

Article

Cell-Based Optimization of Air Traffic Control Sector Boundaries Using Traffic Complexity

César Gómez Arnaldo *, José María Arroyo López, Raquel Delgado-Aguilera Jurado , María Zamarreño Suárez, Javier Alberto Pérez Castán and Francisco Pérez Moreno 

Department of Aerospace Systems, Air Transport and Airports, ETSIAE, Universidad Politécnica de Madrid (UPM), 28040 Madrid, Spain; jose.alopez@alumnos.upm.es (J.M.A.L.); raquel.djurado@upm.es (R.D.-A.J.); maria.zamsuarez@upm.es (M.Z.S.); javier.perez.castan@upm.es (J.A.P.C.); francisco.perez.moreno@upm.es (F.P.M.)

* Correspondence: cesar.gomez.arnaldo@upm.es

Abstract

The increasing demand for air travel has intensified the need for more efficient air traffic management (ATM) solutions. One of the key challenges in this domain is the optimal sectorization of airspace to ensure balanced controller workload and operational efficiency. Traditional airspace sectors, typically static and based on historical flow patterns, often fail to adapt to evolving traffic complexity, resulting in imbalanced workload distribution and reduced system performance. This study introduces a novel methodology for optimizing ATC sector geometries based on air traffic complexity indicators, aiming to enhance the balance of operational workload across sectors. The proposed optimization is formulated in the horizontal plane using a two-dimensional cell-based airspace representation. A graph-partitioning optimization model with spatial and operational constraints is applied, along with a refinement step using adjacent-cell pairs to improve geometric coherence. Tested on real data from Madrid North ACC, the model achieved significant complexity balancing while preserving sector shapes in a real-world case study based on a Spanish ACC. This work provides a methodological basis to support static and dynamic airspace design and has the potential to enhance ATC efficiency through data-driven optimization.

Keywords: air traffic; complexity; heuristic search; occupancy; optimization; cell; sector

1. Introduction

Air Traffic congestion has emerged as a critical challenge in modern air traffic management. Imbalances between demand and capacity at various elements of the air transportation system are a major cause of flight delays worldwide [1,2]. Although delay mitigation measures are implemented, due to the dynamic and stochastic nature of the Air Traffic Management (ATM) system capacity, it is difficult to balance capacity with ever-increasing demand [3,4].

In order to maintain, as far as possible, a balance, Air Traffic Flow Management (ATFM) is created. Flow management ensures that current airspace capacity is used efficiently so that air traffic demand can be met [5]. The main ATFM measure is to reduce delays by regulating traffic flow through optimized rerouting [6,7]. Nevertheless, the capacity of the ATFM is limited, and it is necessary to develop new technologies and procedures to help the Air Traffic Control (ATC) service increase its capacity and match the ever-increasing demand [8]. Conventional ATFM measures primarily rely on occupancy-based



Academic Editor: Michael Schultz

Received: 22 December 2025

Revised: 18 January 2026

Accepted: 19 January 2026

Published: 20 January 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

indicators, which, while effective for global demand regulation, do not fully capture workload variations driven by traffic interactions and sector geometry.

One variable that determines air traffic capacity is the workload of Air Traffic Controllers (ATCOs) [9]. The ability of ATCOs to manage growing air traffic determines the capacity of the system and its resilience to increasing demand [10]. ATCOs manage traffic in fixed ATC sectors. Sectors have essentially remained unchanged over time in terms of geometric shape and the total number of sectors inside a specific airspace. However, with a rapidly increasing air traffic demand, fixed sectors cannot accommodate air traffic anymore [11], thus raising ATCOs' workload. By saturated sectors inducing ATCOs' workload, it is understood that an optimization of the sectors to avoid this saturation can reduce the workload of controllers [12]. According to [13], reshaping the ATC sectors' structure could balance the airspace capacity to meet the cognitive workload of ATCOs. Current airspace sectorization is based on having the best possible distribution of air traffic flows [14]. Therefore, a change in airspace sectorization must consider the air traffic flows that have evolved over time and adapt to this evolution.

Several studies have been based on the prediction of air traffic trajectories and flows [15–17]. But the influence of traffic flows alone can give incomplete information about how air traffic is structured. Air Traffic Complexity gives a detailed insight into the characteristics of air traffic [18]. Air Traffic Complexity has become the critical indicator to reflect ATCOs' workload in ATM system [19]. Although many complexity indicators are used to optimize the ATCOs' tasks and reduce their workload [20], Air Traffic complexity indicators could also help ATFM to manage air traffic flows [21].

The aim of this paper is to develop a sector optimization module using a complexity indicator. According to [22], airspace design changes could help to reduce ATCOs' workload. Air Traffic complexity could determine which airspace areas are more congested and motivate changes in airspace structure [23]. For this purpose, a process is proposed in which the complexity of the airspace is evaluated macroscopically but the airspace is divided into cells of limited size. The macroscopic evaluation is focused on airspace structural characteristics, so it can provide guidance for airspace design [24]. Once the complexity has been evaluated in the airspace, an optimization process of the shape of the ATC sectors, based on the complexity indicator, is intended to be carried out. Specifically, the aim is to optimize the shape of the sectors to balance the Air Traffic complexity between the different sectors created. In this work, sector occupancy is considered an implicit outcome of the complexity-balancing process, since cell reassignments directly modify the distribution of aircraft across sectors.

The creation of this optimization model has several advantages.

- The creation of sectors has been done through expert knowledge. As in other ATFM processes, such as the development of complexity indicators, expert judgment may be associated with bias [25]. To eliminate this bias, a sector shape optimization process is proposed that is influenced exclusively by complexity. The complexity in turn is based on air traffic data, so the data on which this model is based will be the air traffic flown.
- The aim of this paper is to show a method to optimize the shape of the sectors of an airspace according to the complexity of air traffic. This modification of the shape of the sectors is expected to be static. However, this work can lay the foundations for an application of Dynamic Airspace Configuration (DAC) [26]. There are studies based on the development of DAC using optimization algorithms [27]. This paper, through the creation of a proprietary algorithm, can help to advance this research. Although the objective differs slightly, thanks to the flexibility of the model, varying only the time horizon of the input data and making small adaptations, this model could be applied for DAC.

- As discussed, airspace design changes could help in reducing ATCOs' workload. Specifically, the model in this paper attempts to balance the complexity of ATC sectors in the sample airspace. This, given the relationship of complexity to controller workload, may help in balancing controller workload. This could have a beneficial effect on the ATM system by allowing for better management of human and technological resources [25].

In addition to these advantages, the optimization model developed in this paper has been tested with real traffic data, so that more clear and real conclusions can be drawn about its feasibility.

2. Materials and Methods

The need for sector-level optimization of airspace has been highlighted in several recent studies [28] as an effective way to improve the efficiency of the ATM system. Such optimization seeks to balance complexity across sectors, thereby reducing the likelihood of congestion and improving the allocation of operational resources. This work proposes an optimization methodology applicable to real airspace configurations.

The methodology used to develop this optimization is outlined below. First, the process of adapting the airspace to a cell-based structure is described, followed by the calculation of the complexity associated with each cell. Next, the formulation of the optimization problem is detailed, including the operational constraints imposed and the algorithm employed to achieve the optimal sectorization for a fixed number of sectors.

2.1. Methodology of the Pre-Sectorization Optimization Phase

The proposed sectorization-optimization methodology builds on an established framework that begins with airspace analysis at the cell level. Starting from a two-dimensional grid of the airspace, traffic databases are processed and adapted to that format.

With this information in hand, a complexity indicator suitable for a fixed-cell partition of the airspace is applied, and an optimization problem is formulated to distribute complexity more evenly across sectors. This involves performing sector optimization based on complexity by reallocating cells to sectors, followed by validation of the resulting sectorization.

2.1.1. Pre-Optimization Considerations

Before undertaking the optimization process, a series of preliminary factors must be considered, as they shape both the overall approach and the decisions made during development:

- First, the sectorization produced by the optimization must retain a certain structural similarity to the sector configuration currently in operation. Preserving this operational coherence makes any transition to the new design easier and avoids drastic changes that could jeopardize safety, control efficiency, or controller workload. The goal is not to reinvent airspace organization from scratch, but to improve it so it better fits today's traffic patterns.
- Second, although the real problem is three-dimensional—upper and lower sectors are differentiated—the optimization is carried out on a two-dimensional (2D) projection. This simplification lowers computational cost and offers clearer visual interpretation but comes with limitations. In this version of the methodology, the focus is on 2D sector design; shapes are varied in the horizontal plane to adapt to traffic complexity. For that reason, existing integrated-sector configurations already deployed in the airspace are used. This keeps the model coherent with real-world structure, avoids explicitly handling verticality, and simplifies the application of optimization algorithms.
- Lastly, because the starting point is an integrated (non-vertically split) sectorization, the time periods chosen for analysis must be selected carefully. Time windows with

moderate traffic levels are preferred, and periods of high (calculated) complexity are avoided. Without the customary vertical split between sectors, the simplified model might fail to capture real dynamics in heavy-traffic situations. To keep the adopted approach valid and prevent the simplification from introducing significant errors, data from times of day when traffic volume and calculated complexity are low enough to remain representative are prioritized.

Taken together, these considerations ensure that the optimization process remains consistent with the proposed methodology. They also allow for viable results that align with real-world operations, despite the existing limitations.

2.1.2. Airspace Complexity Calculation

The optimization relies on the COMETA algorithm, which measures sector complexity in terms of equivalent flights [29,30]. The algorithm determines sector complexity using six factors, each of which contributes a value to the overall complexity calculation depending on the prevailing conditions. Every flight is assigned a complexity score—obtained by adding the contributions of all six factors—and the sector's total complexity is computed as the sum of the individual complexities of all flights.

Cell-level complexity is computed from extensively processed raw traffic data using the COMETA algorithm, applied over fixed one-hour time windows, as defined in the referenced literature.

2.1.3. Gridding and Adaptation of Complexity Calculation at Cell Level

The forthcoming sector-optimization process begins with an extensive preparation phase that includes a detailed analysis of the associated airspace. The goal is to transform the available information into a format suitable for applying cell-movement-based optimization algorithms. This preliminary phase can be divided into three main parts: (1) designing the airspace grid, (2) characterizing traffic at the cell level, and (3) calculating complexity with the COMETA algorithm adapted to the cell scale.

Airspace Gridding

As a first step toward optimization, a two-dimensional grid of the study airspace is designed, allowing it to be divided into elemental volumes called cells. The proposed 2D grid consists of square cells measuring 30 NM by 30 NM, whose upper and lower vertical limits coincide with those of the study airspace. These cell dimensions are considered appropriate for two main reasons. First, they yield a grid with a manageable number of cells, making it easier to undertake an initial computational approach to the sectorization-optimization problem.

Secondly, the 30 NM $\times$ 30 NM size has an operational rationale: it is roughly equal to the distance an aircraft flying at a representative speed of Mach 0.75 at flight level 350 (FL350) covers in five minutes. This offers a realistic, operationally consistent basis for the grid design.

Because the Earth is spherical, this grid for Madrid North ACC must take latitude into account. To ensure the cells are truly square—30 NM by 30 NM in actual distance—their width in degrees of longitude must be adjusted according to the latitude at which each cell's horizontal side lines.

One nautical mile is approximately equal to one arc-minute of latitude and, consequently, 60 NM equal 1° of latitude anywhere on Earth. By contrast, 1° of longitude equals roughly 60 NM only at the Equator and shrinks progressively toward the poles. Therefore,

the following approximation (1) is adopted to convert distances in nautical miles to degrees of longitude, ensuring that the cells remain square in terms of true distance:

$$\Delta \text{longitude}[^{\circ}] = \frac{\text{distance}[\text{NM}]}{\cos(\text{latitude}[^{\circ}]) \times 60'} \quad (1)$$

According to the approximation given by expression (1), the cell dimensions—expressed in geographic coordinates—are calculated dynamically. In other words, applying that expression performs a dynamic adjustment of the cell's size in degrees of longitude, since that size is computed using the current latitude of the horizontal side being evaluated.

As an example, Figure 1 shows the resulting grid of the airspace associated with Madrid North ACC. The grid is made up of 115 regular cells in total (except for those that border the ACC boundary). Dividing the selected study airspace into this regular grid of cells serves as the foundation for the optimization process. Although it is not represented in Figure 1, the airspace can be divided into a different number of sectors, and in reality Madrid North ACC is divided into 5 different sectors (as it is later shown in the upcoming figures).

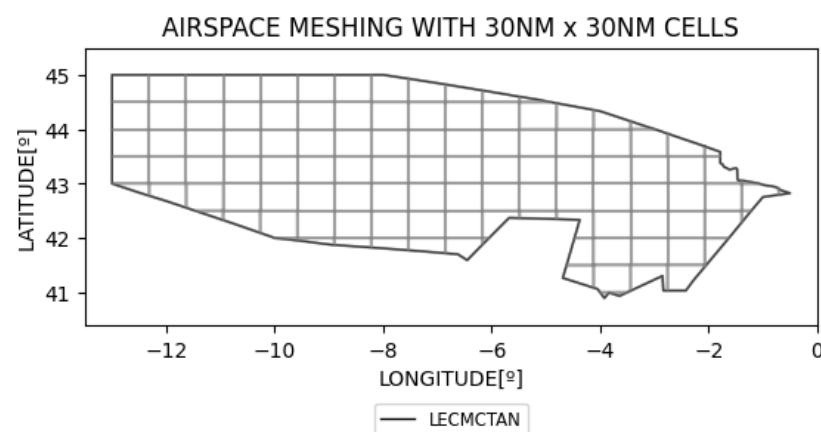


Figure 1. Grid of the airspace associated with Madrid North ACC based on square cells measuring 30 NM × 30 NM.

Characterization of Traffic at Cell Level

Once the grid had been created, the traffic data is adapted to the new elemental unit—the cell. The first step determines the ordered sequence of cells that each clustered flight traverses. To streamline the process, traffic is structured into air-traffic flows using the algorithm detailed in [13]. This makes it possible to classify flights into flows according to their similar trajectories. The inverse relationship is also constructed: for each cell, the flights that pass through it are identified and likewise classified into air-traffic flows. This information allows trajectory prediction to be framed as a succession of cells rather than sectors.

Figures 2 and 3 illustrate, respectively, graphical examples of the analysis of the flows that traverse a cell and of the cells that a given flow traverses, using the Madrid North ACC case.

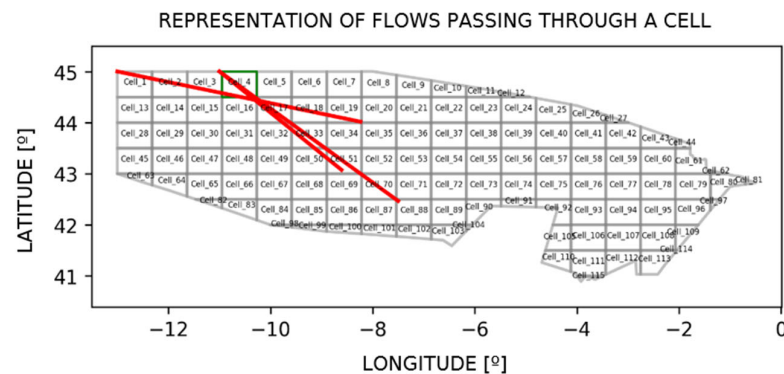


Figure 2. Graphical representation associated with the analysis of the flows that cut through a cell.

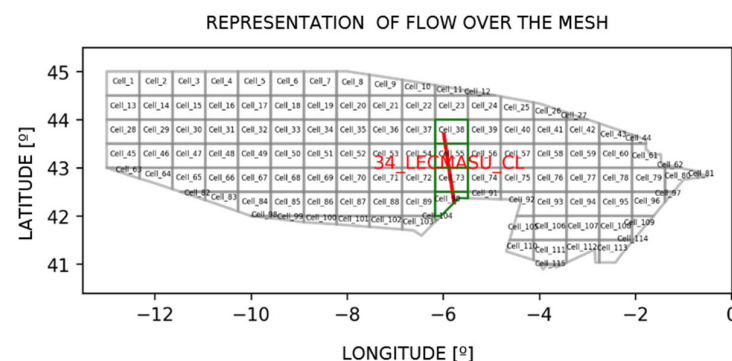


Figure 3. Graphical representation associated with the analysis of the cells which a flow cuts through.

COMETA Complexity Calculation Adapted to COMETA Algorithm

With all the data adapted, a modified version of the COMETA algorithm is applied to calculate complexity in each cell of the grid. This adaptation means that, instead of operating at sector level, the calculations are carried out for each individual cell. The key modification is the computation of a flow-interaction matrix for every cell in the grid; this matrix records the level of interaction between each pair of flows that pass through the same cell.

For each cell and time window, once the matrix has been computed, the six complexity factors of the COMETA indicator described by [29] are calculated. The aggregate complexity for a cell is obtained by summing the individual complexities of all flights present in that cell during the interval. The result of this phase is a cell-level complexity map. Figure 4 illustrates a sample calculation for Madrid North ACC. This complexity map will serve as the starting point for the sectorization-optimization phase.

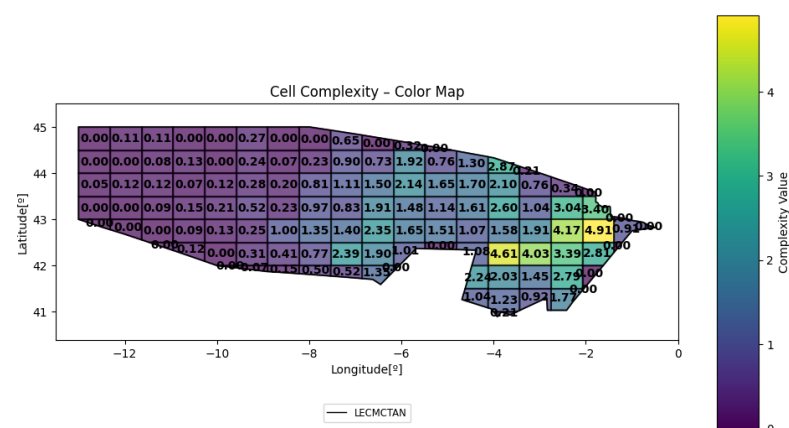


Figure 4. Complexity per cell in the 6 a.m.–7 a.m. time window on 1 June 2022.

Using the resulting complexity values, the optimization process—based on grouping cells—is applied, aiming to balance workload across sectors while maintaining spatial connectivity. This procedure transforms operational airspace and traffic information into a cell-level format. In subsequent phases, this model will serve as the basis for defining new sectorizations in which the sectors have similar complexity values.

2.2. Methodology of the Sector-Optimization Process

The proposed sector-optimization approach begins by subdividing the selected study airspace into smaller elemental volumes: cells. Each cell has an associated complexity value, computed from the traffic it contains over a given period. Sector complexity is estimated as the sum of the complexities of the cells that compose it. The objective is to create clusters of adjacent cells with similar complexity levels, thereby defining the new sectors. The aim throughout the process is to ensure an even distribution of operational workload.

To solve this redistribution problem, we use an iterative optimization algorithm grounded in classical graph-partitioning techniques with topological constraints [31,32], which preserve spatial connectivity between cells. The progressive-improvement scheme adopted here shares its foundations with other optimization approaches applied to complex networks [33] and combinatorial systems, where stochastic mechanisms—such as random restarts—are employed to escape local optima [34,35]. These strategies have proved effective in settings where the solution space is large and the objective function is non-convex, as is the case for our problem.

The optimization begins with a user-specified target sectorization (i.e., the desired number of sectors). The algorithm then performs successive moves of cells between sectors while respecting operational and connectivity constraints. The outcome is an alternative, optimized, and operationally viable sectorization derived from real airspace data.

2.2.1. General Structure of the Optimization Process

The sectorization-optimization process consists of a series of methodological steps designed to balance and efficiently optimize complexity across sectors. Each stage of this process is described in detail below:

1. Grid and input data: Each cell is defined by its polygon and a previously calculated complexity value. The “real” sector configuration is also available and always serves as the starting point for the optimizations.
2. Initial assignment: Each cell is assigned to its “real” sector by computing the polygon intersections and choosing the sector with the largest intersection area.
3. Adjacency-graph construction: Each cell is a node, and two nodes are connected if they share an edge. Because the imposed connectivity criterion is edge-based (to avoid cells connected only at a vertex or disconnected “jumps” in the sectorization), neighboring cells must share a full edge.
4. Objective function definition: The main objective function combines (i) the sample variance of complexity across sectors—this criteria drives an even distribution of complexity—and (ii) the difference between the maximum and minimum sector complexity—this criteria pushes the algorithm to lower the complexity of the busiest sector while increasing that of the least loaded one. Both variables are of the same order of magnitude, so normalization is unnecessary.
5. Optimization: A new assignment of cells to sectors is sought to balance total complexity among sectors by minimizing the objective function. This is accomplished by transferring cells between sectors while respecting the connectivity constraints described above.

2.2.2. Global Optimization Constraints

To preserve connectivity between cells within each sector, maintain spatial continuity, and ensure consistency with real-world operations, a set of global constraints is defined for the optimization; these constraints are applied during every cell reassignment. The constraints are:

- Movement of border cells: Only cells that have at least one neighboring cell in a different sector are considered move candidates, so as not to break internal continuity. In other words, only cells sharing an edge with two or three sectors at once can be moved. Cells located in the core of a sector cannot be moved, preventing cell “jumps” or discontinuous sectorizations. Let:

- $c \in C$: a cell from the total set of cells.
- $S(c)$: sector to which cell c belongs.
- $N(c)$: set of neighboring cells of c .
- Cell c may be considered for relocation if and only if:

$$\exists c' \in N(c) \text{ so that } S(c') \neq S(c) \quad (2)$$

- Movement to adjacent sectors: For every cell on a sector boundary, the only sectors it may belong to during optimization are those of its neighboring cells in the graph. The algorithm identifies boundary cells and transfers a cell only to one of the sectors with which it shares an edge.

$$S_c = \{S(c') | c' \in N(c)\} \quad (3)$$

- New sector assignment must therefore satisfy:

$$S_{\text{new}}(c) \in S_c \quad (4)$$

- Valid neighborhood: Ensures that every move preserves edge contact between the moved cell and the sector it departs from. In this way, sector connectivity is maintained even after cells are reassigned. Let

$$N_S(c) = \{c' \in N(c) | S(c') = S(c)\} \text{ (neighbors in the same sector)} \quad (5)$$

- After the change, for the neighborhood to remain valid

$$|N_{\text{Sold}}(c)| \geq 1 \quad (6)$$

- Preserved connectivity: Checks that, after moving a cell, both sectors—the one that absorbs the cell and the one that loses it—remain connected. This ensures that cell transfers maintain the desired spatial continuity. Let

$$G_s = (C_s, E_s) \text{ be the sub-graph of cells belonging to sector } S, \text{ with } C_s \subseteq C \quad (7)$$

- Being $C_s \subseteq C$ the set of cells in the sector and S and E_s the set of edges between neighboring cells inside S . After moving cell c , both the source sector S_{old} and the destination sector S_{new} must still satisfy that their graphs $G_{s \text{ old}}$ and $G_{s \text{ new}}$ are connected. In other words, the number of connected components must remain equal to 1:

$$\text{Components}(G_s) = 1 \quad (8)$$

- Limit on changes per sector: Caps the number of cells that can be removed from each original sector. This is a tunable decision parameter. Let

- ΔS : be the number of cells that have left sector S
- L_s be the maximum number of cells allowed to leave sector S

$$\Delta S \leq L_s \forall S \in S \quad (9)$$

2.2.3. Implementation of the Sector-Optimization Algorithm

The initial optimization algorithm operates through an iterative process that continually seeks to improve the assignment of cells to sectors, minimizing the objective function described in earlier sections and linked to problem complexity [33]. The algorithm's basic operation can be broken down into several stages that repeat until a solution is reached in which no further cell movement improves the objective function.

In each iteration, the algorithm begins by randomly selecting a border cell—one that is in contact with another sector, meaning one or more of its edges belong to two different sectors. Once the cell has been chosen, one of the neighboring sectors to which the cell could be transferred is also picked at random (if several are available). If there is only one possible candidate, the cell is moved to that sector directly. This movement is treated as a proposed change in the cell assignment.

To determine whether a move is valid, the algorithm must verify that it satisfies all predefined constraints and that it actually lowers the objective-function value. If both conditions are met—that is, the move is valid and improves the objective function—the change is accepted and the cell assignment in the system is updated accordingly. Otherwise, if the move fails to improve the objective function or violates any constraints, it is rejected and the algorithm proceeds to try again with a new cell and a new neighboring sector.

This cycle of random selection, proposed movement, constraint validation, and acceptance or rejection of the move is repeated continuously throughout each iteration of the algorithm. The iteration ends when no moves can be found that improve the objective-function value—that is, when no further improvements can be made to the cell assignment.

However, to keep the algorithm from getting stuck in local optima—that is, sub-optimal solutions that are not necessarily the best achievable—a restart strategy is employed [34,35]. At each restart, the entire process begins anew, but with a different random seed. This ensures that every restart explores a different search path in pursuit of better solutions.

The number of restarts is another tunable decision parameter. After all restarts have been completed, the algorithm compares the final objective-function values obtained in each run. The cell assignment that achieves the lowest objective-function value—i.e., the best balance in terms of complexity—is selected as the final solution. This approach not only produces an effective solution but also helps ensure that the chosen solution is robust and less likely to be trapped in sub-optimal configurations.

2.2.4. Additional Cell-Movement Step with Geometric Considerations

To reinforce the geometric consistency of the sectors produced by the optimization process—and to anticipate undesirable configurations—an extra step has been added to the methodology. This step is designed to prevent isolated cells or spatial arrangements that are not coherent and that could hinder real-world operations.

The mechanism restricts certain cell movements, allowing only those that involve pairs of adjacent cells within the same sector. First, all border pairs that share an edge and belong to the same sector are identified. For each pair, the neighboring sectors that both cells have in common are then determined as potential valid destinations.

Before any move is applied, the algorithm verifies that all the constraints defined in the overall optimization are satisfied. An additional condition is added: the move must not

create any isolated cell in any sector. Only if this condition is met—and the change further reduces the objective function—is the move accepted.

This additional step not only preserves the spatial consistency of the model but also seeks to lessen the likelihood of problematic re-entries. By moving two contiguous cells instead of one, the air traffic travels a greater distance before re-entering its original sector. This gives the controller of the intermediate sector more time to manage potential conflicts, thereby reducing the operational workload of the initial sector.

This mechanism is conceived as a natural extension of the original optimization, providing greater robustness to the model without compromising the goal of balancing complexity across sectors.

3. Results

After establishing the complexity-based sector-optimization methodology, this section presents the optimization results for a chosen case study. The input databases used for the optimization correspond to real traffic over Spanish airspace in 2022 [30]. The data has been obtained by the Spanish service provider ENAIRE, and processed and validated by the company C.R.I.D.A (Madrid, Spain).

Regarding the database structure, each flight has as many records as the number of sectors it crosses within the ACC. This format is convenient when adapting the traffic database to the cell level. The selected case study focuses on the time window from 6 a.m. to 7 a.m. on 1 June 2022. The airspace examined during this period is that of Madrid North ACC, so the optimization will be performed on sectorizations within this ACC.

As an example, Figure 5 shows the real geometry of sectorization CNF5A for Madrid North ACC, while Figure 6 presents its representation after assigning the cells to their corresponding sectors. In this second image, besides the geometry, the sector-complexity values are also included, calculated as the sum of the complexities of the cells that make up each sector.

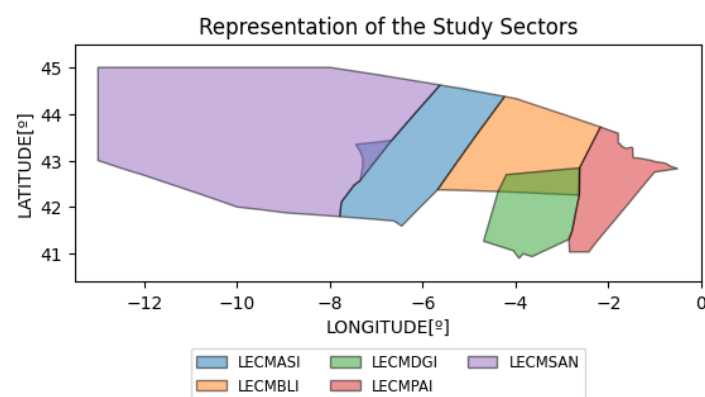


Figure 5. Configuration CNF5A for ACC Madrid North.

The initial sectorization, along with its sector-complexity values for the optimization process, is the one shown in Figure 6. Its sector complexity values differ greatly from one sector to another, indicating that the current sectorization can be optimized for this time window.

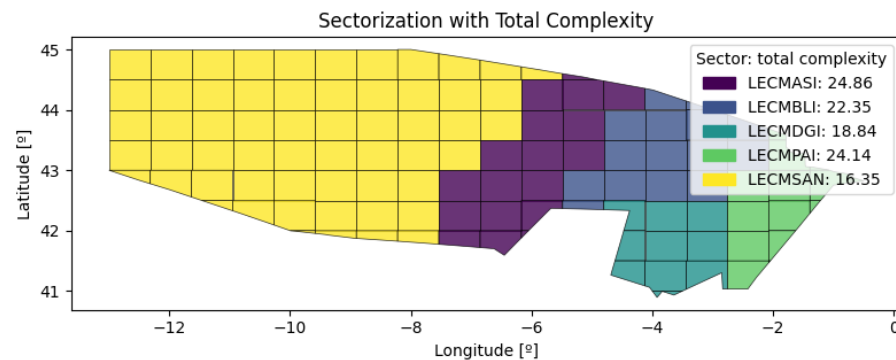


Figure 6. Complexity values for sectorization CNF5A after cell assignment.

3.1. Optimization Using Individual Cells

This subsection presents the first results obtained after applying the optimization algorithm to a specific sector configuration of Madrid North ACC. The main goal of this initial phase is to assess the algorithm's ability to redistribute complexity among sectors while respecting the imposed constraints, such as cell connectivity and the maximum number of changes allowed per sector. For each case, we display the evolution of the sectorization after optimization, together with the resulting complexity values for each sector. These initial results will lay the groundwork for introducing additional constraints in later phases, aimed at guaranteeing more operationally realistic sector geometries.

3.1.1. Optimization Results for a Five-Sector Configuration with One Allowed Cell Change per Sector

To begin the validation and presentation of results, we analyze the most restrictive case of this optimization, which occurs when the maximum cell change per sector is limited to one. The number of iterations is fixed at ten optimization runs, allowing several search paths to be explored without excessively increasing total runtime. The parameter for the maximum cell changes per sector is denoted *max_change*, and the number of optimization iterations is denoted *n*. Figures 7 and 8 display the initial optimization results for a five-sector configuration.

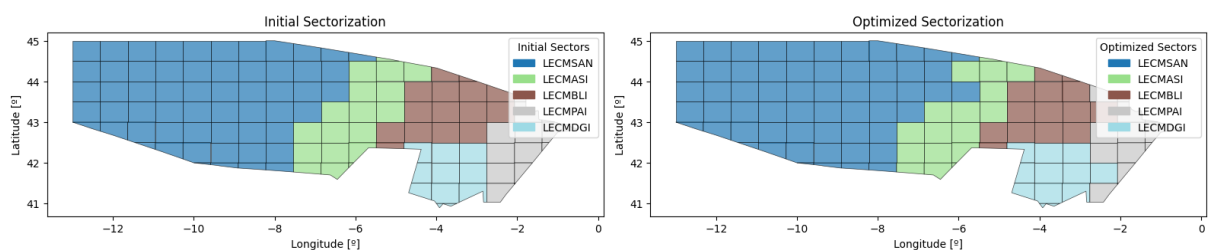


Figure 7. Initial sectorization versus optimized sectorization (*max_change* = 1, *number_sectors* = 5 and *n* = 10).

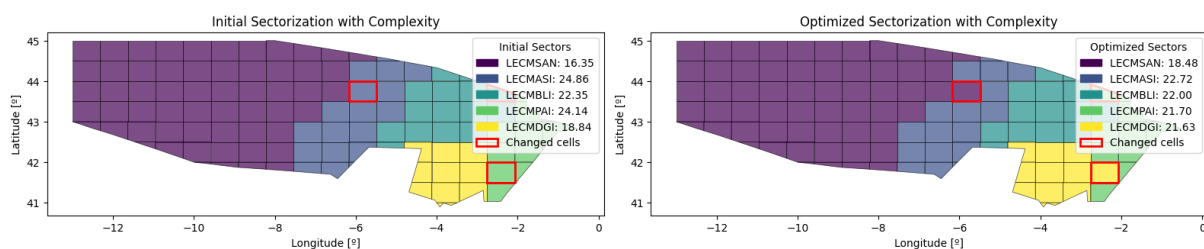


Figure 8. Initial sectorization versus optimized sectorization with the corresponding complexity values per sector (*max_change* = 1, *number_sectors* = 5 and *n* = 10).

The figures compare the initial cell-level sectorization with the optimized one. The initial configuration shown in Figure 7 is generated using the initial-assignment procedure described earlier. The optimization process is then applied while respecting the connectivity constraints. Figure 8 displays the optimized sectorization, where it can be seen at a glance that no sector absorbs more than one extra cell—the imposed limit. The improvement in complexity resulting from these cell moves is also evident: shifting the cells substantially redistributes complexity among sectors, lowering the maximum complexity and raising the minimum, precisely as required by our objective function.

However, setting $\text{max_change} = 1$ prevents the algorithm from continuing to optimize the sectorization—i.e., it is a very restrictive setting. The benefit of fixing such a low maximum cell change per sector arises in situations where one wishes to make only minor adjustments to the current sector geometry (for instance, to ease implementation in real operations, since radical sectorization changes are harder to deploy) or when carrying out early-handover maneuvers for aircraft. In these cases, the option of limiting the maximum change to just one cell is truly useful.

Despite the benefits of this optimization (with max_change set to 1), the geometry of the final sectorization has certain limitations arising from the restrictiveness of the use case. One such limitation is the emergence of isolated cells—cells that, after being transferred from one sector to another, keep connectivity with the new sector through only a single edge. In other words, they are cells surrounded by the original sector, since inter-sector connectivity is defined by shared edges.

In Figure 8 there is a moved cell between sectors LECMSAN and LECMASI that fits this description. Because the cell is surrounded by its original sector, an aircraft following a vertical flow through that area would enter one sector, cross this cell belonging to another sector, and then re-enter the initial sector. In real operations, such a pattern does not align with the flows that respect the current sectorization, so it should be avoided whenever possible. For that reason, new constraints will be introduced to alleviate situations of this kind.

3.1.2. Optimization Results for a Five-Sector Configuration with up to Five Allowed Cell Changes per Sector

The optimization of the case study described above is now analyzed using a parameter of $\text{max_change} = 5$. The results of this optimization are shown in the figures below—Figures 9 and 10.

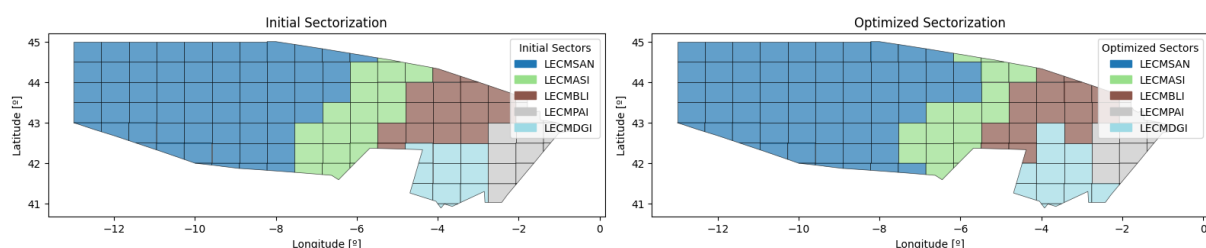


Figure 9. Initial sectorization versus optimized sectorization ($\text{max_change} = 5$, $\text{number_sector} = 5$ and $n = 10$).

As can be seen in the previous figures, the sector-by-sector complexity values are distributed more evenly, which means that the more cells the algorithm is allowed to move between sectors, the better the resulting complexity balance. Nevertheless, isolated cells still appear in the optimized layout because the process optimizes for complexity rather than geometry.

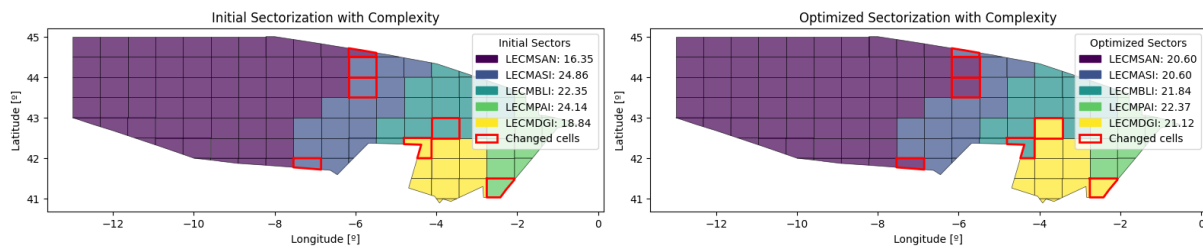


Figure 10. Initial sectorization versus optimized sectorization with the corresponding complexity values per sector (max_change = 5, number_sectors = 5 and n = 10).

3.1.3. Optimization Results for a Five-Sector Configuration with up to Ten Allowed Cell Changes per Sector

Next, the results obtained from optimizing the chosen case study with a parameter of max_change = 10 are analyzed. Figures 11 and 12 present the corresponding optimization outcomes for the case under examination.

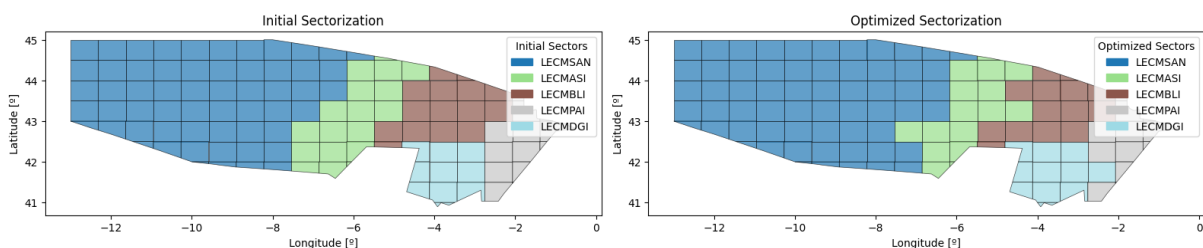


Figure 11. Initial sectorization versus optimized sectorization (max_change = 10, number_sectors = 5 and n = 10).

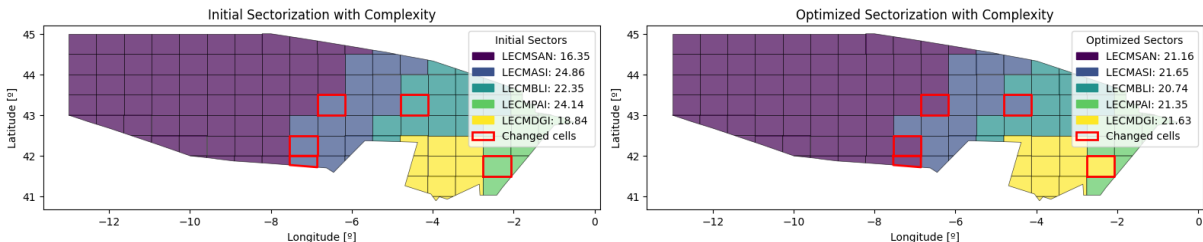


Figure 12. Initial sectorization versus optimized sectorization with the corresponding complexity values per sector (max_change = 10, number_sectors = 5 and n = 10).

As shown in Figure 12, the complexity results achieved with this optimization are excellent. The legend in Figure 12, which lists the complexity value for each sector, reveals that the difference between the most complex and the least complex sector drops from 8.51 to 0.91 in the optimized sectorization. In other words, this optimization minimizes the objective function most effectively and yields the best complexity balance overall.

3.1.4. Optimization Results for a Four-Sector Configuration with up to Ten Allowed Cell Changes per Sector

To explore additional sectorizations and extend the algorithm's application to other scenarios, the figures below present the optimization results for the previously analyzed case—this time using a four-sector configuration, as shown in Figures 13 and 14, alongside the corresponding initial layout based on the real four-sector sectorization.

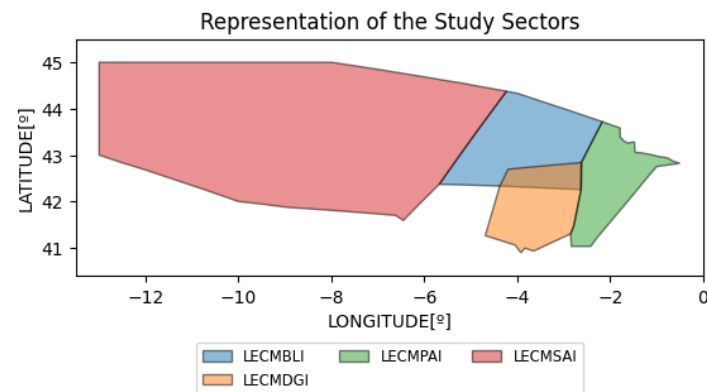


Figure 13. CNF4A configuration of Madrid North ACC.

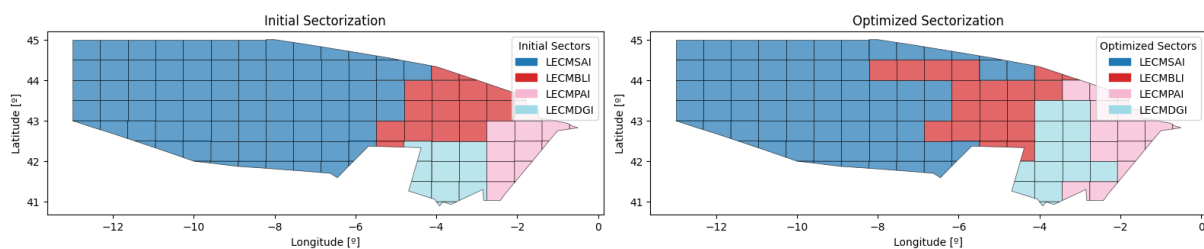


Figure 14. Initial sectorization versus optimized sectorization (max_change = 10, number_sectors = 4 and n = 10).

Just like in the five-sector case, the complexity results obtained with the optimization greatly minimize both the variation in complexity among sectors and the gap between the maximum and minimum sector complexity. It is also apparent that, to optimize this scenario, the algorithm has to move a larger number of cells because complexity is more evenly spread across the sectors. Although the complexity figures presented in Figure 15 are excellent, the geometry of the resulting layout is not fully consistent with real-world operations, as part of sector LECMBLI ends up surrounded by sector LECMSAI.

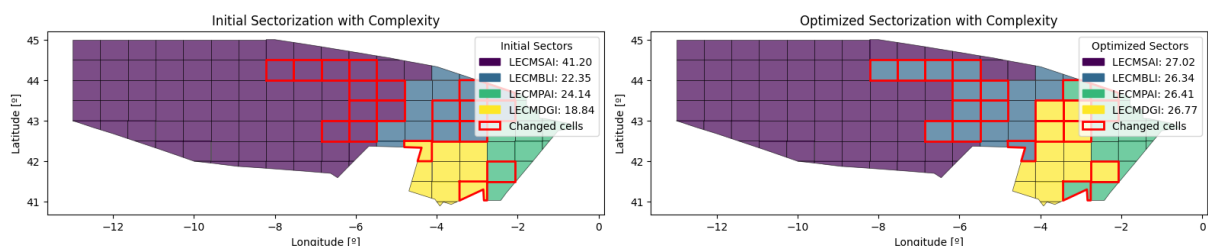


Figure 15. Initial sectorization versus optimized sectorization with the corresponding complexity values per sector (max_change = 10, number_sectors = 4 and n = 10).

Because this optimization does not account for geometry (beyond the required connectivity), we will next explore the same optimization but with an additional constraint on cell movement, aiming to achieve complexity values that are distributed across sectors while maintaining operationally suitable geometries.

3.2. Sector Optimization with Geometric Considerations

This subsection presents the results obtained after applying the additional optimization step based on moving pairs of adjacent cells. This phase complements the initial methodology by incorporating extra constraints aimed at improving the geometric coherence of the resulting sectorizations.

The aim of this optimization is to assess the new approach's ability to resolve the problematic configurations identified in the previous stage, such as the appearance of isolated cells or irregular geometries. To this end, the algorithm has been applied while allowing only moves of boundary cell pairs that belong to the same sector and share an edge, ensuring that the changes preserve connectivity and do not create spatial discontinuities.

The results obtained after applying this optimization to the same configurations analyzed earlier are presented below. Both the resulting sector-complexity values and the improvements in the geometric structure of the sectors are examined, highlighting the operational benefits that this approach introduces in terms of alignment with real-world sectorizations.

3.2.1. Result of the Adjacent-Cell-Pair Optimization for an Initial Five-Sector Configuration

The following section provides a detailed analysis of the optimization results, which significantly redistribute complexity among sectors while preserving the original operational coherence and sector geometry. This sector-optimization problem inevitably involves a trade-off between the geometry and operational aspects of the sectors and their complexity levels. The optimization outcomes for the case study are presented in the figures below.

As Figure 16 shows, the complexity results achieved with this optimization improve markedly over the original assignment. Moreover, the resulting sector geometry remains similar to the starting configuration while still offering a better complexity balance. Although the optimization greatly evens out complexity, it does not reach the highly precise values obtained in the initial, geometry-agnostic run; the added geometric constraint narrows the solution space. Even so, the figures confirm that sector complexity is greatly redistributed, with the maximum gap across the five sectors shrinking from 8.51 to 1.93.

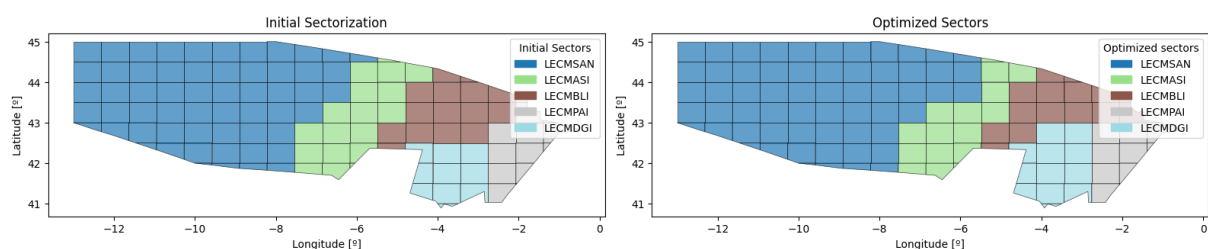


Figure 16. Initial sectorization versus optimized sectorization by pairs of adjacent cells (number_sectors = 5).

The resulting geometry is far more compact and consistent with the initial sectorization than the geometries produced by the first optimization. Although the distribution of sector complexity is slightly less uniform, the geometry benefits greatly. Additional cases will be examined to verify the complexity values optimized with this approach in different scenarios, but Figure 16 already suggests that this method may strike the ideal balance between preserving the initial geometry and achieving a workable complexity distribution. Two further cases will now be analyzed, each starting from an initial sectorization smaller than the one evaluated in Figure 16.

3.2.2. Result of the Adjacent-Cell-Pair Optimization for an Initial Four-Sector Configuration

The following figures—Figures 17 and 18—show the optimization results for the scenario with a smaller number of sectors, namely four.

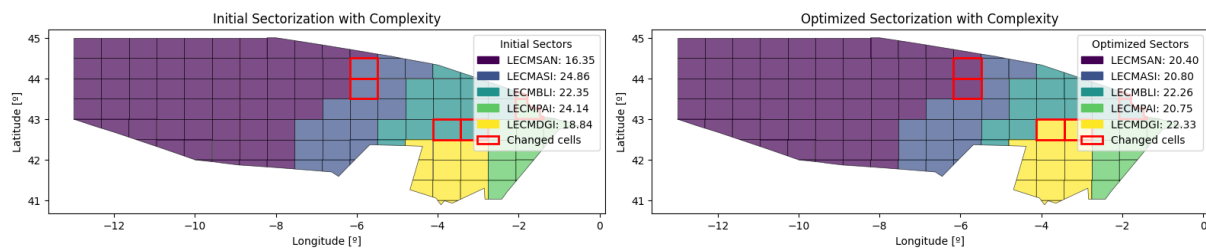


Figure 17. Initial sectorization versus optimized sectorization by pairs of adjacent cells, with the corresponding complexity values per sector (number_sectors = 5).

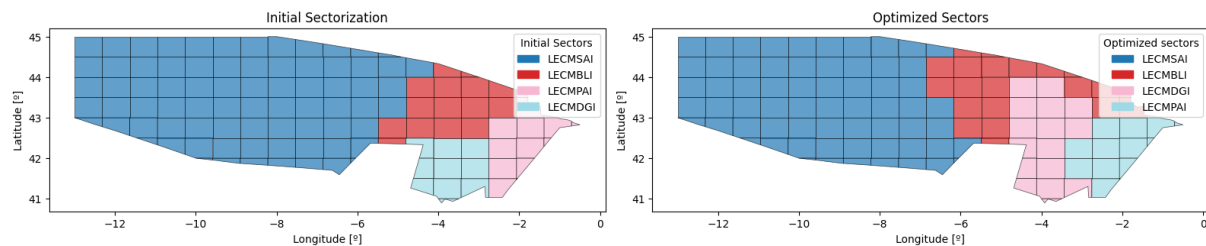


Figure 18. Initial sectorization versus optimized sectorization by pairs of adjacent cells. (number_sectors = 4).

As Figure 18 shows, the algorithm has to move a larger number of cells in order to distribute and balance complexity. In the earlier, less-constrained optimization, moving that many cells would have produced geometries that diverged more from the original layout. However, thanks to the new restriction, the resulting sector is geometrically cleaner than its counterpart. Moreover, the maximum complexity gap in Figure 19 falls from 22.36 to 0.74, demonstrating that the algorithm can indeed redistribute sector complexity effectively.

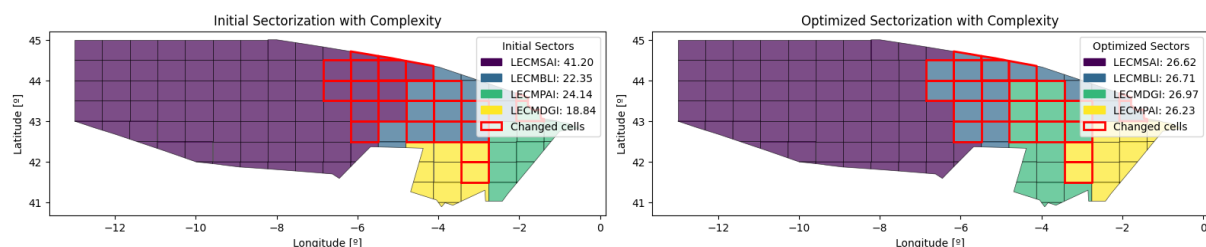


Figure 19. Initial sectorization versus optimized sectorization by pairs of adjacent cells, with the corresponding complexity values per sector (number_sectors = 4).

3.2.3. Result of the Adjacent-Cell-Pair Optimization for an Initial Three-Sector Configuration

Finally, the case featuring an initial configuration of three sectors will be analyzed, as shown in Figure 20. This will serve as the definitive test to determine whether the optimization algorithm preserves geometries consistent with those of real-world sectors in day-to-day operations. The associated complexity results produced by the optimization will also be evaluated to assess their optimality.

As shown in Figures 21 and 22, a compact sector distribution is maintained, with no isolated cells and all sectors in contact with the ACC boundary—none are fully surrounded by other sectors. As with previous optimizations using this algorithm, the complexity values displayed in Figure 22 remain optimal, with the maximum complexity difference dropping from 44.71 to 0.90.

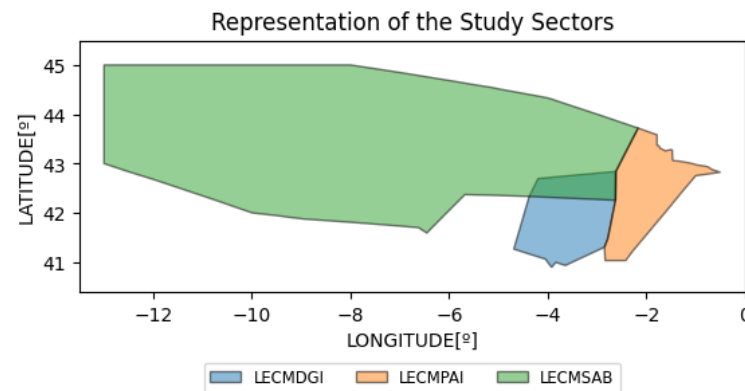


Figure 20. CNF3A configuration of Madrid North ACC.

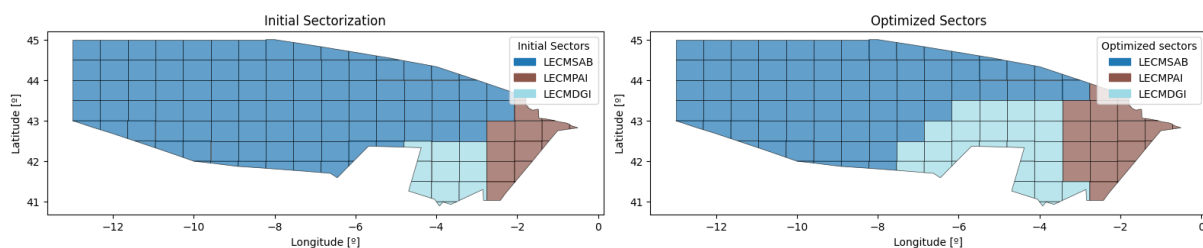


Figure 21. Initial sectorization versus optimized sectorization by pairs of adjacent cells (number_sectors = 3).

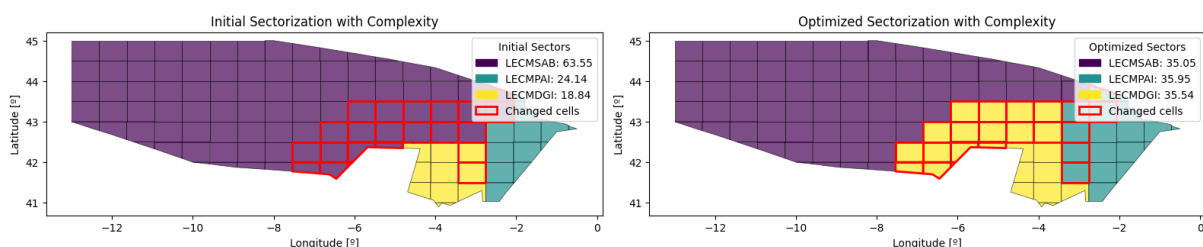


Figure 22. Initial sectorization versus optimized sectorization by pairs of adjacent cells, with the corresponding complexity values per sector (number_sectors = 3).

4. Discussion

The results obtained in this study demonstrate that airspace sector geometries can be effectively optimized using air traffic complexity as the primary driver, leading to a substantial improvement in workload balance across sectors. By formulating the sectorization problem at the cell level and applying a graph-partitioning optimization framework, the proposed methodology successfully redistributes complexity while preserving essential operational constraints such as sector connectivity and spatial coherence.

4.1. Effectiveness of Complexity-Based Sector Optimization

Across all tested configurations, the optimization algorithm consistently reduced both the variance of sector complexity and the gap between the most and least complex sectors. This confirms that cell-level complexity, computed through an adapted COMETA framework, is a suitable macroscopic indicator for guiding airspace design decisions. The results reinforce previous findings that air traffic complexity is closely linked to controller workload and capacity limitations, and that balancing complexity across sectors is a viable strategy for mitigating overload conditions.

In the most permissive optimization scenarios—where a higher number of cell movements was allowed—the algorithm achieved near-uniform complexity distributions. For

instance, in the five-sector case with a high maximum cell-change threshold, the difference between maximum and minimum sector complexity was reduced by an order of magnitude. These results highlight the theoretical potential of complexity-driven sectorization to achieve optimal workload balance when geometric constraints are relaxed.

However, such configurations also reveal an important limitation: optimizing solely for complexity can lead to sector geometries that are operationally unrealistic. The appearance of isolated cells or narrow sector intrusions, while mathematically valid, could introduce frequent sector re-entries and increase coordination demands between controllers. This confirms that complexity balance alone is not sufficient as a single optimization objective in real-world ATM applications.

4.2. Trade-Off Between Complexity Optimization and Geometric Coherence

As the optimization becomes more aggressive—either by increasing the allowed number of cell movements or by reducing constraints—the resulting sector shapes progressively deviate from established operational practices.

This trade-off is particularly evident when comparing single-cell movement optimization with the adjacent-cell-pair strategy. While the former achieves the lowest possible objective-function values, it often produces fragmented geometries. In contrast, the adjacent-cell-pair optimization yields slightly less optimal complexity distributions but significantly improves sector compactness and coherence. Importantly, the latter eliminates isolated cells and produces sector shapes that remain recognizable and consistent with current ACC layouts.

From an operational perspective, this balance is crucial. Sector geometry has a direct impact on controller mental models, coordination patterns, and handover stability. The results therefore suggest that an optimal sectorization solution should not be defined by complexity metrics alone, but rather by a compromise between workload balance and geometric robustness.

4.3. Influence of Sector Count and Operational Context

The analysis conducted for configurations with different numbers of sectors (three, four, and five) further illustrates the adaptability of the proposed methodology. As the number of sectors decreases, the algorithm must redistribute increasingly large amounts of complexity, resulting in more pronounced cell movements. Despite this, the adjacent-cell-pair approach maintains geometrically sound sectorizations even in highly aggregated configurations.

These findings emphasize the importance of selecting appropriate baseline sectorizations and time windows. The methodology is most meaningful when applied to configurations that reflect realistic operational modes for the analyzed traffic demand. Optimizing a low-sector configuration during high-demand periods, or vice versa, may yield mathematically balanced but operationally impractical results. This underscores the need for integrating traffic demand context and operational constraints when interpreting optimization outcomes.

4.4. Implications for Static and Dynamic Airspace Design

Although the proposed methodology is designed primarily for static sector optimization, the results suggest clear potential for application in DAC. By adjusting the temporal scope of the input data and tuning the maximum allowed sector modifications, the algorithm could support pre-tactical or tactical sector reconfiguration strategies.

Notably, the ability to limit the number of cell movements per sector provides a practical mechanism for gradual transitions between configurations, which is essential for maintaining controller situational awareness and trust. This feature aligns well with

operational requirements for DAC, where abrupt or large-scale airspace changes are generally undesirable.

5. Conclusions

After developing and analyzing the results produced by the different sector-optimization strategies applied to Madrid North ACC under a representative low-to-moderate complexity time window, the main conclusions of this study—regarding complexity distribution, geometric coherence of the sectors, and the operational applicability of the results within this scope—are summarized below:

- **Sector-complexity optimization:** The proposed methodology effectively distributes complexity among sectors, minimizing the gap between the most- and least-complex sectors. This improvement is most pronounced when geometric constraints are relaxed and a larger number of cell transfers is allowed.
- **Complexity–geometry trade-off:** A clear trade-off exists between optimizing complexity and preserving the geometric coherence of sectors. The more aggressively the objective function is optimized—and the more cells are moved—the greater the tendency to produce geometries that are less compatible with real-world operations, such as “isolated cells” or fragments of one sector surrounded by others.
- **More balanced optimization strategies:** The strategy of moving pairs of adjacent cells offers the best balance between achieving an efficient complexity distribution and maintaining coherent sector geometry. It prevents isolated cells and keeps sectors compact while still delivering substantial complexity optimization.
- **Importance of time-window adaptability:** The algorithm can tailor sectorization to real traffic levels in different time windows. For instance, during low-demand periods (e.g., 2–3 a.m.), it may be feasible to operate with fewer optimized sectors. Operational realities must always be considered when interpreting results: if day-to-day operations normally use nine sectors and the optimization begins from a configuration with only two sectors, the resulting complexity values will be extremely high, rendering that analysis meaningless.
- **Practical applications:** The algorithm is useful not only for handling high-complexity situations but also for improving day-to-day efficiency by distributing workload more fairly across sectors—even under less demanding conditions. By limiting the maximum number of cells that may be moved between sectors, it additionally supports pre-emptive aircraft-control actions.

Building on the progress made in this study, several future lines of work are proposed to enhance the representativeness, adaptability, and efficiency of airspace sectorization based on complexity values:

- Explore alternative optimization algorithms that could improve solution quality and reduce computation times.
- Extend the current two-dimensional framework toward a layered three-dimensional approach, progressively incorporating vertical stratification and flight-level-dependent sectorization.
- Investigate the most suitable cell size to balance geometric precision and computational cost. Future work will also explore the definition of quantitative geometric and operational indicators—such as sector compactness or isolated-cell metrics—to complement the qualitative assessment of operational robustness presented in this study.
- Include the explicit reporting of aggregated optimization metrics (minimum, maximum, and variance of sector complexity), together with a detailed analysis of computational performance for both complexity calculation and optimization, in order to assess scalability and suitability for pre-tactical applications.

- Conduct multi-scenario validations, including different ACCs, time periods, and higher-complexity traffic conditions, to evaluate the behavior of the methodology under more demanding operational settings and to assess its applicability to dynamic airspace configuration (DAC) contexts.

As a broader methodological improvement, a relevant extension of the proposed approach involves the treatment of cell complexity during the optimization process. In the current implementation, cell-level complexity is computed once at the beginning of the analysis and remains fixed throughout all optimization iterations. While this approach enables a tractable and robust redistribution of workload at a macroscopic level, future work will focus on developing enhanced complexity models capable of accounting for sector-dependent interaction effects. Recalculating—or efficiently updating—cell complexity after sector reassignment would allow the model to capture border-traffic interactions more accurately, improving realism in highly dynamic or congested scenarios.

These topics open up interesting opportunities for future research and applied developments.

Author Contributions: Conceptualization, J.M.A.L. and F.P.M.; methodology, C.G.A.; software, C.G.A.; validation, R.D.-A.J.; formal analysis, J.A.P.C.; investigation, M.Z.S.; resources, M.Z.S.; data curation, J.A.P.C.; writing—original draft preparation, C.G.A.; writing—review and editing, F.P.M.; visualization, J.M.A.L.; supervision, R.D.-A.J.; project administration, F.P.M.; funding acquisition, R.D.-A.J. All authors have read and agreed to the published version of the manuscript.

Funding: Acknowledgement to the Ministerio de Ciencia, Innovación y Universidades of Spain, as this study is part of the project AirST-FRAME (Grant PID2023-150782OB-C21 funded by MICIU/AEI/10.13039/501100011033 and FEDER, UE), a project funded as part of the Spanish program to foster, promote and encourage innovation and scientific work in universities.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not available for this article.

Acknowledgments: Acknowledgement to CRIDA for the collaboration on the project in which this research paper has been developed. We would also like to express our gratitude to CRIDA for providing the data necessary to carry out the work and obtain the results.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Xia, H.; Yao, W.; Zheng, Y.; Zhang, C.; Jiang, C. Airport Capacity Prediction and Optimization Based on Semi-Varying Coefficient and Dynamic Programming. *IEEE Access* **2025**, *13*, 67908–67924. [CrossRef]
2. Antulov-Fantulin, B.; Juričić, B.; Radišić, T.; Četek, C. Determining Air Traffic Complexity—Challenges and Future Development. *Promet—Traffic Transp.* **2020**, *32*, 475–485. [CrossRef]
3. dos Santos, R.; Rocha Murça, M. Airport capacity prediction and optimal allocation for strategic air traffic flow management at Sao Paulo/Guarulhos International Airport. *Case Stud. Transp. Policy* **2025**, *20*, 101414. [CrossRef]
4. Mogford, R.H.; Guttman, J.A.; Morrow, S.L.; Kopardekar, P. *The Complexity Construct in Air Traffic Control: A Review and Synthesis of the Literature*; CTA Incorporated: McKee City, NJ, USA, 1995. Available online: https://hf.tc.faa.gov/publications/1995-the-complexity-construct-in-air-traffic-control/full_text.pdf (accessed on 28 November 2025).
5. Schultz, M.; Lubig, D.; Asadi, E.; Rosenow, J.; Itoh, E.; Athota, S. Implementation of a Long-Range Air Traffic Flow Management for the Asia-Pacific Region. *IEEE Access* **2021**, *9*, 124640–124659. [CrossRef]
6. Ma, C.; Guleria, Y.; Alam, S.; Li, M. Deep reinforcement learning-based air traffic flow coordination in flow-centric airspace. *Adv. Eng. Inform.* **2025**, *65*, 103342. [CrossRef]
7. Balakrishnan, H.; Chandran, B. *Optimal Large-Scale Air Traffic Flow Management*; Massachusetts Institute of Technology: Cambridge, MA, USA, 2014. Available online: https://www.mit.edu/hamsa/pubs/BalakrishnanChandran_ATFM.pdf (accessed on 2 December 2025).
8. Gómez, S.; Goron, A.; Groza, A.; Letia, I. Assuring safety in air traffic control systems with argumentation and model checking. *Expert Syst. Appl.* **2016**, *44*, 367–385. [CrossRef]

9. Hilburn, B. *Cognitive Complexity in Air Traffic Control: A Literature Review*; Center for Human Performance Research: Houston, TX, USA, 2004.
10. Xia, Z.; Chen, C.; Hsieh, M.; Ma, G.; Liu, B.; Yu, X.; Xin, S.; Dong, L. A systematic review on human-AI hybrid systems and human factors in air traffic management. *J. Eng. Des.* **2025**, 1–49. Available online: <https://www.tandfonline.com/doi/full/10.1080/09544828.2025.2509056?af=R> (accessed on 3 December 2025). [CrossRef]
11. Zou, X.; Cheng, P.; An, B.; Song, J. Sectorization and Configuration Transition in Airspace Design. *Math. Probl. Eng.* **2016**, 2016, 6048326. [CrossRef]
12. Hind, H.; Omri, A.; Abghour, N.; Moussaid, K.; Rida, M. Dynamic airspace configuration: Review and open research issues. In Proceedings of the 2018 4th International Conference on Logistics Operations Management (GOL), Le Havre, France, 10–12 April 2018; pp. 1–7. Available online: <https://ieeexplore.ieee.org/document/8378093> (accessed on 30 November 2025).
13. Xu, C.; Zhang, K.; Liu, D.; Song, H. Trust of Airspace Configuration Transition Concerning the Fluctuation of Air Traffic. In Proceedings of the 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC), San Antonio, TX, USA, 3–7 October 2021; pp. 1–7. Available online: <https://ieeexplore.ieee.org/document/9594393> (accessed on 15 November 2025).
14. Xu, Y.; Prats, X.; Delahaye, D. Synchronised demand-capacity balancing in collaborative air traffic flow management. *Transp. Res. Part C Emerg. Technol.* **2020**, 114, 359–376. [CrossRef]
15. Memon, S.; Kim, M.; Shin, M.; Son, H. Dynamic based trajectory estimation and tracking in an uncertain environment. *Expert Syst. Appl.* **2021**, 177, 114919. [CrossRef]
16. Xuhao, G.; Junfeng, Z.; Zihan, P. Trajectory clustering for arrival aircraft via new trajectory representation. *J. Syst. Eng. Electron.* **2021**, 32, 473–496. [CrossRef]
17. Pérez Moreno, F.; Gómez Comendador, V.; Delgado-Aguilera Jurado, R.; Zamarreño Suárez, M.; Janisch, D.; Arnaldo Valdés, R. Methodology of air traffic flow clustering and 3-D prediction of air traffic density in ATC sectors based on machine learning models. *Expert Syst. Appl.* **2023**, 223, 119897. [CrossRef]
18. Pérez Moreno, F.; Ibáñez Rodríguez, F.; Gómez Comendador, V.; Delgado-Aguilera Jurado, R.; Zamarreño Suárez, M.; Arnaldo Valdés, R. Prediction of air traffic complexity through a dynamic complexity indicator and machine learning models. *J. Air Transp. Manag.* **2024**, 119, 102632. [CrossRef]
19. Xiao, M.; Zhang, J.; Cai, K.; Cao, X. ATCEM: A synthetic model for evaluating air traffic complexity. *J. Adv. Transp.* **2016**, 50, 315–325. [CrossRef]
20. Jurinic, T.; Juricic, B.; Antulov-Fantulin, B.; Samardzic, K. Defining Terminal Airspace Air Traffic Complexity Indicators Based on Air Traffic Controller Tasks. *Aerospace* **2024**, 11, 367. [CrossRef]
21. Brázdilová, S.; Cásek, P.; Kubalčík, J. Air traffic complexity for a distributed air traffic management system. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2011**, 225, 665–674. [CrossRef]
22. Kopardekar, P.; Schwartz, A.; Magyarits, S.; Rhodes, J. Airspace Complexity Measurement: An Air Traffic Control Simulation Analysis. *Int. J. Ind. Eng.-Theory Appl. Pract.* **2009**, 16, 61–70.
23. Göppel, S.; Schultz, M.; Itoh, E. Complexity Metrics for Moving Sectors. *IEEE Access* **2024**, 12, 139181–139198. [CrossRef]
24. Wang, H.; Song, Z.; Wen, R.; Zhao, Y. Study on evolution characteristics of air traffic situation complexity based on complex network theory. *Aerosp. Sci. Technol.* **2016**, 58, 518–528. [CrossRef]
25. Pérez Moreno, F.; Gómez Comendador, V.; Delgado-Aguilera Jurado, R.; M. Zamarreño Suárez, M.; Janisch, D.; Arnaldo Valdés, R. Determination of Air Traffic Complexity Most Influential Parameters Based on Machine Learning Models. *Symmetry* **2022**, 14, 2629. [CrossRef]
26. Klein, A.; Rodgers, M.; Keing, H. Dynamic FPAs: A new method for dynamic airspace configuration. In Proceedings of the 2008 Integrated Communications, Navigation and Surveillance Conference, Bethesda, MD, USA, 5–7 May 2008; pp. 1–11. Available online: <https://ieeexplore.ieee.org/document/4559176> (accessed on 14 November 2025).
27. Seergeva, M.; Delahaye, D.; Mancel, C.; Vidosavljevic, A. Dynamic airspace configuration by genetic algorithm. *J. Traffic Transp. Eng.* **2017**, 4, 300–314. [CrossRef]
28. Pérez Moreno, F.; Gómez Comendador, V.; Delgado-Aguilera Jurado, R.; Zamarreño Suárez, M.; Janisch, D.; Arnaldo Valdés, R. Dynamic model to characterise sectors using machine learning techniques. *Aircr. Eng. Aerosp. Technol.* **2022**, 94, 1537–1545. [CrossRef]
29. Criscuolo, P.; Perrotta, L.; Di Bitonto, G.; Grani, G. Enhanced Dynamic Airspace Configuration Algorithm. In Proceedings of the SESAR Innovation Days 2024, Rome, Italy, 12–15 November 2024. Available online: https://www.sesarju.eu/sites/default/files/documents/sid/2024/papers/SIDs_2024_paper_053%20final.pdf (accessed on 4 November 2025).
30. Ibáñez-Gijón, J.; Travieso, D.; Navia, J.; Montes, A.; Jacobs, D.; Frutos, P. Experimental Validation of COMETA Model of Mental Workload in Air Traffic Control. *J. Air Transp. Manag.* **2024**, 108, 102378. [CrossRef]

31. López de Frutos, P.; Rodríguez Rodríguez, R.; Zheng Zhang, D.; Zheng, S.; Cañas, J.; Muñoz de Escalona, E. COMETA: An Air Traffic Controller's Mental Workload Model for Calculating and Predicting Demand and Capacity Balancing. In *Lecture Notes in Computer Science (LNCS)*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 5–7.
32. Fiduccia, C.; Mattheyses, R. A Linear-time Heuristic for Improving Network Partitions. In Proceedings of the 19th Design Automation Conference (DAC), Las Vegas, NV, USA, 14–16 June 1982; pp. 175–181. Available online: <https://limsk.ece.gatech.edu/course/ece6133/papers/fm.pdf> (accessed on 19 November 2025).
33. Schulz, C. *High Quality Graph Partitioning*; Karlsruhe Institute of Technology (KIT): Karlsruhe, Germany, 2013. Available online: <https://publikationen.bibliothek.kit.edu/1000035713> (accessed on 24 November 2025).
34. D. Blondel, V.; Guillaume, J.; Lambiotte, R.; Lefebvre, E. Fast unfolding of communities in large networks. *arXiv* **2008**, arXiv:0803.0476. [[CrossRef](#)]
35. Gendreau, M.; Potvin, J. *Handbook of Metaheuristics*; Springer: Berlin/Heidelberg, Germany, 2010.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.