Incorporating Optimisation in Strategic Conflict Resolution Service in U-space

Yiwen Tang, Yan Xu, Gokhan Inalhan
School of Aerospace, Transport and Manufacturing
Cranfield University
Bedford, United Kingdom

***Abstract*—This study presents an approach to incorporate optimisation in the strategic conflict resolution service in U-space. A conventional approach in line with the First-Come, First-Served (FCFS) rule is introduced, following the generation of two types of flight plans (i.e., *linear* and *area* operations) with uncertainty buffers further taken into account. This approach is based on iteratively checking the availability of the shared airspace volumes. Next, an optimisation model is formulated, using the same concept of common airspace representation, aiming at minimising the overall delay subject to operational constraints including a time-based separation minima. Some potential implementations are also envisioned for the optimisation model under plausible operational scenarios. Finally, simulation experiments are performed where four different case studies are designed, including FCFS and optimisation, as well as a hybrid use of both depending on the flight plans' submission time. Results suggest that, compared to FCFS, a notable delay reduction can be achieved with the optimisation approach incorporated, which is due to the FCFS prioritisation scheme that is often not efficient.**

***Keywords*—UTM; U-space; strategic conflict resolution; first-come, first-served (FCFS); optimisation.**

I. Introduction

The seamless integration of Unmanned Traffic Management (UTM) and Air Traffic Management (ATM) is critical to fully unlock the potential benefits of unmanned aircraft systems (UAS) applications and to contribute to the safety, efficiency, equity and reduced environmental impact of the aviation sector. The current approach for such integration relies on airspace segregation as a first step to ensure that UAS remain well clear of all other traffic. To integrate UAS operations alongside manned aircraft in non-segregated airspace, it is critical to enhance UAS safety levels to match manned aviation, which in particular builds upon the separation assurance and collision avoidance capabilities.

In ATM, the hierarchical conflict management consists of three layers: strategic conflict management (through airspace organisation and management, demand and capacity balancing, and traffic synchronisation); separation provision; and collision avoidance [1]. These continual layers work as a whole to eventually ensure the safety to a designated level. Similarly, there are also three respective layers of conflict management defined in U-space, including strategic (pre-tactical) de-confliction; tactical separation provision; and collision avoidance. Accordingly, a strategic conflict resolution service and a tactical conflict resolution service are proposed to realise at U2 and U3 stage in U-space respectively [2].

Concretely, strategic conflict resolution service is referred to as the initiatives aiming to reduce the need of tactical deconfliction and collision avoidance capabilities. Those initiatives usually occur at the pre-flight phase, which typically involve conflict detection and then resolution. It is linked with the operation planning processing service. As described in [3], this involves Operators sharing drone operation plans to relevant parties and reducing any potential loss of separation by planning routes that are unlikely to cause interactions with other airspace users. For instance, when an operation plan is shared, it is compared to other known plans (i.e., 4D trajectories/volumes) in order to exam if there is any potential conflict. If a conflict is identified then a tentative change of the plan is proposed, potentially followed by an iterative negotiation and resubmission process until there are no predicted conflicts.

Early efforts towards the U-space implementation have demonstrated several initial solutions for the strategic conflict resolution service. A consolidated report of relevant research activities can be found in [4]. For instance, DOMUS explained the central place of operation plan processing in strategic conflict detection. It also studied the interconnection of U-space service providers in such a process and provided an achieved reaction time of under 2 seconds [5]. EuroDrone developed a strategic deconfliction tool that maps UAS missions, and analyses and detects potential conflicts with other UAS missions, manned aircraft trajectories and non-flying area. The detected conflict will be subsequently resolved by means of proposing a departure shift within a pre-defined interval of time [6].

Similar initiatives exist in the FAA/NASA UTM programme, with an equivalent service named strategic deconfliction being developed. Initial demonstrations have applied a first-come, first-served (FCFS) approach, with operations required to resolve known conflicts prior to departure [7]. A set of relevant system-level requirements were presented in [8], addressing three core operating principles including prioritisation, negotiation facility and capability to allow for intersecting operations. Having a similar federated architecture applied, an Open-Access UTM framework has been demonstrated in the UK. The strategic deconfliction is facilitated through the discovery and synchronisation service and a local UTM network that enables inter USSP communications in a standardised manner [9].

In the past few years, an increasing body of research can be also noticed aiming to develop robust, advanced and scalable approaches to support the strategic conflict management. Simulation has been used to explore how a FCFS scheme (based

on when operators file their flight plans) performs in terms of fairness [10]. In combination with a collection of UTM services, simulation has been also performed to assess how well the requirements for strategic deconfliction developed by standards groups can support end-to-end safety [11]. An innovative lane-based approach was proposed and compared with more conventional volume-based strategic deconfliction, which uses one-way lanes, and roundabouts at lane intersections to allow a much more efficient analysis and guarantee of separation safety [12]. An airspace reservation method was studied in [13], where the pre-flight deconfliction problem is considered as a multi-agent path finding problem.

This paper proposes an optimisation model for strategic conflict resolution that is aimed to improve the performance of the existing FCFS approach. We introduce the flight planning of two commonly-seen UAS operation types, extended with specific uncertainty buffers. Leveraging on the common information of airspace volumes' spatial-temporal occupancy, potential conflicts of those flight plans are checked and resolved by the FCFS approach. An optimisation model is then presented to minimise the required delay for the concerned flights whilst resolving all the conflicts. Numerical experiments are conducted with four case studies in which FCFS and optimisation are both examined, along with their hybrid implementations for real-world applications.

II. First-Come, First-Served Approach

This section introduces a conventional strategic deconfliction approach that follows the First-Come, First-Served (FCFS) principle. We first briefly discuss how the flight plans are generated including their uncertainty buffers, which are the key input to strategic deconfliction. The FCFS algorithm is then presented, based on a common airspace representation that synchronises the flight plan processing.

A. Types of operation

As introduced in our previous work [14], two types of commonly-seen UAS operations, namely *area* and *linear* flights, are considered in this paper. Concretely, with the *area* type of operation, certain airspace volumes may be revisited by the UAS for multiple times, while with the *linear* type, the trajectory traverses each volume for only one time. Some intuitive examples can be found in Fig. 1.

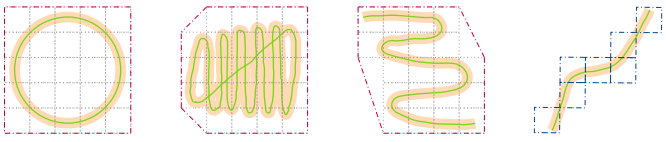


Figure 1. Schematic of *area* and *linear* UAS operations with uncertainty buffers, adapted from [3].

Following such characteristics, this paper further extends the previous definition of operations from 2D to 3D, adding altitude to the associated trajectory/volume. With regard to the *linear* operation, its trajectory is included with a top of climb and a top of descent, assuming vertical take-off and landing phases and a constant cruise phase. For the *area* operation, its operational area is set with an upper bound of altitude, being the ground level as the lower bound. Specifically, a delta time is computed and attached to each airspace volume traversed along the *linear* trajectory, whereas all the concerned volumes for an *area* operation have the same timestamps which involve only the planned taking off and landing times.

B. Airspace constraints

Geozone is a virtual three-dimensional perimeter around a geographic point [15], which is often used as a strategic measure to restrict access to drones for safety, security, privacy or environmental reasons. In addition to the geozones that are more of a long-term measure, there is another restriction closer to the short-term phase, namely the geo-fences that are defined by the (dynamic) geo-fencing service. They are geographic boundaries which should be respected during the UAS operations. The geo-fences' information will be provided through the geo-awareness service, and will have to be shared with the operators as early as possible. In this paper, both geozones and geo-fences are considered in a similar way as done in [14], with altitude bounds further added to restrict UAS from entering in a specific area at a certain altitude.

C. Flight plan generation

To generate the flight plan for this study, a basic 3D airspace model is first built. The airspace is decomposed into a set of airspace volumes of the same dimension. For each flight, a volume is connected to its adjacent in 10 directions (8 horizontal and 2 for up and down) except for those crossing the airspace constraints or on the boundary. The central point of each airspace volume is used when generating the flight path. Fig. 2 shows some instances of the generated *linear* flights (blue lines) and *area* flights (green areas), with also geozones (cylinder) and geo-fences (cuboid), as well as the volume central points (black dots).

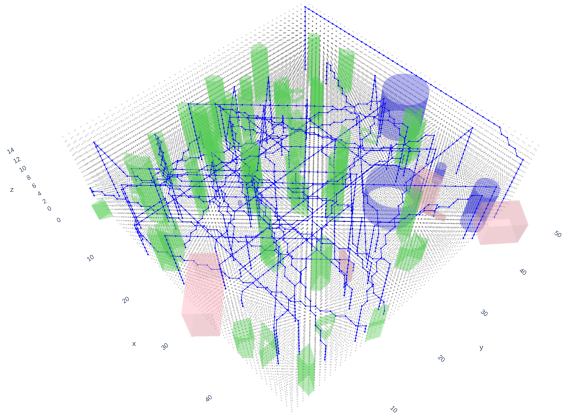


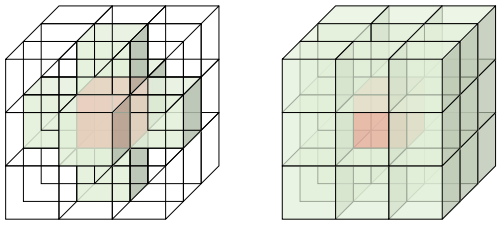
Figure 2. Generated UAS trajectories in the 3D airspace model, including *linear* and *area* flights avoiding airspace constraints.

For the *linear* flight, the classical A^* algorithm is applied to search for the shortest path (bypassing geozones and geo-fences). Specifically, it is assumed that after the UAS takes off, it vertically reaches to the target altitude and keeps cruising until arriving at the destination, followed by vertical landing eventually. On top of the generated path, timestamps are

attached from the take-off time at its first point and iterated over each trajectory segment at a given speed until the last point for landing time. Regarding the *area* flight, a set of vertices with altitude bounds are used to define the boundary of an operation and all the airspace volumes within that boundary are captured as part of its flight plan. Each edge is similar to the segment of a *linear* trajectory which is subject to the check of airspace constraints. Considering that some airspace volumes might be revisited during the operation, its take-off and landing times are set to be the same across all the concerned volumes which define the operation's duration.

D. Uncertainty buffers

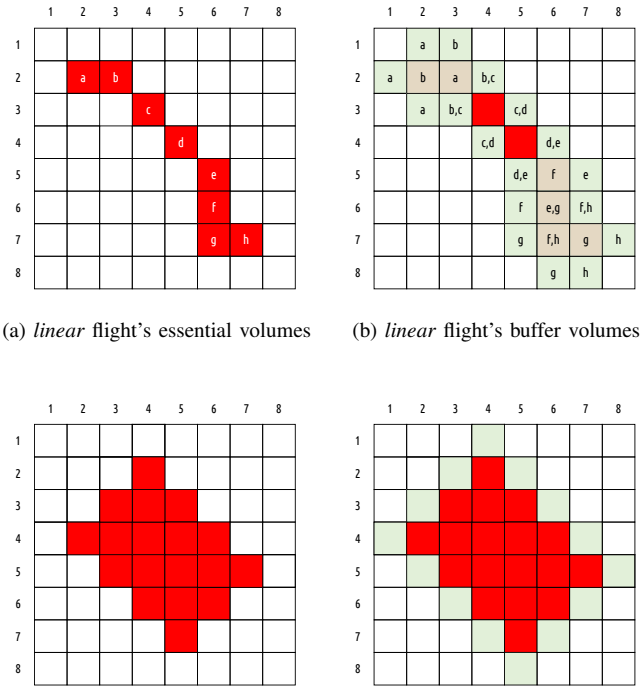
The current UAS techniques may involve significant operational uncertainties that could lead to undesirable consequences, e.g., the UAS experiencing a loss of datalink or a failure of flight controller, or on the ground the remote pilot committing a critical human error. As a result, once the nominal trajectory is generated, we further include some safety buffers as part of the flight plan (recall Fig. 1) to provide redundant protective space in response to potential uncertainties [16]. To this end, a group of buffer volumes are attached to each airspace volume that is planned to be traversed. There are two typical ways of identifying those buffers in open literatures, as shown in Fig. 3, with 6 most adjacent volumes in Fig. 3a and 26 all adjacent ones in Fig. 3b.



(a) 6 volumes (b) 26 volumes

Figure 3. Typical ways of identifying adjacent volumes as uncertainty buffers.

Next is to synchronise the times of the original essential volumes and their associated buffer volumes. A typical example of the essential volumes in 2D for *linear* and *area* flights can be seen in Fig. 4a and Fig. 4c respectively. A specific volume can be an essential volume and a buffer volume at the same time for a *linear* flight. The idea is to enforce that the timestamps of the buffer volumes remain the same as their corresponding essential volumes. Fig. 4c illustrates this situation using the buffer definition of Fig. 3a. For instance, at position (6,6), it is the essential volume f of the original trajectory. With buffers included, it is also the buffer volume corresponding to both essential volumes e and g, which means that three different timestamps are associated with this particular volume within the flight plan. The case for *area* flight is straightforward (see Fig. 4d), as all the volumes (both essential and buffer) are considered to have the same starting and ending times. Thus, the buffer volumes for *area* flights can be simply considered as a set of extended essential volumes.



(a) *linear* flight's essential volumes (b) *linear* flight's buffer volumes

(c) *area* flight's essential volumes (d) *area* flight's buffer volumes

Figure 4. Examples of relationship between essential and buffer volumes.

E. Pseudo algorithm

Given the above flight plans, we then use the FCFS approach to perform strategic conflict resolution. Recall that the flight plans are processed according to their request/submission order, rather than their scheduled take-off time, so this approach may be also referred to as First-Submitted/Requested, First-Served. In such a case, the flight plan submitted early will have a high priority to be processed. The pseudo code of this approach is presented in Algorithm 1.

The fundamental of this algorithm is to have a common information to represent the occupancy status of the airspace volumes. Once a group of volumes have been occupied by some early-submitted flight plans during the specific time periods (with separation minima included), they will be not available for the other flight plans submitted subsequently. Thus, those flights need to be delayed step by step, until all their concerned airspace volumes are repeatedly checked to be clear. Contingency events may occur with some pop-up flight demand of higher priority, which may require replanning the time slots following the updated order. However, this topic is beyond the discussion of this paper.

III. Optimisation Approach

This section introduces an optimisation model for strategic deconfliction that could be incorporated with the FCFS approach. The plausible operational scenarios are envisioned that show the potentials of implementing the proposed approach. The mathematical model is then formulated, with a special focus on the constraints of time-based separation assurance.

Algorithm 1 FCFS pseudo algorithm.

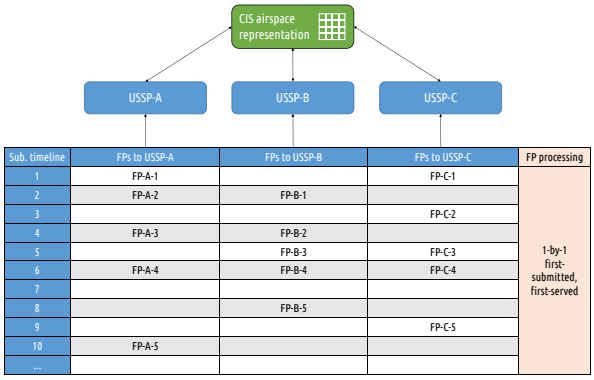
while $f \in F$ **do**
  if f is *linear* flight **then**
    for (ess, t), $(buff_{ess}, t) \in f_l$ **do**
      if (ess, t) or $(buff_{ess}, t)$ is blocked **then**
        Delay f_l for 1 time unit
        Break
      else
        Insert (ess, t) and $(buff_{ess}, t)$ to $list_l$
    for (ess, t) and $(buff_{ess}, t) \in list_l$ **do**
      Occupy (ess, t), $(ess, t \pm sep)$
      Occupy$(buff_{ess}, t), (buff_{ess}, t \pm sep)$
  if f is *area* flight **then**
    for $(j, [t_{ini}, t_{end}]) \in f_a$ **do**
      if $(j, [t_{ini}, t_{end}])$ is blocked **then**
        Delay f_a for 1 time unit
        Break
      else
        Insert $(j, [t_{ini}, t_{end}])$ to $list_a$
    for $(j, [t_{ini}, t_{end}]) \in list_a$ **do**
      Occupy $(j, [t_{ini}, t_{end}])$
      Occupy $(j, [t_{ini} \pm sep, t_{end} \pm sep)$

A. Operational scenarios

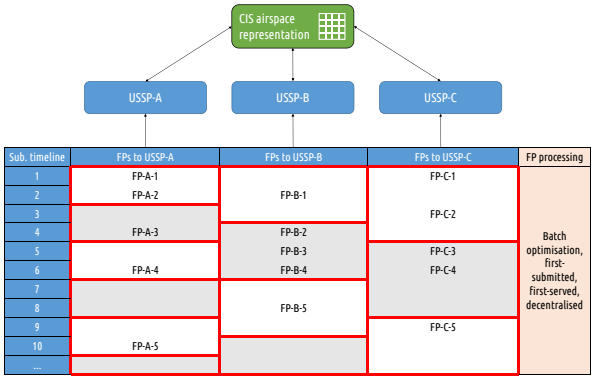
As part of the on-going research work within the AMU-LED project, this paper explores potential operational scenarios of the proposed approach. The detailed architecture, along with specific actors and responsibilities and data flow among them, is beyond the scope of this paper. The reader is directed to the AMU-LED ConOps [17] and the forthcoming deliverables for detailed operational models. In this paper, we will only exam it from the methodological wise, with a special focus on how the flight plans are processed through this service, as illustrated in Fig. 5.

Fig. 5a represents how the flight plan is processed one-by-one following the FCFS rule. Intuitively, each USSP "pulls" the information of airspace occupancy status from the Common Information Service (CIS) provider, and checks if any delay is needed for the particular flight. Once the (potentially delayed) flight plan does not incur any conflicts, it can be "pushed" back through CIS to update the common airspace representation. This can be done in parallel where as many USSPs as possible can be involved handling their respective flight plans. While this might be an over-simplified description, it could provide us some general insights into the process. We note that in real world more complex negotiation schemes and data exchange protocols are being developed and tested.

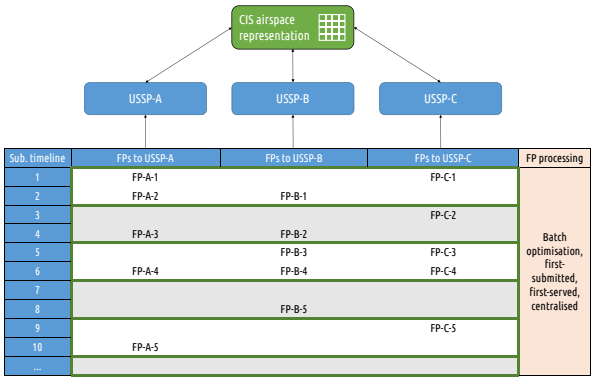
It is obvious that having the flight plans processed one-by-one would be usually less efficient than processing them as a whole, from the resource allocation point of view (where the airspace occupancy in time and space are the resources). As a result, this paper proposes an optimisation model that is aimed to deal with a batch of flight plans and thus produces the optimal delay allocation amongst those flights. The optimisation model is also based on the concept of common



(a) FCFS rule



(b) Decentralised batch optimisation



(c) Centralised optimisation

Figure 5. Potential operational scenarios for processing flight plans.

airspace representation, and thus is fully interoperable with the previous FCFS approach. In addition, there could be different ways to form such batches flexibly, for instance, by periodically handling the submitted flight plans every time when a resolution decision needs to be made, as shown with Figs. 5b and 5c. Specifically, the former takes a decentralised implementation in the same way as FCFS, whereas the latter assumes centralisation where flight plans from various USSPs are processed all together.

B. Model formulation

We present the mathematical model for incorporating optimisation in strategic conflict resolution. The model is formulated with mixed integer linear programming, as follows:

1) Decision variables:

$$x_{l,t}^{j} = \begin{cases} 1, & \text{if } \textit{linear} \text{ flight } l \text{ arrives at volume } j \\ & \text{by time } t \\ 0, & \text{otherwise} \end{cases}$$

$$m_{a,t}^{j} = \begin{cases} 1, & \text{if } \textit{area} \text{ flight } a \text{ enters in volume } j \\ & \text{by time } t \\ 0, & \text{otherwise} \end{cases}$$

Note that if the entrance time for an *area* flight has been determined, the exit time will be known as the flight duration d_a is known given the submitted flight plan.

2) Objective function: The objective function is to minimise the overall delay for all *linear* and *area* flights.

$$\begin{aligned} \min \sum_{l \in L} \sum_{j \in J_l^{(1)}} \sum_{t \in T_l^{J_l^{(1)}}} (t - r_l^{J_l^{(1)}})(x_{l,t}^{j} - x_{l,t-1}^{j}) + \\ \sum_{a \in A} \sum_{j \in J_a^{(1)}} \sum_{t \in T_a^{J_a^{(1)}}} (t - r_a^{J_a^{(1)}})(m_{a,t}^{j} - m_{a,t-1}^{j}) \end{aligned} \tag{1}$$

where T_l^j, T_a^j are manually defined subsets of time moments feasible for delay assignment. The initially scheduled take-off times are depicted by $r_l^{J_l^{(1)}}, r_a^{J_a^{(1)}}$. In this paper, each flight is assumed to be of the same priority, but the model will allow flight prioritisation by specifying their weighted costs of delay.

3) Flight operations constraints: These constraints are associated with the operational limits with regard to each individual *linear* or *area* flight.

$$x_{l,\underline{T}_l^j - 1}^{j} = 0, \; x_{l,\bar{T}_l^j}^{j} = 1 \quad \forall l \in L, \forall j \in J_l \tag{2}$$

$$x_{l,t}^{j} - x_{l,t-1}^{j} \geqslant 0, \quad \forall l \in L, \forall j \in J_l, \forall t \in T_l^j \tag{3}$$

$$\begin{aligned} x_{l,t+\hat{t}_l^{jj'}}^{j'} - x_{l,t}^{j} = 0, \quad \forall l \in L, \forall t \in T_l^j, j = J_l^{(i)}, \\ j' = J_l^{(i+1)} : \forall i \in [1, n_l) \end{aligned} \tag{4}$$

Constraint 2 states that *linear* flight should arrive at volume j within the time window, whose upper and lower bound are depicted by $\bar{T}_l^j$ and $\underline{T}_l^j$ respectively. Constraint 3 enforces the continuity of timeline by specifying the relationship between the status at any time and its previous time. Constraint 4 stipulates that the controlled flight time between any segment (j, j') of a *linear* flight remain unchanged than initially scheduled.

$$m_{a,\underline{T}_a^j - 1}^{j} = 0, \; m_{a,\bar{T}_a^j}^{j} = 1, \quad \forall a \in A, \forall j \in J_a \tag{5}$$

$$m_{a,t}^{j} - m_{a,t-1}^{j} \geqslant 0, \quad \forall a \in A, \forall j \in J_a, \forall t \in T_a^j \tag{6}$$

$$\begin{aligned} m_{a,t}^{j'} - m_{a,t}^{j} = 0, \quad \forall a \in A, \forall t \in [\underline{T}_a^j, \bar{T}_a^j], j = J_a^{(i)}, \\ j' = J_a^{(i+1)} : \forall i \in [1, n_a) \end{aligned} \tag{7}$$

$$n_{a,t}^{j} = m_{a,t-d_a}^{j}, \quad \forall a \in A, \forall j \in J_a, \forall t \in [\underline{T}_a^j + d_a, \bar{T}_a^j + d_a] \tag{8}$$

$$m_{a,t}^{j} = 1, \quad \forall a \in A, \forall j \in J_a, \forall t \in (\bar{T}_a^j, \bar{T}_a^j + d_a] \tag{9}$$

$$n_{a,t}^{j} = 0, \quad \forall a \in A, \forall j \in J_a, \forall t \in [\underline{T}_a^j, \underline{T}_a^j - 1 + d_a] \tag{10}$$

$$z_{a,t}^{j} = m_{a,t}^{j} - n_{a,t}^{j}, \quad \forall a \in A, \forall j \in J_a, \forall t \in [\underline{T}_a^j, \bar{T}_a^j + d_a] \tag{11}$$

Similar to *linear* flight, Constraint 5 specifies the arrival time slot set T_a^j. Constraint 6 enforces the continuity of the timeline. Constraint 7 stipulates the controlled times to be the same at all volumes covered by that flight. Constraint 8 shows that the duration of any *area* flight does not change from the initially planned. Specifically, as the flight duration d_a is fixed, the lower and upper bound of feasible time moments at the flight exit volume can be expressed by $\underline{T}_a^j + d_a$ and $\bar{T}_a^j + d_a$, where $[\underline{T}_a^j, \bar{T}_a^j]$ are the feasible time moments defined at the arrival volume. Constraint 9 and 10 are linked with Constraint 11, where $z_{a,t}^j$ specifies the occupancy status of flight a at volume j at time t.

4) Separation minima constraints: The overall idea of this strategic deconfliction approach is that, to guarantee safe separation, a certain volume can be only occupied by one flight within a certain period of time.

$$\sum_{l \in L^j} c_{l,k}^{j} + \sum_{a \in A^j} c_{a,k}^{j} \leqslant 1 \tag{12}$$

The separation requirement is specified in Constraint 12, where $c_{l,k}^j$ and $c_{a,k}^j$ are the occupancy status of *linear* flight and *area* flight at volume j respectively. k is the sliding time window whose moving step is the unit time step while the look-ahead horizon equals to the separation minima.

$$c_{l,k}^{j} = \begin{cases} 1, & \sum_{t}^{t+s} x_{l,t}^{j} - x_{l,t-1}^{j} \geqslant 1 \\ 0, & \sum_{t}^{t+s} x_{l,t}^{j} - x_{l,t-1}^{j} < 1 \end{cases} \tag{13}$$

For *linear* flights, the occupancy status at a certain volume is determined by the flight's arrival status. As shown in Eq.13, at volume j, if more than or equal to one time slot is occupied by a flight within that time window (recall that there might be several timestamps associated with one volume for *linear* flights due to the buffer volumes), this volume will be regarded as being occupied, and then $c_{l,k}^j = 1$, otherwise $c_{l,k}^j = 0$.

Worth noting that Eq.13 is a piecewise function and should be converted into a linear function. Thus, a group of auxiliary variables and constraints are added, as shown below:

$$\alpha_{l,k}^{j} = \sum_{j \in J^l} \sum_{t \in [k, k+s) \cap [\underline{T}_l^j, \bar{T}_l^j]} x_{l,t}^{j} - x_{l,t-1}^{j}, \quad \forall l \in L, \forall j \in J_l \tag{14}$$

$$c_{l,k}^{j} = {\lambda_1}_{l,k}^{j} + {\lambda_2}_{l,k}^{j}, \quad \forall l \in L, \forall j \in J_l \tag{15}$$

$${\beta_1}_{l,k}^{j} + {\beta_2}_{l,k}^{j} = 1, \quad \forall l \in L, \forall j \in J_l \tag{16}$$

$$\alpha_{l,k}^{j} = {\lambda_1}_{l,k}^{j} + M{\lambda_2}_{l,k}^{j}, \quad \forall l \in L, \forall j \in J_l \tag{17}$$

$${\lambda_0}_{l,k}^{j} + {\lambda_1}_{l,k}^{j} + {\lambda_2}_{l,k}^{j} = 1, \quad \forall l \in L, \forall j \in J_l \tag{18}$$

$${\beta_1}_{l,k}^{j} \leqslant {\lambda_0}_{l,k}^{j} + {\lambda_1}_{l,k}^{j}, \quad \forall l \in L, \forall j \in J_l \tag{19}$$

$${\beta_2}_{l,k}^{j} \leqslant {\lambda_1}_{l,k}^{j} + {\lambda_2}_{l,k}^{j}, \quad \forall l \in L, \forall j \in J_l \tag{20}$$

where ${\lambda_0}_{l,k}^{j}, {\lambda_1}_{l,k}^{j}, {\lambda_2}_{l,k}^{j} \in \mathbb{R}^{+}$, ${\beta_1}_{l,k}^{j}, {\beta_2}_{l,k}^{j} \in \{0,1\}, c_{l,k}^{j} \in \{0,1\}, \alpha_{l,k}^{j} \in \mathbb{R}^{+}$. These variables are the auxiliary decision variables for *linear* flight. J_l represents the collection of volumes which includes essential volume and its related buffer volumes. With the help of Constraint 14 - 20, the relationship between $c_{l,k}^{j}$ and the number of *linear* flight arrives at volume j can be known via a linear formulation.

$$c_{a,k}^{j} = \begin{cases} 1, & \sum_{t}^{t+s} z_{a,j}^{t} \geqslant 1 \\ 0, & \sum_{t}^{t+s} z_{a,j}^{t} < 1 \end{cases} \tag{21}$$

The value of occupancy status $c_{a,k}^{j}$ for *area* flight is determined by $\sum_{t}^{t+s} z_{a,j}^{t}$. Similarly, as Eq.21 shows, when $\sum_{t}^{t+s} z_{a,j}^{t} \geqslant 1$, $c_{a,k}^{j} = 1$, otherwise, $c_{a,k}^{j} = 0$. It means that, if an *area* flight occupies one or more than one time slot at a volume within the time window, the volume should be regarded as being occupied. Similar to *linear* flight, the piecewise equation Eq.21 can be converted to linear functions in the same way:

$$\alpha_{a,k}^{j} = \sum_{t \in [k, k+s) \cap [\underline{T}_a^j, \bar{T}_a^j + d_a]} z_{a,t}^{j}, \quad \forall a \in A, \forall j \in J_a \tag{22}$$

$$c_{a,k}^{j} = {\lambda_1}_{a,k}^{j} + {\lambda_2}_{a,k}^{j}, \quad \forall a \in A, \forall j \in J_a \tag{23}$$

$${\beta_1}_{a,k}^{j} + {\beta_2}_{a,k}^{j} = 1, \quad \forall a \in A, \forall j \in J_a \tag{24}$$

$$\alpha_{a,k}^{j} = {\lambda_1}_{a,k}^{j} + s{\lambda_2}_{a,k}^{j}, \quad \forall a \in A, \forall j \in J_a \tag{25}$$

$${\lambda_0}_{a,k}^{j} + {\lambda_1}_{a,k}^{j} + {\lambda_2}_{a,k}^{j} = 1, \quad \forall a \in A, \forall j \in J_a \tag{26}$$

$${\beta_1}_{a,k}^{j} \leqslant {\lambda_0}_{a,k}^{j} + {\lambda_1}_{a,k}^{j}, \quad \forall a \in A, \forall j \in J_a \tag{27}$$

$${\beta_2}_{a,k}^{j} \leqslant {\lambda_1}_{a,k}^{j} + {\lambda_2}_{a,k}^{j}, \quad \forall a \in A, \forall j \in J_a \tag{28}$$

where ${\lambda_0}_{a,k}^{j}, {\lambda_1}_{a,k}^{j}, {\lambda_2}_{a,k}^{j} \in \mathbb{R}^{+}$, ${\beta_1}_{a,k}^{j}, {\beta_2}_{a,k}^{j} \in \{0,1\}, c_{a,k}^{j} \in \{0,1\}, \alpha_{a,k}^{j} \in \mathbb{R}^{+}$. These variables are the auxiliary decision variables. With the help of Constraint 22 - 28, the relationship between $c_{a,k}^{j}$ and $\sum_{t}^{t+s} z_{a,j}^{t}$ can be known via a linear approach.

5) Decision variable conditions:

$$x_{l,t}^{j} \in \{0,1\}, \forall l \in L, \forall j \in J_l, \forall t \in T_l^j \tag{29}$$

$$m_{a,t}^{j} \in \{0,1\}, \forall a \in A, \forall j \in J_a, \forall t \in T_a^j \tag{30}$$

$$n_{a,t}^{j} \in \{0,1\}, \forall a \in A, \forall j \in J_a, \forall t \in T_a^j \tag{31}$$

$$z_{a,t}^{j} \in \{0,1\}, \forall a \in A, \forall j \in J_a, \forall t \in [\underline{T}_a^j, \bar{T}_a^j + d_a] \tag{32}$$

Finally, Constraints 29 - 32 state the binary constraints and domains of the primary decision variables used in the model. In addition, $n_{a,t}^{j}$ and $z_{a,t}^{j}$ are auxiliary variables associated with *area* flights.

IV. Experiments and Discussions

This section presents the numerical experiments with an illustrative U-space scenario. We look into an airspace that covers a 3D space of 2.5*2.5 km^2 with 150 m altitude, divided to 50*50*5 identical airspace volumes, leading to each volume of 50*50*30 m^3 size. Four different case studies have been conducted, applying both FCFS and optimisation approaches, as well as their combined variants.

A. Experiment setup

The traffic sample includes 200 flights (100 *linear* and 100 *area*) through the 24 hours' overall duration. There are 30 airspace constraints (15 geozones and 15 geo-fences) across this airspace. Trajectory buffers are considered for each *linear* flight in a way that each essential volume is associated with 6 surrounding buffer volumes. The generated flight plans and the airspace constraints are as shown in Fig. 6.

Besides the model assumptions mentioned in Sec. III-B, some additional assumptions have been made in the experiments: (1) the unit time step is set as 5 min; (2) the speed of *linear* operations are randomly set between 10-15 m/s; (3) the duration of *area* operations are randomly set between 5-30 minutes; and (4) a time-based separation minima is assumed to be 15 min at every airspace volume.

In the experiments, Python 3.8 has been used to develop the FCFS algorithm. For the optimisation approach, GAMS v.25.1 software suite has been used as the modelling tool and Gurobi v.7.5 optimiser as the solver. The experiments have been run on a 64 bit Intel® Core™ i5-9500 CPU @ 3.00GHz 6 Cores computer with 16 GB of RAM and Linux OS.

B. Case studies

The generated 200 flight plans are first sorted based on their submission order (which is defined arbitrarily). Depending on whether the FCFS/optimisation approach is selected to tackle a particular group of flights, four different case studies are performed to produce a set of comparable solutions:

- **Case-1**: full FCFS
- **Case-2**: full optimisation
- **Case-3**: half optimisation + half FCFS
- **Case-4**: batch optimisation

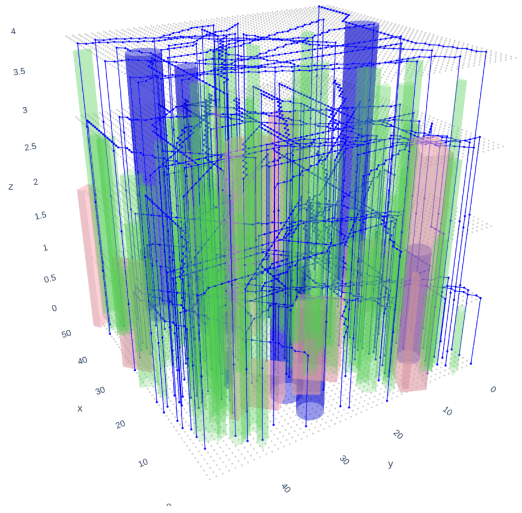


Figure 6. Illustrative example for the numerical experiments.

Concretely, flight plans are solely processed by FCFS and optimisation approach in Case-1 and Case-2 respectively. They are executed for only once with an assumption that all the flight plans are available at the execution time. We note that this may not be the case in real world, where some on-demand flight plans may not be available at an early stage. These two cases are set to evaluate the effectiveness of the two approaches, which can be also used to reveal the upper and lower bounds across all the case studies.

Case-3 concatenates two approaches by applying optimisation to process the first half of the flight plans, and then using FCFS for the rest taking into account the previously occupied airspace volumes. The rationale is that some operations may be scheduled well in advance, whereas others may be submitted shortly before taking off. In Case-4, the flight plans are divided into 5 sequential smaller batches which are then resolved one group by the other, using the optimisation approach.

C. Result comparisons

An overall comparison of the solutions derived from the above four case studies can be found in Table I, summarised in a few key indicators.

TABLE I. RESULT COMPARISONS ACROSS FOUR CASES OF THE STUDY.

KPI	Case-1	Case-2	Case-3	Case-4
Delay (min) - *linear*	890	550	650	830
Delayed flights (a/c) - *linear*	35	37	32	34
Max delay (min) - *linear*	80	55	80	80
Delay (min) - *area*	325	185	350	275
Delayed flights (a/c) - *area*	11	11	13	10
Max delay (min) - *area*	75	45	75	75
Total solution time (sec)	5	1320	310	180

According to this table, we can observe that Case-1 has the highest delay (890 min + 325 min), whereas the delay in Case-2 (550 min + 185 min) is the lowest. Although the number of affected flights is high (48 flights delayed in total), the total delay in Case-2 is almost only half of the delay than what is needed in Case-1. The maximum delay in Case-2 also proves the efficiency of using the optimisation approach to minimise delays. Regarding Case-3, it could be understood that this case combines the benefits of the two approaches, namely reducing delays and improving the flexibility of flight plan processing. Case-4 applies the optimisation approach with a set of flight plans in batches (i.e., around 40 flights). Obviously, if the batch size reduces to only one flight, then the solution should be exactly the same as with Case-1. Thus, the results suggest that some benefits (e.g., 9% delay reduction) could be still achieved even coupling a small number of flights for optimisation, which tends to be easy for implementation.

D. Delay analysis

Fig. 7 shows the exact delay allocated for each flight in four different case studies. There are only 75 delayed flights in total appearing in any of the cases. The flights are arranged in the order in which the flight plans are submitted. For Case-1, since most airspace volumes are available at the early stage, there is no delay for those early submitted flight plans. However, when time elapsed, more airspace resources are occupied, and thus an increasing amount of delay can be found at that time.

With regard to the optimisation approach, the flight delays appear to be evenly distributed, which is due to the fact that the optimisation approach considers the flight prioritisation as a whole (assumed to be equal in this paper although) rather than FCFS that forcefully priorities flights on their submission order. This feature can be also appreciated from Fig. 8 where the take-off time shifting is presented for each flight.

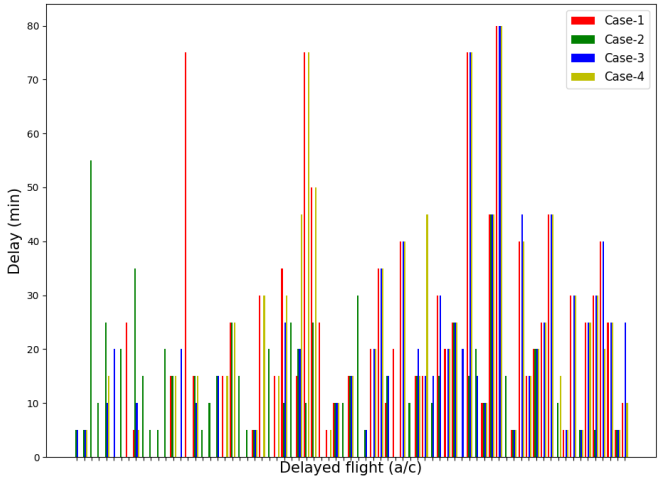


Figure 7. Individual flight delay from each of the four case studies.

Similarly, it can be noticed from the results of Case-3 that the number of delay for the first half of the flights is relative low and the distribution seems even. For the second half of the flights, the delay increases remarkably due to switching to the FCFS approach. The batch optimisation conducted in Case-4 shows some advantage over the FCFS approach, which is expected to be more beneficial if the batch size is increased.

V. CONCLUSIONS

In this paper, we presented an optimisation model for strategic deconfliction with a view to addressing the inefficiency

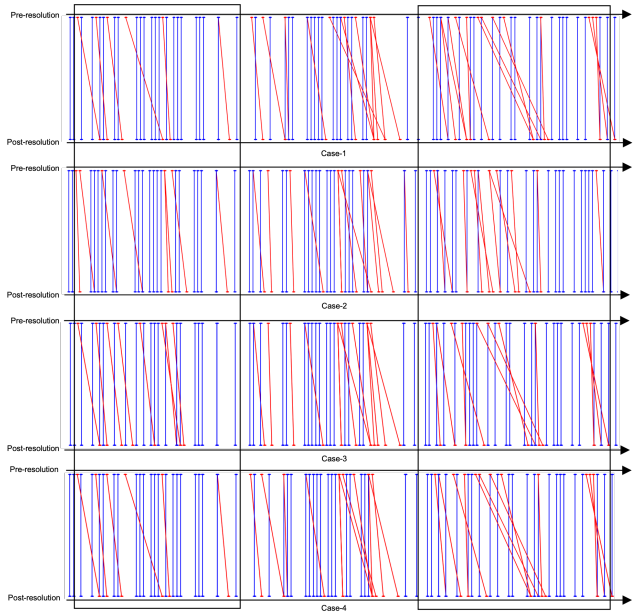


Figure 8. Take-off time shifting from pre-resolution to post-resolution.

of the conventional first-come, first-served (FCFS) approach. Both approaches were examined and compared through an illustrative example where four case studies were performed, including full FCFS, full optimisation, half FCFS and half optimisation, and another batch optimisation, depending on how the flight plans are grouped and tackled. Experimental results show that both approaches can effectively resolve the conflicts, with optimisation producing much less delay than what is required by FCFS. Depending on the availability of flight plans at the time when decisions need to be made, the optimisation approach can be customised jointly with the FCFS approach, or it be can realised through batch computations as a compromised solution. Some open questions still remain, such as the prioritisation scheme within the optimisation model and the associated fairness concerns, as well as how to better address the potential uncertainty factors with respect to the flight plans.

Acknowledgement

This work was partially funded by the SESAR JU under grant agreement No 101017702, as part of the European Union's Horizon 2020 research and innovation programme: AMU-LED (Air Mobility Urban - Large Experimental Demonstrations). The opinions expressed herein reflect the authors view only. Under no circumstances shall the SESAR Joint Undertaking be responsible for any use that may be made of the information contained herein.

Nomenclature

$l \in L$	set of *linear* flights
$a \in A$	set of *area* flights
$j \in J$	set of airspace volumes
$t \in T$	set of time moments
k	sliding time segment
s	safety time separation
L^j	subset of *linear* flight l traversing airspace volume j
A^j	subset of *area* flight a traversing airspace volume j
J_l	subset of airspace volumes that *linear* flight l is planned to traverse
J_a	subset of airspace volumes that *area* flight a is planned to traverse
T_l^j	subset of feasible time moments for *linear* flight l to enter airspace volume j
T_a^j	subset of feasible time moments for *area* flight a to enter airspace volume j
$J_l^{(i)}$	i^{th} $(1 \dots n)$ airspace volume of *linear* flight l
$J_a^{(i)}$	i^{th} $(1 \dots n)$ airspace volume of *area* flight l
$\overline{T}_l^j$	upper bound of feasible time moments T_l^j
$\underline{T}_l^j$	lower bound of feasible time moments T_l^j
$\overline{T}_a^j$	upper bound of feasible time moments T_a^j
$\underline{T}_a^j$	lower bound of feasible time moments T_a^j
r_l^j	scheduled time of *linear* flight l to enter airspace volume j
r_a^j	scheduled time of *area* flight a to enter airspace volume j
d_a	scheduled flight duration of *area* flight a
$\hat{t}_l^{jj'}$	scheduled flight time between segment jj' of *linear* flight l

References

[1] ICAO, "Doc 9854 an/458 global air traffic management operational concept," International Civil Aviation Organization, Tech. Rep., 2005.
[2] SESAR, "European ATM master plan: Roadmap for the safe integration of drones into all classes of airspace," SESAR JU, Tech. Rep., 2018.
[3] CORUS, "U-space concept of operations," SESAR JU, Tech. Rep. Edition 03.00.02, 2019.
[4] SESAR JU, "Supporting safe and secure drone operations in europe: A report of the consolidated sesar u-space research and innovation results," SESAR JU, Tech. Rep., 2020.
[5] DOMUS, "D5.2 final study report," SESAR JU, Tech. Rep., 2020.
[6] V. Lappas, G. Zoumponos, V. Kostopoulos, H. Shin, A. Tsourdos, M. Tantarini, D. Shmoko, J. Munoz, N. Amoratis, A. Maragkakis *et al.*, "Eurodrone, a european utm testbed for u-space," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2020, pp. 1766–1774.
[7] J. Rios, D. Mulfinger, J. Homola, and P. Venkatesan, "Nasa uas traffic management national campaign: Operations across six uas test sites," in *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*. IEEE, 2016, pp. 1–6.
[8] J. Rios, "Strategic deconfliction: System requirements," *NASA UAS Traffic Management (UTM) Project*, 2018.
[9] Connected Places Catapult, "Implementing an Open-Access UTM Framework for the UK," Connected Places Catapult, Tech. Rep., 2020.
[10] A. D. Evans, M. Egorov, and S. Munn, "Fairness in decentralized strategic deconfliction in utm," in *AIAA Scitech 2020 Forum*, 2020, p. 2203.
[11] M. Egorov, A. Evans, S. Campbell, S. Zanlongo, and T. Young, "Evaluation of UTM Strategic Deconfliction Through End-to-End Simulation," *14th USA/Europe ATM R&D Seminar*, 2021.
[12] D. Sacharny, T. C. Henderson, M. Cline, B. Russon, and E. Guo, "Faa-nasa vs. lane-based strategic deconfliction," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, 2020, pp. 13–18.
[13] F. Ho, R. Geraldes, A. Goncalves, B. Rigault, A. Oosedo, M. Cavazza, and H. Prendinger, "Pre-flight conflict detection and resolution for uav integration in shared airspace: Sendai 2030 model case," *IEEE Access*, vol. 7, pp. 170 226–170 237, 2019.
[14] Y. Tang, Y. Xu, G. Inalhan, and A. T. Tsourdos, "An integrated approach for dynamic capacity management service in U-space," *14th USA/Europe ATM R&D Seminar*, 2021.
[15] ICAO, "Unmanned Aircraft Systems Traffic Management (UTM) – A Common Framework with Core Principles for Global Harmonization," Tech. Rep., 2020.
[16] FAA-NASA UTM Research Transition Team, "UAS Traffic Management Conflict Management Model," FAA and NASA, Tech. Rep., 2010.
[17] AMU-LED, "D2.2.010 high level conops - initial," SESAR JU, Tech. Rep. Edition 00.01.00, 2021.