SMARTS Solution for Airspace Design and Configuration

A. Blanch Romero[1], T. De Urrengoechea [2], D. Delahaye[3], A. Guitart[3], M.F. Lema-Esposto[2], G.N. Lui[4], G. Lulli[4], R. Sanaei[5], J. Solzer[5]

[1] ENAIRE - Madrid, Spain
[2] Centro de Referencia I+D+i en ATM (CRIDA) - Madrid, Spain
[3] OPTIM lab, ENAC - Toulouse, France
[4]Lancaster University Management School - Lancaster, United Kingdom
[5] Institute of Air Transport, DLR - Hamburg, Germany

***Abstract*—This paper presents the SMARTS solution for airspace design and dynamic configuration in European air traffic management. The approach addresses the critical need for increased airspace capacity through intelligent sector management using artificial intelligence and optimization algorithms. The SMARTS solution comprises three main components: Basic Volume Design; Sector Design employing mixed integer linear programming for optimal sector boundaries; and Dynamic Configuration using graph-based optimization for near real-time configuration planning. Validation was conducted at Madrid Area Control Center (ACC) using 2023 traffic data. The results show that the SMARTS solution successfully balances air traffic controller workload, improves capacity utilization, and introduces a local resilience KPI for monitoring robustness toward a reliable framework for future European airspace architecture. The system delivers operationally viable configurations that align with real-world air traffic management needs while maintaining flexibility for various operational scenarios.**

***Keywords*—airspace design; dynamic airspace configuration; optimization; resilience; machine learning; validation**

I. Introduction

Dynamic Airspace Configurations (DAC) is at the core of the current and future European air traffic system, as it is the medium to increase airspace capacity. The SESAR Concept of Operations recommends DAC as the first means to resolve demand capacity imbalances, and it is also envisioned as a major cornerstone of the future architecture of the European airspace described in the Airspace Architecture Study. Enabling additional airspace capacity is a key factor to address the significant capacity challenges already faced in the recent past and to cope with the (expected) significant growth in air traffic, while maintaining safety, improving flight efficiency and reducing environmental impact. In line with the strategic vision provided by the European ATM Master Plan and the SESAR Strategic Research and Innovation Agenda, the main objective of the SMARTS research project is delivering the right amount of capacity, at the right moment and with the maximum efficiency to better serve the air traffic demand [1]. More specifically, it aims to make the airspace design and configuration process more efficient taking full advantage of the airspace potential.

Corresponding author: G. Lulli, g.lulli@lancaster.ac.uk

The ambition is to identify sectors and sector configurations that use the Air Traffic Control (ATC) available resources – i.e., Air Traffic Controllers (ATCo) – efficiently. In other words, sectors and sector configurations should ensure that Air Traffic Controllers can handle the associated workload comfortably. However, given the varying characteristics of the air traffic flows, there may not be a single sector configuration that will fulfill the desired objective over all the time. The key component of the proposed solution is the smart sectors. Smart in the sense that sectors are aware of the environment (traffic and complexity prediction, capacity estimation, impact on other sectors), can act and adapt to improve the environment (create a sector design that produces a desired outcome in terms of workload/complexity), and can communicate with relevant actors (both local and network nodes). The smart sector – which is engendered by the design of basic volumes, i.e., the elementary building blocks of the smart sector – enables the improvement of the overall dynamic airspace configuration process and the identification of better capacity actions. More in particular, smart sectors provide the basis for an optimal distribution of workload, tailored around specific safety and operational requirements including complexity. As a by-product, the application of cost-efficient capacity actions allows a more accurate DCB planning in the early INAP phases thus reducing the number of required demand measures.

Compared to current practice on airspace configuration, which is largely based on the experience of operators and heuristic methods, the proposed approach is underpinned by models and algorithms that fall in the broad area of artificial intelligence and analytics. With respect to models and algorithms for sectorization and dynamic airspace configuration recently developed, the proposed approach not only includes the option of creating new sectors but also provides the capability to identify the best-fit operational sector configuration.

The paper is structured as follows: Section II describes in details the SMARTS' approach. Section III presents the experiment setup for the validation exercise, and the data

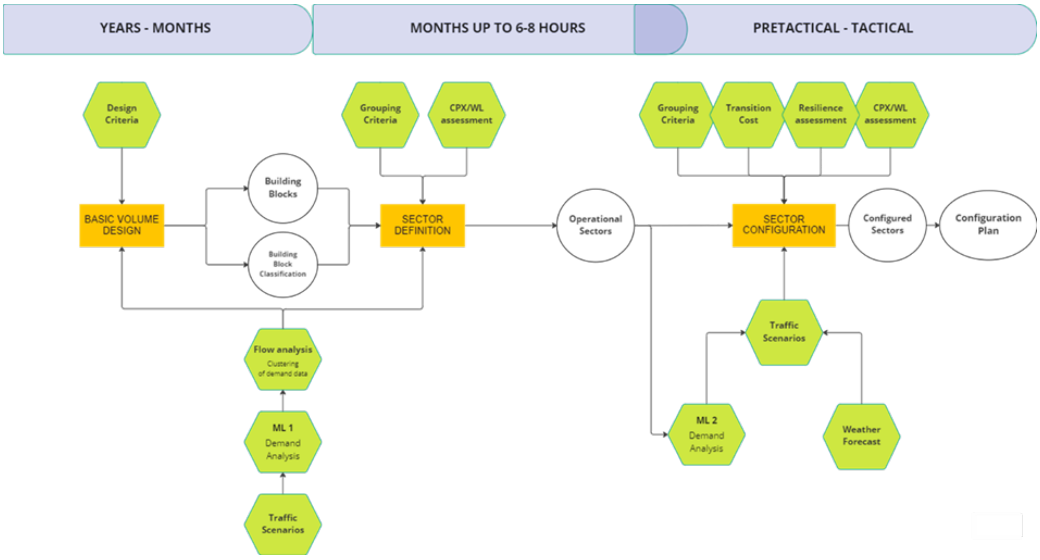


Figure 1: Architecture of the SMARTS solution.

description. Section IV delivers the results and discussion while Section V provides the concluding remarks.

II. SMARTS SOLUTION

The SMARTS approach is schematically depicted in Figure 1. The diagram illustrates the workflow of the dynamic airspace configuration process, highlighting its three main components: Basic Volume Design, Sector Definition, and Sector Configurations and Configuration Plan.

All the DAC components require input data from several sources including prediction models for air traffic flows and air traffic demand. These models provide relevant input for both the basic volume design (e.g., traffic scenarios for flow analysis) and the sector configuration plan definition (e.g., traffic demand prediction). The following subsections provide a detailed description of the methodology applied to each component of the dynamic airspace configuration process, as well as the demand analysis and prediction component.

A. Basic volume design

The first module of the SMARTS solution is the basic volume design. The goal of this module is to generate a set of blocks that will be used to design the sectors. Two types of blocks are considered:

- **airspace block** (AB), i.e., a primary volume of airspace which has to be configured to build workable sectors of control defined as configured sectors in this concept (CS).
- **shareable airspace block** (SAB), i.e., non-workable volume of airspace that can be dynamically configured (attached) in a pre-defined way to any adjacent Airspace Block (AB) to build Configured Sectors (CS). These blocks are smaller than the ABs and have lower traffic volumes.

The generation of the blocks are based on Simulated Annealing [2] and Voronoi diagram [3]. A first version of the proposed approach considering only ABs is described in [4].

1) Complexity metric: To design the blocks we use the history of the flight trajectories. From these trajectories, we calculate convergence zones, represented by high complexity. The blocks must be balanced in complexity and centred around these points of convergence. In this study, we consider that ABs have a higher complexity than SABs. In SMARTS, we decided to use the metric developed in [5]. This metric evaluates the total conflict duration for heading maneuvers between $-\Psi$ and Ψ, integrating all possible positions of flights i and j around the reference with a constant speed and a continuous time uncertainty:

$$w_p = \int_{-\Psi}^{\Psi} \mathcal{T}_{ij}(\psi)\, d\psi, \tag{1}$$

where $\mathcal{T}_{ij}(\psi)$ is the integral for a specific heading change and equals:

$$\mathcal{T}_{ij}(\psi) = kT(z(\overline{u})) - kT(z(\underline{u})),$$

where k is the conflict duration's ellipse area divided by π, $z(\underline{u})$ (resp. $z(\overline{u})$) is linked to the minimal (resp. maximal) time uncertainty and:

$$T(z) = \begin{cases} 0 & \text{if } z \leq -1 \\ \frac{2+z^2}{3}\sqrt{1-z^2} + z\frac{\pi}{2} + z\arcsin(z) & \text{if } z \in (-1,1) \\ \pi z & \text{if } z \geq 1 \end{cases}$$

2) Mathematical model: To compute the airspace blocks, we discretize the space using a regular grid, which serves two purposes: it defines the positions of the block centers and stores the corresponding complexity values.

Once the grid is defined, all 4D trajectory points are projected onto it. For each cell (i,j), we assign a complexity value $w_{i,j}$ equal to the sum of the complexities w_p of all trajectory points located within that cell:

$$w_{i,j} = \sum_{p \in \chi_{i,j}} w_p, \tag{2}$$

where $\chi_{i,j}$ denotes the spatial extent of cell (i,j):

$$\chi_{i,j} = \{(x,y) \,|\, x_i \leq x < x_{i+1}, y_i \leq y < y_{i+1}\}, \quad (3)$$

with (x_i, y_i) representing the coordinates of the bottom-left corner of the cell. Figure 2 illustrates an example in which three trajectories intersect a grid. The trajectories are discretized every ten seconds so that interactions between aircraft can be evaluated at regular intervals within each grid cell. This temporal resolution offers a good balance between computational tractability and the ability to capture interactions occurring in close succession; such events therefore produce high complexity values in the corresponding cells.

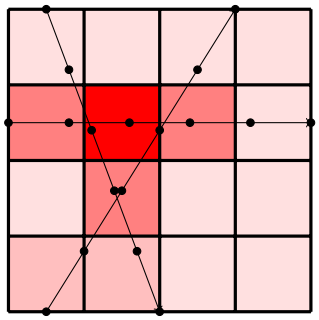


Figure 2: Example of a complexity grid: the redder the cell, the higher its complexity.

In this model, the total number of blocks, the number of ABs (n^{AB}) and the number of SABs (n^{SAB}) are considered constant. The only decision variables are therefore the positions of the centers:

$$P = \left\{p_b = (x_b, y_b), (x_b, y_b) \in \mathbb{N}^2, b \in \left\{1, \ldots, n^{AB} + n^{SAB}\right\}\right\}. \quad (4)$$

A solution is represented by a list of cell identifying numbers in the grid. The first n^{AB} elements are the centres of ABs and the next n^{SAB} are the centres of SABs. An example of solution is given in Figure 3.

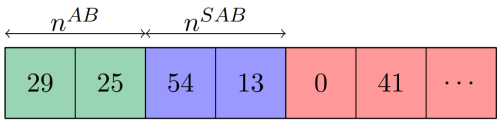


Figure 3: Example of a solution with two ABs (29 and 25) and two SABs (54,13)

The objective function is composed of three different functions. The first one represents the imbalance of complexity between blocks and is defined as follows:

$$f^b = \sum_{b=1}^{n^{AB}+n^{SAB}} \frac{|w_b - W_e|}{W_e}, \quad (5)$$

where W_e is the average value of the complexity. The second objective is to minimize the maximum complexity of blocks. This function is defined as follows:

$$f^c = \max_{b \in \{1,\ldots,n^{AB}+n^{SAB}\}} \frac{w_b \left(n^{AB} + n^{SAB}\right)^2}{W} \quad (6)$$

The difference with the model presented in [4] concerns the addition of a third objective, which corresponds to the difference in complexity between the lowest one of ABs and the highest one of SABs. This function is defined as follows:

$$f^d = \max\left(\max_{b \in \{n^{AB}+1,\ldots,n^{SAB}+n^{AB}\}} w_b - \alpha \min_{b \in \{1,\ldots,n^{AB}\}} w_b, 0\right), \quad (7)$$

where α is the desired minimum percentage difference between the lowest complexity of ABs and the highest one of SABs. This criterion is introduced because SABs correspond to airspace blocks characterized by low traffic density and low operational complexity, which makes them appropriate for generating sector configurations.

Finally, the objective function f is defined by:

$$f = \lambda f^b + (1-\lambda) f^c + M f^d \quad (8)$$

where $\lambda \in [0,1]$ is a compromise coefficient between the two criteria and $M \in \mathbb{R}$ is a big value to guarantee that the solution meets the constraints of minimum complexity difference between ABs and SABs.

3) Block design: To construct the blocks, we use a Voronoi diagram [3], which partitions a plane into regions based on proximity to a given set of objects. In its simplest form, these objects are a finite set of points—called seeds, sites, or generators—each associated with a Voronoi cell, containing all points closer to that seed than to any other. In our approach, we first generate a two-dimensional Voronoi diagram, which is then extruded into the third dimension (see Figure 4). This process yields a set of vertically stacked airspace units that collectively cover the entire designated volume. While the horizontal footprint of each unit remains constant across altitude layers, the complexity of the resulting geometry varies with height.

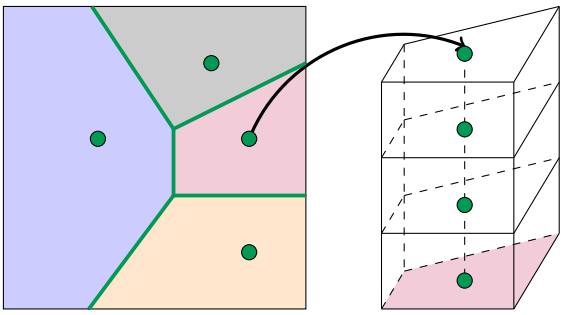


Figure 4: Construction of the Voronoi diagram and 3-dimensional projection.

4) Solution approach: The solution approach is based on Simulated Annealing, originally introduced by Kirkpatrick *et al.* [2], inspired by the physical annealing process of materials. The SA algorithm consists of two main phases: a heating phase that brings the solid into a high temperature, followed by a gradual cooling phase to achieve a solid state with minimal energy ([6]). A key strength of Simulated Annealing lies in its capacity to accept transitions that temporarily degrade the objective function. At the outset, the algorithm operates at a high temperature T, enabling

the acceptance of moves with substantial deterioration in the criterion and thereby promoting extensive exploration of the solution space. As T decreases, the algorithm becomes more selective, accepting primarily improvements or minor degradations. In the final stages, when T approaches zero, no deterioration is permitted, and the algorithm's behavior converges to that of a Monte Carlo method. The SA algorithm is particularly well suited to addressing large-scale problems of this nature.

The main part of the Simulated Annealing is the neighborhood operator. In this study, it consist in changing randomly the center position of one block (AB or SAB). It just therefore consisting in exchanging two elements in the solution list. The blocks obtained are used as input by the sector design module.

B. Sector design

Mathematical programming provides the methodological foundation for sector design. In particular, mixed-integer linear programming (MILP) formulations have been developed to address both two-dimensional case (defined on the latitude–longitude plane) and three-dimensional case. These formulations are described in detail in [7] and [8] respectively. The proposed approach enables the design of effective sectors which satisfy all the operational requirements, as depicted in the exemplar below.

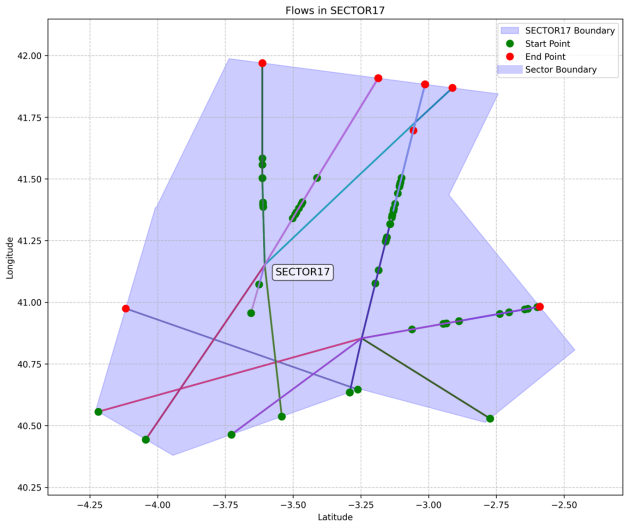


Figure 5: Sector from the weekday morning category. The optimization algorithm has successfully ensured that each flight path traverses the sector exactly once (flow convexity), eliminating the operational inefficiencies and coordination challenges that would arise from trajectory fragmentation.

In accordance with the requirements set by operational stakeholders, for the validation scenarios considered within SMARTS in Madrid ACC airspace, all airspace blocks between FL325 and FL375 are classified a s s hareable. Consequently, in the vertical dimension of the airspace, only two ABs remain: one below FL325 and one above FL375. Since each workable sector must contain at least one AB, this implies that at least one of these two blocks—below FL325, above FL375, or both—must be included in the designed sector. This requirement allows us to apply the 2D formulation using the workflow illustrated in Figure 6.

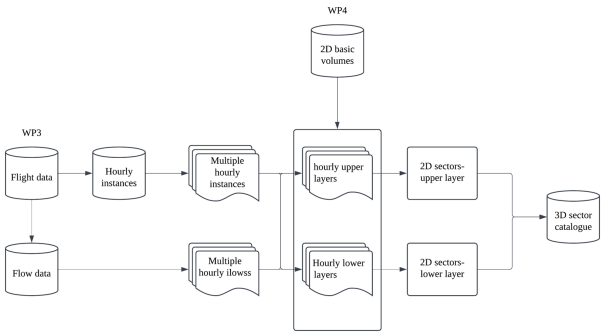


Figure 6: Sector design workflow with hourly average in-stances for each category.

Our approach defines four traffic scenario categories based on weekday/weekend and morning/afternoon combinations. For each category, we compute hourly average instances and traffic fl ows to ca pture re presentative op erational conditions. We conducted experiments with a varying number of sectors K (with $K \in 2, 4, 6, 8$) under two distinct traffic scenarios corresponding to different airspace portions: lower levels and upper levels.

The experimental design employs a two-phase optimization strategy. First, we perform 2D sector optimization to establish horizontal boundaries. Second, we apply systematic vertical layer assignment using a hierarchical protocol:

1) Upper layer sectors: Progressive assignment following the sequence Flight Level (FL) 375+, FL365+, then FL355+
2) Lower layer sectors: Reverse hierarchical assignment pattern of FL355-, FL345-, then FL335-

This methodology significantly reduces computational overhead compared to simultaneous 3D optimization while preserving the essential spatial relationships required for effective airspace management. The outputs of sector design are sector catalogs for each categories, cleaned from those sectors that do not meet the complexity requirements, as in Equation (10), which will be used in the dynamic airspace configuration process.

C. Airspace configuration

The computational framework designed and developed for hosting the SMARTS airspace configuration optimizer is known as Puzzle. Puzzle is a CRIDA-developed Flight Management Position (FMP) tool that contains the SMARTS Sector Configuration Service, which supports air traffic flow and capacity management by providing visualization and optimization tools. The service is structured around two complementary functions:

1) Air Traffic Demand Interface; and
2) Airspace Configuration Plan.

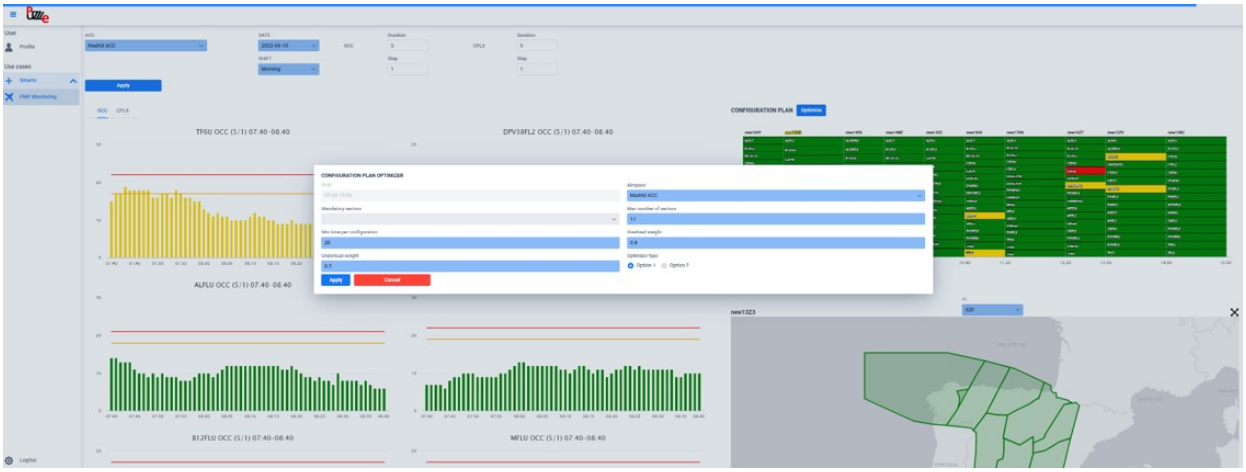


Figure 7: Puzzle FMP tool interface

The **Air Traffic Demand Interface** function provides a comprehensive operational picture of expected traffic demand for a given shift.

The function integrates multiple datasets, including the following:

- Sectorization data: XML files that describe the geographic boundaries and vertical limits of each sector.
- Capacity thresholds: CSV files specifying peak and sustained occupancy thresholds per sector.
- Traffic flows: XML files detailing the main flows within each sector. Each flow entry contains the geometry of the route and potential interactions with other flows.
- Predicted traffic demand: XML files containing flight-level trajectories and flow assignments for all flights crossing a sector during a shift. Each record includes aircraft identifier, entry/exit times, flight levels, and assigned flow.
- Sector Complexity metrics: Complexity metrics per sector based on the type of traffic and the flows interactions.

The complexity of sectors c_s is formally defined as:

$$c_s = c_{base} + c_{evolution} + c_{oceanic} + c_{flow} + c_{OSF} \quad (9)$$

The formulation considers a baseline sector complexity c_{base} together with additional terms capturing flights in evolution (climbs/descents) $c_{evolution}$, oceanic entries $c_{oceanic}$, flow interactions c_{flow}, and flights deviating from standard flows c_{OSF}. Each factor is quantified from traffic data and combined additively, providing a comprehensive indicator of sector difficulty that reflects both structural characteristics and dynamic traffic behaviour.

The calculation of the peak capacity threshold C_i^P and the sustain capacity threshold C_i^S within SMARTS followed the Warning Density Area method derived from the Hourly Entry Count (HEC) versus Occupancy Count (OCC) relationship. Using calibrated abacuses, two critical occupancy values were defined: a Sustain Occupancy, representing the maximum level of traffic that can be managed safely and consistently, and a Peak Occupancy, indicating the upper limit before sector overload requires regulatory measures. The corresponding HEC values were then extracted from traffic data in the Madrid ACC sectors, with expert-defined thresholds mapped to HEC values. Building on this, an automatic process—developed according to the methodology described in [9] was applied, using empirical analysis of HEC→OCC distributions to derive linear trends of maximal and standard occupancy for OCC values. The interval between these thresholds defines the Warning Density Area, where traffic monitoring and tactical interventions ensure sector stability while maintaining a quantifiable link between strategic capacity planning and real-time ATFCM regulation.

In this interface, the user has the flexibility to fix the following parameters:

1) **Overload penalty weight w_o:** Applied when configuration complexity exceeds the sustained threshold. For configurations satisfying $C_i > C_i^S$, the objective penalty is computed as $w_o \times (C_i - C_i^S)$.
2) **Underload penalty weight w_u:** Applied to configurations operating significantly below capacity. When $C_i \leq 0.4 \cdot C_i^P$, the objective penalty becomes $w_u \times (0.4 \cdot C_i^P - C_i)$. The sum of overload penalty weight and underload penalty weight equals to 1: $w_u + w_o = 1$.
3) **Maximum sectors constraint N_{max}:** Defines the upper bound on the number of sectors permitted within any configuration plan.
4) **Permanence duration t_p:** Specifies the minimum activation period for any configuration following transition, ensuring operational stability.

The **Airspace Configuration Plan** facilitates the resolution of detected capacity-demand imbalances through an airspace configuration optimizer that systematically evaluates alternative sector configurations against demand predictions and capacity constraints. The optimal configuration plan is obtained by invoking the configuration optimizer hosted by the Lancaster University Web Services.

Upon receiving optimization requests from the PUZZLE interface, the configuration service returns a comprehensive solution containing time-indexed optimal configurations, associated cost metrics, and detailed sector compositions. This solution replaces the reference configuration within the vi-

sualization environment, enabling direct comparative analysis of demand-capacity balance between baseline and optimized configurations. The iterative nature of the system permits parameter adjustment and re-optimization to achieve desired operational outcomes.

The Dynamic Airspace Configuration (DAC) optimizer provides two algorithmic options for various operational requirements. The first option employs a deterministic graph-based approach using shortest path algorithms [10], while the second option utilizes a deterministic Integer Linear Programming (ILP) model [11] incorporating sector permanence constraints for more realistic operational scenarios. The complete mathematical formulation is detailed in the aforementioned reference. Both the options use complexity values computed by Puzzle as criterion of performance. The complexity of configuration i is formally defined as:

$$C_i = \sum_{j=1}^{N} c_{s_j}, \quad \forall j \in N \tag{10}$$

where configuration i comprises N sectors and c_{s_j} represents the complexity metric for sector s_j. The sector-level complexity metric c_{s_j} was computed using the a formula to calculate complexity metrics per sector, which aggregates multiple contributing factors into a single score.

For Optimizer, we leverage NetworkX [12] for efficient graph modeling and shortest path computation, minimizing external dependencies while ensuring fast execution times suitable for real-time applications. We also integrate Gurobi Python [13] for its optimization capabilities and proven convergence stability in complex mixed-integer programming problems.The system architecture is built using FastAPI, a modern Python web framework that provides automatic API documentation, request validation, and high-performance asynchronous processing. FastAPI enables seamless RESTful API integration with PUZZLE while maintaining type safety and providing interactive API documentation for easier integration and testing.

The DAC Optimizer follows a streamlined workflow where client requests are received through an API layer that handles validation and parameter extraction. The system then loads the appropriate configuration based on time-of-day and day-of-week parameters, utilizing sector catalogs provided in Section II-B.

III. Validation Exercise

A. Data and scenarios

The validation exercise was conducted using the EUROCONTROL RNEST 2 platform at Madrid ACC (LECMCTA), covering airspace sectors LECMCTAN and LECMCTAS from FL 245. The study utilized 12 selected dates from June and July 2023, alongside predicted traffic for 2030 using STATFOR forecasts. Filed flight plans (M1 data) from the Demand Data Repository (DDR2) served as the primary data source, representing the best approximation of airspace users' preferences [14].

B. Demand Analysis

Demand analysis underpins both the design of sectors and their subsequent configuration. In the project, we employed flight plan data from the LECMCTA over the six-month summer season of 2023. A clustering machine learning model is used to identify traffic flows in this dataset, which comprises approximately 444,000 flights.

The clustering model employs Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) due to its capacity to extract clusters of varying scales and densities. Drawing on established literature [15], HDBSCAN excels at discovering clusters with heterogeneous densities and arbitrary shapes and is relatively robust to hyperparameter selection.

In this application, the clusters identified via HDBSCAN demonstrate higher quality (per defined clustering metrics such as compactness, separation, or stability) compared to those produced by Gaussian mixture models. However, a considerable fraction — approximately 29% of all flights — is assigned to an outlier (noise) cluster under the HDBSCAN regime. For alternative use cases in which it is undesirable to have such a high proportion of unclustered or outlier data, one might accept somewhat inferior cluster quality by using a Gaussian clustering paradigm (e.g. Gaussian mixture models), which forces assignment of all data points to clusters.

By computing each flight's distance to its cluster medoid, cluster quality was assessed. In the vertical dimension, 91.37% of the flights lie within ±600 feet of their medoids. In the full three-dimensional space, 88.44% of the flights are at most 10 NM away from their medoids. These results attest to the high internal consistency of the 1,700 clusters identified (each cluster containing at least 60 flights). Figure 8 displays the top 10 clusters, with medoids distributed across the LECMCTA.

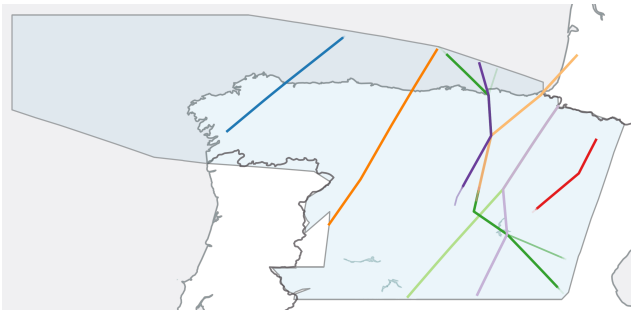


Figure 8: Distribution of the 10 most used traffic flows in LECMCTA airspace. Each color represents a different flow with it´s medoid as the thick line.

C. Resiliency

As a contribution to the SMARTS initiative aimed at enhancing the resiliency of sectors and their configuration plans (Figure 1), we propose a methodology for assessing the resilience of configuration plans using three performance metrics that collectively define a Key Performance Indicator (KPI). The local resilience KPI (11) evaluates each configuration plan through a composite metric that quantifies the

remaining flexibility to absorb disruptions of varying characteristics. These disruptions are captured through indicators related to capacity, demand, and local state conditions (e.g. weather):

$$\text{Local Resiliency}_t = \text{LSD}_t * \frac{\text{LCR}_t}{\max\{1, \text{LDD}_t\}}. \quad (11)$$

where LCR, LDD and LSD denote Local Capacity Resiliency (12), Local Demand Disruption (14) and Local State Disruption (15). The first indicator (LCR) captures the absolute deviation (reserved buffer) between planned and actual occupancy counts by quantifying the standardized distance to its designated thresholds for each active sector in a given configuration. Note that the standardized delta values makes the LCR range to be lower or equal to one:

$$LCR = 1 - |\text{LCR}_{\text{Plan}(t)} - \text{LCR}_{\text{Actual}(t)}| \quad (12)$$

$$\text{LCR}_{x(t)} = \left(\frac{\sum_i \Delta^x_{i,t}}{\frac{1}{3}\sum_i (P_i + 2\,U_i)} \right) \quad (13)$$

where:

- i = Enumerator for Sectors
- t = Time interval
- U_i = Sustained Threshold for Sector i
- P_i = Peak Threshold for Sector i
- $\Delta_{i,t}$ = Sector's Sustained Threshold ($U_{i,t}$) minus Planned occupancy at time t
- $\Delta'_{i,t}$ = Sector's Sustained Threshold ($U_{i,t}$) minus Predicted occupancy at time t

The demand indicator (LDD) quantifies the divergence between planned traffic and predicted traffic under a specific configuration plan:

$$\text{LDD}_t = \sum_i |(\text{Demand}_{i,t} - \text{PredictedDemand}_{i,t})| \quad (14)$$

and the last term of equation (11), quantifies the impact of convective weather:

$$\text{LSD}_t = \frac{\sum_i (1 - \text{PCT}_{overlap(i,t)})}{\text{S}_{count(t)}} \quad (15)$$

This metric aggregates the percentage of intact area across each sector i, and subsequently computes the mean value over S_{count}, the total number of sectors, during the time interval t.

The use of absolute values in the equations above is deliberate in order to account for both positive and negative disruptions [16] - i.e. a negative disruption may correspond to unused capacity. Furthermore, by incorporating components for capacity (LCR) and demand (LDD), the key performance indicator is constructed to provide an equitable measure of both types of deviations. Finally, adverse weather impact (LSD) is modeled to affect demand and capacity simultaneously.

The resilience KPI's maximum theoretical value is 1 (note that LCR delta values are always smaller than LDD non-normalized values) that represents the configuration plan in absence of any convective weather ($LSD = 1$), while the planned occupancy counts (in pre-tactical phase) match exactly the calculated counts in tactical phase ($\text{LCR}_{\text{Plan}(t)} = \text{LCR}_{\text{Actual}(t)}$). However, the practical ceiling of local resilience KPI (< 1) is the largest recorded value for a pool of available configuration plans in post-operation data denoting the maximum value for a given set of days and airspace (here LECMCTA). This case represents the most flexible configuration plan consistent to the definition of resilient system [17, 18].

IV. Results and Discussion

We first conduct a numerical analysis to evaluate the SMARTS solution. Setting $w_o = 0.6$ and $w_u = 0.4$ with an additional penalty for over-peak scenarios, we examine the performance of the optimal configuration plan in Figure 9, using 5-minute time intervals. The results demonstrate that

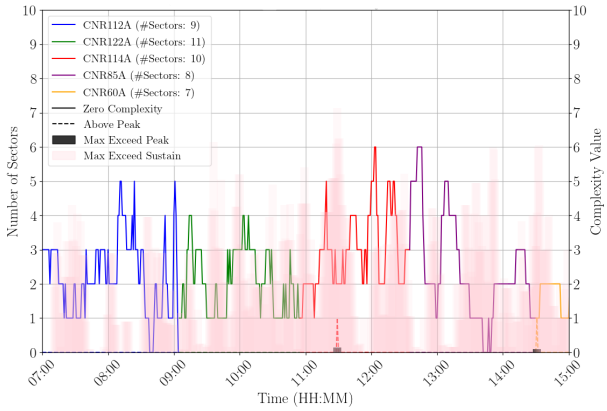


Figure 9: Key metrics from the optimal configuration plan. The solid line indicates the number of sectors with zero complexity, while the dashed line shows the number of sectors exceeding peak capacity. The dark boxes represent the maximum complexity above peak capacity among all sectors for each configuration, and the pink boxes indicate the maximum complexity above sustained capacity.

over-peak scenarios are largely avoided, with only one sector experiencing slight over-peak complexity at approximately 11:30 am and 2:30 pm. This demonstrates the optimizer's capability to prevent excessive workload for ATCOs. Additionally, we observe that some zero-complexity sectors are unavoidable despite workload balancing efforts, due to temporal variations in active traffic flows and input data quality limitations.

A. Performance achievements

The system consistently delivered acceptable solutions when configured with 12-16 sectors, with users finding the traffic distribution and sector balancing generally effective. The model emerged as the standout approach, representing

the preferred solution by FMP operators and consistently delivering superior operational results.

Under adverse conditions, the system demonstrated marked improvement when utilizing updated traffic models (M1). This enhancement resulted in significantly better sector utilization and traffic distribution, with optimizer complexity metrics aligning well with user expectations. The system's ability to adapt to challenging operational scenarios represents a key strength. FMP operators expressed clear satisfaction with solutions providing 8-16 sectors featuring balanced workload distribution. The system successfully navigated the operational sweet spot, avoiding both excessive overloading from too few sectors and unnecessary complexity from too many underloaded sectors.

When performing optimally, the system generated practical configuration plans that aligned with real-world operational needs. Traffic flow distribution was well-managed, with critical traffic points appropriately centered and positioned away from sector boundaries. Multiple scenarios demonstrated the system's ability to generate operationally viable solutions that controllers found acceptable for implementation. The system showed good responsiveness to traffic variations, with complexity metrics that made operational sense and traffic distribution improvements evident when appropriate models were applied. The evaluation revealed strong user confidence in the system's core capabilities, with controllers and FMP operators recognizing its potential for operational deployment. The positive feedback on the generated configurations particularly indicates that the algorithmic approach successfully balances the complex trade-offs inherent in airspace management.

V. Concluding remarks

The SMARTS solution herein presented addresses the critical challenge of dynamic airspace configuration through an integrated three-component approach combining artificial intelligence and optimization algorithms. The initial validation exercise at Madrid Area Control Center demonstrated exceptional performance achievement, with superior capability in avoiding over-peak complexity and finding workload balance, evaluated by the FMP operators.

While the system demonstrates strong foundational capabilities, targeted improvements would maximize operational effectiveness. The sector catalog would benefit from expansion to provide enhanced splitting and merging options. Additionally, specific geographic hotspots require enhanced sector management capabilities, and the flow prioritization functionality needs refinement to consistently generate viable solutions.

A new validation exercise is scheduled in the coming weeks to verify whether some of the identified issues have been addressed. As part of this exercise, a comparative analysis will be conducted between SMARTS and a solution developed by Eurocontrol. It is worth noting that Eurocontrol's approach demonstrated remarkable congestion reduction, achieving a 98.5% improvement compared to the S44 reference baseline.

Acknowledgment

This paper is an output of the SMARTS project. SMARTS has received funding from the SESAR 3 Joint Undertaking (JU) under grant agreement No 101114686. The JU receives support from the European Union's Horizon Europe research and innovation programme and the SESAR 3 JU members other than the Union. UK participants in SMARTS receive funding from UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee [grant numbers 10086651 (Lancaster University)] and 10091277 (NATS). Opinions expressed in this work reflect the authors' views only, and the SESAR 3 JU and UKRI are not responsible for any use that may be made of the information contained herein.

References

[1] Horizon Europe, *Smart sectors: Optimising airspace design to meet air traffic demand efficiently*, 2023. DOI: 10.3030/101114686.

[2] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983. DOI: 10.1126/science.220.4598.671.

[3] F. Aurenhammer, "Voronoi diagrams—a survey of a fundamental geometric data structure," *ACM Computing Surveys (CSUR)*, vol. 23, no. 3, pp. 345–405, 1991.

[4] A. Guitart, J. Lanvandier, and D. Delahaye, "Voronoi diagrams and Simulated Annealing for airspace block optimization," in *International Workshop on ATM/CNS (IWAC2024)*, 2024.

[5] J. Lavandier, "Une métrique rapide et précise pour évaluer et optimiser la complexité du trafic aérien.," Ph.D. dissertation, Ecole Nationale de l'Aviation Civile, 2025.

[6] D. Delahaye, S. Chaimatanan, and M. Mongeau, "Simulated annealing: From basics to applications," in *Handbook of metaheuristics*, Springer, 2019, pp. 1–35. DOI: 10.1007/978-3-319-91086-4_1.

[7] G. N. Lui, G. Lulli, M. F. Lema-Esposto, and R. L. Martinez, "Airspace sector design: An optimization approach," in *SESAR Innovation Days*, 2024.

[8] G. N. Lui and G. Lulli, "A mixed integer programming approach for airspace sector design problem," in *The 12th Triennial Symposium on Transportation Analysis (TRISTAN XII)*, 2025.

[9] EUROCONTROL, "Hourly entry count versus occupancy count relationship – robustness, calibration, application (ii)," Eurocontrol Experimental Centre, Brétigny-sur-Orge, France, Tech. Rep. EEC Note No. 16/07, 2007.

[10] G. N. Lui, G. Lulli, L. D. Givanni, M. Galeazzo, R. L. Martinez, and I. G.-O. Carro, "A robust optimization approach for dynamic airspace configuration," in *Proceedings of the 1st US-Europe Air Transportation Research & Development Symposium*, 2025, pp. 1–11.

[11] M. Galeazzo, L. De Giovanni, M. F. Lema-Esposto, and G. Lulli, "An Integer Programming approach to Dynamic Airspace Configuration," in *ICRAT*, 2024.

[12] A. Hagberg, P. Swart, and D. S Chult, "Exploring network structure, dynamics, and function using NetworkX," Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 2008.

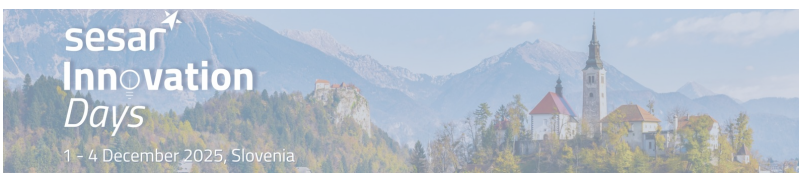
[13] Gurobi Optimization, LLC, *Gurobi Optimizer Reference Manual*, 2025. [Online]. Available: https://www.gurobi.com.

[14] Eurocontrol, *Demand data repository*, 2025. [Online]. Available: https://www.eurocontrol.int/ddr.

[15] R. J. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Pacific-Asia conference on knowledge discovery and data mining*, Springer, 2013, pp. 160–172.

[16] R. Sanaei, A. Lau, and V. Gollnick, "A study of capacity regulations to define european air traffic management network states," *Transportation Planning and Technology*, vol. 44, no. 4, pp. 337–355, 2021. DOI: 10.1080/03081060.2021.1919346.

[17] R. Sanaei, B. A. Pinto, and V. Gollnick, "Toward ATM Resiliency: A Deep CNN to Predict Number of Delayed Flights and ATFM Delay," *Aerospace*, vol. 8, no. 2, 2021, ISSN: 2226-4310. DOI: 10.3390/aerospace8020028. [Online]. Available: https://www.mdpi.com/2226-4310/8/2/28.



[18] R. Francis and B. Bekera, "A metric and frameworks for resilience analysis of engineered and infrastructure systems," *Reliability Engineering & System Safety*, vol. 121, pp. 90–103, 2014, ISSN: 0951-8320. DOI: https://doi.org/10.1016/j.ress.2013.07.004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0951832013002147.