Journal of Air Transport Management 132 (2026) 102953



Contents lists available at ScienceDirect

Journal of Air Transport Management

journal homepage: www.elsevier.com/locate/jairtraman



Robust 3D dynamic airspace sectorization: A multilayer graph-based approach

Tianyu Zhao *, Jose Escribano, Arnab Majumdar, Washington Yotto Ochieng

Centre for Transport Engineering and Modelling, Imperial College London, London SW7 2AZ, United Kingdom

ARTICLE INFO

Keywords:
Airspace management
Dynamic airspace sectorization
Chaos analysis
Spectral clustering
Traffic complexity

ABSTRACT

This paper presents a robust algorithm for three-dimensional dynamic airspace sectorization, introducing for the first time multilayer traffic networks in the study field. Distinct from widely used meta-heuristic algorithms, this approach delivers consistent results for the same traffic scenario, avoiding the instability of stochastic search techniques. This approach uses a graph-based model, taking the air traffic network as input, based on which we calculate traffic complexity. To quantify the complexity assigned to the network, we employ two parameters: one derives from the traffic scenarios based on flight vectors, and the other from the network topology. Using this complexity-weighted network as input, a multi-layer spectral clustering algorithm is applied to generate the desired number of communities. To achieve an ideal sector structure, we introduce a boundary refinement framework to produce smooth and tightly connected three-dimensional sectors. The performance of the proposed algorithm is validated using three Key Performance Indicators (KPIs): workload, sector flight time, and dynamic density, demonstrating its capability to generate more load-balanced sector configurations compared to both the current UK operational sectors and the widely used Voronoi diagram-based methods. The performance of the algorithm is evaluated through eight experiments under both peak and off-peak traffic conditions, including four-hour short-term and four six-hour long-term scenarios, with the number of target sectors kept consistent with the operational configuration. The reduced standard deviations and coefficients of variation of the KPIs indicate that the proposed sectorization achieves a more balanced distribution of traffic loads across sectors. This research provides Air Navigation Service Providers (ANSPs) with an automatic tool for three-dimensional airspace sectorization, enabling more balanced workload distribution while adapting to evolving air traffic flow patterns.

1. Introduction

Aviation plays a vital role in connecting international and domestic passengers and freight, propelling business, tourism, and economic growth. In 2019, global airlines operated 46.8 million scheduled commercial flights, carrying 4.5 billion passengers and contributing $3.5 trillion to the global economy (Air Transport Action Group, 2020). Although COVID-19 led to a temporary decline in flights, air traffic demand has rebounded worldwide such as the United States and Europe, where flight volumes now match pre-pandemic levels (Federal Aviation Administration, 2024; EUROCONTROL, 2022). Unfortunately, punctuality performance has deteriorated, underscored by higher rates of delays and cancellations (EUROCONTROL, 2022; US Department of Transportation, 2025). This intensifying congestion highlights a critical imbalance between growing demand and finite airspace capacity allocated, creating substantial operational challenges for ANSPs.

In order to meet this challenge, the International Civil Aviation Organization (ICAO) has identified Airspace Organization and Management (AOM) as one of the key pillars of the Global Air Traffic Management Operational Concept early. The primary goal is to optimize airspace structures to accommodate growing air traffic activities (International Civil Aviation Organization, 2005). Correspondingly, airspace management was integrated into the future air transport development plans of various countries and regions, such as the US Next Generation Air Transportation System (NextGen) and the Single European Sky Air Traffic Management Research (SESAR), target adaptable and integrated airspace arrangements to support rising traffic needs.

This can also be seen in the case of the UK, where since 2010 there has been substantial growth in the demand for commercial air transport, culminating in a peak in 2019 with more than 2.6 million commercial flights operating within UK airspace (Airspace Change Organising Group, 2022). National Air Traffic Services En route (NERL),

* Correspondence to: Exhibition Rd, South Kensington, London SW7 2AZ, United Kingdom.
E-mail address: t.zhao21@imperial.ac.uk (T. Zhao).

https://doi.org/10.1016/j.jairtraman.2025.102953
Received 13 June 2025; Received in revised form 3 November 2025; Accepted 3 December 2025
Available online 4 December 2025
0969-6997/© 2025 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

the UK en route air traffic services provider, forecasts this figure will increase nearly 30% by 2040 (Airspace Change Organising Group, 2022). The UK's airspace is among the most complex in the world. However, the core infrastructure of the system has remained largely unchanged since its initial design for over 70 years (Civil Aviation Authority, 2018). Given the traffic volumes seen in 2019 and the continuous growth trend from 2024, the outdated system is increasingly struggling to meet demand while ensuring safe and efficient operations. Consequently, the airspace change masterplan was relaunched in 2021 and this included the objective of releasing constraints on sectors' ability to safely manage traffic flow, highlighting the necessity to upgrade sectors so that controllers can safely manage rising traffic flow (Airspace Change Organising Group, 2022).

Sectors, as the minimal units through which controllers provide navigational services to flights, directly influence airspace capacity. If the workload generated by the traffic flow within a sector exceeds the threshold that controllers can manage, flight delays and cancellations are highly likely to occur. With the robust recovery and growth in flight volumes, there is an urgent need for adjustments in sector structure to accommodate the large and continuously increasing traffic. This sectorization has been a longstanding focus of research in relieving air traffic demand and capacity imbalance. By proactively modifying sector shapes and sizes, Dynamic Airspace Sectorisation (DAS) not only enhances the utilization of available airspace but also alleviates controllers' cognitive load, helping relieve capacity constraints considered in many solutions from the air traffic flow management perspective (Chen et al., 2017; Dal Sasso et al., 2018; Xu et al., 2020; Guo et al., 2022). While this research has seen a number of alternative methods employed, they increasingly rely upon stochastic search processes (Zou et al., 2016; Gerdes et al., 2018; Sergeeva et al., 2017, 2015; Wong et al., 2020; Zhang et al., 2023). However, such meta-heuristic methods often suffer from reproducibility, parameter tuning, and struggle with three-dimensional sector design. In response to this gap, our paper develops a method capable of generating three-dimensional airspace sector configurations that balance inter-sector workloads and can adjust sector structures in response to changes in traffic flow, helping to relieve capacity bottlenecks. Thus, our paper presents the following contributions:

i. It introduces a novel method for constructing FRA networks. This method can also be used to generate networks from traffic demands.
ii. It develops a unique metric based on the Largest Lyapunov Exponent (LLE) of aircraft trajectories and network connectivity features, capable of assessing the degree of chaotic behavior in traffic flows and the network's ability to propagate this chaos.
iii. For the first time, it incorporates multilayer networks and multilayer graph-cut algorithms for three-dimensional sectorization, which can produce consistent results under the same traffic scenario and overcome a poor reproducibility problem of stochastic search algorithms.

The rest of this paper is organized as follows. Section 2 categorizes existing research on DAS, discusses the advantages and disadvantages of these methods and explains the reason for employing the graph cut approach in this study. Section 3 describes how to establish a complexity-weighted UK air traffic network (ATN), including the complexity metrics and the supplement and simplification of the ATN. Section 4 introduces the algorithm for generating initial sectors based on the weighted ATN, which involves multi-layer spectral clustering and the generation of proper sector shapes. Section 5 validates and presents the algorithm's performance through experiments using historical data. Section 6 contains conclusions and discussions.

2. Literature review

Current research on DAS is classified into four categories: region-based, trajectory-based, Voronoi diagram-based, and graph-based methods. Region-based models divide the airspace into predefined geometrical blocks and generate sectors by combining different blocks. This approach was first applied to the US airspace (Yousefi and Donohue, 2004), using clustering methods to create two-dimensional sectors from hexagonal blocks. Subsequent research explored various techniques, including seed growth algorithms (Klein, 2005) and Mixed Integer Programming (MIP) (Granberg et al., 2019), to optimize sector formation and workload balancing. Later studies have extended this concept to three-dimensional sectorization (Tang et al., 2012), employing metrics such as aircraft counts and sector flight times. However, while region-based models are relatively straightforward to implement, they face several key limitations. The initial airspace segregation significantly impacts the method's effectiveness (Yousefi and Donohue, 2004; Klein, 2005; Granberg et al., 2019). These models struggle with scalability, particularly in three-dimensional scenarios, due to high computational demands (Tang et al., 2012). Moreover, the approach fails to directly account for the complex structure of air route networks.

Trajectory-based models represent a less explored approach in airspace sectorization, utilizing historical flight data to inform sector boundaries based on traffic distribution. Early work (Brinton and Pledgie, 2008) performs cluster analysis based on different aircraft state parameters in flight and generates sectors based on these cluster centers. A more comprehensive approach was proposed by Basu et al. (2009), combining binary space partitions (BSP) and pie-partition algorithms in a heuristic framework. Trajectory-based models have shown limitations in generating optimal sector shapes. Their performance is heavily dependent on the quality and representativeness of the available trajectory data. Moreover, the research in this area remains relatively sparse, indicating a need for further investigation to fully realize the potential of this approach in practical airspace management scenarios.

Voronoi diagram-based models have been extensively explored in DAS research. These models typically employ a two-step approach: initial sector generation using Voronoi diagrams, followed by boundary adjustments through meta-heuristic algorithms. This method's popularity stems from its versatility and ability to incorporate various quantitative metrics as optimization objectives. For instance, Gerdes et al. (2018) used controller communication time as a complexity metric, while Sergeeva et al. (2015) employed a combination of workload distribution, overloads, flight transfers, reentry events, and short transits. The optimization process often involves evolutionary algorithms (Gerdes et al., 2018; Sergeeva et al., 2015; Wong et al., 2020), with some research considering sector similarity to minimize drastic changes (Wong et al., 2020). Furthermore, climate impacts have also been integrated into these models by driving sector structure adjustments through climate-friendly rerouting of flight paths, with human-in-the-loop simulations validating the feasibility and practicality of this approach in managing climate-sensitive areas (Ahrenhold et al., 2023a,b).

Recent work has extended this concept to three-dimensional sectorization (Zhang et al., 2023), incorporating sector height as an additional gene in genetic algorithms. This ensures that resulting three-dimensional sectors maintain a right prism shape. However, the scalability and stability of this approach in complex airspace systems remain to be fully validated. While Voronoi diagram-based models offer flexibility through meta-heuristic optimization, they face several challenges: the outcomes depend on the initial solutions generated by Voronoi diagrams, potentially leading to local optima. Furthermore, the outputs from each iteration typically necessitate manual evaluation to ensure they meet acceptable standards, as emphasized by Sergeeva et al. (2015). Another drawback is the limited replicability and parameter tuning due to the nature of meta-heuristic algorithms, which

can yield varying results. Additionally, these models require substantial computational power, especially when dealing with complex airspace configurations.

Graph-based partitioning algorithms incorporate network structural characteristics during computations. This method has been explored through various techniques, including spectral bisection for decomposing overburdened sectors (Martinez et al., 2007), spectral clustering for partitioning weighted networks combined with heuristic algorithms to meet convexity and minimum distance constraints (Li et al., 2010; Wang et al., 2010), adaptive clustering with constraint-embedded graphs for workload balancing (Feng et al., 2023), and hybrid graph-cut algorithms integrating multiple techniques to ensure sector cell connectivity (Chen and Zhang, 2014). Beyond graph cut approaches, Lui et al. (2024) integrates graph theory with MIP by modeling airspace as a graph where nodes represent basic airspace volumes and edge weights represent traffic flows between volumes. The MIP model constructs decision variables to represent the sector affiliation between volumes. It also introduces flow and connectivity constraints to ensure that each sector is geographically continuous and that traffic flows in the same direction.

The primary advantage of graph partitioning algorithms lies in their ability to leverage network structural information, thereby avoiding the geometric constraint issues faced by other techniques. Moreover, these methods, based entirely on adjacency matrices, can produce consistent outcomes under the same conditions. However, despite these strengths, graph-based approaches have largely been confined to horizontal divisions, which have not been verified in three-dimensional applications. A key limitation of such models is scalability, as algorithm performance tends to decline with increasing network size. This is a challenge effectively addressed by metaheuristics and machine learning techniques.

Additionally, it is important to mention the exploration of machine learning algorithms in sector partitioning issues. Machine learning methods are used to predict controller workload (Wong et al., 2019), forecast trajectories (De Leege et al., 2013; Gallego et al., 2019), or directly apply to the prediction and partitioning of sector configurations (Gianazza, 2010; Zhou et al., 2023). Their advantage lies in the ability to capture complex nonlinear relationships between data distributions and actual flight conditions. However, this is limited to the range of data. Typically, the datasets used for training and validation in research are confined to specific periods, but the performance of machine learning under scenarios of disruptive, unforeseen situations and the long-term landscape of rapidly growing air traffic has not yet been validated. Furthermore, under air traffic operations where safety is the primary goal, the uninterpretability of machine learning poses a potential risk. Thus, their application still requires further validation (Graña, 2019).

Despite these limitations, recent hybrid approaches that integrate machine learning with optimization techniques have provided new directions for DAS. These approaches rely on machine learning to forecast traffic conditions or complexity, followed by optimization techniques to design sector configurations. For example, Zhou et al. (2023) proposed the AirFusion framework, which combines Temporal Fusion Transformer-based demand and capacity forecasting with graph partitioning to adapt sector boundaries in real time. Xu et al. (2024) proposed a WP-ConvLSTM model to predict spatial–temporal air traffic workload, represented by dynamic density grids. These predictions are then transformed into spatial point data and processed using constrained K-means clustering, SVM-based boundary generation, and evolutionary optimization to derive adaptive and balanced airspace sector configurations. Similarly, the enhanced DAS algorithm integrates machine learning-based workload estimation with operational constraints such as convective weather and military areas, supporting real-time reconfiguration validated through human-in-the-loop simulations at Milan ACC (Criscuolo et al., 2024). In parallel, Yang et al. (2022) combined complexity indicator-based workload prediction using XGBoost with a two-stage BSP and A*-based boundary optimization method, demonstrating how complexity modeling can be systematically coupled with heuristic optimization to balance sector workload under demand uncertainty. These studies have opened new directions by showing that, compared with conventional methods that rely on historical traffic patterns for sectorization, integrating complexity prediction models with optimization enables a representation of future traffic situations, thereby supporting more forward-looking sectorization strategies (Xu et al., 2023).

In the literature above, three key metrics are predominantly employed to quantify traffic complexity and evaluate model performance, which are also adopted in this study:

1. **Monitoring Workload, Coordination Workload, and Conflict Workload:**
    - **Monitoring Workload** refers to the amount of work generated by controllers in a sector as they monitor and provide navigational services to aircraft. This workload is represented by the number of aircraft within the sector.
    - **Coordination Workload** involves the work required for controllers to handover aircraft between sectors. This is quantified by the number of aircraft cross the sector boundary.
    - **Conflict Workload** refers to the work in resolving potential conflicts between flight pairs that are too close and at risk of collision. The number of flight pairs that are in close proximity and potentially conflicting is used to represent this workload.
2. **Sector Flight Time:** This metric measures the accumulated time aircraft spend within a sector. A longer sector flight time means that aircraft remain within that sector for a longer time, which can be due to a better alignment of the sector shape with the primary flight path or a larger sector size. This is an intuitive manifestation of monitoring workload. Regular traffic flows do not typically generate excessive complexity. Consequently, this KPI can assist us in analyzing the factors contributing to traffic complexity within a sector.
3. **Dynamic Density (DD):** Dynamic density is a classic and comprehensive metric of air traffic complexity (Laudeman et al., 1998), which consists of nine indicators, including aircraft flight heading, speed, altitude change, aircraft number, close flight pairs within 5, 10 nm, and flights predicted to be in conflicts within 25, 40, 70 nm. It provides a detailed distribution of the monitoring workload and conflict workload of controllers within a sector.

For a complex airspace system such as the UK, two-dimensional sector partitioning is clearly insufficient. Traffic in UK airspace is intensive and complicated, including numerous intersecting flight paths and varying altitude profiles. Two-dimensional sector designs are unable to account for these vertical complexities. The ideal DAS algorithm for conducting three-dimensional sectorization should satisfy the following characteristics:

i. The computation time and burden should be within reasonable limits to allow for partitioning of the large-area airspace.
ii. The algorithm should produce consistent results under the same traffic scenarios. Thus, metaheuristic algorithms that generate different solutions in each search process are not desired.
iii. The results generated by the algorithm should consider the network structure and traffic distribution, and meet geometric constraints that ensure safe operation.
iv. The algorithm should have the capability to read prior knowledge to adapt to the operational constraints and requirements of the specific airspace.

Based on the requirements above, this paper adopts a graph based model for three-dimensional sector partitioning. Graph based model calculations are entirely based on mathematical programming, providing robust solutions. Moreover, the graph model incorporates information about the network topological structure. These include the re-entry constraint, which means that flights should not enter the same sector multiple times; connectivity constraints, ensuring that sector boundaries are tightly connected without overlaps or gaps; and minimum distance constraints, ensuring that critical nodes to sector boundaries maintain a minimum safety margin, allowing controllers sufficient time to manage transfers and resolve potential conflicts. Most importantly, we introduced a multi-layer network graph partitioning for the first time, which allowed us to generate three-dimensional sectors, enabling our algorithm to adapt to complicated airspace.

3. Complexity weighted graph modeling

This section describes building a complexity-weighted network for the graph cut algorithm. This process includes three main parts: (1) construct the UK air traffic network, (2) define and quantify traffic chaos and network dissemination capabilities, and (3) combine two features into complexity and expand it to the whole network.

Due to the implementation of Free Route Airspace (FRA), where users can freely plan routes between defined entry and exit points without adhering to the conventional Air Traffic Services (ATS) route network (EUROCONTROL, 2024), it is necessary to construct an ATN based on actual flight trajectory data. This approach generalizes well to future scenarios where FRA is increasingly adopted. Through a map-matching algorithm, a network that covers the whole UK airspace is constructed. The detailed modeling process is described in Section 3.1. Subsequently, the network is refined to include only the main traffic structures to relieve the computation burden on the graph cut algorithm, details shows in Section 3.2.

The complexity indices comprises two components: one based on the flight trajectories and the other based on network topological characteristics. The detailed calculation process is described in Section 3.3. Trajectory-based complexity is represented by the Lyapunov exponent. In complex air traffic systems, the relative positions, directions, and speeds of numerous aircraft combine to create scenarios that impose a significant workload on controllers. The degree of interaction among these vectors can be depicted through chaos analysis (Delahaye and Puechmorel, 2010). A negative LLE indicates that small initial differences among aircraft trajectories converge over time, leading to trajectory aggregation and higher traffic complexity, which increases controllers' cognitive workload. Conversely, a positive LLE reflects trajectory divergence, implying reduced workload. Unlike traditional metrics such as aircraft count, LLE captures the system's sensitivity to initial conditions. Here, trajectory-based complexity is assessed through phase space reconstruction followed by LLE computation.

Additionally, the consideration of network topological characteristics stems from the design of the existing UK airspace system, which is based on the positions of ground navigation beacons. Aircraft are required to fly towards and over the physical locations of each beacon, often leading to multiple routes converging at the same place (Airspace Change Organising Group, 2022). Such network intersections naturally create complex traffic scenarios, and congestion is also more easily spread. Therefore, incorporating network topological characteristics enables the algorithm to focus on more high-complexity hubs within the ATN.

Based on the combined metrics, we can determine the traffic complexity under certain scenarios. Establishing a vector field that covers the UK airspace allows us to ascertain the traffic complexity at any point on the network. This part will be elaborated in detail in Section 3.4.

3.1. Graph modeling

In this study, we establish a multilayer air traffic network, which is constructed by stacking two-dimensional air traffic networks across different flight levels. The mathematical formulation of the multilayer network is provided in Section 4, while this section focuses on building the two-dimensional network structure. At the layer level, the two-dimensional network model $G = (V, E)$ aims to replicate the UK ATN structure, where V represents the network's airports and waypoints, and E represents the connecting air routes. It is important to note that UK airspace is composed of: the London FIR and the Scottish FIR. According to the UK FRA Deployment plan, by the end of 2024, all of the Scottish FIR EGPX and part of the London FIR EGTT will have achieved FRA implementation (NATS, 2021b). This means that the structured network will be phased out, allowing aircraft operations greater freedom. As of September 2019, FRA operation had been progressively introduced within the EGPX (NATS, 2021a). The disappearance of structured networks poses significant challenges in establishing the UK ATN. In the EGPX region, a total of 176 waypoints and 1951 potential route connections were identified. The specific steps of the map matching algorithm to construct the ATN are shown in Algorithm 1 (Zhao et al., 2024).

Algorithm 1 Trajectory Matching Algorithm

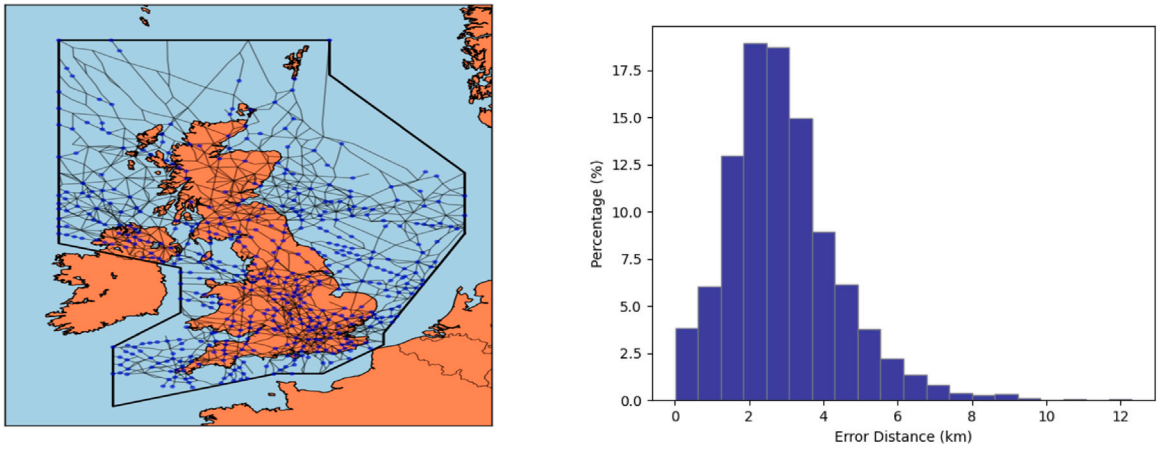
Input: ADS-B Data, Scottish FIR waypoints

1: Connect all possible routes for waypoint pairs in the Scottish FIR, as long as they are within the region. This would serve as a full connection network.
2: For each trajectory point, find all candidate routes within a certain range.
3: For each identified candidate route, compute the observation probability.
4: Repeat the above process for all the trajectory points.
5: Determine the sequence of routes with the highest probability.

Output: FRA network with only used routes

This algorithm consists of two main parts. The first is the Hidden Markov Model (HMM) (Zhao et al., 2024), utilized to match ADS-B trajectory data to the airway network to get potential results. This model employs aircraft position data (longitude and latitude) as observation variables, and the corresponding air routes as hidden variables. The HMM comprises three types of probabilities: state transition probabilities (A), observation probabilities (B), and the initial state distribution (π), each crucial for the results.

1. **Initial State Probabilities** (π): The initial state distribution provides the likelihood of selecting each candidate segment. We assume that the initial probabilities are uniformly distributed across all candidates.
2. **Observation Probabilities** (B): The observation probabilities consist of two components: the emission probability and a corrective factor. The emission probability is calculated based on a Gaussian distribution, focusing on the shortest distance from the current trajectory point to the nearest candidate segment. The corrective factor considers the angle between the aircraft's heading and the direction of the airway, adjusting the emission probability using the angle to adapt to areas with dense air routes.
3. **State Transition Probabilities** (A): State transition probabilities describe the likelihood of transitioning from one observation point's candidate segment to another. This probability is estimated by comparing the similarity between the actual distances of consecutive observation points and the shortest paths between their candidate segments, including an assessment of the trend in distance error changes over time.



(a) (b)

Fig. 1. UK ATN (a) and Trajectory matching error distribution (b).

In this study, we set the distance threshold to 5 km and the Gaussian parameter to 100 m for the observation model. Finally, using the Viterbi algorithm (Zhao et al., 2024), the most probable sequence of airway segments corresponding to the observed ADS-B flight trajectory data is determined. This algorithm employs dynamic programming to filter all results in HMM, identifying the most likely sequence of hidden states for a given sequence of observations. By starting with the highest probability of each segment, the algorithm recursively calculates the maximum probability for the next segment, with path optimality determined through backtracking. The specific steps are shown in Algorithm 2.

Algorithm 2 Viterbi Algorithm

Input: Observations $y_1, y_2, \ldots, y_T$, State space S, Transition probabilities A, Emission probabilities B, and Initial probabilities π

for each $s_i \in S$ **do**
  $Viterbi[s,1] = \pi_i \cdot b_i(y_1)$
  $Backtrack[s,1] = 0$
end for
for $t = 2, 3, ..., T$ **do**
  for each $s_j \in S$ **do**
    $Viterbi[s,t] = \max_{s' \in S}(Viterbi[s', t-1] \cdot A_{s',s}) \cdot B_s(y_t)$
    $Backtrack[s,t] = \arg\max_{s' \in S}(Viterbi[s', i-1] \cdot A_{s',s}) \cdot B_s(y_t)$
  end for
end for
for $t = T, T-1, ..., 1$ **do**
  $x_t = \arg\max_{s \in S} Viterbi[s,t]$
end for

Output: Most probable state sequence $x_1, x_2, \ldots, x_T$

The resulting UK ATN is shown in Fig. 1(a), with an average map matching error of 1.99 km, which aligns with the Strategic Lateral Offset Procedure (SLOP) tolerances, the distribution of which is in Fig. 1(b). However, such a complex route network imposes a significant computational burden on graph cut algorithm and the final sectorization refinement process.

3.2. Graph refinement

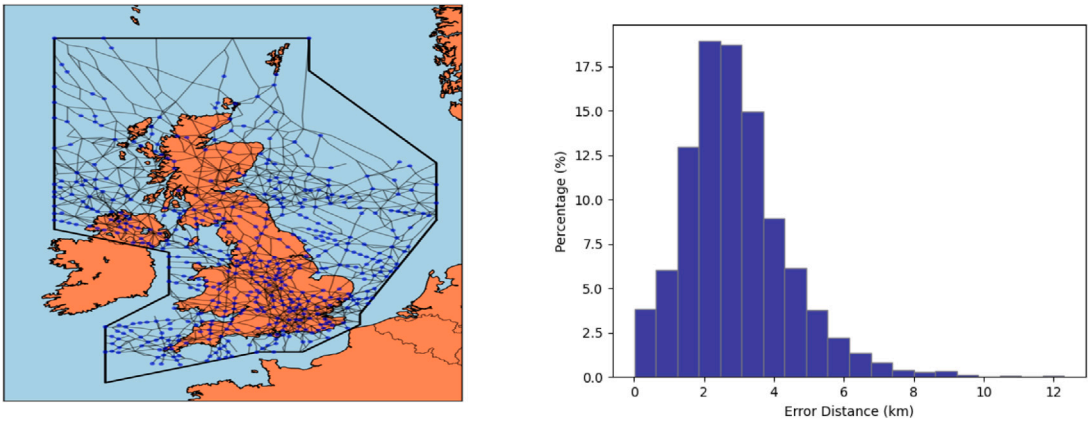
Given the initial network's complexity and the excessive distance spanned by some edges, graph cut algorithms are unsuitable. Therefore, this section introduces a novel refinement algorithm designed to preserve the fundamental structure of the network while reducing the computational burden for sectorization algorithms. Initially, we categorize waypoints based on their significance and implemented varying merge distances based on the utilization rates. The steps are:

1. Firstly, we add two types of waypoints, intersection points and midpoints added every 10 nautical miles (nm) along the routes. These points assist in preserving the fundamental structure of the network during the simplification.
2. Next, the waypoints are categorized into four types: airports, waypoints, intersection points, and midpoints, which were assigned importance levels from highest to lowest of 4, 3, 2, and 1, respectively.
3. Based on their importance, higher-importance points can merge those of lower importance, but not vice versa. The specifics are:
   (a) If $Importance_i < Importance_j$, Point j is merged into point i, whose coordinates remain unchanged, and succeeds all connections of point j.
   (b) If $Importance_i = Importance_j$, a midpoint between the two points is computed as the new vertex and points i and j are removed from the original ATN. It is worth noting that the position of the airports cannot change even if the distances are close.

In addition, when the importance levels are identical, the sequence of merging can also result in variations in the final ATN structure. To ensure that the refined network does not cause excessive drift from the original network, this paper prioritizes processing based on utilization frequency.

Therefore, we sort the usage frequency of all nodes in the ATN and rank them from highest to lowest. Waypoints with the top 50% of frequency are processed first, followed by the bottom 50%. Here, we choose to differentiate the usage frequencies based on the median because, statistically, the usage frequency of nodes in the lower 50% is mostly less than twice. Waypoints not used during the experimental dates are processed last. The merge logic adheres to the importance level rules previously described; however, points with higher frequencies are prioritized for merging, so that these points are more likely to retain their original positions, thus better preserving the network structure.

In this paper, the merge distances utilized are: 8NM for waypoints with high usage frequencies, 12NM for waypoints that have been used, and 16NM for waypoints that have not been used. Since 10NM represents the distance a commercial flight typically cruises in 1 min, we slightly adjust the merge distance to ensure a reasonable margin of error. The resulting air route network is depicted in Fig. 2(a),



(a) (b)

Fig. 2. Refined ATN (a) and Merge error distribution (b).

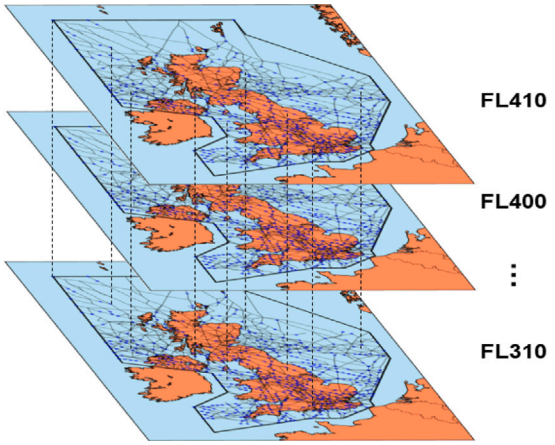


Fig. 3. Multilayer air traffic network example.

with an average map matching error denoted as 2.92 km, and its distribution displayed in Fig. 2(b). This approach not only ensures the structure while simplifying the ATN but also guarantees that the errors post-merging remain within an acceptable range.

Here, we record the route obtained after map matching for each flight and the corresponding merged route after refinement. Therefore, we define the loss here as the error in projecting the flights onto the corresponding merged routes, rather than the projection distance to the nearest graph connection. We use the traffic projection from 8 September 2019, as an example. As can be seen in Fig. 2(b), after the merge, a long tail emerges which was not presented in Fig. 1. Around 80% of the flights have an error distance within 4 km, which shows the same trends as the origin error distribution. The average new error increased less than 1 km, and less than 3% of flights have an error over 6 km, indicating that the new network manages to represent the main network structure and maintain loss within a rational range. Then this network is expanded to a multilayer network with each layer representing a flight level, as shown in Fig. 3. Although establishing the original two-dimensional network does not distinguish traffic between different altitudes, the subsequent assignment of complexity values to network edges ensures that any unused edges in a given layer will be assigned a minimal complexity value, guaranteeing that their negligible contribution does not adversely affect the overall sectorization.

3.3. Complexity metrics

The LLE quantitatively describes the exponential divergence or convergence of neighboring orbits in the phase space of a complex system, and it is one of the most important parameters for describing the chaotic dynamics of a system (Abarbanel, 2012). It provides a deep understanding of how small initial perturbations in aircraft trajectories can grow exponentially. This characteristic is central to chaotic dynamics in busy air traffic scenarios, where minor deviations can increase rapidly, adding to controllers' workload. Traditional metrics such as aircraft count do not capture this exponential divergence, so LLE is particularly suitable for demonstrating unpredictability in dense traffic flows. Trajectory-based complexity calculation needs to reconstruct the phase space and then calculate the Lyapunov exponent in the phase space.

3.3.1. Phase space reconstruction

Aircraft within a sector can be seen as moving particles, and within the traffic system composed of aircraft, the aircraft state vector field determines the complexity of this dynamical system.

A vector field $\mathbf{F}$ in an n-dimensional Euclidean space $\mathbb{R}^n$ can be expressed as follows:

$$\mathbf{F} : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n, \quad \mathbf{F}(\mathbf{x}, t) = (\mathbf{F}_1(\mathbf{x}, t), \mathbf{F}_2(\mathbf{x}, t), \ldots, \mathbf{F}_n(\mathbf{x}, t)) \tag{1}$$

Here, $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ represents trajectory points in space, t denotes time, and $\mathbf{F}_i(\mathbf{x}, t)$ is the vector of the ith component that varies with time and space. The chaotic dynamics of aircraft operation in vector fields can be studied on the time series of the flight's position. All the nonlinear time series analyses based on chaos theory are performed in phase space, so the reconstruction of the phase space is a necessary precondition for calculating the Lyapunov exponent.

Reconstructing the phase space requires determining the delay time τ and the embedding dimension m. For this purpose, the C–C method, which simultaneously determines the embedding dimension and delay time, is employed (Kim et al., 1999). This method first defines the correlation integral, then constructs a statistic $S(m, N, r, t)$ representing the correlation of the nonlinear time series. Based on the statistic, the optimal values of the embedding dimension m, the time series length N, and the neighborhood radius r are used to estimate the optimal delay τ_d and embedding window τ_w. The correlation integral is defined as follows:

$$C(m, N, r, t) = \frac{2}{M(M-1)} \sum_{1 \le i \le j \le M} \Theta(r - \left\| \mathbf{x}_i - \mathbf{x}_j \right\|), \quad r > 0,$$
$$\Theta(a) = \begin{cases} 0 & a < 0 \\ 1 & a \ge 0 \end{cases} \tag{2}$$

Here, t is the index lag, and M represents the number of data points embedded in the m-dimensional space, calculated as $M = N - (m-1)t$. The correlation integral is essentially a cumulative distribution function

that represents the probability that the distance between any two points in the phase space is less than the radius r. Based on the correlation integral, the statistic is defined as follows:

$$S(m,N,r,t) = C(m,N,r,t) - C^m(1,N,r,t) \tag{3}$$

In the calculations, the time series can be divided into t non-overlapping sub-sequences to eliminate nonlinear dependence. Consequently, we obtain:

$$\begin{aligned} \mathbf{X}_1 &= \{x_1, x_{t+1}, \ldots, x_{N-t+1}\} \\ \mathbf{X}_2 &= \{x_2, x_{t+2}, \ldots, x_{N-t+2}\} \\ &\ldots \\ \mathbf{X}_t &= \{x_t, x_{2t}, \ldots, x_N\} \end{aligned} \tag{4}$$

By calculating the statistic S for each sub-sequence and combining them, we can obtain:

$$S(m,N,r,t) = \frac{1}{t}\sum_{s=1}^{t}\left[C_s(m,\frac{N}{t},r,t) - C_s^m(1,\frac{N}{t},r,t)\right] \tag{5}$$

When $N \to \infty$, the formula (5) can be expressed as

$$S(m,r,t) = \frac{1}{t}\sum_{s=1}^{t}\left[C_s(m,r,t) - C_s^m(1,r,t)\right], \quad m = 2,3,\ldots \tag{6}$$

For a fixed m and t, for all r, if the data are independently and identically distributed when $N \to \infty$, then $S(m,r,t) = 0$. However, actual datasets may be finite and exhibit serial correlations, thus generally $S(m,r,t) \neq 0$. In such cases, the optimal delay τ is the first zero of $S(m,r,t)$ as a function of t or the time point where the differences for all radial r are minimized, as this indicates the points in the reconstructed phase space are closest to a uniform distribution. The difference is defined as:

$$\Delta S(m,t) = max\left\{S(m,r_j,t)\right\} - min\left\{S(m,r_j,t)\right\} \tag{7}$$

$\Delta S(m,t)$ measures the deviation of $S(m,r,t) \sim t$ across all radials. The local maximum time should be the zero of $S(m,r,t)$ or the minimum of $\Delta S(m,t)$. The ideal delay time corresponds to the first of these local maximum times. Based on Brock–Dechert–Scheinkman (BDS) statistics, it is found that m should be between 2 and 5, and r should be between $\sigma/2$ and 2σ. Within this range, the asymptotic distributions were well approximated by finite time series. Here, σ is the standard deviation of the statistics data. Based on the above parameters, the averages of $S(m,r_j,t)$ and $\Delta S(m,t)$ can be calculated as follows:

$$\overline{S}(t) = \frac{1}{\|m\| \cdot \|j\|}\sum_m\sum_j S(m,r_j,t) \tag{8}$$

$$\Delta\overline{S}(t) = \frac{1}{\|m\|}\sum_m \Delta S(m,t) \tag{9}$$

The first local minimum of $\Delta\overline{S}(t)$ or the first zero of $\overline{S}(t)$ corresponds to the time delay τ.

$$S_{cor}(t) = \Delta\overline{S}(t) + \left|\overline{S}(t)\right| \tag{10}$$

The optimal value for the delay window $\tau_w = (m-1)\tau$ corresponds to the time when $S_{\mathrm{cor}}(t)$ attains its minimum value. From this, we can simultaneously determine the required τ and m for the reconstruction of the phase space.

3.3.2. Largest Lyapunov exponent

To capture the intrinsic organization and operational difficulty of high-density air traffic, this study adopts the LLE as a measure of local traffic complexity. The LLE quantifies how the distance between nearby flight trajectories changes over time, thereby reflecting the order or disorder in the traffic flow. When the LLE is negative, trajectories tend to converge, indicating that aircraft are moving towards the same region or intersection. Although such convergence implies locally organized motion in a dynamical sense, it actually corresponds to a highly complex and workload-intensive situation for controllers, as aircraft are becoming closer and potential conflicts increase. Conversely, when the LLE is positive, trajectories diverge, representing traffic dispersion and reduced interaction among aircraft. In this case, the flow becomes less complex and easier to manage from an operational perspective. Therefore, in this interpretation, larger negative LLE values correspond to stronger convergence and higher operational complexity, aligning with the notion that controller workload increases when the traffic flow exhibits spatial concentration behavior.

Here, we employ a method for calculating the largest Lyapunov exponent using a small data set (Rosenstein et al., 1993). Given that the time series data used to measure the system's complexity at any given time, namely the trajectory data of aircraft, should not be extensive, our focus is on short-term traffic complexity representing the chaos degree of sample points. In the calculations, it is not necessary to compute the complete spectrum of Lyapunov exponents, observing whether the largest Lyapunov exponent is lower than zero is sufficient to determine if the system's motion is converging.

First, perform a Fast Fourier Transform on the time series to compute the average period p. For the reconstructed time series data, find the nearest point $Y_{j'}(t_0)$ to the jth point $Y_j(t_0)$ on the trajectory, which is given by

$$d_j(0) = min\left\|Y_j(t_0) - Y_{j'}(t_0)\right\|, \quad \left|j - j'\right| > p \tag{11}$$

The largest Lyapunov exponent can be represented by the average divergence rate of each trajectory point and its nearest neighbor. Therefore, the largest Lyapunov exponent can be expressed as (Sato et al., 1987)

$$\lambda(i) = \frac{1}{i\Delta t}\frac{1}{M-i}\sum_{j=1}^{M-i} ln\frac{d_j(i)}{d_j(0)} \tag{12}$$

where Δt is the sampling period of the time series, and $d_j(i)$ is the distance between the jth pair of nearest points after i discrete time steps. To quantify the divergence characteristics of the initial trajectory, the formula is as follows:

$$d_j(i) \approx C_j e^{\lambda(i\cdot\Delta t)} \tag{13}$$

where C represents the initial distance and $C_j = d_j(0)$. By taking the logarithm, equation (13) can be simplified as

$$lnd_j(i) = lnC_j + \lambda(i\cdot\Delta t), \quad j = 1,2,\ldots,N-(m-1)\tau \tag{14}$$

Finally, by fitting a regression line using the least squares method to obtain the slope, which corresponds to averaging $\ln d_j(i)$ over all j for each i, the following expression for the largest Lyapunov exponent of the time series data can be derived:

$$y_j(i) = \frac{1}{p\Delta t}\sum_{j=1}^{p} lnd_j(i) \tag{15}$$

3.3.3. Network topology characteristic weight

In addition to the complexity derived from aircraft trajectories, this paper also utilizes an intrinsic complexity based on network topological features. Due to the characteristics of air traffic, airspace capacity depends on the design of the air route network and the ability of controllers to provide safe guidance (Bauranov and Rakas, 2021). The existing UK ATN is designed based on the positions of ground-based navigation beacons. Aircraft are required to fly towards or over these beacons. Currently, there are about 40 beacons operational across the UK (Airspace Change Organising Group, 2022), making it highly possible that multiple air routes converge at the same point. Even if the same number of flights operates above these regions as in other areas, scenarios such as convergences or altitude adjustments to avoid conflicts are more likely to occur, resulting in higher complexity.

Under the existing air traffic management methods, air traffic controllers manage a large volume of traffic through vectoring, with instructions verbally transmitted to pilots via radio. Pilots read back

each instruction to verify correct reception (Airspace Change Organising Group, 2022). The more instructions issued, the more congested the radio frequency becomes. During peak times, the radio frequency can reach its limit, unable to issue or receive further instructions. Additionally, aircraft cannot stop in mid-air to wait for instructions. At these times, air traffic controllers experience a high workload and must prioritize critical flights for control, thereby increasing complexity. Therefore, when planning sectors, it is essential to consider the potential load that this part of the network might cause and provide a reasonable configuration.

This paper employs a centrality metric that does not require user-defined parameters to measure the complexity of network topology (Meng et al., 2022). This includes three main components: the H-index, k-shell iteration factor, and clustering coefficient.

For a node v in a graph $G(V, E)$, the H-index is the maximum integer h such that node v has at least h neighbors, each with a degree of at least h. This measures the influence of a node within the network by the degree of its neighbors, combining aspects of degree centrality and coreness. It considers not only the connectivity of the node itself but also the characteristics of its neighborhood. The k-shell iteration index $k_s(v)$ is determined through the k-shell decomposition method, where nodes are removed in increasing order of their degrees until all nodes with degree less than k are removed. Compared to the k-shell, the k-shell iteration factor can more accurately characterize the distinguishing ability of a node.

The clustering coefficient is calculated as follows:

$$C_v = \frac{2T_v}{d_v(d_v - 1)} \tag{16}$$

where T_v is the number of three-point clusters formed among the neighbors of node v, and d_v is the degree of node v. The clustering coefficient measures the closeness of the interconnections between the neighbors of the nodes. A higher clustering coefficient indicates that the node is more likely to form local clusters with its adjacent nodes, facilitating easier propagation of congestions.

Based on these three parameters, the network topology centrality metric for edges is presented as follows, integrating topological information such as node connectivity, neighborhood data, and coreness:

$$T_{ij} = \frac{H(i) \times KI(i)}{1 + C(i)} + \frac{H(j) \times KI(j)}{1 + C(j)} \tag{17}$$

where i and j are connected nodes, and H, KI, and C represent the topological parameters of each node, namely H-index, k-shell iteration factor, and clustering coefficient, respectively. With this, we can determine the topological complexity of each edge in the UK ATN.

A topological heatmap has been developed to visually represent the complexity across different areas, particularly before any merging of data points. This visualization prominently shows that the London area and routes to the central UK exhibit a higher level of complexity due to its dense network connections and high node centrality values. This high complexity is likely due to the significant air traffic and numerous navigation beacons concentrated in this area, which necessitate intricate air traffic control measures. Besides, the FRA routes consisting of a large number of east–west flights through the UK airspace also show a high level of complexity (see Fig. 4).

3.4. Total complexity on the traffic network

Based on the calculations from Section 3.3.3, we can determine the topological complexity and derive the trajectory complexity from flight positions. To integrate these and cover the entire UK airspace, it is necessary to introduce a method for vector interpolation and the establish a vector field.

Each aircraft's movement direction and speed at a specific moment can be considered as a vector, denoted by $V(A) = (u(A), v(A), w(A))$ where $A = (x, y, z)$. To establish the vector field, it is essential to account

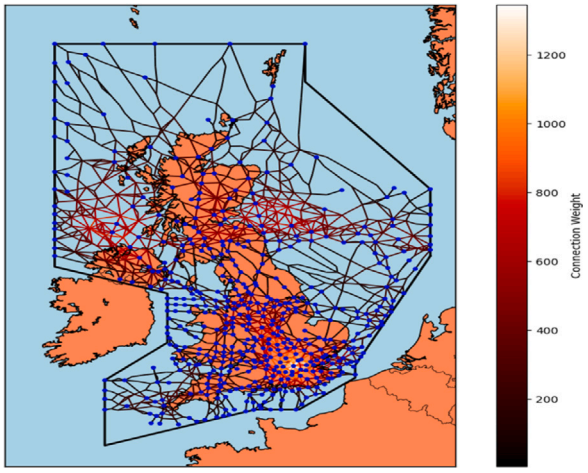


Fig. 4. UK ATN topology characteristics.

for the energy dissipation in the divergence and vorticity fields. These fields are represented as (Amodei and Benbourhim, 1991):

$$\begin{aligned} divV &= \partial_x u + \partial_y v + \partial_z w \\ vorV &= \begin{pmatrix} (vorV)_x \\ (vorV)_y \\ (vorV)_z \end{pmatrix} = \begin{pmatrix} \partial_y w - \partial_z v \\ \partial_z u - \partial_x w \\ \partial_x v - \partial_y u \end{pmatrix} \end{aligned} \tag{18}$$

In order to smooth the interpolation of the vector field, it is necessary to minimize the energy loss in both the divergence and vorticity fields, which can be represented as (Amodei and Benbourhim, 1991):

$$\begin{aligned} J_{\alpha,\beta_x,\beta_y,\beta_z}(V) = \alpha &\int_{\mathbb{R}^3} \|\nabla divV\|^2 \, dx\,dy\,dz \\ &+ \beta_x \int_{\mathbb{R}^3} \left\|\nabla (vorV)_x\right\|^2 \, dx\,dy\,dz \\ &+ \beta_y \int_{\mathbb{R}^3} \left\|\nabla (vorV)_y\right\|^2 \, dx\,dy\,dz \\ &+ \beta_z \int_{\mathbb{R}^3} \left\|\nabla (vorV)_z\right\|^2 \, dx\,dy\,dz \end{aligned} \tag{19}$$

Here, ∇ represents the three-dimensional nabla operator, and $\|\cdot\|$ denote the Euclidean norm of the $\mathbb{R}^3$ vector field. α and β are positive constants, $\alpha = \beta_x = \beta_y = \beta_z$ was chosen to balance the effects of the two fields. Consequently, we can obtain vector interpolations at any waypoints based on existing vectors and establish the vector field for the UK ATN.

Combining the two types of complexity, the traffic complexity for any edge in the UK ATN can be represented as

$$W_{ij} = \frac{H(i) \times KI(i) \times L(i)}{1 + C(i)} + \frac{H(j) \times KI(j) \times L(j)}{1 + C(j)} \tag{20}$$

However, it is important to note that the Lyapunov exponent is calculated based on the vector field at a specific moment. To represent the network edge weights over a period of time, we employ the following formula:

$$\omega_{ij} = \mu_{ij} + \eta\sigma_{ij} \tag{21}$$

where μ represents the average complexity of all sampling moments on a network edge within the specified time period, and σ denotes the standard deviation of these averages. η is user defined constants. Consequently, we obtain the network weights over a time interval. By stratifying the network according to flight levels and considering only the flight information of adjacent altitude layers, we can derive the complexity weight matrix for any specific layer of the UK ATN.

$$\mathrm{Mcut}(A)=\sum_{i\in A}\left(\frac{\sum_{\alpha\in L}\sum_{e_{uv}^{\alpha\alpha}\in E^{\alpha\alpha}}\gamma_{uv}^{\alpha\alpha}w_{uv}^{\alpha\alpha}(1-\delta_i(u^\alpha,v^\alpha))+\sum_{\alpha\in L}\sum_{\beta\in L,\beta\neq\alpha}\sum_{e_{uv}^{\alpha\beta}\in E^{\alpha\beta}}\gamma_{uv}^{\alpha\beta}w_{uv}^{\alpha\beta}(1-\delta_i(u^\alpha,v^\beta))}{\sum_{\alpha\in L}\sum_{e_{uv}^{\alpha\alpha}\in E^{\alpha\alpha}}\gamma_{uv}^{\alpha\alpha}w_{uv}^{\alpha\alpha}\delta_i(u^\alpha,v^\alpha)+\sum_{\alpha\in L}\sum_{\beta\in L,\beta\neq\alpha}\sum_{e_{uv}^{\alpha\beta}\in E^{\alpha\beta}}\gamma_{uv}^{\alpha\beta}w_{uv}^{\alpha\beta}\delta_i(u^\alpha,v^\beta)}\right) \tag{22}$$

Box I.

4. Graph partitioning

4.1. Multilayer spectral clustering

This section discusses the partitioning of the multi-layer network established in Section 3 into n disjoint subgraphs, with the goal of evenly distributing the workload among the subgraphs. Here, we utilize a Multilayer Spectral Clustering (MSC) algorithm, starting with the introduction of key parameter definitions. The multi-layer network can be represented as $M=(V_M,E_M,V,L)$, where V^α denotes the nodes in the α layer of the network, and L represents the set of network layers. All nodes V can be partitioned into k disjoint communities, as shown by $A=\{A_1,A_2,\ldots,A_k\}$, where $\bigcup_{i=1}^{k}A_i=V$, $A_i\cap A_j=\varnothing$ and $\forall i\neq j$ (Karaaslanli and Aviyente, 2019). The multi-layer spectral clustering algorithm extends the weighted minimum cut problem to multi-layer networks. The cost function of this algorithm is represented as follows (see the equation in Box I):

In the cost function, the numerator's first term calculates the cut within the same layer α, while the second term involves the cut between layer α and β. The denominator represents the weight of the ith cluster connections. The constant γ acts as a multiplier that adjusts the relative importance of intra-layer and inter-layer edges. Specifically, for intra-layer edges, $\gamma_{uv}^{\alpha\alpha}$ is where users can incorporate prior knowledge to assign higher importance to specific edges, making them more costly to cut and thus discouraging partitions from occurring along critical operational structures. In this study, $\gamma_{uv}^{\alpha\alpha}$ is set to 1, meaning that all edges within the same layer are treated equally, and only their traffic complexity weights determine their contribution to the cut cost. For inter-layer edges, $\gamma_{uv}^{\alpha\beta}$ is derived from the strength distribution of the connected nodes across layers:

$$\gamma_{uv}^{\alpha\beta}=\frac{s_d^\beta(u^\alpha)+s_d^\alpha(v^\beta)}{2},\quad \text{for } \alpha,\beta\in L \tag{23}$$

which reflects the operational coupling between flight levels. The term w is the complexity on the edge, and δ_i is an indicator that represents two points u and v connected or not. Prior knowledge can be used to determine user-defined parameters, thereby achieving the anticipated clustering structure. From Eq. (22), it is evident that both terms in the numerator correspond to the sector handover workload, while the denominator corresponds to the surveillance and control tasks within a sector. The sectorization problem, aimed at balancing sector workload and minimizing coordination efforts, is thus transformed into a graph partitioning problem where the balance of subgraph weights is maximized and the edge weights between subgraphs are minimized, aligning with our objectives. The specific steps are shown in algorithm 3.

4.2. Airspace sectorization

Based on the above clustering results, the final step is to divide them into regular shaped sectors that are disjoint and have no overlapping space. The complete process is described in the following subsections.

4.2.1. Initial sector boundary generation

To ensure that sector boundaries maintain a certain distance from intersections and air routes to avoid potential cross-sector conflicts, it is necessary to introduce two types of protection points, as shown in Fig. 5 (Li et al., 2010). One type centers around the intersection points,

Algorithm 3 Multilayer spectral clustering algorithm

Input: Multilayer traffic network M, number of communities k

1: Calculate the strength distribution of a multilayer network S
2: Construct the supra-adjacency matrix $\hat{A}$ from S
3: Degree matrix $\hat{D}=\mathrm{diag}\left(\sum_j \hat{A}_{ij}\right)$ for each i
4: Calculate the Laplace matrix $\hat{L}=\hat{D}-\hat{A}$
5: Construct the normalized Laplace matrix $\hat{\mathcal{L}}=\hat{D}^{-\frac{1}{2}}\hat{L}\hat{D}^{-\frac{1}{2}}$
6: The solution matrix can be found as $\arg\min_H \mathrm{Tr}(H^T\hat{\mathcal{L}}H)$ s.t. $H^TH=I$
7: Result $A=\mathrm{kmeans}(U^*,k)$

Output: Disjoint cluster structure A

establishing a square protective zone with the square's four vertices added to the UK ATN. The distance from the intersection point to each vertex of the square protective zone is set at $\sqrt{2}d_m$, thereby ensuring that the square encloses a circular area with a radius of d_m, adhering to the minimum distance constraint. For air routes, protection points are added on both sides of each air route point, at a distance of d_m parallel to the direction of the air route. In the partitioning of sector configurations, protection points and corresponding waypoints must remain within the same subgraph. Protection points have no links to any other points.

In generating the initial horizontal boundaries for each cluster, we considered the convex hull due to its ability to directly produce the regular shape required for sectors and avoid the re-entry problem. However, convex hulls often encompass a significant amount of extra space. In the UK ATN, the convex hulls formed by clusters frequently overlap with those of adjacent clusters. Given the restriction that protection points and waypoints must be in the same subgraph, it is challenging to remove overlaps in the areas where two convex hulls intersect, even if there are not many points. Therefore, we introduce a concave hull algorithm—α-shape. This concave hull algorithm maximizes the release of space, including only the air route points in the subgraph within the protective zones. The only concern is to avoid the re-entry problem. Thus, the α-shape polygon is applied after verifying with clusters that there is no re-entry problem; otherwise, convex hull is the only choice.

The overlap of the initial sector configurations can be minimized by using a combination of both shells. However, some overlap still remains. To address this, a heuristic algorithm is used to remove overlapping areas. The subgraphs of the sectors are first sorted by complexity, and iterations are performed in this order to check for overlaps between sectors. If two sectors overlap, the points from the higher complexity overlap area of one subgraph are moved to the lower complexity subgraph, and the protective areas are regenerated. This process is repeated until all overlapping areas are eliminated. However, in complex environments, due to the presence of protective points, iterations using this method might result in a deadlock of overlap removal. Therefore, a fast retreat method is necessary. When the above method fails to eliminate an overlapping area, the points within the overlapping area are reassigned to the subgraph with fewer points. Note that this may cause potential imbalance in sector complexity and should only be used when a non-overlapping protective zone cannot be achieved.

To generate proper sector configurations, for each cluster, the first step is to obtain its horizontal subgraphs at each height layer, and then



Fig. 5. Two types of protection points.

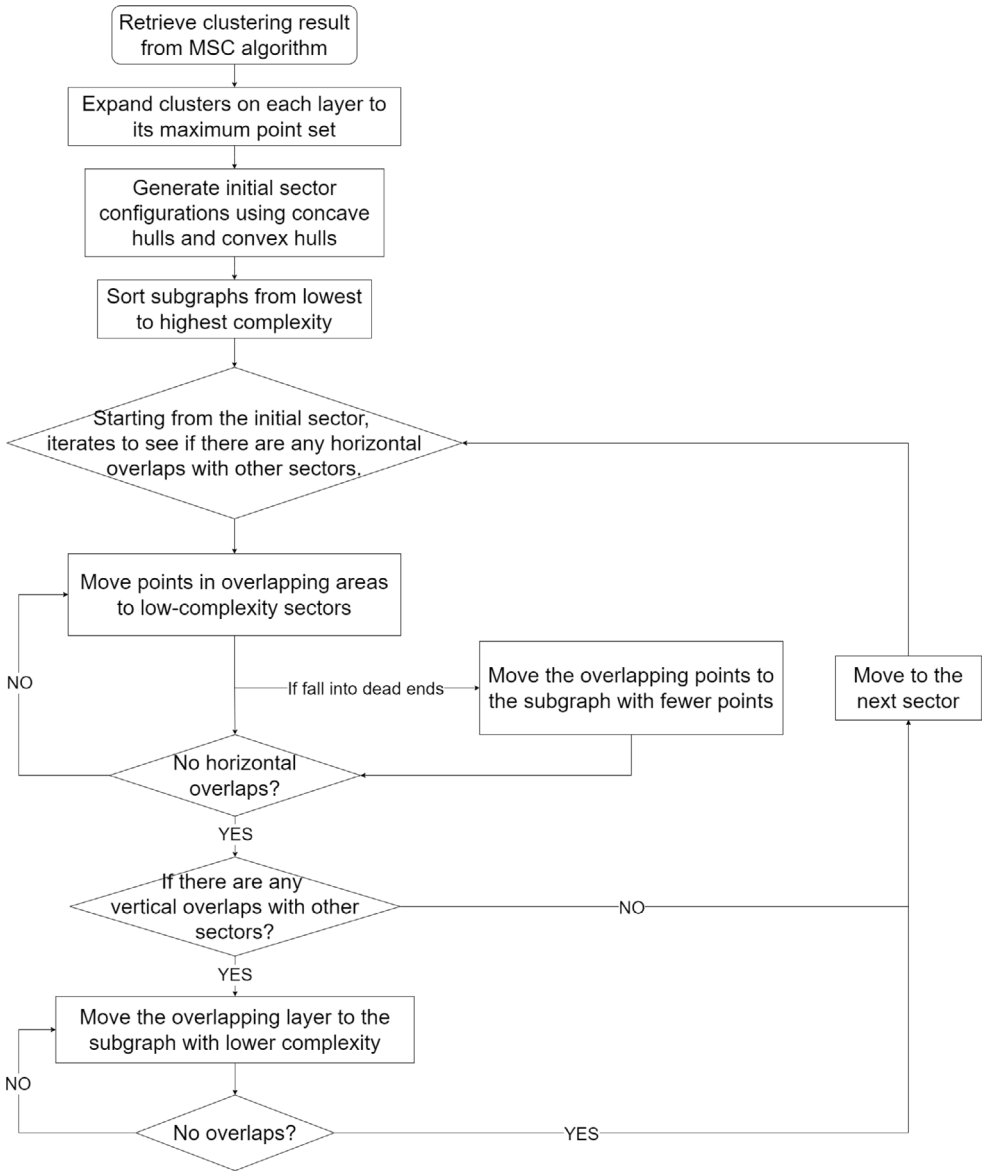


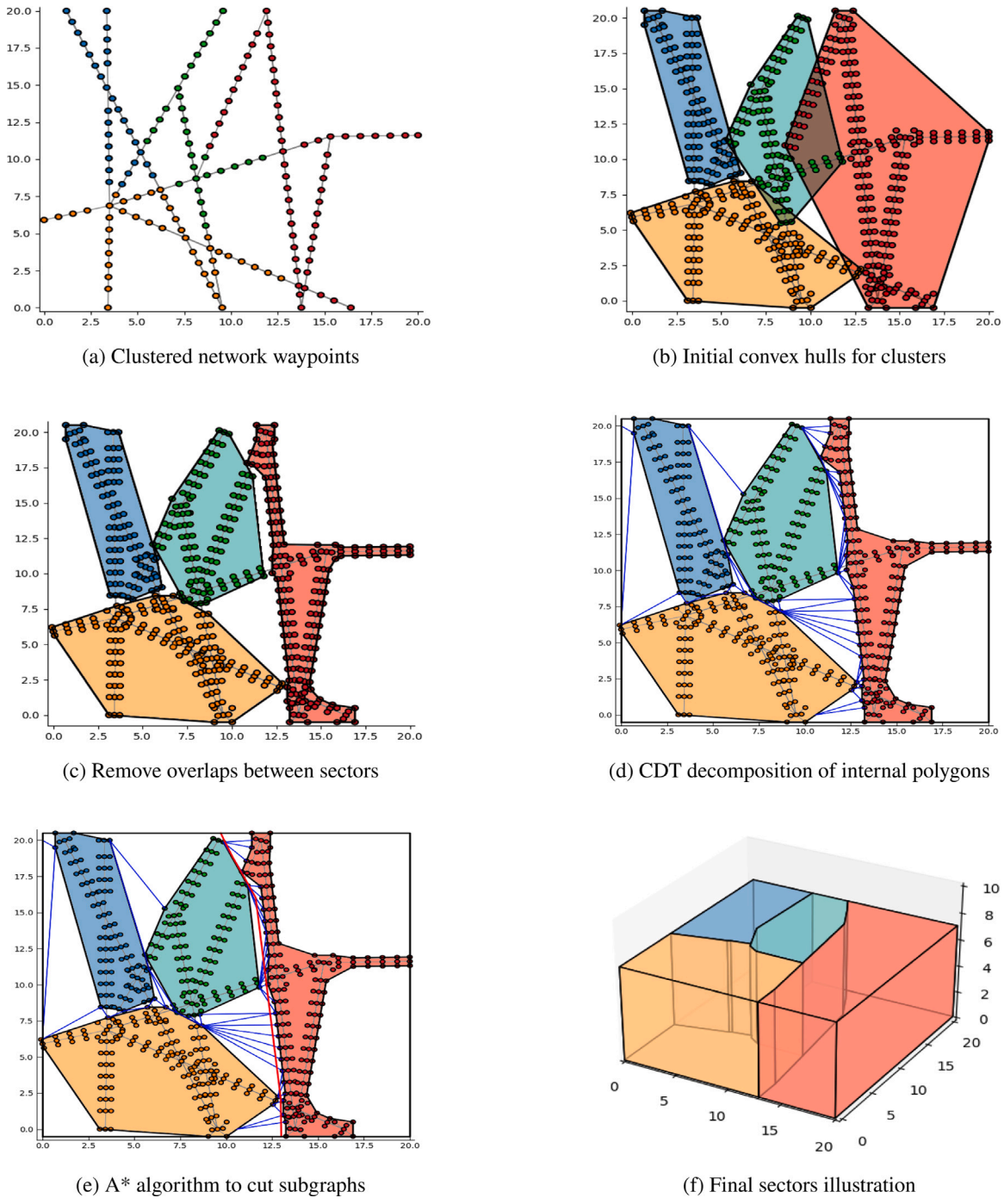
Fig. 6. The flow of removing overlaps among any sectors.

cover all the layers in which it is located with its maximum point set extension. By doing so, the task of three-dimensional sector partitioning is converted into several two-dimensional problems. The process starts with the subgraph having the least complexity, and horizontal retreats are performed until there are no overlapping horizontal sections. In terms of layer allocation, if there is any overlap between subgraphs, the complexity of the two subgraphs is assessed, and the overlapping area at that altitude layer is assigned to the sector with lower complexity. The complete retreat process is illustrated in Fig. 6.

4.2.2. Sector configuration refinement

In the previous section, we obtained non-overlapping sector boundaries. However, these subgraphs do not entirely cover the three-dimensional airspace. Therefore, we propose an airspace partitioning method that ensures each proper sector contains and only contains one type of clustering result. All sectors are interconnected, non-overlapping, and completely cover the three-dimensional airspace of the UK region.

Given that we have already achieved non-intersecting protective areas, there should be a polygon that, after removing all subgraphs, represents the structure of the remaining polygon, bounded by the two UK FIRs. To determine the primary structure of the polygon, we used Constrained Delaunay Triangle (CDT) algorithm, an extended method of Delaunay triangle that allows the inclusion of specific, constraint edges that must be preserved during the triangulation process. This feature enables CDT to capture the fundamental structure of the polygon. By linking the midpoints of triangle edges and sector vertices, we can obtain the connection graph. Based on this connectivity, we



(a) Clustered network waypoints

(b) Initial convex hulls for clusters

(c) Remove overlaps between sectors

(d) CDT decomposition of internal polygons

(e) A* algorithm to cut subgraphs

(f) Final sectors illustration

Fig. 7. Airspace sectorization algorithm steps. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

employ the A* algorithm for partitioning. The cost function of the A* search is defined as $f(n) = g(n) + h(n)$, where both the cost $g(n)$ and heuristic cost $h(n)$ are calculated using Euclidean distances within the constrained triangulated domain. The A* algorithm calculates the shortest path to partition the airspace with the objective of ensuring that only one cluster exists within the partitioned space. The area is then separated from the rest of the airspace and the process is repeated until all airspace has been completely partitioned, with each subgraph containing only one clustering result. Finally, the Douglas–Peucker algorithm is applied to smooth the boundaries if necessary. The resulting sector boundaries fully consider the clustering results and are smooth enough to completely cover all airspace.

Fig. 7 shows an example of the proper sector configuration generation process. Most of the steps here are shown in two dimensions, because the multilayer traffic network in three dimensions is complex and not as effective to visualize. Fig. 7(a) shows the initial multilayer spectral clustering algorithm results. Here all the network nodes are divided into four communities. First we tried to generate initial convex-shape sectors as shown in Fig. 7(b). But it is found that there is overlapping. In 3D space, these hulls are computed per layer and then stacked together. Fig. 7(c) shows the result of removing the overlapping areas. Here we use concave hull and reassigned waypoints in the overlappings according to the complexity of the overlapping sector pairs. Based on this result, we can identify the interior polygon

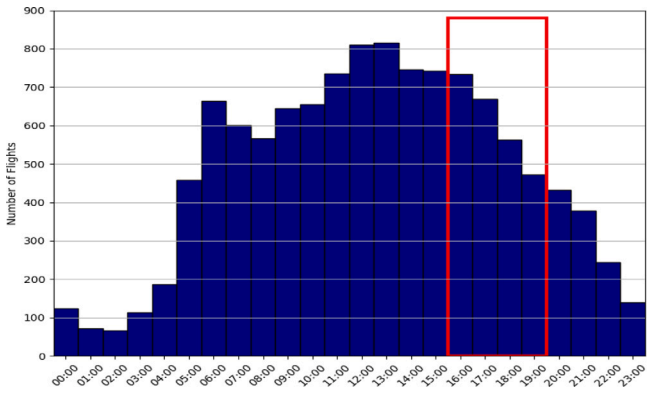


Fig. 8. UK air traffic hourly volume on the day of experiment.

with the largest area outside the initial sector removed within the square boundary and perform a CDT decomposition on it as shown in Fig. 7(d). Subsequently based on the connected graph, we use the A* algorithm to divide this polygon with the aim that there exists and only one clustering result in each subgraph. The preliminary results of the A* algorithm are shown as the red solid line in Fig. 7(e). After dividing all subgraphs, the final three-dimensional sector configuration structure is shown in Fig. 7(f). The result is a set of disjoint sectors that are connected, and meet operational shape requirements, such as right-prism shapes and a minimum distance from critical nodes as mentioned.

5. Results

In this section, we test the effectiveness of our algorithm using real flight trajectory data within the UK airspace. We obtained ADS-B data for the UK and surrounding areas for September 2019 (Schäfer et al., 2014), which include basic information such as flight speed, altitude, and heading. We stratified the network into 11 layers by flight level from FL310 to FL410. For each layer, the constructed route network consists of 1126 nodes and 2403 undirected edges. To ensure spatial safety and avoid sector-boundary conflicts, the minimum safety distances are set to 15 nm between airports and sector boundaries, 9 nm between waypoints and sector boundaries, and 3 nm between air routes and sector boundaries. The experiments were conducted on a 64-bit Linux workstation equipped with two AMD EPYC 7502 32-Core processors and 251 GB of RAM.

The airspace configuration used for comparison was derived from the Aeronautical Information Publication (AIP) and reflects the actual sector configurations on the date of the experiment. Here, we opted to quantify sector load using three KPIs described in Section 2, instead of graph weights, to verify the robustness of the algorithm.

September is one of the busiest months for air traffic in the UK. Taking the experiment on 8 September 2019 as an example, the hourly air traffic volume distribution is illustrated in Fig. 8. During peak hours, the UK airspace needs to simultaneously accommodate over 800 flights per hour. To validate the effectiveness of the algorithm, both short-term and long-term experiments were conducted. The short-term experiments are conducted hourly from 16:00–20:00, traffic volume is delineated within the red box. As time progresses, there is a gradual decline in traffic volume, which correspondingly manifests in the progressively diminishing workloads calculated across the subsequent four-hour experiments. Long-term six-hourly experiments are conducted on the same day, four times to cover all day 24 h.

Fig. 9 presents the UK's current sectors for comparison. The results in Figs. 10 and 12 show that the algorithm successfully provides sector shapes with smooth boundaries. Numbers on two-dimensional representations of sectors are used for indexing. Visually, the operational sector shapes tend to be smaller and more densely packed, particularly in the southeastern region above London area. From two-dimensional representations, this part of the airspace is fragmented into many pieces, potentially causing a significant coordination workload. The new sector structure primarily differentiates in altitude, favoring a larger horizontal extent. The following calculations shown in Table 2, 3, 4, 6, 7, and 8 represent the average values and standard deviations of three KPIs reflecting workload, with experiments conducted at one-hour intervals and six-hour intervals. Here we have highlighted standard deviations, which represent the degree of balance of KPIs between different sectors, and use the coefficient of variation (CV) to quantify this imbalance. This is defined as the ratio of the standard deviation σ to the mean μ as in Eq. (24).

$$C_v = \frac{\sigma}{\mu} \tag{24}$$

In the result tables, our approach is compared not only against the operational sector configuration but also against a modern hybrid Voronoi diagram and evolutionary optimization method (Zhang et al., 2023). To mitigate the stochasticity inherent in meta-heuristic algorithms, the evolutionary algorithm was run 10 times, with the average results presented in the tables. Specific parameters used in these experiments are outlined in Table 5. The results indicate that, in certain intervals, the Voronoi diagram-based method performs even worse than the current operational sectors. This discrepancy can be attributed to several factors:

1. **Limited performance in multi-sector airspace:** The Voronoi diagram-based method struggles in multi-sector airspace. Although the evolutionary algorithm can adjust positions of boundary vertices to align with optimization objectives, the extent of vertex movement is inherently constrained in multi-sector airspace. Large movements often result in overlapping sector coverages or invalid configurations, forcing the model to take smaller, incremental steps. Consequently, this limitation leads to minimal improvements in such complex airspace scenarios.
2. **Ineffectiveness in imbalanced airspace:** The method performance degrades in the highly imbalanced UK airspace, where traffic is heavily concentrated in the southern regions while the northern part remains relatively uncongested. Previous studies evaluated this framework in traffic balanced airspace (Gerdes et al., 2018; Sergeeva et al., 2017; Zhang et al., 2023), where sector areas are relatively uniform. In contrast, the middle and southern UK airspace are divided into small areas, and our multi-layer spectral clustering algorithm can distribute workload across different flight levels. However, the evolutionary algorithm fails to significantly reduce sector sizes in congested areas, as excessive vertex movement leads to invalid configurations.
3. **Challenges in incorporating prior information for three-dimensional sectorization:** This framework requires predefined altitude attributes to perform three-dimensional DAS. Although altitude ranges can be adjusted within the chromosome of the evolutionary algorithm, the vertical division of specific two-dimensional sectors must be determined beforehand. This presents a challenge in defining altitude attributes to achieve the desired outcomes.
4. **High computational burden:** The method demands significant computational resources and long iteration times. Across all experiments, an average of 658 min was required to complete 300 iterations, primarily due to the large number of vertices requiring optimization. A detailed runtime breakdown is provided in Table 1, which shows that most of the computational cost arises from the evolutionary search loop. This is because the complexity metrics are recalculated during each iteration of the evolutionary algorithm, whenever the sector boundaries are adjusted, leading to substantial cumulative overhead. In contrast, our multilayer spectral clustering algorithm completes the entire sectorization process in 29.82 min, with a predictable and balanced distribution of computational effort across modules.

Table 1
Runtime breakdown of MSC and Voronoi diagram based method pipelines.

Method	Module	Runtime (avg.)
MSC	Data preprocessing and network construction	0.8 min
	Complexity metric calculation	18.4 min
	Spectral clustering	9.2 min
	Sector boundary generation and refinement	1.42 min
	Total	**29.82 min**
Voronoi diagram based method	Initial clustering and Voronoi generation	0.12 min
	Vertex extraction and boundary mapping	0.63 min
	NSGA-II evolutionary search loop	656.44 min
	Cell reconstruction and final output	1.45 min
	Total	**658.64 min**

Table 2
Short-term experiment workload comparison between benchmark, our algorithm, and Voronoi-based algorithm.

Time, hrs	Benchmark			Our algorithm			Voronoi-based algorithm		
	Mean	Std	CV (%)	Mean	Std	CV (%)	Mean	Std	CV (%)
1600–1700	683.05	634.83	92.94	657.46	**334.11**	**50.82**	716.66	539.05	75.22
1700–1800	610.33	494.22	80.98	560.03	**273.37**	**48.81**	624.19	582.68	93.35
1800–1900	497.15	461.02	92.73	494.18	**306.16**	**61.95**	529.84	458.56	86.55
1900–2000	449.78	494.45	109.93	399.33	**315.25**	**78.94**	464.32	402.15	86.61

Table 3
Short-term experiment sector flight time comparison between benchmark, our algorithm, and Voronoi-based algorithm.

Time, hrs	Benchmark			Our algorithm			Voronoi-based algorithm		
	Mean	Std	CV (%)	Mean	Std	CV (%)	Mean	Std	CV (%)
1600–1700	59.26	54.35	91.67	58.72	**28.91**	**49.23**	58.71	38.96	66.36
1700–1800	48.61	44.39	91.30	47.63	**22.69**	**47.64**	47.96	44.26	92.29
1800–1900	45.59	38.12	83.60	45.30	**28.18**	**62.21**	45.35	28.61	63.09
1900–2000	43.76	43.66	99.77	42.88	30.11	70.22	42.89	**29.03**	**67.68**

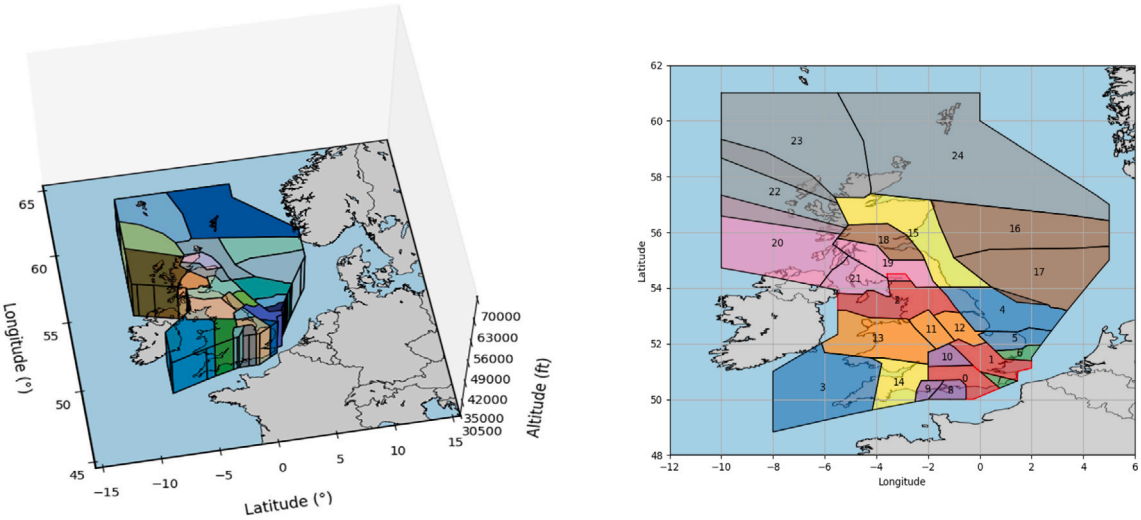


Fig. 9. Two-dimensional and three-dimensional presentation of current sectors.

These findings underscore the need for improvements to this method to better accommodate the UK airspace. Following analysis focus on comparing the performance of our algorithm with the current operational sectors.

Subsequently, we will analyze each of the three metrics in detail. Fig. 11(a) represents the total workload, including the monitoring workload, the coordination workload, and the conflict workload in both the operational and new sectors over four hours. As shown in Table 2, we can observe that the standard deviation of the workloads in the new configuration is consistently lower than that in the current structure and have fewer outliers. This is an average improvement of 34% over the current sectors and outperforms the Voronoi algorithm's 25.3%. Over the four hours, there are two area sectors consistently bearing the highest loads, even though the overall traffic volume is gradually decreasing. One of these two is located in the southwest area of the London FIR, being the largest sector above the London area. The other one covers the central part of the UK, including major airports Manchester and Liverpool. The Manchester is one of the most busiest area in the UK, making the sectors covering its airspace naturally busy (NATS, 2024). Additionally, sectors below the 25th percentile belong to the Scottish FIR, which carries less traffic compared to London, hence consistently bearing loads below the average.

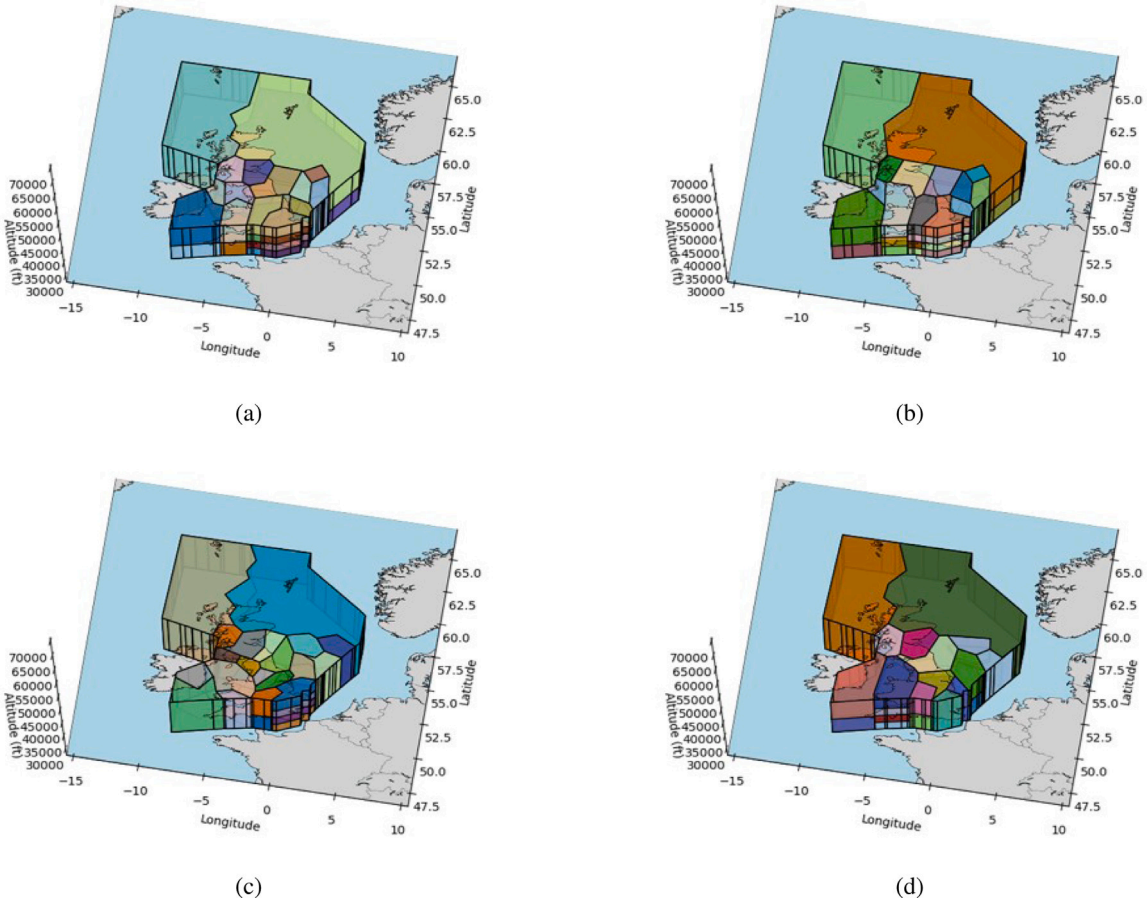


Fig. 10. Short-term new sector configurations: (a) 1600–1700 (b) 1700–1800 (c) 1800–1900 (d) 1900–2000 h, UTC.

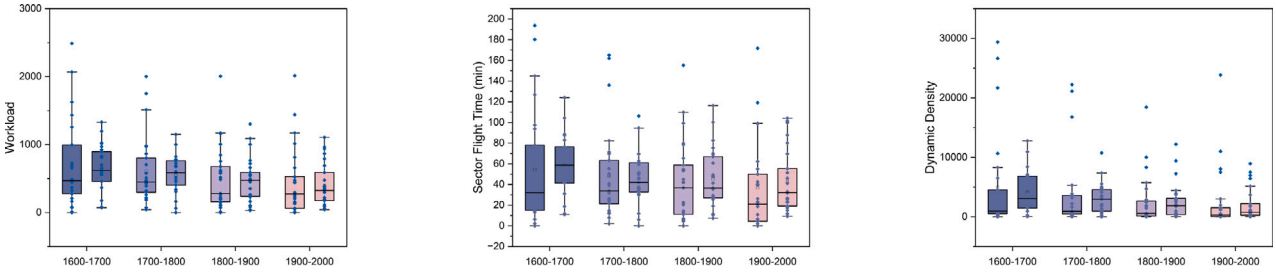


Fig. 11. KPI performances for short-term experiments: (a) workload (b) sector flight time (c) dynamic density.

Table 4
Short-term experiment dynamic density comparison between benchmark, our algorithm, and Voronoi-based algorithm.

Time, hrs	Benchmark			Our algorithm			Voronoi-based algorithm		
	Mean	Std	CV (%)	Mean	Std	CV (%)	Mean	Std	CV (%)
1600–1700	5127.28	8369.23	163.27	4151.98	**3378.87**	**81.34**	4428.19	5345.57	120.72
1700–1800	3708.40	6277.36	169.27	3186.66	**2459.26**	**77.20**	4218.29	6937.76	164.47
1800–1900	2678.90	4214.94	157.30	2613.13	**2984.33**	114.22	3065.55	3146.19	**102.63**
1900–2000	2741.60	5356.80	195.40	1999.85	**2679.58**	133.90	2344.87	2780.71	**118.59**

Table 5
Parameter settings for the Non-dominated Sorting Genetic Algorithm II (NSGA-II) algorithm.

Parameters	Generations	Population size	Crossover rate	Mutation rate	Crossover index	Mutation index
Values	300	15	100%	10%	4	10

Table 6
Long-term workload comparison between benchmark, our algorithm, and Voronoi-based algorithm.

Time, hrs	Benchmark			Our algorithm			Voronoi-based algorithm		
	Mean	Std	CV (%)	Mean	Std	CV (%)	Mean	Std	CV (%)
0000–0600	978.10	851.63	87.07	1058.14	**416.36**	**39.35**	996.18	608.49	61.08
0600–1200	3917.05	3114.90	79.52	3712.80	**1767.00**	**47.59**	4285.67	3169.68	73.96
1200–1800	4285.67	3193.15	74.50	4036.84	**2187.96**	**54.22**	4408.74	3415.59	77.47
1800–2400	2091.86	1901.56	90.91	2038.43	**1188.41**	**58.31**	2155.01	1665.80	77.30

Table 7
Long-term sector flight time comparison between benchmark, our algorithm, and Voronoi-based algorithm.

Time, hrs	Benchmark			Our algorithm			Voronoi-based algorithm		
	Mean	Std	CV (%)	Mean	Std	CV (%)	Mean	Std	CV (%)
0000–0600	120.12	122.45	101.93	120.53	**74.95**	**62.22**	120.38	106.15	88.18
0600–1200	358.99	319.60	89.01	356.82	**168.99**	**47.36**	354.68	286.27	80.71
1200–1800	341.78	272.74	79.80	340.86	**190.54**	**55.89**	350.50	233.24	66.54
1800–2400	184.53	164.70	89.27	184.62	**117.60**	**63.71**	181.93	144.93	79.66

Table 8
Long-term dynamic density comparison between benchmark, our algorithm, and Voronoi-based algorithm.

Time, hrs	Benchmark			Our algorithm			Voronoi-based algorithm		
	Mean	Std	CV (%)	Mean	Std	CV (%)	Mean	Std	CV (%)
0000–0600	4947.32	6998.98	141.44	4429.50	**3440.32**	**77.68**	5317.31	5084.52	95.62
0600–1200	37 289.25	57 877.03	155.19	30 809.59	**22 798.12**	**74.04**	35 387.93	47 738.31	134.90
1200–1800	37 761.55	54 180.96	143.45	37 877.72	**35 520.42**	**93.78**	48 040.70	51 338.43	106.86
1800–2400	11 986.88	19 800.16	165.16	11 263.71	**12 924.90**	**114.72**	12 449.42	18 678.71	150.03

In contrast, the algorithm tends to divide airspace more based on altitude layers rather than horizontal regions. For the London area, the algorithm generates large areas horizontally. This is because the air traffic network over London is the most complex in the UK, with the network's topological parameters significantly contributing to its complexity. Any cuts along the edges of such a complex network in this region result in imbalanced normalized cuts in the spectral clustering algorithm. However, partitioning by multiple altitude layers effectively reduces the overall traffic load and contributes to a balanced distribution of load between sectors. The highest load areas shown in the new sector configuration load line graphs all originate from the London area.

Both the current and new sector configurations have sectors with very low load or even zero load. This means some of the routes are infrequently used and might not have been used during the short time intervals. Therefore, although these air route points are assigned complexity during vector field calculations, they do not experience traffic within the experiments. We did not use only the traffic trajectories available during the experiment to establish the network because ADS-B trajectories are the posteriori knowledge, resulting from controller decisions. However, in real operations, controllers guide aircraft according to existing traffic and network structure. Thus, an air route without passing flights does not mean it is outside of the controllers' consideration; such routes can still potentially cause cognitive load due to latent thinking.

From the second KPI performance shown in Fig. 11(b), our algorithm is greatly optimized in terms of balancing the sector time of flight by 64.11%, outperforming the Voronoi algorithm's 15%. By locating the sectors with the longest flight time, it is found that the sectors with the highest load as indicated in the first KPI do not have the longest flight times; instead, their flight durations are closer to the average or even lower. This suggests that regular traffic flow does not lead to excessive workload. Additionally, while busy sectors have a higher number of flights, their sector flight time is not the highest. This is due to the excessive workload generated by deconfliction potential and handover flights, so sectors in busy areas are designed to have a limited volume so that sectors have a small flight time and a small monitoring load to better balance the load between sectors. This also indicates the advantage of using our complexity rather than the flight counts.

Furthermore, by comparing the average sector flight times over four hours, we observe that the average flight times in the new sector configuration are very close to, yet slightly less than, those in the current sectors. This is because the current sector structure provided by AIP has some overlaps in certain areas, whereas the new sector configuration is tightly interconnected without any overlappings. This is one of the advantages of the sector configuration derived from the algorithm's calculations.

The evaluation of DD considers more aspects than workloads, thus with a larger instability interval. DD metrics do not reflect coordinated loads, so internally busy sectors are more disparate from others. During these four experiments, the main contribute sectors to the complexity are sectors located in the London area and the middle of the UK. Sector 1 and 2, as shown in 9, are the largest sectors in London area, thus has more potential conflict pairs at different distance thresholds. Structurally, Sector 2 also has more intersections and denser network structure, leading to higher complexity. The DD in the new sector is more evenly distributed, with the peak values also belonging to different altitude ranges of the London area. However, the results obtained from the algorithm exhibit a smaller standard deviation, indicating a more balanced distribution compared to the existing sectors.

To further validate the capability and robustness of the proposed model, four six-hour long-term experiments are conducted across different times on the same day. Consistent with traffic volume distribution, the performance of each KPI was observed to be lower during the early morning and late-night periods, and higher during peak daytime hours. Table 6, 7 and 8 demonstrated the MSC algorithm's superior performance in generating more balanced sector configurations compared to both the current operational sectors and the Voronoi-based algorithm. Specifically, the MSC algorithm achieved an average improvement in balance of 33.13% for workload, 32.71% for sector flight time, and 61.26% for dynamic density relative to current operational sectors. This significantly outperforms the Voronoi algorithm, which achieved respective improvements of 22.59% for workload, 21.48% for sector flight time, and 31.80% for dynamic density.

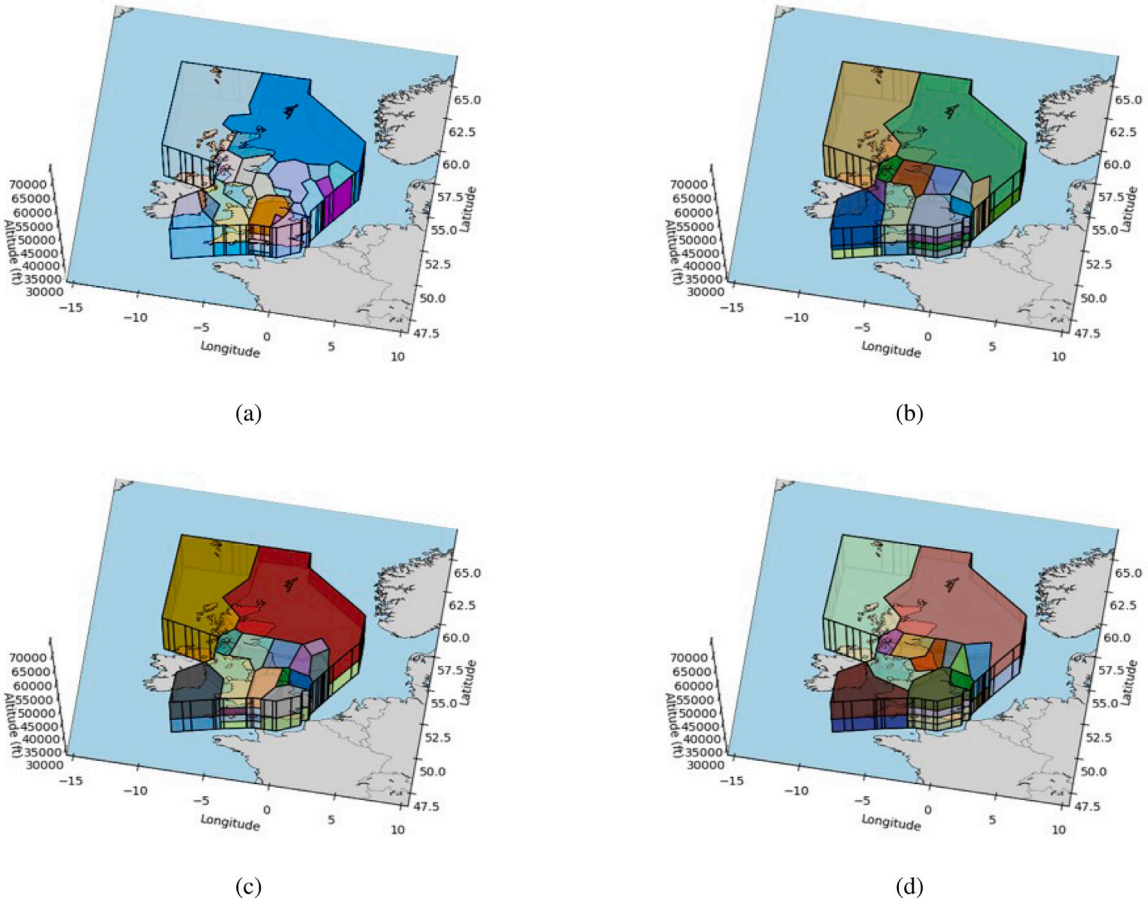


Fig. 12. Long-term new sector configurations: (a) 0000-0600 (b) 0600-1200 (c) 1200-1800 (d) 1800-0000 h, UTC.

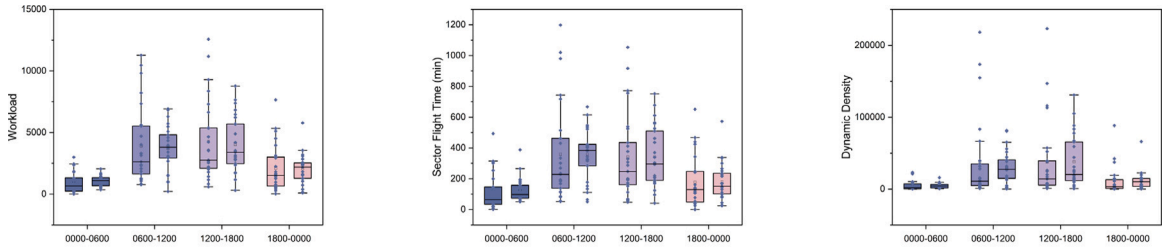


Fig. 13. KPI performances for long-term experiments: (a) workload (b) sector flight time (c) dynamic density.

The results clearly indicate that the MSC algorithm effectively adapts to different scale of fluctuations in air traffic demand throughout the day, providing robust and balanced sectorization under varying operational conditions. One point worth mentioning is that conventional graph-based methods rely on underlying route networks. While this is straightforward in most airspaces, it poses challenges in FRA where fixed route structures are not mandatory. In our framework, we can mitigate this by constructing the network from flight plans, thereby ensuring that the method remains applicable in FRA settings. As trajectory-based operations become more prevalent, the availability of high-resolution 4D trajectory data will further enhance this flexibility. By updating the network model with detailed trajectory information, the proposed algorithm can be seamlessly adapted to future operational concepts without altering its core partitioning mechanism (see Fig. 13).

6. Conclusions

The airspace over the United Kingdom is amongst the most complex in the World, but unfortunately its ageing infrastructure and accompanying procedures, is not only associated with numerous current problems, but also threatens efforts to accommodate future predicted growth.

To tackle this situation, the UK has launched an airspace and a major proposal contained within it is to re-sectorize the current airspace structure so as to achieve a more optimal demand and capacity balance than at present. While numerous studies have attempted to assess the impacts of this, they all suffer from their reliance on methods based on the stochastic search processes.

Therefore in order to overcome this barrier, in this paper we propose, a three-dimensional DAS algorithm based on a graph cut algorithm as a means to overcome these challenges. The UK airspace

network is complemented by structured networks and flight trajectories. After being filtered through the map-matching results also satisfy the minimum distance requirement from the important nodes and the main route to the sector boundary.

The proposed algorithm was tested during the busiest month, in one of the world's most complex airspaces. We were particularly careful with these tests to reflect the situation prior to the onset of the Covid-19 pandemic, which had major disruptive efforts on air traffic globally and especially in the UK. The results demonstrated that the calculated outcomes could more evenly distribute the sector complexity, particularly in the busiest areas of the southwestern and central regions of the London FIR. The algorithm effectively avoided cases where individual sectors would bear significantly higher loads. Furthermore, this network adjacency matrix-based algorithm overcomes the drawbacks of random optimization inherent in metaheuristic methods, providing stable and feasible results. In particular, these improvements were achieved without any effort for parameter tuning, highlighting just how robust the method is and its practical applicability in different traffic scenarios.

A key advantage of the proposed approach is its adaptability to future air traffic management paradigms, trajectory based operations. This concept emphasizes user-preferred trajectories and system wide optimization. The proposed model can be dynamically updated based on the trajectory information, allowing the algorithm to reflect changes in traffic patterns. In addition, the multi-layer network framework is inherently scalable. This data-driven and modular design makes the approach well-suited for long-term implementation.

A promising future direction for our approach is the incorporation of prior knowledge into the sectorization process. By integrating pre-existing constraints or preferences in the adjacency matrices for sector partitioning, the adjacency matrix to better align with specific operational requirements. Such advancements can significantly improve the algorithm's utility for any Air Navigation Service Providers in their efforts to achieve a better balance between demand and capacity than at present.

Given the results of our testing of the UK's airspace, we recommend our approach for the management and redesign of complex and dense airspace anywhere in the World.

CRediT authorship contribution statement

Tianyu Zhao: Writing – original draft, Software, Methodology, Formal analysis, Conceptualization. **Jose Escribano:** Writing – review & editing, Validation, Supervision, Project administration. **Arnab Majumdar:** Writing – review & editing, Supervision. **Washington Yotto Ochieng:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix. Sensitivity and ablation analysis

To examine how different edge weighting schemes influence the quality of the sectorization results, we carried out comparative experiments using four types of weighting. These were workload-based weighting, topology-only weighting, LLE-only weighting, and the proposed fused metric. The evaluation covered eight scenarios, including four short planning windows and four extended windows, which match the main experimental setup.

Before proceeding with the quantitative analysis, it is important to emphasize that, from a theoretical perspective, the LLE-based approach provides several distinct advantages over traditional workload metrics, which typically include monitoring, coordination, and conflict-resolution workloads.

1. First, the LLE approach does not rely on predefined conflict detection thresholds. Instead, it derives a local vector field from the observed trajectories and evaluates how they evolve over time. This enables the identification of regions where vectors begin to converge, even before any pairwise distance violations occur. Consequently, the LLE acts as a proactive geometric sensor, capable of detecting latent hotspots and signaling rising complexity prior to explicit conflict formation.
2. Second, the LLE inherently captures the dynamic influence of aircraft speed. A larger absolute value of LLE indicates faster state changes. Under convergent conditions, controllers must react more quickly and perform larger maneuvers to maintain separation. Thus, the LLE links the kinematic intensity of the flow field to the cognitive and coordination workload that conventional models cannot represent.
3. Finally, the LLE metric is fully data-driven and free from subjective weighting. It depends solely on the geometric and kinematic properties of real trajectories, ensuring that complexity assessments remain objective, reproducible, and transferable across different airspace structures and traffic densities.

The fused weighting consistently produced the lowest variability in all three key performance indicators, as shown in Table A.9. This pattern was stable across both short and extended time horizons. Its strength becomes particularly evident during periods with strong traffic flows and frequent crossing interactions, when both structural characteristics and local dynamic variations play significant roles. Under these conditions, using only structural weighting fails to capture transient convergence and divergence patterns, while using only dynamical weighting lacks the broader structural context needed to place sector boundaries effectively. The fused weighting scheme overcomes both of these shortcomings by integrating the two perspectives.

The workload-weighted baseline shows competitive behavior in specific cases in most long-term scenarios. In these situations, its results for workload and sometimes flight time are close to the best outcomes. However, its performance in dynamic density is less stable. This is because workload-based weighting reflects traffic intensity and handovers, but it does not fully represent complex flow structures. It may underestimate convergence patterns in the vector field, which affects the overall interaction level. As a result, even when sector workloads appear balanced, dynamic interactions remain uneven.

The LLE-only weighting shows different characteristics. Its performance improves noticeably in extended planning horizons, especially for dynamic density. In many long-duration scenarios, LLE-only achieves the lowest variability in dynamic density and often performs competitively for workload and flight-time balance. This is because the LLE metric captures local flow instability and highlights areas of convergence and divergence, which become more apparent when traffic patterns evolve over longer periods.

We also compared multilayer sectorization with a single-layer spectral clustering baseline, shown in Table A.10. In short planning windows and in extended intervals with sustained utilization of upper flight levels, the multilayer formulation reduces variability in all three KPIs relative to single-layer clustering, indicating that vertical structuring can redistribute managing pressure. In extended periods with low traffic density, the single-layer approach occasionally gives lower workload variability. This is expected because in lightly loaded situations, vertical separation adds granularity to already balanced sectors. Even in these cases, the multilayer formulation continues to improve flight-time balance and dynamic density, which shows that vertical structuring captures the interaction geometry more accurately than a purely horizontal view.

In summary, these results suggest a robust operational rule. The fused metric is an appropriate default across both short and extended windows because it couples structural and dynamical cues and therefore minimizes variability jointly in workload, flight time, and dynamic

Table A.9
Comparison of different edge weighting schemes (short-term hourly windows and long-term 6-hour windows).

Time	Weighting scheme	Workload (CV)	Flight time (CV)	Dynamic density (CV)
1600–1700	Workload-weighted	0.84	0.92	1.19
	Topology-only	0.74	0.67	0.92
	LLE-only	0.73	0.78	0.89
	Fused	**0.51**	**0.49**	**0.81**
1700–1800	Workload-weighted	0.75	0.78	1.37
	Topology-only	0.65	0.60	0.89
	LLE-only	0.74	0.65	0.68
	Fused	**0.49**	**0.48**	**0.77**
1800–1900	Workload-weighted	1.06	1.12	1.23
	Topology-only	0.83	0.81	1.20
	LLE-only	0.89	0.86	**1.03**
	Fused	**0.62**	**0.62**	1.14
1900–2000	Workload-weighted	0.85	0.89	1.52
	Topology-only	1.04	0.92	1.34
	LLE-only	1.00	0.89	**1.32**
	Fused	**0.79**	**0.70**	1.34
0000–0600	Workload-weighted	0.87	**0.62**	1.08
	Topology-only	0.54	0.71	1.20
	LLE-only	0.64	0.66	1.12
	Fused	**0.39**	**0.62**	**0.78**
0600–1200	Workload-weighted	0.70	0.74	1.12
	Topology-only	0.69	0.65	1.13
	LLE-only	0.77	0.63	0.97
	Fused	**0.48**	**0.47**	**0.74**
1200–1800	Workload-weighted	0.72	0.76	0.95
	Topology-only	0.73	0.79	1.15
	LLE-only	0.80	0.81	0.98
	Fused	**0.54**	**0.56**	**0.94**
1800–2400	Workload-weighted	0.65	0.70	1.30
	Topology-only	0.83	0.78	1.27
	LLE-only	**0.52**	0.65	**0.84**
	Fused	0.58	**0.64**	1.15

Table A.10
Comparison between 2D and 3D sectorization results.

Time	Model	Workload (CV)	Flight time (CV)	Dynamic density (CV)
1600–1700	2D (single-layer)	0.66	0.65	1.10
	3D (multi-layer)	**0.51**	**0.49**	**0.81**
1700–1800	2D (single-layer)	0.89	0.89	1.65
	3D (multi-layer)	**0.49**	**0.48**	**0.77**
1800–1900	2D (single-layer)	0.76	0.78	1.40
	3D (multi-layer)	**0.62**	**0.62**	**1.14**
1900–2000	2D (single-layer)	0.95	0.96	1.74
	3D (multi-layer)	**0.79**	**0.70**	**1.34**
0000–0600	2D (single-layer)	**0.71**	0.75	1.03
	3D (multi-layer)	0.87	**0.62**	**0.78**
0600–1200	2D (single-layer)	0.82	0.70	1.19
	3D (multi-layer)	**0.80**	**0.47**	**0.74**
1200–1800	2D (single-layer)	0.87	0.93	1.63
	3D (multi-layer)	**0.75**	**0.56**	**0.94**
1800–2400	2D (single-layer)	**0.85**	0.90	1.37
	3D (multi-layer)	0.91	**0.64**	**1.15**

density. By contrast, purely workload-weighted or purely structural weightings are suitable as diagnostic references but are not preferred as primary designs when interaction geometry is complicated or when flows concentrate along altitude-differentiated corridors.

Data availability

The authors do not have permission to share data.

References

Abarbanel, H., 2012. Analysis of Observed Chaotic Data. Springer Science & Business Media.

Ahrenhold, N., Gerdes, I., Mühlhausen, T., Temme, A., 2023a. A human-in-the-loop study on the dynamic sectorization concept for air traffic control in Shannon airspace. In: 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference. DASC, IEEE, pp. 1–7.

Ahrenhold, N., Gerdes, I., Mühlhausen, T., Temme, A., 2023b. Validating dynamic sectorization for air traffic control due to climate sensitive areas: Designing effective air traffic control strategies. Aerospace 10 (5), 405.

Air Transport Action Group, 2020. Aviation Benefits Beyond Borders. Technical Report, URL: https://aviationbenefits.org/media/167517/aw-oct-final-atag_abbb-2020-publication-digital.pdf.

Airspace Change Organising Group, 2022. UK airspace change masterplan.

Amodei, L., Benbourhim, M.N., 1991. A vector spline approximation. J. Approx. Theory 67 (1), 51–79.

Basu, A., Mitchell, J.S.B., Sabhnani, G., 2009. Geometric algorithms for optimal airspace design and air traffic controller workload balancing.

Bauranov, A., Rakas, J., 2021. Designing airspace for urban air mobility: A review of concepts and approaches. Prog. Aerosp. Sci. 125, 100726. http://dx.doi.org/10.1016/j.paerosci.2021.100726, URL: https://www.sciencedirect.com/science/article/pii/S0376042121000312.

Brinton, C.R., Pledgie, S., 2008. Airspace partitioning using flight clustering and computational geometry. In: 2008 IEEE/AIAA 27th Digital Avionics Systems Conference. (ISSN: 2155-7209) http://dx.doi.org/10.1109/DASC.2008.4702800, URL: https://ieeexplore.ieee.org/abstract/document/4702800. 3.B.3–1–3.B.3–10.

Chen, J., Chen, L., Sun, D., 2017. Air traffic flow management under uncertainty using chance-constrained optimization. Transp. Res. Part B: Methodol. 102, 124–141.

Chen, Y., Zhang, D., 2014. Dynamic airspace configuration method based on a weighted graph model. Chin. J. Aeronaut. 27 (4), 903–912. http://dx.doi.org/10.1016/j.cja.2014.06.009, URL: https://www.sciencedirect.com/science/article/pii/S1000936114001174.

Civil Aviation Authority, 2018. Airspace mordernisation strategy. URL: https://www.caa.co.uk/publication/download/20428. abstractTranslation: abstractTranslation:.

Criscuolo, P., Perrotta, L., Bitonto, G.D., Grani, G., 2024. Enhanced dynamic airspace configuration algorithm. Rome.

Dal Sasso, V., Fomeni, F.D., Lulli, G., Zografos, K.G., 2018. Incorporating stakeholders' priorities and preferences in 4D trajectory optimization. Transp. Res. Part B: Methodol. 117, 594–609.

De Leege, A., van Paassen, M., Mulder, M., 2013. A machine learning approach to trajectory prediction. In: AIAA Guidance, Navigation, and Control (GNC) Conference. p. 4782.

Delahaye, D., Puechmorel, S., 2010. Air traffic complexity based on dynamical systems. In: 49th IEEE Conference on Decision and Control. CDC, (ISSN: 0191-2216) pp. 2069–2074. http://dx.doi.org/10.1109/CDC.2010.5718004, URL: https://ieeexplore.ieee.org/abstract/document/5718004.

EUROCONTROL, 2022. EUROCONTROL aviation outlook 2050 main report. URL: https://www.eurocontrol.int/publication/eurocontrol-aviation-outlook-2050.

EUROCONTROL, 2024. EUROCONTROL free route airspace (FRA) (FRA) | eurocontrol. URL: https://www.eurocontrol.int/concept/free-route-airspace.

Federal Aviation Administration, 2024. FAA aerospace forecasts. URL: https://www.faa.gov/data_research/aviation/aerospace_forecasts.

Feng, K., Liu, D., Liu, Y., Liu, H., Song, H., 2023. Graphdac: A graph-analytic approach to dynamic airspace configuration. In: 2023 IEEE 24th International Conference on Information Reuse and Integration for Data Science. IRI, IEEE, pp. 235–241.

Gallego, C.E.V., Comendador, V.F.G., Carmona, M.A.A., Valdés, R.M.A., Nieto, F.J.S., Martínez, M.G., 2019. A machine learning approach to air traffic interdependency modelling and its application to trajectory prediction. Transp. Res. Part C: Emerg. Technol. 107, 356–386.

Gerdes, I., Temme, A., Schultz, M., 2018. Dynamic airspace sectorisation for flight-centric operations. Transp. Res. Part C: Emerg. Technol. 95, 460–480. http://dx.doi.org/10.1016/j.trc.2018.07.032, URL: https://www.sciencedirect.com/science/article/pii/S0968090X18310520.

Gianazza, D., 2010. Forecasting workload and airspace configuration with neural networks and tree search methods. Artificial Intelligence 174 (7–8), 530–549. http://dx.doi.org/10.1016/j.artint.2010.03.001, URL: https://linkinghub.elsevier.com/retrieve/pii/S0004370210000275.

Graña, M., 2019. Dynamic airspace configuration: A short review of computational approaches. In: Nguyen, N.T., Chbeir, R., Exposito, E., Aniorté, P., Trawiński, B. (Eds.), Computational Collective Intelligence. In: Lecture Notes in Computer Science, Springer International Publishing, Cham, pp. 486–497. http://dx.doi.org/10.1007/978-3-030-28377-3_40.

Granberg, T.A., Polishchuk, T., Polishchuk, V., Schmidt, C., 2019. Integer programming–based airspace sectorization for terminal maneuvering areas with convex sectors. J. Air Transp. 27 (4), 169–180. http://dx.doi.org/10.2514/1.D0148, URL: https://arc.aiaa.org/doi/10.2514/1.D0148.

Guo, Y., Hu, M., Zou, B., Hansen, M., Zhang, Y., Xie, H., 2022. Air traffic flow management integrating separation management and ground holding: An efficiency-equity bi-objective perspective. Transp. Res. Part B: Methodol. 155, 394–423.

International Civil Aviation Organization, 2005. Global air traffic management operational concept.

Karaaslanli, A., Aviyente, S., 2019. Strength adjusted multilayer spectral clustering. In: 2019 IEEE 29th International Workshop on Machine Learning for Signal Processing. MLSP, IEEE, Pittsburgh, PA, USA, pp. 1–6. http://dx.doi.org/10.1109/MLSP.2019.8918845, URL: https://ieeexplore.ieee.org/document/8918845/.

Kim, H.S., Eykholt, R., Salas, J.D., 1999. Nonlinear dynamics, delay times, and embedding windows. Phys. D: Nonlinear Phenom. 127 (1), 48–60. http://dx.doi.org/10.1016/S0167-2789(98)00240-1, URL: https://www.sciencedirect.com/science/article/pii/S0167278998002401.

Klein, A., 2005. An efficient method for airspace analysis and partitioning based on equalized traffic mass. In: Proceedings of the 6th USA/Europe Air Traffic Management R & D Seminar. George Mason University George, VA, USA.

Laudeman, I.V., Shelden, S.G., Branstrom, R., Brasil, C., 1998. Dynamic Density: An Air Traffic Management Metric. Technical Report.

Li, J., Wang, T., Savai, M., Hwang, I., 2010. Graph-based algorithm for dynamic airspace configuration. J. Guid. Control Dyn. 33 (4), 1082–1094. http://dx.doi.org/10.2514/1.47720, URL: https://arc.aiaa.org/doi/10.2514/1.47720.

Lui, G.N., Lulli, G., Lema-Esposto, M.F., Martinez, R.L., 2024. Airspace sector design: An optimization approach. SESAR Innovation Days.

Martinez, S., Chatterji, G., Sun, D., Bayen, A., 2007. A weighted-graph approach for dynamic airspace configuration. In: AIAA Guidance, Navigation and Control Conference and Exhibit. American Institute of Aeronautics and Astronautics, Hilton Head, South Carolina, http://dx.doi.org/10.2514/6.2007-6448, URL: https://arc.aiaa.org/doi/10.2514/6.2007-6448.

Meng, L., Xu, G., Yang, P., Tu, D., 2022. A novel potential edge weight method for identifying influential nodes in complex networks based on neighborhood and position. J. Comput. Sci. 60, 101591. http://dx.doi.org/10.1016/j.jocs.2022.101591, URL: https://linkinghub.elsevier.com/retrieve/pii/S1877750322000266.

NATS, 2021a. Free route airspace deployment 2.

NATS, 2021b. Free route airspace deployment 3&4.

NATS, 2024. Airport factsheets.

Rosenstein, M.T., Collins, J.J., De Luca, C.J., 1993. A practical method for calculating largest Lyapunov exponents from small data sets. Phys. D: Nonlinear Phenom. 65 (1–2), 117–134. http://dx.doi.org/10.1016/0167-2789(93)90009-P, URL: https://linkinghub.elsevier.com/retrieve/pii/016727899390009P.

Sato, S., Sano, M., Sawada, Y., 1987. Practical methods of measuring the generalized dimension and the largest Lyapunov exponent in high dimensional chaotic systems. Progr. Theoret. Phys. 77 (1), 1–5.

Schäfer, M., Strohmeier, M., Lenders, V., Martinovic, I., Wilhelm, M., 2014. Bringing up OpenSky: A large-scale ADS-B sensor network for research. In: IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks. pp. 83–94. http://dx.doi.org/10.1109/IPSN.2014.6846743, URL: https://ieeexplore.ieee.org/document/6846743.

Sergeeva, M., Delahaye, D., Mancel, C., Vidosavljevic, A., 2017. Dynamic airspace configuration by genetic algorithm. J. Traffic Transp. Eng. (English Edition) 4 (3), 300–314. http://dx.doi.org/10.1016/j.jtte.2017.05.002, URL: https://www.sciencedirect.com/science/article/pii/S2095756417301927.

Sergeeva, M., Delahaye, D., Zerrouki, L., Schede, N., 2015. Dynamic airspace configurations generated by evolutionary algorithms. In: 2015 IEEE/AIAA 34th Digital Avionics Systems Conference. DASC, (ISSN: 2155-7209) pp. 1–14. http://dx.doi.org/10.1109/DASC.2015.7311520.

Tang, J., Alam, S., Lokan, C., Abbass, H.A., 2012. A multi-objective approach for dynamic airspace sectorization using agent based and geometric models. Transp. Res. Part C: Emerg. Technol. 21 (1), 89–121. http://dx.doi.org/10.1016/j.trc.2011.08.008, URL: https://www.sciencedirect.com/science/article/pii/S0968090X11001185.

US Department of Transportation, 2025. Air travel consumer report.

Wang, T., Li, J., Hwang, I., 2010. A sectorization method for dynamic airspace configuration. In: AIAA Guidance, Navigation, and Control Conference. American Institute of Aeronautics and Astronautics, Toronto, Ontario, Canada, http://dx.doi.org/10.2514/6.2010-8292, URL: https://arc.aiaa.org/doi/10.2514/6.2010-8292.

Wong, C.S.Y., Sundaram, S., Sundararajan, N., 2019. CDAS: A cognitive decision-making architecture for dynamic airspace sectorization for efficient operations. IEEE Trans. Intell. Transp. Syst. 20 (5), 1659–1668. http://dx.doi.org/10.1109/TITS.2018.2833151, Conference Name: IEEE Transactions on Intelligent Transportation Systems.

Wong, C.S.Y., Suresh, S., Sundararajan, N., 2020. A rolling horizon optimization approach for dynamic airspace sectorization. IFAC J. Syst. Control. 11, 100076. http://dx.doi.org/10.1016/j.ifacsc.2020.100076, URL: https://www.sciencedirect.com/science/article/pii/S2468601820300031.

Xu, Y., Dalmau, R., Melgosa, M., Montlaur, A., Prats, X., 2020. A framework for collaborative air traffic flow management minimizing costs for airspace users: Enabling trajectory options and flexible pre-tactical delay management. Transp. Res. Part B: Methodol. 134, 229–255.

Xu, Q., Pang, Y., Liu, Y., 2023. Air traffic density prediction using Bayesian ensemble graph attention network (BEGAN). Transp. Res. Part C: Emerg. Technol. 153, 104225.

Xu, Q., Pang, Y., Liu, Y., 2024. Dynamic airspace sectorization with machine learning enhanced workload prediction and clustering. J. Air Transp. Manag. 121, 102683.

Yang, L., Huang, J., Gao, Q., Zhou, Y., Hu, M., Xie, H., 2022. Dynamic boundary optimization of free route airspace sectors. Aerospace 9 (12), 832.

Yousefi, A., Donohue, G., 2004. Temporal and spatial distribution of airspace complexity for air traffic controller workload-based sectorization. In: AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum. American Institute of Aeronautics and Astronautics, Chicago, Illinois, http://dx.doi.org/10.2514/6.2004-6455, URL: https://arc.aiaa.org/doi/10.2514/6.2004-6455.

Zhang, W., Hu, M., Yin, J., Li, H., Du, J., 2023. Multi-objective 3D airspace sectorization problem using NSGA-II with prior knowledge and external archive. Aerospace 10 (3), 216. http://dx.doi.org/10.3390/aerospace10030216, URL: https://www.mdpi.com/2226-4310/10/3/216.

Zhao, T., Escribano-Macias, J., Zhang, M., Fu, S., Feng, Y., Elhajj, M., Majumdar, A., Angeloudis, P., Ochieng, W.Y., 2024. Enhancing air traffic operational efficiency by reducing network scale. Aerosp. Traffic Saf..

Zhou, W., Pham, D.T., Alam, S., 2023. Airfusion: A machine learning framework for balancing air traffic demand and airspace capacity through dynamic airspace sectorization. In: 2023 IEEE 26th International Conference on Intelligent Transportation Systems. ITSC, IEEE, pp. 5324–5331.

Zou, X., Cheng, P., An, B., Song, J., 2016. Sectorization and configuration transition in airspace design. Math. Probl. Eng. 2016, e6048326. http://dx.doi.org/10.1155/2016/6048326, URL: https://www.hindawi.com/journals/mpe/2016/6048326/.