in blue, the closest neighbouring trajectory in orange and the predicted trajectory in grey.

which the controller used coarse trajectory predictions with large uncertainty values. If we intend to use the resulting catalogue of deconfliction actions to learn an uncertainty model for the trajectory prediction, filtering out these deconflictions might bias the resulting model towards low uncertainty values. For such an application, it might be interesting to set *Sep* at a greater value than 8 NM, even though we might include situations that are not actual deconflictions (false positives).

On the opposite, if the intended application of the catalogue requires more certainty that the extracted situations are actually solved conflicts, we might want to choose the *Sep* threshold between 5 and 8 NM, at the expense of a higher rate of false negatives (*i.e.* actual deconflictions that we exclude from the catalogue).

The filtering threshold *Sep* can be adjusted depending on the use we want to make of the catalogue. Though reducing the threshold may help filter out some of these instances, it would also strip our dataset from some of its diversity. On the other hand, increasing the threshold will include more deconflictions, but would also increase the risk to include false positives.

While our methodology has shown promise in identifying deconfliction situations, there are opportunities for improvement. Our parameters can be adjusted to filter results more precisely, and the heuristic method could be extended to handle unexpected situations, such as successive deviations in the same trajectory. Additional variables could be extracted from our data, such as geometrical properties of the deviations and their position inside control sectors, or the time to the closest point of approach.

For this study, we conducted a manual validation of a sample consisting of 100 results to assess the performance of our method, but this validation was primarily qualitative and not formal. To ensure the reliability of the extracted conflict resolution situations and the robustness of our methodology, further formal validation is required. A more extensive and systematic validation process involving operational expertise and a larger dataset is necessary to provide statistical significance to our findings and to validate the method's effectiveness on a broader scale. For instance, we could use a labelled dataset like [24] to verify that the count of deviations we identify aligns with the count of heading change instructions generated for an equivalent number of flights. This comparison would constitute an interesting partial validation.

We might also apply unsupervised machine learning algorithms, such as clustering, to our result dataset. This approach could serve a dual purpose in our research. First, it could assist us in the identification of distinct aircraft manoeuvre patterns, potentially revealing different types of conflict resolution strategies adapted to different situations. Secondly, it has the potential to uncover corner cases that can help us further refine our methodology.

7. Conclusion and Perspectives

In this study, we have presented a heuristic method to extract conflict resolution situations from historical ADS-B data. We focused exclusively on lateral deconflictions where aircraft undergo trajectory deviations to resolve conflicts.

Our method, as presented in Section 3, can be summarized as follows. For each aircraft that is not aligned with any navaid of its planned route for a significant amount of time, we find the closest neighbouring flight, considering the separation between flown trajectories. We then compute a trajectory prediction for the deviated flight, following the planned route after the beginning of the deviation. If this trajectory prediction comes closer than a chosen threshold value from the closest neighbour, and if the actual separation between flown trajectories is greater than the predicted separation, then we consider the deviation as resulting from a deconfliction action.

Using a Median K-Nearest Neighbours (KNN) Regression method on our data, we showed that a threshold value of 8 nautical miles seems adequate for filtering out lateral deviations unrelated to deconfliction actions (see Section 4.3). We discussed the relevance of this value in Section 6, depending on the intended application of the resulting catalogue of deconflicted traffic situations.

With a separation threshold of 8 NM, we were able to extract $2,352$ lateral deconflictions from an initial dataset of $78,316$ ADS-B trajectories in the upper airspace of Bordeaux ACC (see Section 4). A preliminary visual exploration (in Section 5) of a small subset of 100 traffic situations of this catalogue showed that all the lateral deviations in this subset seemed to be related to a deconfliction. Among these, three pairs of flights seemed to have undergone several deviations to avoid a same conflict.

Once refined and validated, we believe that the methodology presented in this study can serve as a foundation for future research in the field of air traffic management and provide a valuable resource for developing and testing advanced conflict resolution algorithms that align with human decision patterns and improve airspace safety.

In future work, we intend to extract valuable information from our catalogue of deconflicted situations on the controllers' operational patterns, habits, and uncertainties, ultimately leading to the creation of improved assistance tools for air traffic management.

Author contributions

- K. Gaume: Conceptualization, Methodology, Software, Data curation, Investigation, Writing–Original draft
- X. Olive: Conceptualization, Methodology, Supervision, Sofware, Writing–Review & editing
- D. Gianazza: Conceptualization, Methodology, Supervision, Writing–Original draft, Conceptualization, Writing –Review & editing
- R. Alligier: Conceptualization, Methodology, Software, Supervision, Conceptualization, Writing –Review & editing

- N. Durand: Conceptualization, Writing –Review & editing

Open data statement

The ADS-B data used in this study is publicly available on the OpenSky Network website https://opensky-network.org/. Supplementary data include flight plans and navaids locations provided by DSNA, the French Air Navigation Service Provider. They are not publicly available, but are a common resource in the Air Traffic Management (ATM) research community. The catalogue of deconflicted air traffic situations resulting from our study is available to the public, on the following web page: https://github.com/Vibujor/deconflictions (*results* directory).

Reproducibility statement

Provided the supplementary data described in the previous section can be obtained from ANSP services, the work presented in this paper can be reproduced using a) ADS-B data available from the OpenSky Network b) The *python* traffic library available at https://traffic-viz.github.io/ c) the specific source code developed for this study, available at the following site: https://github.com/Vibujor/deconflictions.

References

[1] Russell A. Paielli and Heinz Erzberger. “Conflict Probability Estimation for Free Flight”. In: *Journal of Guidance, Control, and Dynamics* 20.3 (May 1997), pp. 588–596. ISSN: 0731-5090. DOI: 10.2514/2.4081.

[2] M. Prandini, J. Hu, J. Lygeros, and S. Sastry. “A probabilistic approach to aircraft conflict detection”. In: *IEEE Transactions on Intelligent Transportation Systems* 1.4 (Dec. 2000), pp. 199–220. ISSN: 1558-0016. DOI: 10.1109/6979.898224.

[3] Zhengmao Chen, Dongyue Guo, and Yi Lin. “A Deep Gaussian Process-Based Flight Trajectory Prediction Approach and Its Application on Conflict Detection”. In: *Algorithms* 13.11 (Nov. 2020), p. 293. ISSN: 1999-4893. DOI: 10.3390/a13110293.

[4] Cheng Cheng, Liang Guo, Tong Wu, Jinlong Sun, Guan Gui, Bamidele Adebisi, Haris Gacanin, and Hikmet Sari. “Machine-Learning-Aided Trajectory Prediction and Conflict Detection for Internet of Aerial Vehicles”. In: *IEEE Internet of Things Journal* 9.8 (Apr. 2022), pp. 5882–5894. ISSN: 2327-4662. DOI: 10.1109/JIOT.2021.3060904.

[5] Javier Alberto Pérez-Castán, Luis Pérez-Sanz, Lidia Serrano-Mira, Francisco Javier Saéz-Hernando, Irene Rodríguez Gauxachs, and Víctor Fernando Gómez-Comendador. “Design of an ATC Tool for Conflict Detection Based on Machine Learning Techniques”. en. In: *Aerospace* 9.2 (Feb. 2022), p. 67. ISSN: 2226-4310. DOI: 10.3390/aerospace9020067.

[6] Lucia Pallottino, Eric M Feron, and Antonio Bicchi. “Conflict resolution problems for air traffic management systems solved with mixed integer programming”. In: *IEEE transactions on intelligent transportation systems* 3.1 (2002), pp. 3–11.

[7] Sonia Cafieri and Nicolas Durand. “Aircraft deconfliction with speed regulation: new models from mixed-integer optimization”. en. In: *Journal of Global Optimization* 58.4 (Apr. 2014), pp. 613–629. ISSN: 1573-2916. DOI: 10.1007/s10898-013-0070-1.

[8] Thibault Lehouillier, Jérémy Omer, François Soumis, and Guy Desaulniers. “Two decomposition algorithms for solving a minimum weight maximum clique model for the air conflict resolution problem”. In: *European Journal of Operational Research* 256.3 (2017), pp. 696–712.

[9] Nicolas Durand, Jean-Marc Alliot, and Joseph Noailles. “Automatic aircraft conflict resolution using Genetic Algorithms”. In: *SAC 1996, 11th Annual ACM Symposium on Applied Computing*. Philadelphia, United States, Feb. 1996, pp 289–298. DOI: 10.1145/331119.331195.

[10] Anoop Sathyan, Nicholas Ernest, Loic Lavigne, Franck Cazaurang, Manish Kumar, and Kelly Cohen. "A Genetic Fuzzy Logic Based Approach to Solving the Aircraft Conflict Resolution Problem". en. In: *AIAA Information Systems-AIAA Infotech @ Aerospace*. Grapevine, Texas: American Institute of Aeronautics and Astronautics, Jan. 2017. ISBN: 978-1-62410-449-7. DOI: 10.2514/6.2017-1751.

[11] Huaxian Liu, Feng Liu, Xuejun Zhang, Xiangmin Guan, Jun Chen, and Pascal Savinaud. "Aircraft conflict resolution method based on hybrid ant colony optimization and artificial potential field". In: *Science China Information Sciences* 61.12 (Dec. 2018), p. 129103. ISSN: 1674-733X, 1869-1919. DOI: 10.1007/s11432-017-9310-5.

[12] Cyril Allignol, Nicolas Barnier, Nicolas Durand, and Jean-Marc Alliot. "A new framework for solving en route conflicts". In: *Air traffic control quarterly: an international journal of engineering and operations* 21.3 (2013), pp. 233–253.

[13] Esa M. Rantanen and Ashley Nunes. "Hierarchical Conflict Detection in Air Traffic Control". In: *The International Journal of Aviation Psychology* 15.4 (Oct. 2005), pp. 339–362. ISSN: 1050-8414. DOI: 10.1207/s15327108ijap1504_3.

[14] Phuoc H. Dang, Phu N. Tran, Sameer Alam, and Vu N. Duong. "A machine learning-based framework for aircraft maneuver detection and classification". In: 2021. URL: https://dr.ntu.edu.sg/handle/10356/152776 (visited on 05/16/2023).

[15] Xavier Olive, Jeremy Grignard, Thomas Dubot, and Julie Saint-Lot. "Detecting Controllers' Actions in Past Mode S Data by Autoencoder-Based Anomaly Detection". In: *SID 2018, 8th SESAR Innovation Days*. Salzburg, Austria, Dec. 2018. URL: https://hal.archives-ouvertes.fr/hal-02338690 (visited on 11/15/2022).

[16] Brian Hilburn, Carl Westin, and Clark Borst. "Will Controllers Accept a Machine That Thinks like They Think? The Role of Strategic Conformance in Decision Aiding Automation". en. In: *Air Traffic Control Quarterly* (Aug. 2016). DOI: 10.2514/atcq.22.2.115.

[17] Adan Vela, William Singhose, Karen Feigh, John-Paul Clarke, and Eric Feron. "Evaluation of conflict-resolution policies on controller taskload". In: *Chinese Journal of Aeronautics* 34.5 (May 2021), pp. 315–330. ISSN: 1000-9361. DOI: 10.1016/j.cja.2020.12.006.

[18] Phu N. Tran, Duc-Thinh Pham, Sim Kuan Goh, Sameer Alam, and Vu Duong. "An Interactive Conflict Solver for Learning Air Traffic Conflict Resolutions". In: *Journal of Aerospace Information Systems* 17.6 (2020), pp. 271–277. DOI: 10.2514/1.I010807.

[19] Kwangyeon Kim, Raj Deshmukh, and Inseok Hwang. "Development of data-driven conflict resolution generator for en-route airspace". In: *Aerospace Science and Technology* 114 (July 2021), p. 106744. ISSN: 1270-9638. DOI: 10.1016/j.ast.2021.106744.

[20] Jean-Baptiste Gotteland, Sarah Degaugue, and Nicolas Durand. "Learning Uncertainty Parameters for Tactical Conflict Resolution". In: *Proc. of the 14th ATM Research and Development Seminar, 2021*. New Orleans, United States, Sept. 2021. URL: https://hal.science/hal-03484004.

[21] Karim Mehadhebi. "Identification of critical scenarios of risk: An operational approach". In: ATM 2015, 11th USA/Europe Air Traffic Management Research and Development Seminar. 2015.

[22] Matthias Schäfer, Martin Strohmeier, Vincent enders, Ivan Martinovic, and Matthias Wilhelm. "Bringing up OpenSky: A large-scale ADS-B sensor network for research". In: *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*. IEEE. 2014, pp. 83–94.

[23] Xavier Olive. "traffic, a Toolbox for Processing and Analysing Air Traffic Data". In: *Journal of Open Source Software* 4.39 (July 2019). DOI: 10.21105/joss.01518.

[24] Jens Nilsson and Jonas Unger. "Swedish civil air traffic control dataset". In: *Data in Brief* 48 (2023), p. 109240. ISSN: 2352-3409. DOI: https://doi.org/10.1016/j.dib.2023.109240.