Dynamic Capacity Management for Air Traffic Operations in High Density Constrained Urban Airspace

Patrinopoulou, Niki; Daramouskas, Ioannis; Badea, Calin Andrei; Veytia, Andres Morfin; Lappas, Vaios; Ellerbroek, Joost; Hoekstra, Jacco; Kostopoulos, Vassilios

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Article

Dynamic Capacity Management for Air Traffic Operations in High Density Constrained Urban Airspace

Niki Patrinopoulou [1,*], Ioannis Daramouskas [1], Calin Andrei Badea [2], Andres Morfin Veytia [2], Vaios Lappas [3], Joost Ellerbroek [2], Jacco Hoekstra [2] and Vassilios Kostopoulos [1]

[1] Applied Mechanics Lab, University of Patras, 26504 Patras, Greece; daramousk@ceid.upatras.gr (I.D.); kostopoulos@upatras.gr (V.K.)
[2] Control and Simulation, Faculty of Aerospace Engineering, Delft University of Technology, 2629 HS Delft, The Netherlands; c.badea@tudelft.nl (C.A.B.); a.morfinveytia@tudelft.nl (A.M.V.); j.ellerbroek@tudelft.nl (J.E.); j.m.hoekstra@tudelft.nl (J.H.)
[3] Department of Aerospace Science & Technology, National Kapodistrian University of Athens, 10563 Athens, Greece; valappas@aerospace.uoa.gr
* Correspondence: n.patrinopoulou@upnet.gr

Abstract: Unmanned Aircraft Systems (UAS) Traffic Management (UTM) is an active research subject as its proposed applications are increasing. UTM aims to enable a variety of UAS operations, including package delivery, infrastructure inspection, and emergency missions. That creates the need for extensive research on how to incorporate such traffic, as conventional methods and operations used in Air Traffic Management (ATM) are not suitable for constrained urban airspace. This paper proposes and compares several traffic capacity balancing methods developed for a UTM system designed to be used in highly dense, very low-level urban airspace. Three types of location-based dynamic traffic capacity management techniques are tested: street-based, grid-based, and cluster-based. The proposed systems are tested by simulating traffic within mixed (constrained and open) urban airspace based on the city of Vienna at five different traffic densities. Results show that using local, area-based clustering for capacity balancing within a UTM system improves safety, efficiency, and capacity metrics, especially when simulated or historical traffic data are used.

1. Introduction

There is increasing interest in incorporating Unmanned Areal Vehicle (UAV) flights in urban environments to reduce ground congestion on city streets. Companies such as Amazon [1], Alphabet [2], and Walmart [3] have explored drone-based delivery for small packages. This application is predicted to become the largest source for urban air operations demand, as ground vehicles could be substituted by small unmanned aerial systems (UAS), reducing city traffic and CO_2 emissions [4]. Such systems could also be used to reliably transport sensitive medical supplies, as highlighted by Zipline [5] and UPS [6], which use drones to deliver medicine and COVID-19 vaccines, respectively.

To accommodate the demand for such operations, we aim to investigate the design of an Unmanned Traffic Management (UTM / U-space) system capable of serving a large number of flights in an urban environment. In 2014, NASA defined the purpose of UTM as a system to enable low-altitude airspace operations in a safe and efficient manner [7]. It should provide services such as airspace design, traffic congestion management, route planning and replanning, strategic and/or tactical separation management, dynamic geofencing, etc. The present research is intended to support the transition to the third level of U-space (U3), as defined by SESAR in [8]. At the U3 level, advanced services support complex operations in highly dense areas by including capacity management, conflict detection assistance, and automated detect and avoid (DAA) functionalities.

The research at hand is a follow-up study to the Metropolis II project [9]. The project aimed to create a unified approach to the design of a UTM system for very low-level, highly dense urban airspace and investigate the effect of the degree of centralisation of separation management on the performance of the system. The results of the project show that a hybrid approach (both centralised and decentralised components) achieves the greatest mission safety and efficiency.

Thus, the aim of this research is to determine and propose a dynamic capacity management solution for a hybrid UTM system. A central flow control module is developed to enable capacity balancing by identifying high-traffic density areas and informing the aircraft. This allows aircraft to individually and tactically replan their routes to avoid congested areas. The objective of developing the flow control module is to actuate a more homogeneous redistribution of traffic, which would have benefits the safety and efficiency of the flights. Flow control operates in flow groups, which are predefined regions where traffic density is measured.

The main scope of this work is to study location-based dynamic traffic capacity management techniques. This is conducted by creating flow groups with three different techniques, (1) street-based, (2) grid-based, and (3) cluster-based. The grid-based and cluster-based techniques are designed in two different flow group sizes to test the impact of the flow group size. A common flow control module acts as a traffic regulator for the three different techniques. This traffic regulator can be part of a future hybrid concept with decentralised flight planning and tactical conflict resolution. The flow control module is implemented and tested using all the proposed flow group definitions. The developed systems are compared to evaluate their performance across different metrics, emphasising safety and performance efficiency.

The article at hand is structured as follows: related work that contributed to the development of the methods at hand is presented in Section 2. An overview of the components and design considerations for a UTM concept of operations is presented in Section 3. The flow control design and methodology are presented in Section 4. The experimental process and results are given in Section 5, and the analysis and interpretation of the latter are presented in Section 6.

2. Related Work

Dynamic Airspace Management (DAM) intends to improve the demand-capacity balance of the airspace; this can be enabled by using adaptable airspace sectors for Air Traffic Management (ATM) [10]. The airspace is partitioned into control sectors based on the current locations of the aircraft to ensure that the aircraft count in each sector is lower than a predefined value. The DACUS project [11] investigated methods for determining the maximum capacity of the airspace and solutions for very high demands. It also identified the benefits of designing capacity metrics that consider the highly dynamic nature of U-space. An airspace 3D discretisation technique considering Communication, Navigation, and Surveillance (CNS) performance was introduced in [12]. The proposed technique enables dynamic management of airspace resources in urban airspace. The position and the estimated CNS performance of each aircraft are used to assign occupancy grids online.

The decentralisation of certain components of a UTM system offers better workload distribution and lowers the dependency on ground infrastructure, decreasing the chance for single point of failure occurrences [13]. A decentralised, hierarchical approach for a UAS ATM system allows scalability and robustness for high traffic densities [14]. Vertiports and vertihubs were defined as control entities in [14], with vertihubs controlling the flow of aircraft in and out of their sectors of responsibility, while vertiports control the takeoff and landing procedures of individual aircraft. Unmanned traffic can be structured using the Dafermos algorithm designed to avoid traffic congestion and complexity [15]. Thus, a network to describe the airspace and an air traffic assignment model should be defined.

A feasibility assessment of organising unmanned straight-line flights in densely populated areas demonstrated a noticeable increase in the number of inter-vehicle conflicts

in large clusters over densely populated areas [16]. The San Francisco Bay Area is used as the experimental area, in which up to 1,000,000 flights per day were simulated. The performance was evaluated using a count and frequency of large deconfliction problems. These results show that free unmanned flights are feasibly safe for up to 10,000 flights per day, while more elaborate traffic management is required for larger numbers of flights. Their results suggest that the number of conflicts in urban airspace can be reduced by applying airspace structure and capacity balancing techniques in the system.

Air traffic management literature includes different metrics capable of describing the traffic in the airspace. Those metrics may be used to develop better-performing traffic management systems. In 1998, NASA introduced a dynamic density metric for air traffic [17], including traffic density and traffic complexity. It was designed to provide real-time information about air traffic to help in estimating the workload of an air traffic controller. The proposed factors contributing to the dynamic density metric are the traffic density, three dynamic factors describing heading, speed and altitude changes, two aircraft distribution factors counting the aircraft with Euclidean distance shorter than two predetermined limits, and three conflict factors predicting the number of conflicts in specific future time intervals [17]. Aircraft clusters can be used in the future to automatically determine dynamic regions of high traffic density and complexity [18], and special cluster identification methodologies need to be applied.

In 2019, SESAR presented a performance framework for the European ATM Architecture and proposed Key Performance Indicators (KPIs) and Performance Indicators (PIs) over nine Key Performance Areas (KPAs): Safety, Security, Environment, Capacity, Operational Efficiency, Predictability, Cost-effectiveness, Flexibility, and Access and Equity [19]. Several of these indicators were taken into account for the work at hand.

3. Unmanned Traffic Management System Overview

The UTM system studied in this paper is a hybrid system that includes airspace rules, flight planning, tactical separation management, and capacity management functionalities for highly dense urban airspace. The rules and operations are common for all the UAVs of the system. The current section presents an overview of the concept of operations of this system and of its basic modules and design requirements. The research conducted in this paper focuses on the capacity balancing and flow control functionality of the system, while the rest of the system is designed to provide the testing framework for the module.

It is assumed that the UAVs of the system have no information concerning future flights or access to the exactly planned routes of other UAVs. Due to this assumption, strategic pre-flight planning and deconfliction are not included in this system. The absence of strategic planning intensifies the generation of traffic hotspots, which capacity balancing attempts to resolve through flow control. The lack of strategic planning, in combination with the use of high traffic densities during experiments, tests the effectiveness and the limits of the studied flow control implementations.

The proposed hybrid system can be divided into four modules:

1. Airspace rules and structure: responsible for defining a set of flight rules for the aircraft, aiming at increasing the overall efficiency and safety of the system by making traffic aligned and segmented [20,21].
2. Capacity management and flow control: responsible for capacity balancing by collecting position data from the flying aircraft and generating and distributing traffic data.
3. Flight planning: responsible for computing the flight plan of every aircraft based on the origin and destination pair while considering the airspace structure and traffic data sent from flow control.
4. Conflict detection and resolution: responsible for the tactical separation of the aircraft and hence their safety. It detects conflicts and takes the necessary actions to prevent them.

The four main modules and their relationship are depicted in a diagram in Figure 1. The airspace structure and rules are fed as input to all other modules of the system. The

definition of the flow groups takes place before the system initiates its operation. Flight planning is conducted by each UAV before takeoff (i.e., pre-flight) to compute the initial flight plan. As the UAV executes its flight plan, it communicates its state to nearby UAVs to detect and resolve potential conflicts. During the flight, the positions of all flying UAVs are sent to the dynamic capacity balancing module, which computes the traffic information and shares it with the UAVs. The UAVs may use their flight planning module and the updated traffic information to update their routes in order to avoid congested areas.
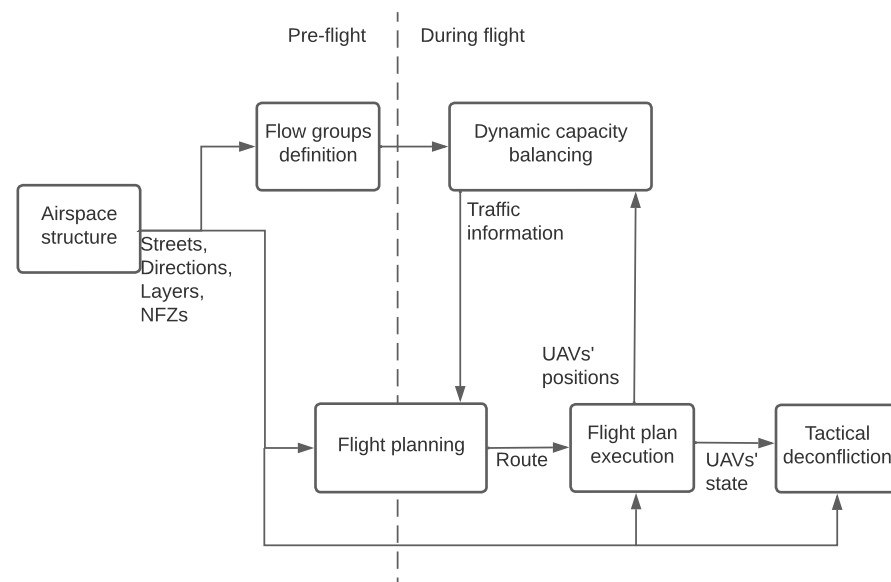
Figure 1. Diagram presenting the modules and functionalities of the studied UAS Traffic Management (UTM) system. The data exchange between modules is indicated by arrows. Modules are grouped depending on if they act pre-flight or during flight.

3.1. Scenarios and Missions

In order to design and test a capacity balancing management module for a UTM system, an urban airspace environment was created based in the city of Vienna. The airspace is divided into two areas: constrained airspace in the city centre, where aircraft must follow the existing street network when cruising, and open airspace on the outskirts of the city, as shown in Figure 2. Open airspace also contains restricted areas (such as public parks and cemeteries) over which flying is prohibited.

Within the defined urban environment, high-density air traffic scenarios were generated. Two types of departure points are included: parcel distribution centres, from which a large volume of aircraft depart, and vertiports, which are scattered around the city. Thus, several types of missions are defined by the function of their characteristics, as presented below:

1. Parcel delivery: Missions that originate from distribution centres and head to vertiports, creating diverging patterns around distribution centres. They account for 40% of the overall traffic.
2. Food delivery: Missions that originate from vertiports and head to vertiports, creating point-to-point traffic patterns. They account for 60% of the overall traffic.
3. Emergency missions: Point-to-point missions of high importance and low frequency. They account for less than 0.1% of the overall traffic.

Not all missions of the systems are assumed to be of the same importance or urgency. A medical delivery mission is reasonably more important and, in many cases, more urgent than commercial package delivery. Priority may be assigned for special missions [22], or depending on the mission type and the mass of the aircraft [23]. For those reasons, three mission priority levels (from 1 to 3) are defined for the experiments of this study. The

urgency of the mission increases with the level of priority. Emergency flights always have a separate priority level, which is significantly more urgent than the mission priority levels. The mission priority levels are equally distributed among the not emergency missions. Finally, two aircraft types based on a simplified DJI Matrice 600 Pro hexacopter drone performance model were used in the simulations. The difference between the two is that one cruises at a speed of 20 kts while the other one at 30 kts.

Figure 2. The Vienna airspace used in our system. The denser city centre is defined as constrained airspace, while the airspace around it is open airspace containing No Fly Zones (NFZ) areas. The overall airspace is a cylinder of an 8 km radius and 500 feet in height.

3.2. Airspace Structure

The airspace structure was developed separately for open and constrained airspace to meet the requirements of each airspace type. The common element between the open and constrained airspace structures is the existence of the same number of altitude layers. In both cases, the airspace is vertically decomposed into a set of layers that span the whole airspace in the horizontal plan and differ in the vertical plan. Layers have been proven to increase the structure of the traffic and decrease potential conflicts by segmenting and aligning aircraft [20]. For open airspace, the layers are used to separate traffic based on where the aircraft is heading. Each layer allows for a limited heading range; thus, only aircraft with similar bearings will fly at a specific altitude.

Layers have two purposes in constrained airspace: (1) vertically separate intersecting traffic and (2) isolate aircraft in the process of turning, as intersecting traffic and turning aircraft provoke a large number of conflicts [24]. To accommodate these purposes, constrained airspace layers include turning layers alongside cruising layers, which aircraft use during the turning process in order to minimise the disruption of the main traffic flows. These turn layers are situated between two cruising layers of different directions. Cruising layers are the main layers while flying in constrained airspace and are grouped into two categories based on their direction. More specifically, consecutive street edges in constrained airspace are grouped into streets or "layer groups" based on the continuity of their bearing. The streets in constrained airspace are one-way streets in a fully connected network. Cruising layers were assigned to layer groups to maximise the separation of cruising layers in intersections. However, cruising layers are not separated in all intersections due to the non-orthogonal geometry of the network.

3.3. Flight Deconfliction Logic

The deconfliction module is composed of two sub-modules: conflict detection and conflict resolution. The conflict detection module is tasked with identifying potential intrusions (when two aircraft are closer than the minimum separation margin, which is set to 32 m for the horizontal plane and 10 m in the vertical axis) that might happen. Potential conflicts are identified using a look-ahead time of ten seconds before a potential loss of separation occurs. The conflict detection method used within the concept at hand is a state-based one, as intent-based decentralised conflict resolution has been found to cause airspace instability [25]. Thus, it was hypothesised that the robustness of the method would overcome the shortcomings given by the geometry of the airspace in which it is employed.

The conflict resolution module is activated once conflicts are detected. Based on the conflict geometry, the mission priorities of the aircraft involved, and the available solution space, actions are performed in order to avoid an intrusion event. The algorithm chooses to either perform a resolution action or not, depending on the priority of the involved aircraft. In the situation where aircraft are one behind the other, it was decided that the aircraft in the front always has priority in order to avoid increasing the complexity of the situation.

Thus, the aircraft with the lowest priority will perform one of three actions: it will continue its trajectory while the intruder resolves, it will ascend or descend to another cruise layer, or it will perform speed-based conflict resolution. The latter consists of using velocity obstacles to calculate the required velocity for the aircraft with the lowest priority so that an intrusion is avoided.

3.4. Flight Planning

Every time a new flight request is generated, the aircraft must plan an efficient route to complete its mission. It is assumed that the individual agents do not have knowledge about other flight requests. The designed flight planning system includes algorithms for path planning in both constrained and open airspace; in constrained airspace, the designed route must follow the street network and its directionality as described from the airspace configuration, while for open airspace, the route should not violate any of the imposed No-Fly Zones (NFZs). The route computation is conducted in 2D, while the aircraft configures its movement in the vertical axis dynamically depending on the airspace structure characteristics of its location. The aircraft will start its flight in the lowest cruise layer. The designed routes are optimised to minimise the flight time and the number of turns, as heading changes are a major cause of conflicts, diminishing the safety of the system [24]. Finally, the system should offer fast replanning capabilities, as without any information concerning other flight requests, it has to act dynamically in problematic situations.

Based on the aforementioned requirements, the D* Lite [26,27] algorithm was selected as the basis of the path planning system. The D* Lite algorithm is capable of finding the most cost-effective path connecting an origin node to a destination node in a graph, with assigned costs for traversing every edge of the graph. D* Lite offers the advantage of fast dynamic replanning. It computes the route cost starting from the destination and moving towards the origin node. The algorithm is terminated when the search reaches the origin node. For every processed node, the cost to reach that node from the goal has been computed and saved to be used during replanning. For the replanning process, the algorithm recomputes the costs only for the nodes between the changed edges and the start point, allowing for fast replanning.

The application of the D* Lite algorithm for constrained airspace is straightforward since the constrained airspace is described as a directed graph, with the street intersections being graph nodes and the street segments connecting them being the edges. Turning had an increased cost over continuing in the current street. This allowed UAVs to prefer routes with fewer turns in constrained airspace, as, based on the literature, turns generate a large number of conflicts [24], and minimising the number of turns increases the efficiency of the route in regards to time and energy [28,29]. In order to apply the same algorithm for open

airspace, a set of nodes and the connection between them has to be defined. The selected approach for creating the nodes in open airspace is the trapezoidal cell decomposition [30,31], which transforms the open airspace into a set of cells. Each cell corresponds to a node point located at the centre of the cell, and a bi-directional graph describing open airspace is created.

4. Dynamic Capacity Management

Dynamic capacity management will be a mandated component of any future UTM system for urban airspace [32]. This module will help mitigate local hotspots and will enable UAVs to react to the actual traffic situation. We propose the inclusion of a flow control module in order to minimise those problems for UTM systems. Flow control is applied only in constrained airspace, in which traffic is congested due to space limitations. A flow control design and different implementations are proposed in this section for this purpose. Constrained airspace is subdivided into flow groups based on different criteria: (1) street-based, (2) grid-based, and (3) cluster-based. A common flow control module is applied to each different design.
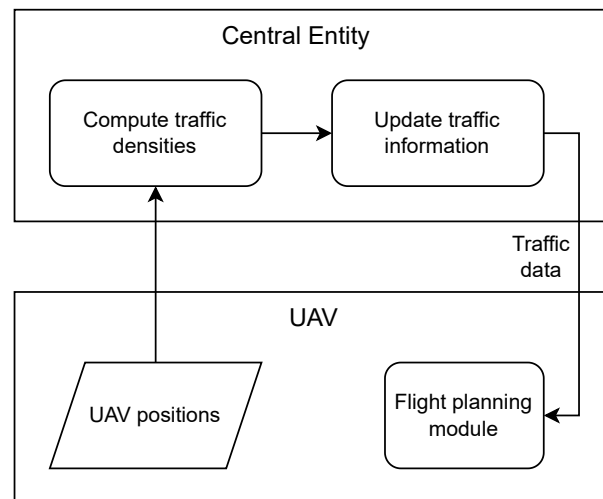
4.1. Flow Control

The flow control module is an active, centralised service in our system. It is a centralised information resource that individual agents can use to replan. It may be compared with the multiple existing mobile traffic and navigation applications (e.g., Google Maps, Apple Maps, Sygic GPS Navigation and Maps), as those apps gather location data from smartphone users and estimate traffic and arrival time at the desired destination. Those applications compute and propose alternative routes to avoid traffic and minimise the travel time, but it is up to the user to decide which route they prefer to follow. Following the same logic, the proposed flow control module gathers location data from the UAVs registered in the system and estimates the traffic density across the streets. That information is shared with the UAV agents, who plan their routes using the traffic data.

4.1.1. Flow Control Module Responsibilities

The flow control module is responsible for gathering traffic data from all the UAVs in the air, processing them, and sharing that information with the UAVs of interest. The idea behind the flow control system is inspired by [33]. They focused on traffic management in a warehouse with autonomous ground vehicles. In their case, the routes were computed from a uni-directional graph with a fixed known cost for each edge. They added a time-varying cost at each edge, as assigned by the traffic management agents. Our work applies a similar strategy. The flow control logic is presented in Figure 3, and the methodology is composed of the following four steps:

1. Data gathering: The flow control module gathers the position of all UAVs every 10 s.
2. Traffic computation: After the data are collected, the UAV positions are grouped to generate the current traffic pattern. A set of predefined flow groups is used, as described in Section 4.2, and each UAV is assigned to one of the groups depending on its position. The number of UAVs in each group is used as a traffic indicator for that group. The traffic density is computed by dividing the UAV number by the summed length of all the streets contained in the group.
3. Traffic information update: Based on the UAV density computed for each group, the flow control system updates the traffic information and imposes some costs for the streets contained in that group. Higher densities correspond to higher costs.
4. Traffic information announcement: The recently updated traffic information needs to be communicated to the affected UAVs so that they can regulate their flight speeds and possibly replan to avoid high traffic.

Figure 3. Diagram showing the operation logic of the flow control module.

4.1.2. Flow Control Speed Limits

When updating the information and cost of a traffic group, two values are changed. Every flow group has two kinds of speed limits: a real speed limit and a virtual speed limit. The real speed limit corresponds to the maximum allowed speed for the streets of the flow group. The virtual speed limit has an effect only while an aircraft is planning a route, as it is the one used as the cost of traversing a street. As a result, a path using a very crowded street appears to have a higher cost during path planning and an alternative path is investigated. Three levels of traffic and speed limits are used:

1. Low traffic: Density of less than 0.005 aircraft per metre with a real and virtual speed limit of 30 knots.
2. Medium traffic: Density between 0.005 and 0.025 aircraft per metre with a real speed limit of 30 knots and a virtual speed limit of 15 knots.
3. High traffic: Density of more than 0.025 aircraft per metre with a real speed limit of 15 knots and a virtual speed limit of 0.1 knots.

The maximum density of the constrained airspace with the imposed airspace structure is achieved for 0.156 aircraft per metre (one aircraft per layer in the five layers with 32 m of separation distance between aircraft in the horizontal plane), such that aircraft do not violate the minimum separation distance. The traffic levels are selected based on that value and are fine-tuned using extensive testing.

After the speed limits are updated, the flow control module sends the new speed limits to the aircraft. The responsibility of utilising that information is assigned to the aircraft. When an aircraft initially plans its flight plan, it uses a copy of the original street graph, assuming that all flow groups have low traffic. Before taking off, the aircraft requests the flow control module to provide it with the current speed limits, and the aircraft updates its path, replans it if needed, and then starts its flight. After take-off, the aircraft keeps updating its graph every time it receives new traffic information from the flow control module.

4.1.3. Flight Replanning

When an aircraft receives the speed limit changes, it updates the cost of the nodes directly affected by the changes, as described by the D* Lite algorithm [26] and the edge speed limits. If there are changes in its graph, the aircraft recomputes its path by re-applying the D* Lite algorithm for the updated weights.

The difference between planning and replanning is found in the initialisation of the graph. Initial planning starts with an original graph, with no computed costs for edges, while in replanning, the costs of the edges are computed based on the current traffic

situation. The computation time of replanning depends on the number of changes and on whether these have an effect on the already explored nodes.

Because the flow control system imposes speed limits in an attempt to regulate traffic reactively, it could cause all the aircraft that found themselves in high traffic to replan using the same alternative route. That behaviour would cause the high-traffic mass to be moved into another flow group instead of being dissolved. An example of oscillatory congestion in traffic networks is presented in [34]. Users of routing apps react in real-time to traffic congestion, and drivers switch between highways trying to avoid heavy traffic. They concluded that the use of routing apps may have a negative effect on the stability of equilibrium points and cause oscillatory traffic patterns. To avoid that phenomenon in our system, aircraft flying in streets with high traffic density are allowed to replan based on their mission priority. In this way, only a portion of the aircraft contribute to high traffic replans, allowing them to escape the traffic area and decreasing traffic density. Our system includes three mission priority levels. Replanning in high-traffic density streets is only allowed for the highest priority flights, representing approximately 33.5% of the total aircraft.
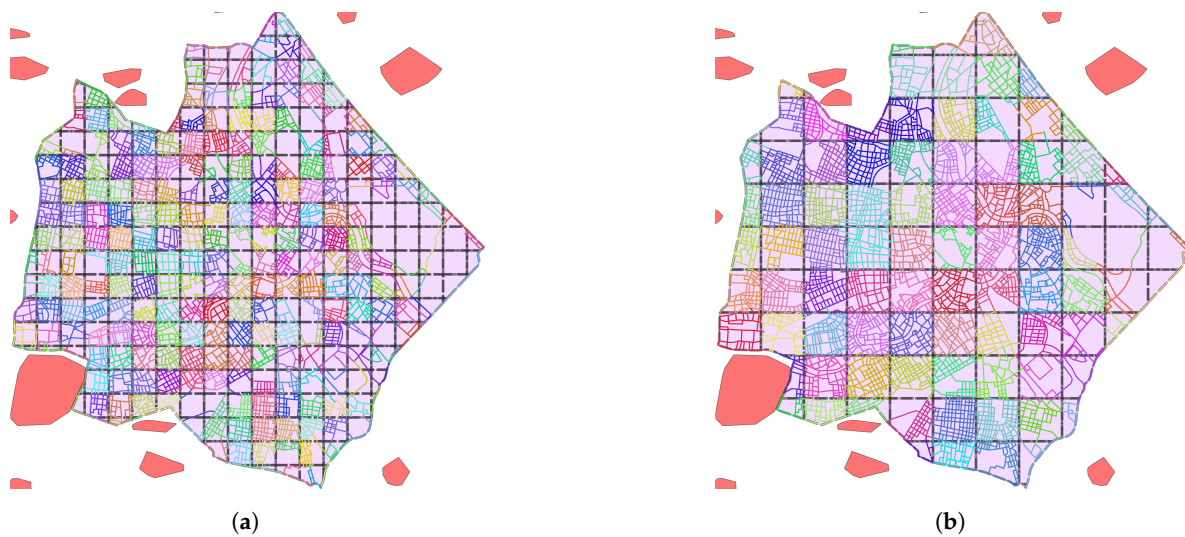
4.2. Flow Groups

The main focus of this study is to investigate how constrained urban airspace should be segmented into flow control sectors or flow groups to facilitate dynamic capacity balancing. Three separate sectoring methods for the definition of flow groups are proposed. The grid-based and cluster-based techniques are implemented for two sizes (small and large) to test the impact of their size. The small and large implementations are designed to have similar sizes for the grid-based and cluster-based cases in order to enable the comparison of the results.

4.2.1. Baseline: Street Groups

The street groups methodology generates flow groups by conjoining sequential and continuous street edges. As described in Section 3.2, the street edges are grouped into layer groups based on their continuity. Two street edges sharing a node (i.e., neighbouring edges) are grouped together if they have similar bearings. The Continuity in Street Networks (COINS) algorithm is used for this purpose [35]. The flow groups are continuous groups of street edges, and they are generated by splitting the layer groups into shorter streets. The final flow groups have an average length of 335.04 m, and each generated flow group contained at least one street edge. In total, 2113 flow groups are defined for constrained airspace after splitting the layer groups. This method was created and used by the decentralised concept of Metropolis II and will be referred to as the baseline for this paper.

4.2.2. Grid Sectors

The grid sectors methodology defines flow groups as a group of neighbouring streets rather than a collection of continuous streets. That is achieved by dividing the airspace area into subareas, each defining a flow group and including all the street edges of the subarea. Two implementations of this approach are generated, one with a grid of 20 by 20 squares of 500 m edge size (small grid sectors) and one with a grid of 11 by 11 squares of 909 m edge size (large grid sectors). The grids are overlaid over the constrained airspace graph. The street edges are distinguished into flow groups based on which square contains each of them. If a street edge belongs to more than one square, then it is assigned to the one in which it has the longest segment. Squares that do not include any street edges at the end are deleted. This method creates 263 flow groups for the small grid sectors implementation and 92 flow groups for the large grid sectors implementation. The resulting flow groups are shown in Figure 4 for the two implementations, in which the street edges are coloured depending on their corresponding flow group.

(**a**) (**b**)

Figure 4. The defined flow groups for the two grid sectors' implementations. The overlaid grid is delineated in black, while each flow group is drawn by a different colour. (**a**) The small grid sectors flow groups; (**b**) The large grid sectors flow groups.
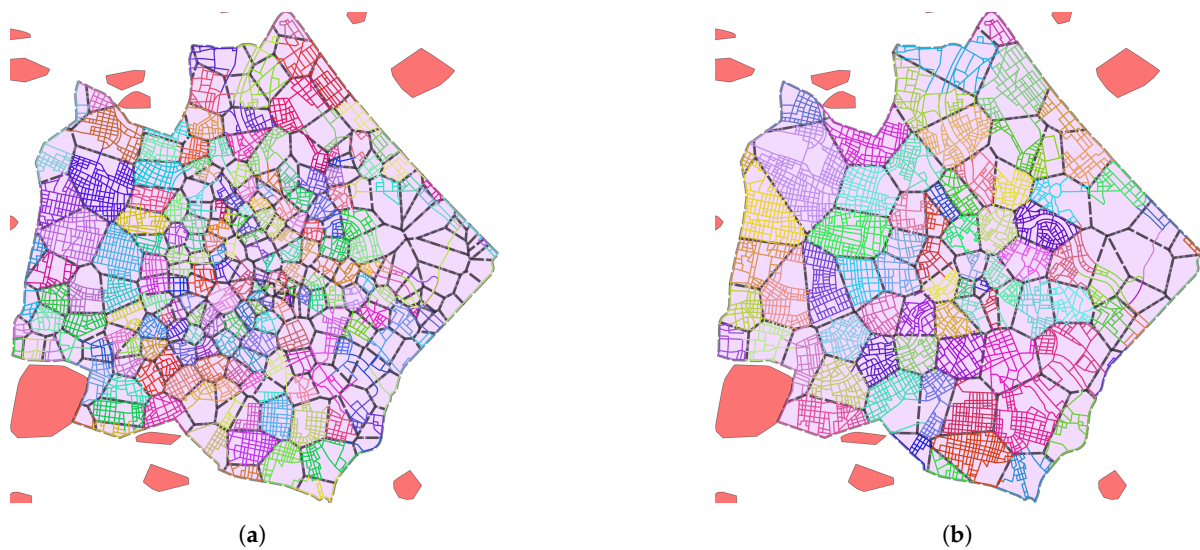
4.2.3. Cluster Sectors

The cluster sectors method requires historical or simulated data in order to use conflicts as input. The conflicts are grouped into clusters based on their location, and the geometric centres of the clusters are used to define the flow groups. We selected to run a test with the flow control module deactivated for a low-density scenario with 3350 flights over 1.5 h, and that created a collection of 12,916 conflicts. The conflicts are grouped using the k-means constrained clustering algorithm [36] and provides the desired number of clusters and maximum size for each cluster. Utilising this technique allows us to generate denser and smaller cluster areas in locations with a large number of conflicts and larger cluster areas in locations with fewer conflicts. After the cluster centres are computed, a Voronoi graph is created based on the centres. The polygons of the Voronoi graph are cropped based on the geometry of the constrained airspace. The street edges contained in a certain cell are used to define a unique flow group. If a street edge belongs to more than one cell, it is matched to the cell in which it has the longest segment, in the same way as in Section 4.2.2.

The cluster sectors method is used to generate two different sets of flow groups, the large cluster sectors and the small cluster sectors. These use the same conflicts as inputs but have different clustering settings in order to vary the average size of a cluster. The selected constants used in the clustering algorithm for the first set were 100 clusters, with a maximum number of 300 conflicts per cluster, which created a number of 80 flow groups, as shown in Figure 5b. For the second set, 300 clusters were used with a maximum cluster size of 100 conflicts. The second set of flow groups is presented in Figure 5a and includes 233 flow groups. In both cases, the flow group size is noticeably smaller in the city centre, where the conflict concentration is larger.

4.2.4. Flow Group Definitions Overview

The following section presents a comparison of the obtained structures for each method. Table 1 presents the number of flow groups for each flow group division method, alongside the minimum, average, and maximum length of the flow groups. Although the average and maximum length of the baseline flow groups are the smallest of the five cases, its minimum length is larger than the grid sectors methods and the small clusters implementations. Furthermore, in the baseline case, the number of flow groups is about 10 times greater than the number of flow groups for the small division implementations and about 25 times greater than the large division implementations.

(**a**)   (**b**)

Figure 5. The defined flow groups for the two cluster sectors implementations. The Voronoi graph is delineated in black, while each flow group is drawn in a different colour. (**a**) The small cluster sectors flow groups; (**b**) The large cluster sectors flow groups.

The graph presented in Figure 6a shows the number of street edges in each group in the five different implementations. The baseline implementation has a noticeably lower dispersion compared to the rest. For the baseline approach, the standard deviation of the number of edges per flow group is 1.2, while it takes the values of 18.67, 53.97, 73.74, and 33.76 for the small grid sectors, large grid sectors, large cluster sectors, and small cluster sectors, respectively. The small grid sectors and the small cluster sectors have similar values for the largest percentage of data, while the latter has some outliers with higher values. The large grid sectors and large cluster sectors also show comparable numbers of edges. The small and large implementations of the grid and cluster sectors are deliberately selected to have similar lengths and numbers of edges. The boxplot presented in Figure 6b shows the length of each flow group in the five different methods. The large cluster sectors and large grid sectors approaches are depicted to have longer flow groups, with more than half of them being longer than 10 km.

(**a**)   (**b**)
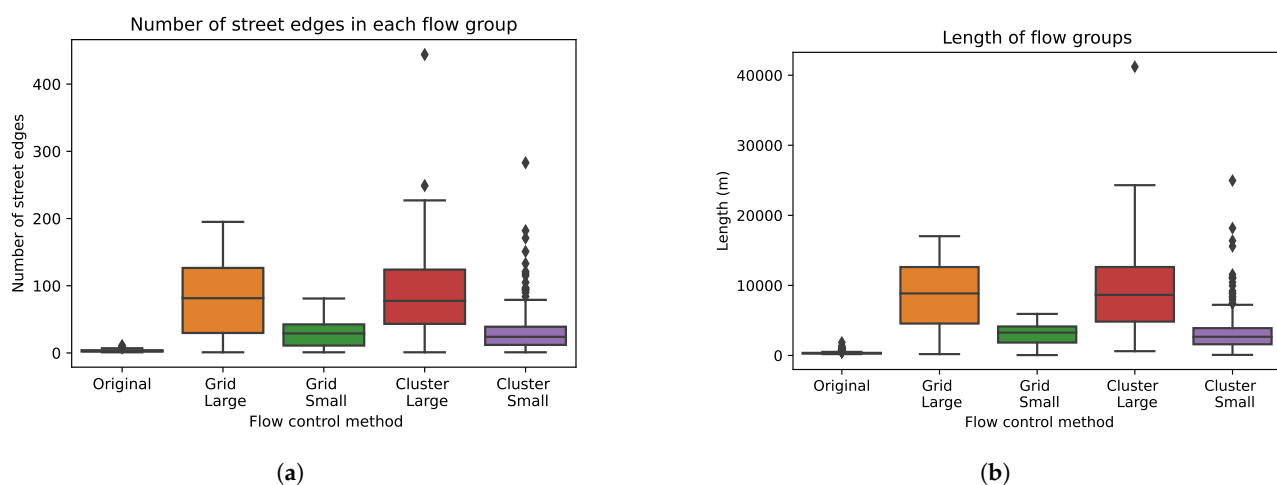
Figure 6. The number of street edges and the total length for per flow group for the five flow control implementations. (**a**) The number of street edges per flow group; (**b**) The sum of the street edges length per flow group.

Table 1. An overview of the different designed flow groups.

Flow Control	Number of Flow Groups	Minimum Length (m)	Average Length (m)	Maximum Length (m)
Baseline	2113	193.22	335.04	1858.1
Grid sectors large	92	191.6	8523.34	17,020.71
Grid sectors small	263	53.14	2981.55	5928.727
Clusters sectors large	80	610.1	9801.84	41,209.44
Clusters sectors small	233	79.93	3365.44	24,964.13

5. Concept Evaluation

The current section includes the description of experiments, the simulation results, and the evaluation of the flow group definition methods.

5.1. Experiment Setup

The flow control experiments require simulations to test the performance of the system across different scenarios. The open-air traffic simulator BlueSky [37] is used to execute simulations with up to 1900 aircraft simultaneously in the air. BlueSky is designed to be highly modular, allowing the implementation of the flow control and the other system modules as programmed plugins.

The flow control approaches are tested using the BlueSky simulator in five levels of flight densities. For every density, nine different traffic scenarios are generated and tested, which share the same experimental parameters but include different flight paths. The results of the nine repetitions are used to create each boxplot. Table 2 presents the five density levels used in the experiments. The values of the average number of aircraft per scenario and the maximum number of concurrent aircraft for each density are provided. These numbers correspond to the total airspace, described as a cylinder with a radius of 8 km and a height of 500 feet.

Table 2. The traffic densities used in the experiments.

Density	Average Number of Aircraft per Scenario	Maximum Number of Concurrent Aircraft
Very Low	1660	350
Low	3340	720
Medium	4990	1100
High	6650	1500
Very high	8290	1900

5.2. Dependent Variables

Seven metrics are used to evaluate the performance of the system across four different categories: safety, efficiency, capacity, and replanning. The seven dependent experiment variables are introduced in Table 3.

One capacity metric is proposed, called average demand delay. The demand delay per flight is measured as the ideal time of arrival subtracted by the actual time of arrival at the destination. Firstly, the ideal time of arrival is computed as the required time of departure plus the ideal flight time, which is calculated as the length of the shortest route connecting the origin to the destination without considering traffic and turns, divided by the cruising speed of the aircraft. A similar metric is employed in [38]. Then, the computed demand delays are averaged to calculate the average demand delay for each scenario.

Efficiency is expressed as horizontal distance and route duration efficiency, like in [39]. Both are provided as a percentage value, where larger values represent better system efficiency. The true values of the horizontal distance and route duration are compared to the ideal to compute the value of efficiency. The metrics are computed by dividing the ideal values by the actual ones.

Table 3. The metrics used to evaluate the flow control approaches.

Metric Name	Metric Category	Metric Description
Average demand delay	Capacity	The average delay in reaching the destination in comparison to ideal arrival time.
Horizontal distance efficiency	Efficiency	The percentage of the sum of the ideal length of all flights over the sum of the actual length of all flights.
Route duration efficiency	Efficiency	The percentage of the sum of the ideal execution time of all flights over the sum of the actual execution time of all flights.
Number of conflicts per flight	Safety	The number of conflicts detected divided by the number of aircraft in the scenario.
Number of intrusions per flight	Safety	The number of intrusions detected divided by the number of aircraft in the scenario.
Number of replans per flight	Replanning	The number of times aircraft modified their flight plan due to replanning divided by the number of aircraft in the scenario.
Number of attempted replans per flight	Replanning	The number of times aircraft attempted to replan and did not modify their flight plan divided by the number of aircraft in the scenario.

The number of conflicts and the number of intrusions per flight are utilised as safety metrics, similar to [40]. A conflict is defined as a predicted situation that will lead to a loss of separation if no action is taken, while an intrusion is a loss of separation. In our experiments, the minimum allowed separation distance between two UAVs is 32 m in the horizontal plane and 10 m in the vertical axis.

Finally, two metrics regarding replanning are introduced to count the number of replans and attempted replans per flight. The replanning metrics are designed specifically for this work. Although those two metrics do not act as an indicator of the performance of the system, they are proposed to quantify the reactiveness of each proposed approach. The replan and attempted replan events are both triggered by the changes in the cost of the path planning graph, applied by the flow control entity due to new traffic data. A replan is considered when an aircraft selects a new route to reach its destination. An attempted replan is considered when an aircraft attempts to compute the best route based on the new traffic data, and the generated route is identical to the one the aircraft was already executing.

5.3. Results

5.3.1. Capacity

The average demand delay is shown in Figure 7 and represents the average delay in reaching the destination for all flights in a scenario. The baseline approach offers the lowest delay for very low traffic density and provides comparable values up to medium density. For the high and very high densities, the results are reversed, with the baseline approach offering the longest demand delays. The implementation with no flow control shows the worst demand delay for densities up to the medium level. For the high and very high density levels, only the baseline approach generates more delay than the system with no flow control. The other methods of flow group definitions (i.e., grid sectors and cluster sectors variations) show the best results for densities between medium and very high. The small cluster sectors approach seems to be the best performing across different densities.
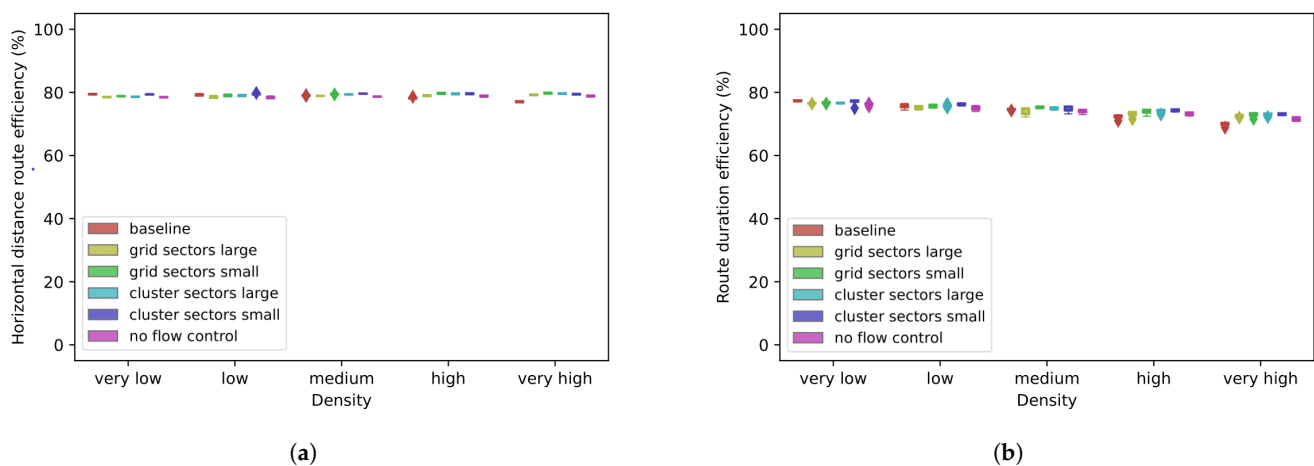
5.3.2. Efficiency

Figure 8 shows the horizontal distance route efficiency and the route duration efficiency metrics. The flow control implementations show similar patterns for the horizontal distance efficiency and the duration efficiency values across all densities. From very low to medium densities, all six implementations have very close values, with the no flow control, the two grid sectors, and the large cluster sectors cases having slightly smaller performance values for the very low density. For the high density, the baseline has the lowest performance values. The very high density presents similar results to the high density, but the differences

between the implementations are larger. The route duration efficiency metric presented in Figure 8b shows similar results.
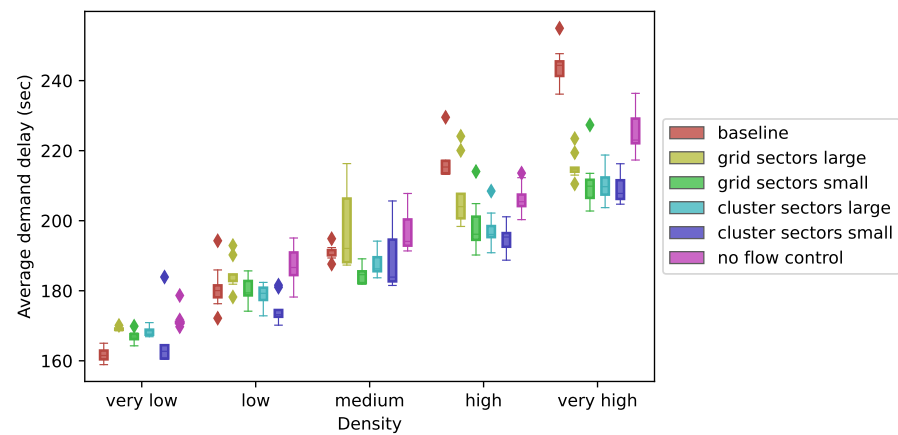

Figure 7. Boxplot graph for the average demand delay, measured in seconds, across the five different density values.

Figure 8. Boxplot graphs of the horizontal distance route and route duration efficiency metrics across the five different density values. (**a**) The horizontal distance route efficiency, measured as a percentage over the ideal horizontal distance route; (**b**) The route duration efficiency, measured as a percentage over the ideal route duration.

5.3.3. Safety

Figure 9 presents the number of conflicts and the number of intrusions, respectively, per flight for the five different density values. The six different flow control approaches show similar patterns for a number of conflicts and intrusions. In very low and low density, the number of conflicts and intrusions remain relatively constant across the six different implementations. This may be because streets are not yet congested, so the flow control system is not very active. In medium density, some variation appears between the results of the different implementations. The system with no flow control has a slightly higher number of conflicts and intrusions, while all alternative flow group definition methods show slightly better results than the baseline. In high and very high densities, the small grid sectors, large cluster sectors, and small cluster sectors show clear improvement over the baseline. The small cluster sectors implementation is the best performing, as it reduces conflicts and intrusions up to 15% from the baseline. In the higher densities, we also notice that the no flow control approach generates a smaller number of conflicts and intrusions than the baseline system, which means that the baseline system has reached its capacity limits.

(**a**)　　　　　　　　　　　　　　　　　　　　　　　　　　　(**b**)
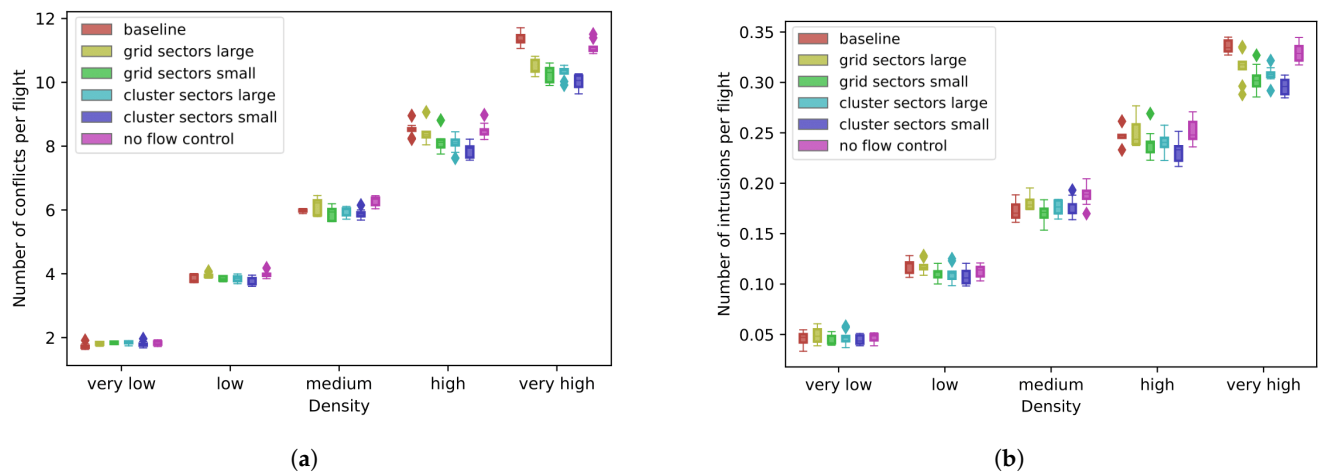
Figure 9. Boxplot graphs of the number of conflicts and the number of intrusions per flight across the five different density values. (**a**) The number of conflicts per flight, measured as the number of detected conflicts. (**b**) The number of intrusions per flight, measured as the number of loss of separation events.

5.3.4. Replanning

Figure 10 shows the average number of replans and attempted replans per flight. Figure 10a demonstrates that the number of replans per flight shows an increasing trend with increasing density levels for all flow control approaches. The baseline approach causes about twice the number of replans than the small cluster sectors approach. The number of attempted replans, from Figure 10b, is relatively constant across the five density levels for the baseline approach, while it presents an increasing trend for the rest of the implementations. The baseline approach shows the greatest number of attempted replans, having comparable values to the small cluster sectors approach for high and very high densities. The high numbers of replans and attempted replans in the baseline approach may be explained by the length of its flow groups. As derived from Table 1, the baseline approach has significantly shorter flow groups by average and small-scale redistribution of traffic might change the costs of the path planning graph. On the other hand, the two grid sectors' implementations have comparable but shorter flow groups to the corresponding cluster sectors' implementations. When comparing the grid sectors small to the cluster sectors small and the grid sectors large to the cluster sectors large, the cluster sectors trigger more replans and attempted replans than the grid sectors.

(**a**)　　　　　　　　　　　　　　　　　　　　　　　　　　　(**b**)
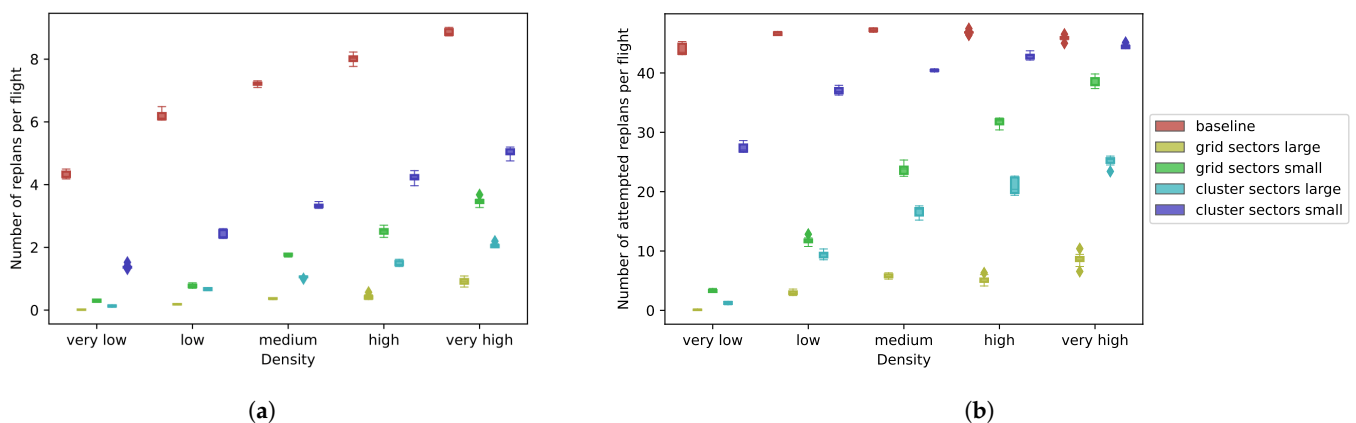
Figure 10. Boxplot graphs of the number of replans and attempted replans per flight across the five different density values. (**a**) The number of replans per flight. (**b**) The number of attempted replans per flight.

6. Discussion

The flow control study provided a number of interesting results. The most immediate result is that the baseline system has an overall worse performance than the case in which no flow control is used. Although the baseline implementation showed improved results over the no flow control implementation for low densities across the capacity, efficiency and safety metrics, the results were reversed for high and very high densities. This indicates that the baseline approach reached its capacity limits and has its bottleneck at a medium traffic density level. On the other hand, the proposed alternative flow control methods were more effective and generated improved or comparable results to the no flow control implementation for all density levels. That shows that the identified bottleneck of the system is dependent on the specific flow group definition method in use.

The small cluster sector method seems to have the best performance across the capacity, efficiency and safety metrics, especially for high and very high-density values. A potential drawback of the cluster sectors approach is derived by its dependency on the available traffic data (simulation or historical). If the data used to produce the flow groups are not indicative of the actual traffic patterns, the efficiency of the method may be diminished. In the case of highly unpredictable or significantly changing traffic patterns, the gird sectors approach might be preferred over the cluster sectors. Another proposed solution would be to apply the cluster sectors approach dynamically and allow the flow groups to adjust to the current traffic pattern.

The cluster sectors implementations, both small and large, generated more replans and attempted replans than the grid sectors implementation of the respective size. The probable reason for that is that the cluster sectors approaches created shorter flow groups in the congested city centre, where the traffic is often redistributed. That phenomenon is desirable, as it should cause replannings to avoid the heavy traffic of the city centre.

After the flow groups are defined, the computational complexity of the proposed dynamic capacity method is independent of the approach used to create the flow groups. During its operation, the flow control module has a computational complexity of $O(n)$, analogous to the number of flow groups, which is not dependent on the number of changes in the traffic values. In the work at hand, the flow control module is a central service, and the number of flow groups investigated are not expected to create any significant delays. However, if the flow control capability is instead implemented in a distributed manner across the UAVs of the system, the computational complexity of the flow control method could be a limiting factor. In this case, the number of flow groups should be tested to ensure the distributed agents do not reach their computational limits and implementations with a larger number of flow groups (e.g., the baseline implementation or the two small sectors) might be avoided. Additionally, the replanning metrics show that those implementations with a larger number of flow groups and, consequently, smaller flow group regions trigger more replan and attempted replans. In every replanning (or attempted replanning) event, the UAV uses the D* Lite algorithm to search for an improved flight plan. As a result, the future UTM designer should consider the computational complexity of the flow control method and the flight replanning method when designing the flow groups.

Capacity, efficiency, and safety metrics were used for the evaluation of the performance of the flow control implementations. The metrics of those categories can easily be used for comparison of different implementations, as their desired values are clear. Capacity, efficiency, and safety are distinct metrics, and their improvement has a direct impact on the overall performance of the system. That is not also the case for the replanning metrics, as it is more difficult to define an ideal value for them. The replanning metrics may be used to evaluate the reactiveness of the system to traffic changes. Although the preferred level of reactiveness cannot be expressed as a number, this study showed that the performance metrics provide the best results for a system with a small, but not zero, number of replans and attempted replans across all densities, which slightly increases with the density level.

7. Conclusions

This paper studies dynamic capacity balancing in UTM/ U-space and utilises a flow control module for this purpose. A flow control module for a hybrid system is presented, and three approaches for defining the flow groups used by the system are described. The different flow groups are compared to each other to identify the best-performing one. The developed system is also compared against a system with no flow control to showcase the advantages offered by its use. The system is tested in highly dense traffic in constrained, urban airspace.

Capacity balancing can be conducted via a dynamic airspace structure or in a centralised proactive way via strategic pre-flight separation management. This work attempts to study how to define a flow control system to support indirect traffic spreading. It tests a system in which a central entity gathers and processes traffic data, and the individual UAVs decide if replanning is efficient for them. Capacity balancing for UTM is an open research subject, and many aspects of it remain to be investigated. In this work, traffic is managed based on the current traffic density of predetermined flow groups. The flow control module is tested in a system including no strategic flight planning or deconfliction to stress the system performance in high traffic densities.

The comparison of the flow control implementations shows the advantage of including such a method in the UTM system and the impact of different designing methods. The baseline implementation had relatively poor results for high and very high air traffic densities, while all the other flow group definition methods proved to have a positive effect on the system capacity, efficiency, and safety across all densities. The small cluster sectors implementation was the best performing of the tested implementations. However, it should be noted that for the design of cluster sector flow groups, simulation and/or historical data are needed. The flight data are used to quantify the risk of different map areas. The map is divided into large flow groups in areas with low risk and smaller, more controllable flow groups in areas with high risk. In this work, the risk was quantified using the conflict events from previous simulation runs. The performance improvement of the system by introducing a flow control module with carefully defined flow groups has been shown. The system efficiency, safety, and capacity were ameliorated by the inclusion of most of the tested flow control implementations over a system with no flow control.

Future research should look into combining traffic density with traffic complexity measurements (e.g., attitude changes, inter-vehicle distances, and predicted conflicts metrics) to estimate dynamic traffic density, near-future traffic prediction based on the intentions of the aircraft, and the use of real-time aircraft clustering for defining flow groups dynamically based on aircraft congested areas. The future designer of such a system should heavily consider the airspace structure and flight planning algorithm used on the system, as they are deeply co-dependent. In a future UTM system, the flow control could work alongside strategic flight planning to further reduce aircraft congestion in specific regions. Furthermore, the proposed flow control module gathers the positions of all UAVs in a predefined interval. A more elaborate version of the system could also include the intentions of the aircraft so that it can estimate their positions in the next time step. That would allow the system to act in a more proactive manner.

Author Contributions: Conceptualisation, N.P., I.D., C.A.B., A.M.V., J.E., J.H., V.L. and V.K.; methodology, N.P., I.D., C.A.B., A.M.V., J.E. and V.L.; software, N.P., I.D., C.A.B. and A.M.V.; investigation, N.P., I.D., C.A.B., A.M.V. and J.E.; resources, J.E. and J.H.; data curation, C.A.B., N.P. and A.M.V.; writing—original draft preparation, N.P.; writing—review and editing, N.P., I.D., C.A.B., A.M.V., J.E., J.H., V.L. and V.K.; visualisation, N.P.; supervision, J.E., J.H., V.L. and V.K.; project administration, J.E., J.H., V.L. and V.K. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The code developed for this research is available at [41,42].

Conflicts of Interest: The authors declare no conflict of interest.The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

ATM	Air Traffic Management
CNS	Communication, Navigation, and Surveillance
DAA	Detect And Avoid
DAM	Dynamic Airspace Management
KPAs	Key Performance Areas
KPIs	Key Performance Indicators
NFZ	No-Fly Zone
PIs	Performance Indicators
UAS	Unmanned Aircraft System
UAV	Unmanned Areal Vehicle
UTM	UAS Traffic Management

References

1. Amazon Staff. Amazon Prime Air Prepares for Drone Deliveries. 2022. Available online: https://www.aboutamazon.com/news/transportation/amazon-prime-air-prepares-for-drone-deliveries (accessed on 6 December 2022).
2. Hawkins, A.J. Alphabet's Wing Kicks off Drone Delivery Service in Dallas on April 7th. 2022. Available online: https://www.theverge.com/2022/4/4/23006894/alphabet-wing-drone-delivery-dallas-kick-off (accessed on 6 December 2022).
3. Guggina, D. We're Bringing the Convenience of Drone Delivery to 4 Million U.S. Households in Partnership with DroneUp. 2022. Available online: https://corporate.walmart.com/newsroom/2022/05/24/were-bringing-the-convenience-of-drone-delivery-to-4-million-u-s-households-in-partnership-with-droneup (accessed on 6 December 2022).
4. Deloison, T.; Hannon, E.; Huber, A.; Heid, B.; Klink, C.; Sahay, R.; Wolff, C. The Future of the Last-Mile Ecosystem. *Tech. Rep.* **2020**, *1*, 1–28.
5. Bellan, R. Zipline's Drones to Deliver Medicine in Salt Lake City Area. 2022. Available online: https://techcrunch.com/2022/10/04/ziplines-drones-to-deliver-medicine-in-salt-lake-city-area/ (accessed on 6 December 2022).
6. UPS. UPS Operates First Ever U.S. Drone COVID-19 Vaccine Delivery. 2021. Available online: https://about.ups.com/ca/en/our-stories/innovation-driven/drone-covid-vaccine-deliveries.html (accessed on 6 December 2022).
7. Kopardekar, P. *Unmanned Aerial System (UAS) Traffic Management (UTM): Enabling Low-Altitude Airspace and UAS Operations*; Technical Report NASA/TM—2014–218299; Ames Research Center: Moffett Field, CA, USA, 2014.
8. SESAR Joint Undertaking. U-Space Blueprint—Brochure. 2017. Available online: https://op.europa.eu/en/publication-detail/-/publication/f8613e25-cf38-11e7-a7df-01aa75ed71a1/language-en (accessed on 13 June 2023).
9. Metropolis II Consortium. Metropolis II. 2022. Available online: https://metropolis2.eu (accessed on 8 February 2022).
10. Lim, Y.; Premlal, N.; Gardi, A.; Sabatini, R. Eulerian Optimal Control Formulation for Dynamic Morphing of Airspace Sectors. In Proceedings of the 31st Congress of the International Council of the Aeronautical Sciences, Belo Horizonte, Brazil, 9–14 September 2018.
11. Janisch, D.; Sánchez-Escalonilla, P.; Gordo, V.; Jiménez, M. UAV Collision Risk as Part of U-space Demand and Capacity Balancing. In Proceedings of the SESAR Innovation Days, Online, 7–9 December 2021.
12. Pongsakornsathien, N.; Bijjahalli, S.; Gardi, A.; Symons, A.; Xi, Y.; Sabatini, R.; Kistan, T. A Performance-Based Airspace Model for Unmanned Aircraft Systems Traffic Management. *Aerospace* **2020**, *7*, 154. [CrossRef]
13. de Oliveira, I.R.; Carlos Pinto Neto, E.; Matsumoto, T.T.; Yu, H. Decentralized Air Traffic Management for Advanced Air Mobility. In Proceedings of the 2021 Integrated Communications Navigation and Surveillance Conference (ICNS), Dulles, VA, USA, 19–23 April 2021; pp. 1–8. [CrossRef]
14. Bharadwaj, S.; Carr, S.; Neogi, N.; Topcu, U. Decentralized Control Synthesis for Air Traffic Management in Urban Air Mobility. *IEEE Trans. Control Netw. Syst.* **2021**, *8*, 598–608. [CrossRef]
15. Wang, Z.; Delahaye, D.; Farges, J.L.; Alam, S. *Adaptive Structuring of Unmanned Traffic: A UTM Concept—Complexity-Optimal Traffic Assignment for Future Urban Airspace*; Dr. Stud. Days ONERA: Toulouse, France, 2021.
16. Bulusu, V.; Sengupta, R.; Zhilong, L. Unmanned Aviation: To Be Free or Not To Be Free? In Proceedings of the 7th International Conference on Research in Air Transportation, Philadelphia, PA, USA, 20–24 June 2016.
17. Laudeman, I.V.; Shelden, S.G.; Branstrom, R.; Brasil, C.L. *Dynamic Density: An Air Traffic Management Metric;* Ames Research Center: Moffett Field, CA, USA, 1998.

18. Bilimoria, K.; Lee, H. Analysis of Aircraft Clusters to Measure Sector-Independent Airspace Congestion. In Proceedings of the AIAA 5th ATIO and 16th Lighter-Than-Air Sys Tech. and Balloon Systems Conferences, Arlington, VA, USA, 26–28 September 2005

19. SESAR Joint Undertaking. PJ19.04: Performance Framework. *Tech. Rep.* **2019**.

20. Sunil, E.; Hoekstra, J.; Ellerbroek, J.; Bussink, F.; Nieuwenhuisen, D.; Vidosavljevic, A.; Kern, S. Metropolis: Relating Airspace Structure and Capacity for Extreme Traffic Densities. In Proceedings of the ATM Seminar 2015, 11th USA/EUROPE Air Traffic Management R&D Seminar, Lisboa, Portugal, 23–26 June 2015.

21. Hoekstra, J.; Ellerbroek, J.; Sunil, E.; Maas, J. Geovectoring: Reducing Traffic Complexity to Increase the Capacity of UAV airspace. In Proceedings of the International Conference for Research in Air Transportation, Barcelona, Spain, 25–29 June 2018.

22. Primatesta, S.; Scanavino, M.; Lorenzini, A.; Polia, F.; Stabile, E.; Guglieri, G.; Rizzo, A. A Cloud-based Vehicle Collision Avoidance Strategy for Unmanned Aircraft System Traffic Management (UTM) in Urban Areas. In Proceedings of the 2020 IEEE 7th International Workshop on Metrology for AeroSpace (MetroAeroSpace), Pisa, Italy, 22–24 June 2020; pp. 309–313. [CrossRef]

23. Rios, J.L.; Homola, J.; Craven, N.; Verma, P.; Baskaran, V. *Strategic Deconfliction Performance: Results and Analysis from the NASA UTM Technical Capability Level 4 Demonstration*; Technical Report NASA/TM-20205006337; Ames Research Center: Moffett Field, CA, USA, 2020.

24. Doole, M.; Ellerbroek, J.; Hoekstra, J. Drone Delivery: Nature of Traffic Conflicts in Constrained Urban Airspace Environments. In Proceedings of the Delft International Conference on Urban Air-Mobility 2021, Delft, The Netherlands, 15–17 March 2021.

25. Hoekstra, J.M.; Ellerbroek, J. Aerial Robotics: State-based Conflict Detection and Resolution (Detect and Avoid) in High Traffic Densities and Complexities. *Curr. Robot. Rep.* **2021**, *2*, 297–307. [CrossRef]

26. Koenig, S.; Likhachev, M. D*lite. In Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, AB, Canada, 28 July–1 August 2002; pp. 476–483.

27. Koenig, S.; Likhachev, M. Improved fast replanning for robot navigation in unknown terrain. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292), Washington, DC, USA, 11–15 May 2002; Volume 1, pp. 968–975. [CrossRef]

28. Nedjati, A.; Izbirak, G.; Vizvari, B.; Arkat, J. Complete Coverage Path Planning for a Multi-UAV Response System in Post-Earthquake Assessment. *Robotics* **2016**, *5*, 26. [CrossRef]

29. Zhang, H.; Yao, Y.; Xie, K.; Fu, C.W.; Zhang, H.; Huang, H. Continuous Aerial Path Planning for 3D Urban Scene Reconstruction. *ACM Trans. Graph.* **2021**, *40*, 225. [CrossRef]

30. Galceran, E.; Carreras, M. A survey on coverage path planning for robotics. *Robot. Auton. Syst.* **2013**, *61*, 1258–1276. [CrossRef]

31. Abbadi, A.; Prenosil, V. Safe path planning using cell decomposition approximation. *Distance Learn. Simul. Commun.* **2015**, *7*, 1–6.

32. Barrado, C.; Boyero, M.; Brucculeri, L.; Ferrara, G.; Hately, A.; Hullah, P.; Martin-Marrero, D.; Pastor, E.; Rushton, A.P.; Volkert, A. U-Space Concept of Operations: A Key Enabler for Opening Airspace to Emerging Low-Altitude Operations. *Aerospace* **2020**, *7*, 24. [CrossRef]

33. Chung, J.J.; Miklić, D.; Sabattini, L.; Tumer, K.; Siegwart, R. The Impact of Agent Definitions and Interactions on Multiagent Learning for Coordination. In Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, Montreal, Canada, 13–17 May 2019; International Foundation for Autonomous Agents and Multiagent Systems: Richland, SC, USA, 2019; pp. 1752–1760.

34. Bianchin, G.; Pasqualetti, F. Routing Apps May Cause Oscillatory Congestions in Traffic Networks. In Proceedings of the 2020 59th IEEE Conference on Decision and Control (CDC), Jeju Island, Republic of Korea, 14–18 December 2020; pp. 253–260. [CrossRef]

35. Tripathy, P.; Rao, P.; Balakrishnan, K.; Malladi, T. An open-source tool to extract natural continuity and hierarchy of urban street networks. *Environ. Plan. B Urban Anal. City Sci.* **2021**, *48*, 2188–2205. [CrossRef]

36. Bennett, K.; Bradley, P.; Demiriz, A. *Constrained K-Means Clustering*; Technical Report MSR-TR-2000-65; Microsoft Research: Redmond, WA, USA, 2000.

37. Hoekstra, J.; Ellerbroek, J. BlueSky ATC Simulator Project: An Open Data and Open Source Approach. In Proceedings of the Conference: International Conference for Research on Air Transportation, Philadelphia, PA, USA, 20–24 June 2016.

38. Li, S.; Egorov, M.; Kochenderfer, M. Analysis of Fleet Management and Infrastructure Constraints in On-Demand Urban Air Mobility Operations. In Proceedings of the AIAA AVIATION 2020 FORUM, Virtual, 15–19 June 2020. [CrossRef]

39. Xue, M.; Rios, J.; Silva, J.; Zhu, Z.; Ishihara, A. Fe3: An Evaluation Tool for Low-Altitude Air Traffic Operations. In Proceedings of the 2018 Aviation Technology, Integration, and Operations Conference, Atlanta, GA, USA, 25–29 June 2018. [CrossRef]

40. Bulusu, V.; Polishchuk, V.; Sengupta, R.; Sedov, L. Capacity Estimation for Low Altitude Airspace. In Proceedings of the 17th AIAA Aviation Technology, Integration, and Operations Conference 2017, Denver, CO, USA, 5–9 June 2017. [CrossRef]

41. Decentralised Research. Available online: https://github.com/amorfinv/decentralised_research (accessed on 13 June 2023).

42. Bluesky. Available online: https://github.com/andubadea/bluesky/tree/M2-sensitivity (accessed on 13 June 2023).