

Integer Programming-Based Airspace Sectorization for Terminal Maneuvering Areas with Convex Sectors

Tobias Andersson Granberg^{*}, Tatiana Polishchuk[†], Valentin Polishchuk[‡] and Christiane Schmidt[§]
*Communications and Transport Systems, Linköping University,
Campus Norrköping, SE-601 74 Norrköping, Sweden,
firstname.lastname@liu.se*

We present an airspace sectorization framework for Terminal Maneuvering Areas based on mixed integer programming. It incorporates an airspace complexity representation, as well as various constraints on the sectors' geometry, e.g., the requirement that points demanding increased attention from air traffic controllers should lie in the sector's interior to allow for enough time to resolve possible conflicts. Our method can enforce convex sectors. In contrast to earlier integer/constraint programming approaches, which used synthesis methods with variables per elementary airspace piece that were glued together to form sectors, our IP formulation uses a variable per potential edge on the sector boundary.

It is also the first step towards an integrated design of routes, the resulting complexity, and a sectorization. We present results for Stockholm Arlanda airport; and compare our results to convex sectorizations obtained by enumerating all possible topologies for a given number of sectors. This yields a proof of concept for the application of our highly flexible approach to Terminal Maneuvering Areas.

Nomenclature

a_0	=	area of the complete TMA
a_{LB}	=	lower bound on sector area
a_s	=	area of sector s
ATCC	=	air traffic control centres
ATCO	=	air traffic controller
\mathcal{C}	=	set of constraints on the sectors
c_1	=	constant in lower bound for sector area
c_2	=	constant in lower bound for sector taskload

^{*} Associate Professor, Communications and Transport Systems, Linköping University, Campus Norrköping, SE-601 74 Norrköping, Sweden

[†] Assistant Professor, Communications and Transport Systems, Linköping University, Campus Norrköping, SE-601 74 Norrköping, Sweden

[‡] Associate Professor, Communications and Transport Systems, Linköping University, Campus Norrköping, SE-601 74 Norrköping, Sweden

[§] Assistant Professor, Communications and Transport Systems, Linköping University, Campus Norrköping, SE-601 74 Norrköping, Sweden

γ	=	factor in objective function
E	=	set of edges
\mathcal{F}	=	objective function
$f_{i,j}$	=	signed area of the triangle formed by edge (i, j) and r
$G = (V, E)$	=	bi-directed graph
$h_{i,j}$	=	signed heat value of the triangle formed by edge (i, j) and r
h_q	=	heat value for heatmap point q
IATA	=	International Air Transport Association
$\ell_{i,j}$	=	length of edge (i, j)
MIP	=	mixed integer programming
$N(i)$	=	neighbors of i in G
$q_{j,m}^s$	=	indicator (switch clockwise/counterclockwise triangles)
$qabs_{j,m}^s$	=	absolute value of $q_{j,m}^s$
P	=	simple polygon
$p_{i,j}$	=	sign of $f_{i,j}$
$p_{i,j,m}$	=	sign of triangle formed by edge (i, j) and r_m
r, r_1, \dots, r_4	=	reference points
\mathcal{S}	=	sectors
$ \mathcal{S} $	=	number of sectors
\mathcal{S}^*	=	sectors incl. S_0
S_0	=	artificial sector that encompasses the complete boundary of P
S_i	=	sector i (subpolygon of P)
SUA	=	Special Use Airspace
t_{LB}	=	lower bound for the taskload in any sector
t_{UB}	=	upper bound for the taskload in any sector
TMA	=	Terminal Maneuvering Area
V	=	set of vertices
$w_{i,j}$	=	edge weight that depends on the heat-values of its endpoints
$y_{i,j,s}$	=	binary indicator for edge (i, j) being a boundary edge for sector s
$z_{i,j,m}^s$	=	$y_{i,j,s} \cdot qabs_{j,m}^s$

I. Introduction

Over the last decades air traffic volumes have increased, and projections indicate that the growth will continue: the International Air Transport Association (IATA) [1] estimates that the number of passengers will double from 2014 until 2034, and the Statistics and Forecasts (STATFOR) unit of EUROCONTROL [2] predicts an increase of 40-120% in flight movements in Europe from 2010 to 2030 (where the range stems from various scenarios from limited resources to strong economic growth). The resulting congestion is particularly concentrated on Terminal Maneuvering Areas (TMAs), that is, the area surrounding one or more neighboring aerodromes, as traffic converges towards a point near the runway. This results in significant delays and operation expenses. A possibility to increase capacity is to use the existing resources more efficiently. Thus, an optimized design of the TMA control sectors is essential for coping with ever increasing numbers of aircraft movements. The constructed sectorization partitions the airspace into a certain number of sectors.

A major challenge is posed by the central role of humans-in-the-loop: each sector is monitored by an air traffic controller (ATCO) who takes the highest level of responsibility to guarantee safe separation of aircraft in the sector at all times. This is in sharp contrast to fully distributed systems of other types of transportation, e.g., road traffic. The mental workload associated with working in such a complex system gives rise to the major constraints of an airspace sectorization: the workload should be balanced and not exceed thresholds for every single ATCO. Moreover, any design must be valid w.r.t. the sector shape and how the sector boundaries interact with standard flows and critical points. The problem of accessing the workload associated with a sector and deciding when it exceeds a safe threshold is a complex question that has been investigated by several researchers, see Loft et al. [3] for an overview.

One common requirement put on sectorizations, mentioned in a variety of sectorization papers (see, for example, [4–7]), is that the resulting sectors should be convex. The rationale behind desiring the convexity is that convex shapes are easy to "grasp" (learn, comprehend) by human controllers. Moreover, a (straight-line) flight cannot enter and leave a convex sector multiple times, which is a useful property of airspace design, since changing sectors involves communication overhead and switching the radio frequency. Many authors suggested automatic design methods for sectorization, but the vast majority cannot enforce the sector convexity. The exception are Voronoi-diagram methods, where a set of points are chosen as Voronoi sites and the sectors are their Voronoi cells; however, such methods can provide only rather crude control over the sector boundaries and do not give enough flexibility to ensure balanced workload among the sectors.

Today, the airspace layout at most airports is done manually, based on expert opinions. However, this will generally not yield an optimal sectorization for given criteria. Many papers presented suggestions for automatic design methods for airspace sectorization, see Section I.B for a detailed review, but the vast majority of approaches concentrates on en-route airspace.

In this paper, we present a general framework for integrating a complexity representation in a TMA's sectorization:

we can incorporate the convexity constraint, while also taking care of all "usual" load balancing requirements. We apply our techniques to split Stockholm TMA into convex sectors of approximately the same workload. To obtain our solution, we use mixed integer programming (MIP). Formulating the sectorization problem as an MIP is by far not new—many earlier works employed MIP or constraint programming to produce sectors, see Section I.B. However, all such prior work used synthesis methods in which the IPs had a variable per elementary piece of airspace, the MIP solution glued the pieces together to form the sectors. In contrast, we use a novel MIP formulation in which there is a variable per potential edge of the sector boundary. As input, we use the coordinates of the TMA, a numerical representation of the complexity, and constraints on the resulting sectorization. Our method is exact.

Our grid-based MIP formulation discretizes the search space by laying out a square grid in the airspace, and connecting each node to its 8 neighbors. We can easily enforce any of the options (a) general sectors without any convexity requirements, (b) sectors with an upper bound on angle deviation from only convex vertices, and (c) convex sectors. To the best of our knowledge, no other sectorization approach generates this flexibility, e.g., Gerdes et al. [8] start with convex Voronoi cells, and then adapt the cells to be non-convex. Adding the convexity constraint in the MIP is possible, but we only have a limited number of edge directions, which yields that each sector is at most an octagon. In addition, it is computationally expensive for many sectors. If we restrict ourselves to TMAs, which have a small number of sectors, we observe that only a limited number of topologies is possible. Thus, we suggest an alternative, computationally less expensive, approach that enumerates all these topologies and chooses the best of the operationally feasible solutions—and compare the solutions obtained by our MIP and by the enumeration.

Our formulation is a MIP, MIPs in general are NP-hard to solve, and the convexity constraints, if applied to a larger number of sectors, increase the computational cost. Hence, the strength of our model lies in the fact that we can formulate a large set of options as direct requirements—and solve these, while this flexibility comes at the cost of increased computation time in comparison to many other models, which are not able to require the full set of our options.

Apart from computing optimal sectorizations based on a given complexity representation, this framework is also the first step towards an integrated design of routes, the resulting complexity, and a sectorization, as we presented in [9]. Our approach is particular fitting for a TMA, as we—on the one hand—deal with a limited number of sectors, which makes solving our MIP computationally feasible, but—on the other hand—we want to be able to have a fine-grained approach that encompasses a multitude of constraints representing the ATCO's operational requirements, and has polygonal sectors (in contrast to elongated sectors with highly parallel traffic patterns where aircraft enter/leave the sector from a common direction).

The general approach can be transferred to en-route sectorizations, however, if these require both a larger grid and more convex sectors, the actual computation may become infeasible. On the other hand, en-route sectors may not require such a fine granularity, and we could use coarser grids. Further investigation is needed to draw a definite conclusion for the applicability for en-route airspace.

A. Roadmap

In the remainder of this section we review related work. In Section II, we discuss the terms taskload and workload, and formally introduce our problem in Section III. In Section IV we present our grid-based MIP formulation of the sectorization problem with various constraints, including the constraints for convex sectors, and review the complete MIP. We detail how the low number of TMA sectors allows us to enumerate the best balanced solutions for all different topologies in Section V. In Section VI, we present experiments for both the MIP and the topologies for Stockholm TMA. We conclude with conclusion and discussion in Section VII.

B. Related Work

Various papers considered automated airspace sectorization, for an extensive survey see Flener and Pearson [4]. Most research concentrated on sectorization of en-route airspace. Authors used fairly different definitions of taskload/workload, as no universal workload metric has been agreed on so far. US-based studies often have a focus on convective weather, which plays a smaller role in European studies.

The approaches can mostly be split in graph- or region-based models, where the former builds on partitioning a graph representing existing trajectories and constructing sector boundaries based on the partition, and the latter partitions the airspace into regions. Various sets of constraints on the resulting sectorization are considered, including constraints on the workload, the sector's geometry and size, and on the interaction with routes, e.g., each trajectory must intersect a sector for a minimum distance.

Kostitsyna [10] proves that most formulations of the airspace sectorization problem are NP-hard. In addition, she presents a method to redesign sectors that improves a given sectorization by locally adjusting sector boundaries. Sabhnani et al. [5] present a flow conforming design, where they consider constraints on flow-sector boundary crossings, on flow-flow crossings, on convexity, and constraints that forbid too acute sector angles; in addition, they integrate constraints on the interplay with Special Use Airspace (SUA). The authors then use a discretized search space—a uniform grid plus nodes approximating the medial axis of standard flows—and search for cuts in the complete graph on these nodes that conform to the constraints.

Recently, Gerdes et al. [8] presented an approach that first clusters flight data using fuzzy clustering, then computes a Voronoi diagram based on the resulting cluster centers, and then takes the controller workload into account using an evolutionary algorithm. They in particular make sure that the convex Voronoi cells are also able to handle non-convexity.

Other authors also presented geometry-based approaches. Xue's [11] design extends a pure Voronoi diagram computation. Brinton et al. [12] give a three stage algorithm that grows cells into clusters and straightens out boundaries in the final step. Their workload definition is based on dynamic density, see Kopardekar and Magyarits [13] for a comparison of four different dynamic density metrics. Conker et al. [14] present another three stage algorithm that uses a modified k -means clustering to obtain an initial sectorization, followed by an SLS heuristic to improve the workload

balance and a final phase that straightens out sector boundaries. Gianazza [15] combines elementary airspace modules for sectors, predicting the workload of configurations with a neural network. Leiden et al. [16] give a method based on a monitor alert parameter: once this threshold is exceeded for a sector it is split, and the algorithm picks the better of the solutions from a greedy bottom-up and a greedy top-down phase. The authors evaluate their sectorizations based on a transition cost, but do not include that cost in computing new sectors. Both Bloem and Gupta [17] and Kulkarni et al. [18] propose approximate dynamic programming methods. Drew [19] gives a MIP-based procedure that combines under-utilized sectors to larger sectors, without integrating shape constraints.

An approach using constraint programming is presented by Jägare [20]: hexagonal cells are merged to build sectors using constraints on the workload, entry points etc.

A graph-based constraint programming method is presented by Trandac et al. [6]. We refer to the survey of Allignol et al. [21] for constraint programming approaches in air traffic management. Various further graph-based approaches have been proposed. One of the oldest works on sectorization by Delahaye et al. [22] uses 3D Voronoi diagrams in post-processing to construct the sector boundaries. Martinez et al. [23] assign spatial cells to the nodes of a flow network, and partition the flow network subsequently until all sub-graphs comply with an upper bound on the workload (measured as a peak traffic count).

Zelinski and Li [24] present a comparison of seven algorithms to a baseline, the current solution: they compare the delay resulting from the constructed sectorization, and the traffic pattern and reconfiguration complexity. All but one algorithm achieve better results for the delay.

In contrast to all these approaches, we focus on a sectorization of a TMA, which has fewer sectors than the en-route airspace of air traffic control centres (ATCC). Thus, it is feasible to choose an approach that may become more computationally complex with increased number of sectors. On the other hand, our method integrates various constraints, and is flexible as it can be based on different complexity representations.

II. Taskload/Workload

ATCOs must first of all ensure safe separation of aircraft (i.e., ensure a minimum safety distance between aircraft). In addition, they enable aircraft to reach their destinations in a timely manner. To do so, they permanently monitor air traffic, anticipate and detect (potential) conflicts and perform various other tasks that drive an ATCO's mental workload. Both taskload and workload reflect the demand of the air traffic controller's monitoring task: the latter measures objective demands, the former reflects the subjective demand experienced during a task; in the remainder of this paper, we will refer to both as taskload, because we estimate this to be the more precise term for the available underlying data. Loft et al. [3] give an overview on different methods that study the elaborate problem of determining the taskload associated with a sector. Meckiff et al. [25] describe workload as a function of the air traffic's geometry, the operational procedures and practices used to handle the air traffic, and the characteristics of individual ATCOs. Yousefi

et al. [26] and Kopardekar and Magyarits [13] compare various air traffic management (ATM) modeling metrics and dynamic density metrics, respectively.

Recently, Zohrevandi et al. [27] presented a model for relating ATCO's taskload to the airspace complexity. They quantify the taskload as a weighted combination of ATCOs' clicks on the radar screen (weight \approx time for the task). Using linear regression the authors were able to explain terminal airspace complexity, given by eight complexity factors, about 70% better than the model by Djokic et al. [28] who used controller pilot data link communication and controller-pilot voice communications (regression analysis factor $R^2 = 0.84$). Thus, the weighted radar screen clicks is a comparatively good model for terminal airspace complexity. Zohrevandi et al. presented heat maps that visualize the density of weighted clicks. We use the heat maps for weighted clicks ([27]) as input for our sectorization. Our model does not depend on these specific maps, it is a general model that integrates complexity. In particular, it can be used as a building block in a common design of routes (with resulting complexity) and sectors (see [9]). Any other numerical representation of controller taskload can be used as a basis for our sectorization framework.

The radar screen clicks are closely related to aircraft trajectories in the considered airspace, thus, when we base our sectorization on the heat maps, we integrate the interaction with routes. This can also reflect different runway configurations: each runway configuration is connected to different routes and, hence, to different heat maps. We can choose to compute different sectorizations for different runway configurations, or we can aim to find the best sectorization for the complete heat map, that is, the time-weighted overlay of all different runway configurations.

III. Notation and Preliminaries

A *simple polygon* P is given by a set of n vertices v_1, v_2, \dots, v_n and n edges $v_1v_2, v_2v_3, \dots, v_{n-1}v_n, v_nv_1$ such that no pair of non-consecutive edges share a point. P is the closed finite region bounded by the vertices and edges. A *sectorization* of a simple polygon P is a partition of the polygon P into k disjoint subpolygons $S_1 \dots S_k$ ($S_i \cap S_j = \emptyset \ \forall i \neq j$), such that $\cup_{i=1}^k S_i = P$. The subpolygons S_i are called *sectors*.

Sectorization Problem:

Given: The coordinates of the TMA, defining a polygon P , the number of sectors $|S|$, a set \mathcal{C} of constraints on the resulting sectors, and possibly an objective function \mathcal{F} .

Find: A sectorization of P with $k = |S|$, fulfilling all constraints in \mathcal{C} , and possibly optimizing \mathcal{F} .

IV. Grid-Based MIP Formulation for the Sectorization Problem

We discretize the search space by laying out a square grid in the TMA. Every grid node is connected to its 8 neighbors. The resulting graph $G = (V, E)$ is bi-directed, i.e., for any two neighbors i and j both edge (i, j) and (j, i) are included in the edge set E . The length of an edge $(i, j) \in E$ is denoted by $\ell_{i,j}$.

The main idea for the sectors is to use an artificial sector, S_0 , that encompasses the complete boundary of P , using

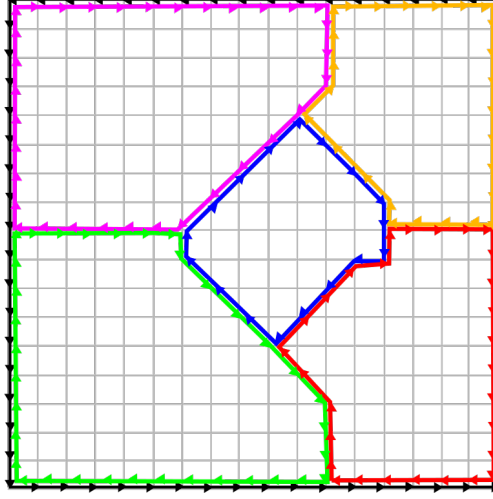


Fig. 1 The artificial sector S_0 is shown in black. A sectorization with five sectors is given. Edges are slightly offset to enhance visibility.

all counterclockwise edges. That is, we use sectors in $\mathcal{S}^* = \mathcal{S} \cup S_0$ with $\mathcal{S} = \{S_1 \dots S_k\}$. For all edges (i, j) used for sector boundary of any sector, we then enforce that also the opposite edge, (j, i) , is used for another sector, see Figure 1. Thus, all edges of an (interior) sector are clockwise.

We use decision variables $y_{i,j,s}$, where $y_{i,j,s} = 1$ indicates that edge (i, j) is a boundary edge for sector s . We add the following constraints:

$$y_{i,j,0} = 1 \quad \forall (i, j) \in S_0 \quad (1)$$

$$\sum_{s \in \mathcal{S}^*} y_{i,j,s} - \sum_{s \in \mathcal{S}^*} y_{j,i,s} = 0 \quad \forall (i, j) \in E \quad (2)$$

$$y_{i,j,s} + y_{j,i,s} \leq 1 \quad \forall (i, j) \in E, \forall s \in \mathcal{S}^* \quad (3)$$

$$\sum_{s \in \mathcal{S}^*} y_{i,j,s} \leq 1 \quad \forall (i, j) \in E \quad (4)$$

$$\sum_{(i,j) \in E} y_{i,j,s} \geq 3 \quad \forall s \in \mathcal{S}^* \quad (5)$$

$$y_{i,j,s} \in \{0, 1\} \quad \forall (i, j) \in E, \forall s \in \mathcal{S}^* \quad (6)$$

Equation (1) ensures that all counterclockwise oriented boundary edges belong to the artificial sector S_0 . Consistency between edges is given by Equation (2): if (i, j) is used for some sector, edge (j, i) has to be used as well. Equation (3) ensures that a sector cannot contain both edges (i, j) and (j, i) , that is, enclose an area of zero. In combination with Equation (2) it ensures that if an edge (i, j) is used for sector S_ℓ , the edge (j, i) has to be used by some sector $S_k \neq S_\ell$. Equation (4) enforces that one edge (i, j) cannot participate in two sectors. Equation (5) enforces a minimum size for

all sectors: each sector consists of at least 3 edges. Moreover, we add constraints on the degree of vertices on sector boundaries:

$$\sum_{l \in V: (l,i) \in E} y_{l,i,s} - \sum_{j \in V: (i,j) \in E} y_{i,j,s} = 0 \quad \forall i \in V, \forall s \in \mathcal{S}^* \quad (7)$$

$$\sum_{l \in N: (l,i) \in E} y_{l,i,s} \leq 1 \quad \forall i \in V, \forall s \in \mathcal{S}^* \quad (8)$$

Equation (7) ensures balance in all nodes, that is, all nodes have the same number of ingoing and outgoing edges. Equation (8) enforces that for each sector a node can have at most one ingoing edge.

A. Sectorization Constraints

The Equations (1)-(8) guarantee that the union of the $|\mathcal{S}|$ sectors completely covers the TMA, that the sectors are pairwise disjoint, and that each of them has a non-zero area. Of course, there are various other constraints for a sectorization, see for example the survey article of Flener and Pearson [4]. The constraints we consider can roughly be split in two categories: **balancing** and **geometric** constraints. The latter class could also incorporate interaction with routes.

1. Balancing Constraints

Balancing constraints are related to two factors: size/area and taskload. We consider the following constraints:

- a) *Balanced size*: The area of each sector, thus, the area that must be monitored by a single air traffic controller must be balanced out with the area of other sectors.
- b) *Bounded taskload*: There is an upper bound of movements that an air traffic controller can handle per time unit (hour), which might differ for air traffic controllers with varying experience. The taskload of each sector may not exceed this upper bound.
- c) *Balanced taskload*: The taskload of each sector, and, thus, of each air traffic controller, must be balanced out with the taskload of other sectors.

For constraint a, balanced size, we need to be able to associate an area with the sector selected by the boundary edges. The area of a polygon P with rational vertices is rational, and can be computed efficiently (see Fekete et al. [29]): we introduce a reference point r , and compute the area of the triangle of each directed edge e of P and r , see Figure 2. We then sum up the triangle area for all edges of the polygon: clockwise (cw) triangles contribute positive, counterclockwise (ccw) triangles contribute negative. Let $f_{i,j}$ denote the signed area of the triangle formed by edge (i, j) and r .

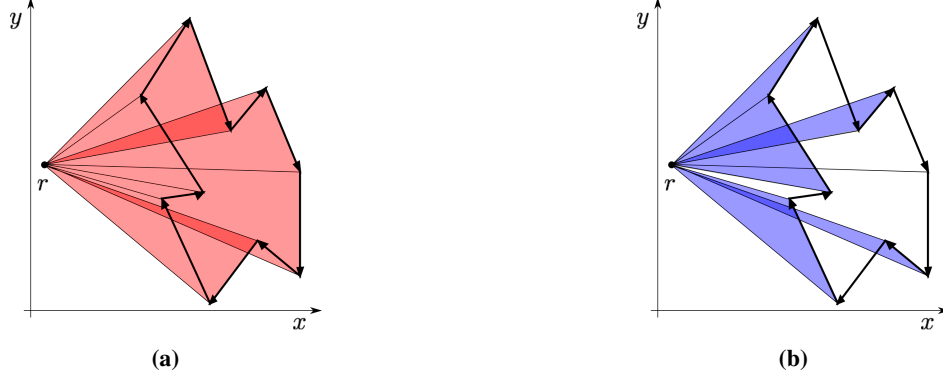


Fig. 2 Area of a polygon P (in bold): each (directed) edge of P forms an oriented triangle with a reference point r . Clockwise triangles contribute positive (a), counterclockwise triangles negative (b).

$$\sum_{(i,j) \in E} f_{i,j} y_{i,j,s} - a_s = 0 \quad \forall s \in \mathcal{S}^* \quad (9)$$

$$\sum_{s \in \mathcal{S}} a_s = a_0 \quad (10)$$

Equation (9) assigns the area of sector s to the variable a_s , equation (10) ensures that the sum of the areas of all sectors equals the area of the complete TMA, a_0 . If we want to balance the sector size, we add the following constraint to the MIP:

$$a_s \geq a_{LB} \quad \forall s \in \mathcal{S} \quad (11)$$

With this constraint we introduce a lower bound on the size of each sector, this could be a constant, or we can choose $a_{LB} = c_1 \cdot a_0 / |\mathcal{S}|$, where we can choose c_1 , e.g., $c_1 = 0.9$.

For constraints b and c we need to be able to associate a taskload with a sector. For this section we assume that a heatmap representing the controller's taskload (or any other complexity representation) is given, see Section II. Given this heatmap we overlay it with a grid, see Figure 3(a), extract the value at the grid points, see Figure 3(b), and use this discretized heatmap for further computations (see Figure 3(c), which shows the discretized heat map for the area of interest for P : the heat values at grid points for all grid points within some triangle of an edge e of P and the reference point r). We associate each discrete heatmap point, q , with a "heat value", h_q . Again, we consider triangles for each directed edge (i, j) of P and the reference point r , see for example Figure 3(d): we sum up the heat values for all grid points within the triangle ($h_q \forall q \in \Delta(i, j, r)$). The sign of the heat value for a triangle is determined by the sign of $f_{i,j}$, denoted by $p_{i,j}$, e.g., the triangle highlighted in Figure 3(d) is oriented cw (indicated by the red boundary), its heat value

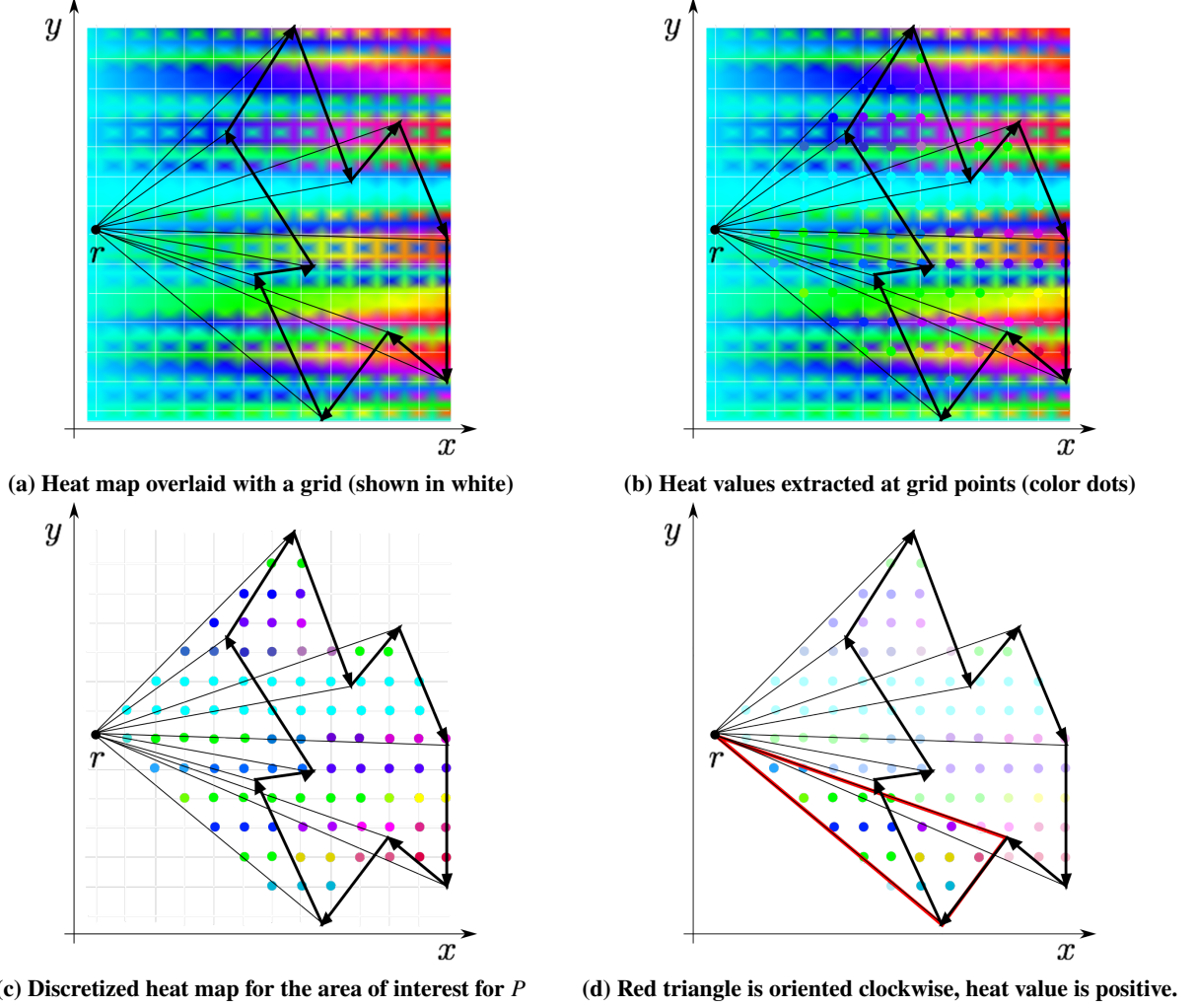


Fig. 3 Artificial heat map.

is positive ($p_{i,j} = +1$). Let $h_{i,j}$ denote the signed heat value of the triangle formed by edge (i, j) and r , that is:

$$h_{i,j} = p_{i,j} \sum_{q \in \Delta(i,j,r)} h_q.$$

If the taskload is of interest, we add Equation (12), which assigns each sector s a taskload t_s . In analogy to the balanced size, we add Equation (13) to achieve a balanced taskload. Here, $t_{LB} = c_2 \cdot t_0 / |\mathcal{S} \setminus S_0|$ with, e.g., $c_2 = 0.9$. For a bounded taskload we add Equation (14), with some fixed value t_{UB} giving the upper bound for the taskload in any sector. In case the upper bounds for different sectors differ because of controllers with varying experience, we can use sector-specific upper bounds t_{UB}^s for all $s \in \mathcal{S} \setminus S_0$.

$$\sum_{(i,j) \in E} h_{i,j} y_{i,j,s} - t_s = 0 \quad \forall s \in \mathcal{S} \quad (12)$$

$$t_s \geq t_{LB} \quad \forall s \in \mathcal{S} \quad (13)$$

$$t_s \leq t_{UB} \quad \forall s \in \mathcal{S} \quad (14)$$

2. Geometric Constraints

- \mathfrak{d}) *Connected sectors*: A sector must be a connected portion of airspace, that is, from each point in a sector each other point in a sector must be reachable via a path that runs only in the same sector. Thus, a sector should not be fragmented into a union of several unconnected airspace units, see Flener and Pearson [4].
- \mathfrak{e}) *Nice shape*: A sector's boundary should not be jagged. The sector's geometric shape should be easy to remember, see Flener and Pearson [4].
- \mathfrak{f}) *Interior Conflict points*: Points that require increased attention from air traffic controllers should lie in the sector's interior, that is within a certain threshold from the sector boundary.
- \mathfrak{g}) *Convex sectors*: The sectors should be convex. Where convexity can, for example, be defined either geometrically, that is, for any pair of points in the sector the straight line connection between these points is also fully contained in the sector, or trajectory-based, that is, no route enters the same sector more than once, see Flener and Pearson [4].

For constraint \mathfrak{d} we chose to use the length of the sector boundary as an objective function instead of using subtour elimination constraints, see Section IV.C.

Constraint \mathfrak{f} essentially asks to have points with higher complexity to be located within the sector's interior. We cannot use an absolute threshold heat value for the complexity of points on the sector boundary: we would like to enforce points of relatively high airspace complexity, represented by heat values, to be in the interior. We again make use of the objective function, see Section IV.C.

We take care of constraint \mathfrak{e} in post-processing: Given our constraint set $\mathcal{C} \subseteq \{\mathfrak{a}, \mathfrak{b}, \mathfrak{c}, \mathfrak{d}, \mathfrak{e}, \mathfrak{f}, \mathfrak{g}\}$, we solve the MIP with $\mathcal{C} \setminus \{\mathfrak{e}\}$ and then use shortcuts by removing vertices as long as the constraints in $\mathcal{C} \setminus \{\mathfrak{e}\}$ are not violated.

B. Constraint \mathfrak{g} : Convex Sectors

We can easily integrate convexity in our approach—a feature many other optimization approaches lack. For a convex sector, constraint \mathfrak{g} , there exist only one connected chain of edges with cw triangles, and one connected chain of edges with ccw triangles, see Figure 4(a). Unfortunately, the only-if part of that statement is not true, see Figure 4(b): there are non-convex polygons with a single chain of edges with ccw triangles and a single chain of edges with cw triangles. We can make use of the fact that we have only eight edge directions. For every direction of an incoming edge, there are

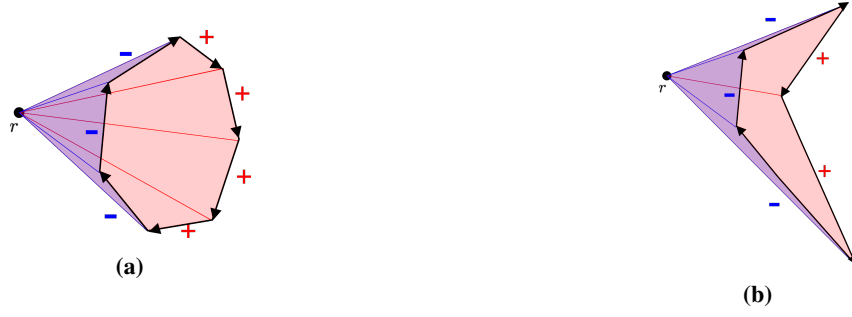


Fig. 4 A convex (a) and a non-convex (b) polygon.

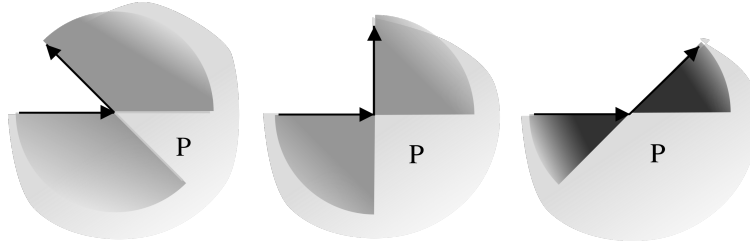


Fig. 5 Three outgoing edge directions yield a non-convex polygon (interior of P below ingoing edge).



Fig. 6 The green reference point (located in the dark-gray cone for the forbidden outgoing edge direction) detects the switch in triangle orientation. Altogether we have two chains of cw and two chains of ccw triangles.

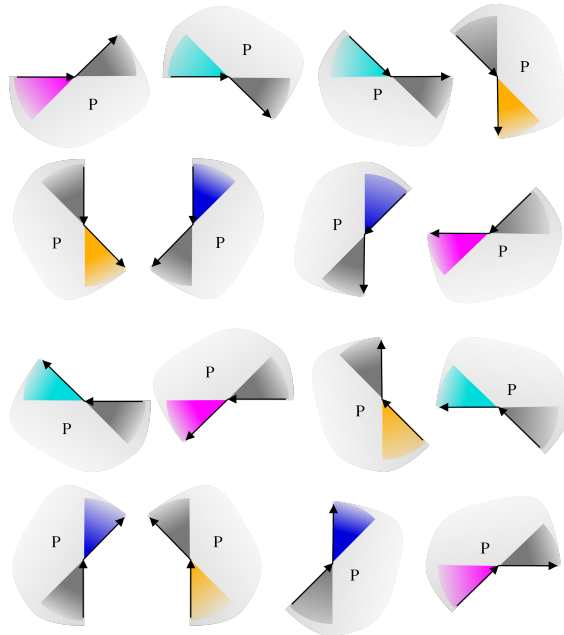


Fig. 7 Four (colored: pink, turquoise, blue, and orange) cones can be used for all edge directions.

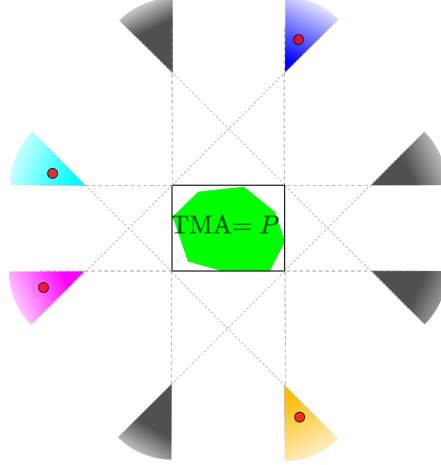


Fig. 8 Four red reference points for detecting non-convexity. The TMA, the polygon, is shown in green, the black box depicts the polygon's axis-aligned bounding box.

three directions of outgoing edges that are forbidden in a convex polygon, see Figure 5: there exist two open cones (indicated in gray) in which a reference point must be located to detect the switch. Thus, any reference point located in the dark gray cones, both the intersection of three cones, yields a switch in the triangle orientation. Consider Figure 6 for an example of a non-convex polygon, where the green reference point can detect the switch in triangle orientation. Hence, that polygon has more than a single chain of cw and a single chain of ccw triangles, and we can detect that it is not convex. If we consider all possible edge directions (eight incoming edge directions with two outgoing edge directions each for the dark gray cones from Figure 5), the cones for the necessary directions overlap, see Figure 7. Thus, we only need four points located in the intersection of these cones to cover all points of the grid, see Figure 8: we place one reference point in each of the four colored cones in Figure 8, and denote them by r_1, \dots, r_4 ($r = r_m$, for some $m \in \mathcal{M} = \{1, \dots, 4\}$). At least one of the r_m will result in a cw/ccw switch for non-convex polygons (as the green reference point in Figure 6 resulted in a cw/ccw switch). Let $p_{i,j,m}$ denote the sign of the triangle of the edge (i, j) and reference point r_m , $m \in \mathcal{M}$. We add:

$$q_{j,m}^s = \frac{1}{2} \left(\sum_{i:(i,j) \in E} p_{i,j,m} y_{i,j,s} - \sum_{l:(j,l) \in E} p_{j,l,m} y_{j,l,s} \right) \quad (15)$$

$$\forall s \in \mathcal{S}, \forall j \in V, \forall m \in \mathcal{M}$$

This equation assigns, for each sector, a value of -1, 0, 1 to each vertex. An interior vertex of either a chain of cw or ccw triangles has $q_{j,m}^s = 0$; if at j a chain with ccw (cw) triangles switches to a chain of cw (ccw) triangles $q_{j,m}^s = -1$ ($q_{j,m}^s = 1$). For a convex sector s , the sum over the $|q_{j,m}^s|$ for all sector vertices j is 2 for all reference points r_m ; for non-convex sectors this value is larger than 2 for at least one reference point r_m . Equations (16), (17) define the absolute

values. To enforce convexity (18) must hold. But, as two variables are multiplied in Equation (18), we cannot add it to the MIP. Instead, we use Equations (19)-(22) to define variables $z_{i,j,m}^s = y_{i,j,s} \cdot qabs_{j,m}^s \forall i, j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M}$, and add Equation (23), a reformulation of Equation (18).

$$qabs_{j,m}^s \geq q_{j,m}^s \quad \forall s \in \mathcal{S}, \forall j \in V, \forall m \in \mathcal{M} \quad (16)$$

$$qabs_{j,m}^s \geq -q_{j,m}^s \quad \forall s \in \mathcal{S}, \forall j \in V, \forall m \in \mathcal{M} \quad (17)$$

$$\sum_{i \in V} \sum_{j \in V} y_{i,j,s} \cdot qabs_{j,m}^s = 2 \quad \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (18)$$

$$z_{i,j,m}^s \geq 0 \quad \forall i, j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (19)$$

$$z_{i,j,m}^s \leq qabs_{j,m}^s \quad \forall i, j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (20)$$

$$z_{i,j,m}^s \leq y_{i,j,s} \quad \forall i, j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (21)$$

$$z_{i,j,m}^s \geq y_{i,j,s} - 1 + qabs_{j,m}^s \quad \forall i, j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (22)$$

$$\sum_{i \in V} \sum_{j \in V} z_{i,j,m}^s = 2 \quad \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (23)$$

Limited Deviation from 180° Interior Angle. We can also limit the angle deviation from only convex vertices. The rationale behind the latter being that an exemplary sectorization as shown in Figure 1 is acceptable. That is, the MIP formulation will allow us not only to strictly enforce the convexity of the sectors, but also to allow having a limited angle deviation from only convex vertices per sector.

If we allow usage of a few reflex vertices, we might limit the total deviation from a maximum interior degree of 180° of reflex vertices per sector (or penalize reflex vertices in the objective function). If we allow at most a deviation by 45°, that is, at most one reflex vertex with interior angle of 225°, we add instead of Equation (23):

$$\sum_{i \in V} \sum_{j \in V} z_{i,j,m}^s \leq 4 \quad \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (24)$$

$$\sum_{m \in \mathcal{M}} \sum_{i \in V} \sum_{j \in V} z_{i,j,m}^s \leq 10 (= 2 + 2 + 2 + 4) \quad \forall s \in \mathcal{S} \quad (25)$$

If we allow at most a deviation by 90°, that is, either one reflex vertex with interior angle of 270° or two reflex vertices with interior angle of 225° each, the upper bound in Equation (25) is 12 (= 2+2+4+4). We denote the resulting constraint “limited interior angle of reflex vertices” by f^r .

C. Objective Function

As opposed to most optimization approaches, in our case, it is not obvious what kind of objective function should be used. Cost functions used in literature, cp. [4], are, e.g., taskload imbalance (constraint a), and number of sectors (which we consider as input).

Because no obvious objective function exists, we consider different functions, all of which integrate constraint b.

1. Basic Objective Function

Our basic objective function is:

$$\min \sum_{s \in \mathcal{S}} \sum_{(i,j) \in E} \ell_{i,j} y_{i,j,s} \quad (26)$$

Minimizing the perimeter, that is using objective function (26), steers towards connected sectors if we only balance the area, that is, $a \in \mathcal{C}$, $b, c \notin \mathcal{C}$.

If we consider taskload, objective function (26) only yields connected sectors if c_2 in $t_{LB} = c_2 \cdot t_0 / |\mathcal{S} \setminus S_0|$ of Equation (13) allows it: for example $c_2 = 0.9$ may not allow a “ c_2 -balanced” sectorization with connected sectors, but if we allow for larger disparities between sectors, making a connected solution feasible, by lowering the parameter, e.g., $c_2 = 0.7$, we again obtain connected sectors. Essentially, this translates to: given the current complexity map a user must allow larger imbalances between controller’s taskload, if having connected sectors is a necessary condition.

2. Integration of Constraint f

In Section IV.A we described that the objective function also takes care of constraint f: interior conflict points. If $f \in \mathcal{C}$ we use the following objective function (an extension of the basic objective function (26)):

$$\min \sum_{s \in \mathcal{S}} \sum_{(i,j) \in E} (\gamma \ell_{i,j} + (1 - \gamma) w_{i,j}) y_{i,j,s}, \quad 0 \leq \gamma < 1 \quad (27)$$

Where $w_{i,j}$ represents an edge weight that depends on the heat-values of its endpoints. We choose:

(I) $w_{i,j} = h_i + h_j$

(II) $w_{i,j} = \sum_{k \in N(i)} h_k + \sum_{l \in N(j)} h_l$, with $N(i)$ denoting the neighbors of i in G (including i itself)

(I) ensures that relatively large heat-values are not located on the sector boundary, (II) pushes larger values further into the interior. An alternative to using objective function (27) instead of objective function (26) is to use a constraint with an upper bound W . This yields another restriction on a possible connected solution: we only obtain an optimal connected solution, if, given c_2 and W , there exists a feasible connected solution.

D. The Complete MIP

To enhance readability, we present the complete MIP in Figure 9. It includes the constraints for convexity, and not those for a limited deviation from an 180° interior angle. The number of variables grows linearly with the number of sectors.

V. Topologies

In a TMA only few sectors are needed, and only a limited number of topologies must be checked for a given number of convex sectors. Because adding the convexity constraints to the MIP is computationally expensive, we compare our MIP results for the convex sectorization of Stockholm TMA to those obtained by enumerating workload balanced convex sectorizations. That is, for each topology we compute the best balanced solution that fulfills all constraints in $\mathcal{C}_0 = \{a, b, c, d\}$. In particular, this approach does not have the constraint of limited (grid) edge directions, on the other hand the number of topologies to check rapidly increases with $|\mathcal{S}|$. Moreover, because we aim for perfect balance, the solutions may be worse than the MIP solutions w.r.t. constraint e.

For $|\mathcal{S}| = 2$ we search for the best chord that connects any two points on the TMA boundary, for $|\mathcal{S}| = 3$ we need to determine the location of four points, either all of them on the boundary (for two chords), or one center point connecting to the sector boundary in a Y-shape, see Figure 10.

VI. Experimental Study: Arlanda Airport

We consider Stockholm TMA, see [30–32] for the TMA boundaries. The model was solved using AMPL and CPLEX 12.6 on a single server with 24GB RAM and four kernels running on Linux. We used an 11×9 grid to yield and edge length of about 10 nm, which reflects operational requirements of TMA optimization recommended by LFV operational experts. In Subsection VI.A we present sectorizations without the convexity constraint g, compare constraints with and without this constraint in Subsection VI.B, show how we can limit the deviation from interior angles in Subsection VI.C, and present a comparison between convex sectorizations obtained with the MIP and based on the topology enumeration in Section VI.D.

With the given computational resources (server size) and in the current implementation for convex sectors the problem was typically solved in 10-30 minutes for 2 sectors, in 2 hours for 3 sectors, in 7-8 hours for 4 sectors, and in about 20 hours for 5 sectors. However, aspects like situational awareness of ATCOs require consistency in sectorizations, that is, we do not expect repeated dynamic recomputations within a day, but the strategic computation of several sectorizations for different traffic densities at least a few days ahead of operation. Hence, even a computation time of 20 hours is feasible. Moreover, currently Arlanda airport is mostly operated with 2-3 sectors, see Dervic and Rank [31] and Nilsson et al. [32].

Additionally, in the future both an optimization of the implementation, as well as a realization on more powerful

$$\begin{aligned}
& \min \sum_{s \in \mathcal{S}} \sum_{(i,j) \in E} (\gamma \ell_{i,j} + (1-\gamma) w_{i,j}) y_{i,j,s} \quad 0 \leq \gamma \leq 1 \quad (26)/(27) \\
& \text{s.t.} \\
& \sum_{s \in \mathcal{S}^*} y_{i,j,s} - \sum_{s \in \mathcal{S}^*} y_{j,i,s} = 0 \quad \forall (i,j) \in E \quad (2) \\
& y_{i,j,s} + y_{j,i,s} \leq 1 \quad \forall (i,j) \in E, \forall s \in \mathcal{S}^* \quad (3) \\
& \sum_{s \in \mathcal{S}} y_{i,j,s} \leq 1 \quad \forall (i,j) \in E \quad (4) \\
& \sum_{(i,j) \in E} y_{i,j,s} \geq 3 \quad \forall s \in \mathcal{S}^* \quad (5) \\
& \sum_{l \in V: (l,i) \in E} y_{l,i,s} - \sum_{j \in V: (i,j) \in E} y_{i,j,s} = 0 \quad \forall i \in V, \forall s \in \mathcal{S}^* \quad (7) \\
& \sum_{l \in V: (l,i) \in E} y_{l,i,s} \leq 1 \quad \forall i \in V, \forall s \in \mathcal{S}^* \quad (8) \\
& \sum_{(i,j) \in E} f_{i,j} y_{i,j,s} - a_s = 0 \quad \forall s \in \mathcal{S}^* \quad (9) \\
& \sum_{s \in \mathcal{S}} a_s = a_0 \quad (10) \\
& \sum_{(i,j) \in E} h_{i,j} y_{i,j,s} - t_s = 0 \quad \forall s \in \mathcal{S} \quad (12) \\
& t_s \geq t_{LB} \quad \forall s \in \mathcal{S} \quad (13) \\
& p_{j,m}^s = \frac{1}{2} \left(\sum_{(i,j) \in E} p_{i,j,m} y_{i,j,s} - \sum_{(j,l) \in E} p_{j,l,m} y_{j,l,s} \right) \quad \forall s \in \mathcal{S}, \forall j \in V, \forall m \in \mathcal{M} \quad (15) \\
& qabs_{j,m}^s \geq q_{j,m}^s \quad \forall s \in \mathcal{S}, \forall j \in V, \forall m \in \mathcal{M} \quad (16) \\
& qabs_{j,m}^s \geq -q_{j,m}^s \quad \forall s \in \mathcal{S}, \forall j \in V, \forall m \in \mathcal{M} \quad (17) \\
& z_{i,j,m}^s \leq qabs_{j,m}^s \quad \forall i,j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (20) \\
& z_{i,j,m}^s \leq y_{i,j,s} \quad \forall i,j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (21) \\
& z_{i,j,m}^s \geq y_{i,j,s} - 1 + qabs_{j,m}^s \quad \forall i,j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (22) \\
& \sum_{i \in V} \sum_{j \in V} z_{i,j,m}^s = 2 \quad \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (23) \\
& y_{i,j,0} = 1 \quad \forall (i,j) \in S_0 \quad (1) \\
& y_{i,j,s} \in \{0,1\} \quad \forall (i,j) \in E, \forall s \in \mathcal{S}^* \quad (6) \\
& 0 \leq z_{i,j,m}^s \quad \forall i,j \in V \forall s \in \mathcal{S}, \forall m \in \mathcal{M} \quad (19)
\end{aligned}$$

Fig. 9 Complete MIP.

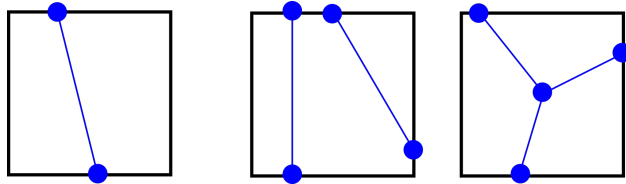


Fig. 10 Different topologies. Left: $|\mathcal{S}| = 2$. Middle, Right: $|\mathcal{S}| = 3$.

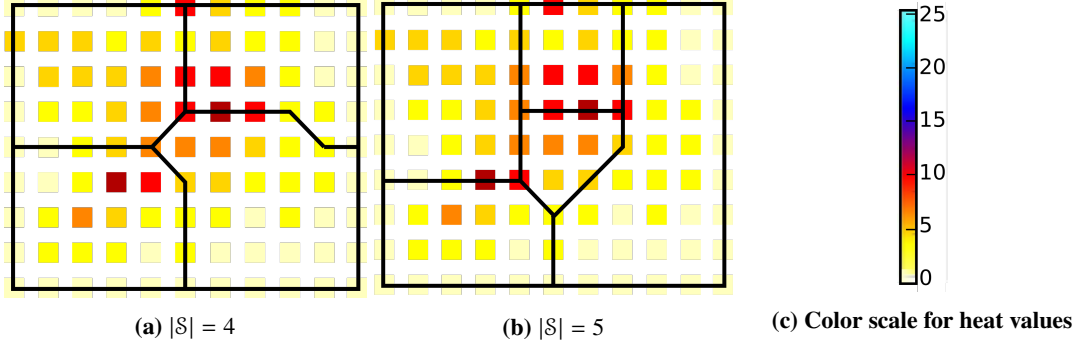


Fig. 11 $\mathcal{C}_1 = \{c, d, e\}$, with the objective function (26).

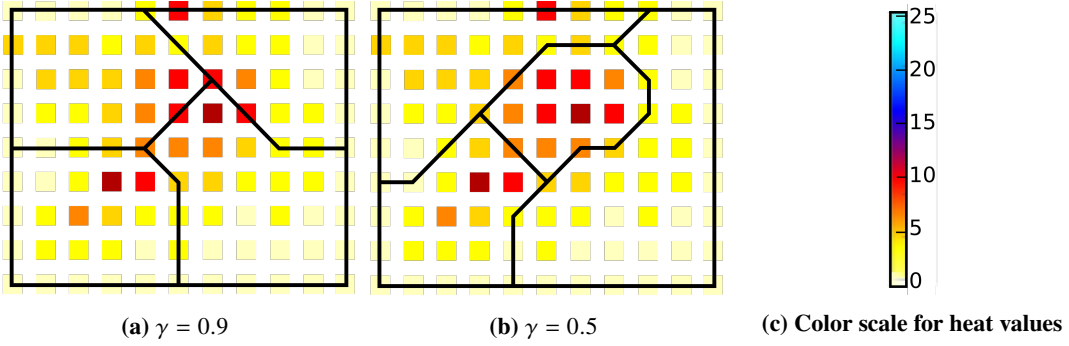


Fig. 12 $|\mathcal{S}| = 4$, and $\mathcal{C}_2 = \{c, d, e, f\}$, with the objective function (27), with $w_{i,j} = h_i + h_j$. In (b) the objective of interior conflict points/hotspot avoidance has a higher weight than in (a).

computational machines is possible. This can, for example, include to switch to a bigger server available at LiU (Tetralith server [33]), utilizing the Intel HNS2600BPB computer nodes with 32 CPU cores, 384 GiB, provided by the Swedish National Infrastructure for Computing (SNIC)).

A. Sectorization without Convexity Constraints

We consider $\mathcal{C}_1 = \{c, d, e\}$ and $\mathcal{C}_2 = \{c, d, e, f\}$, with $w_{i,j} = h_i + h_j$. If not mentioned otherwise we use $c_2 = 0.9$.

Each instance was run until a solution with less than 1% gap had been found, or for a maximum of one CPU-hour. No instance finished with an optimality gap of more than 6%. Figure 11 depicts solutions for $\mathcal{C}_1 = \{c, d, e\}$, that is, objective function (26). In Figure 12 we present our sectorizations for $|\mathcal{S}| = 4$, and $\mathcal{C}_2 = \{c, d, e, f\}$, that is, objective function (27) with $w_{i,j} = h_i + h_j$, for different values of γ .

Comparing Figure 12(a) with Figure 11(a) we can observe that the objective to have interior conflict points avoids the heat value of “10” in the center; for $\gamma = 0.5$, Figure 12(b), both hotspots are avoided by sector boundaries, that is, we yield a sectorization with interior conflict points.

Figure 13(a) shows that this instance does not have a “ c_2 -balanced” solution for $c_2 = 0.9$. On the other hand, if we substitute constraint c by a , we can ensure connected sectors, see Figure 13(b).

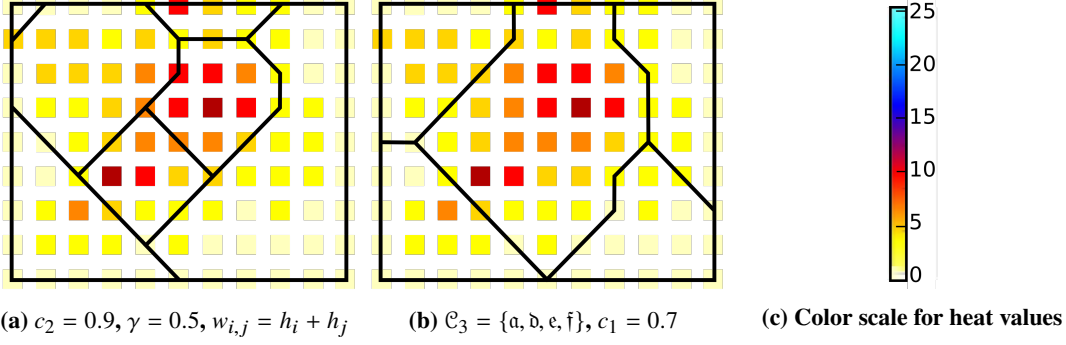


Fig. 13 $|\mathcal{S}| = 5$: (a) No “ c_2 -balanced” sectorization with connected sectors exists for $c_2 = 0.9$. (b) In contrast, if we look at area balance, we obtain connected sectors.

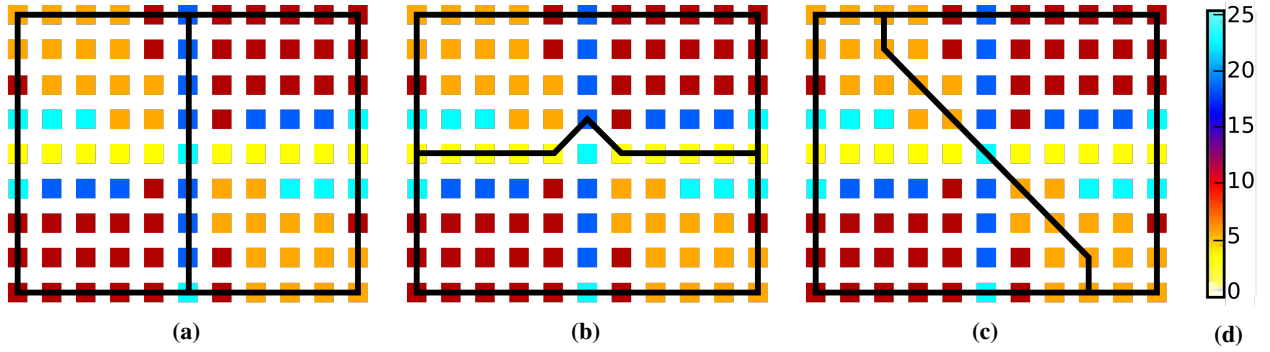


Fig. 14 Example to show the influence of choosing $w_{i,j}$.

Influence of Choosing $w_{i,j}$. In this subsection, we present an instance that is not connected to Stockholm TMA, to highlight the influence of the two definitions for the weight $w_{i,j}$. As this example is chosen to show that relation, we present results for $|\mathcal{S}| = 2$ to ease perception. Consequently, we pick one cut through the rectangle representing the square as the boundary between the two sectors. We consider $\mathcal{C}_2 = \{c, d, e, f\}$, and apply both $w_{i,j} = h_i + h_j$ and $w_{i,j} = \sum_{k \in N(i)} h_k + \sum_{l \in N(j)} h_l$ for $f \in \mathcal{C}_2$. That is, we use $\min_{s \in \mathcal{S}} \sum_{(i,j) \in E} (\gamma \ell_{i,j} + (1 - \gamma) w_{i,j}) y_{i,j,s}$, $0 \leq \gamma < 1$ as objective function. In Figure 14 (a) we use $\gamma = 1$, i.e., we do not try to avoid sector boundaries running through hotspots. Consequently, the cut runs along the shortest connection that balances workload. In Figure 14 (b) and (c) we use $\gamma = 0.5$. So, we use a linear combination of length and hotspot avoidance as our objective function. In (b) we use weight function (I), and in (c) weight function (II). That is, in (b) we want to avoid that the sector boundary runs through hotspots; in the solution, we see that the low heat points of value 2 are chosen. In the center the cut avoids the (turquoise) 25, and runs through one of the (inevitable) 20's (blue) on the central vertical line (as in the objective value we have $\frac{1}{2} \cdot 2 \cdot \sqrt{2} < \frac{1}{2} \cdot 5$). In (c), we also account for the weight of the neighbors of vertices on the cut. Thus, the cut that was optimal for (I) has a high weight, due to the neighboring 20's and 25's. The optimal solution now runs through the areas of low complexity, with heat values of 5 (orange), and avoids the hotspots.

B. Sectorizations with and without Convexity Constraints

We compare sectorizations without the convexity constraints to sectorizations with the convexity constraints. Each instance without the convexity constraints was run until a solution with less than 1% gap had been found, or for a maximum of one CPU-hour. No instance finished with an optimality gap of more than 1%. More sectors, and the convexity constraints made the problem harder to solve. Figure 15, left column, shows sectorizations without the convexity constraints, while the right column shows sectorizations with the convexity constraints. That is, the right column depicts sectorizations obtained by adding constraints (16), (17), and (19)-(23) to the MIP (that is, using $\mathcal{C}_3 = \{c, d, e, f, g\}$). We can observe, that for $|\mathcal{S}| = 4$, $\gamma = 0.2$, and $w_{i,j} = h_i + h_j$, that is, Figure 15(e), the sectorization does not have connected sectors, because of the extended objective function as explained earlier. If we increase γ to 0.8 in Figure 15(g), we obtain a connected sectorization with four sectors.

C. Sectorizations with Limited Interior Angle at Reflex Vertices

Here, we want to show that our approach can also be used to limit the deviation from a maximum interior degree of 180° of reflex vertices per sector. In Figure 16 we allow a deviation of at most 45° . This allows at most one reflex vertex with interior angle of 225° , which in the optimal solution is used by two out of the three sectors.

D. Convex Sectorizations: MIP and Topology Enumeration

Figure 17, left column, shows convex sectorizations obtained by adding the convexity constraints to the MIP. Figure 17, right column, shows sectorizations with convex sectors obtained by enumerating all solutions for the possible topologies ($\mathcal{C}_0 = \{a, b, c, d\}$). All sectorizations obtained by enumeration have perfect workload balance.

The convex sectorizations in the right column of Figure 17 yield a perfect workload balance for two (with two equally good solutions) and three sectors. The comparison of Figure 17(a) and Figure 17(b) shows that the MIP results in the perfectly balanced solution without any edge directions restriction. For larger values of $|\mathcal{S}|$, this restriction no longer gives the perfectly balanced solution, but for 3 sectors we present a solution with $c_2 = 0.95$, that is, the deviation between the sectors is very small. Moreover, the perfectly balanced solutions may be less attractive for constraint e, that is, the location of hot spots.

VII. Conclusion

In this paper we presented a sectorization method that balances or bounds the taskload for the sectors, based on a complexity representation. Moreover, we show how to integrate geometric constraints on the resulting sectors: convexity, connectedness, nice boundary, and interior conflict points.

We are able to easily enforce any of the options (a) general sectors without any convexity requirements, (b) sectors with an upper bound on angle deviation from only convex vertices, and (c) convex sectors.

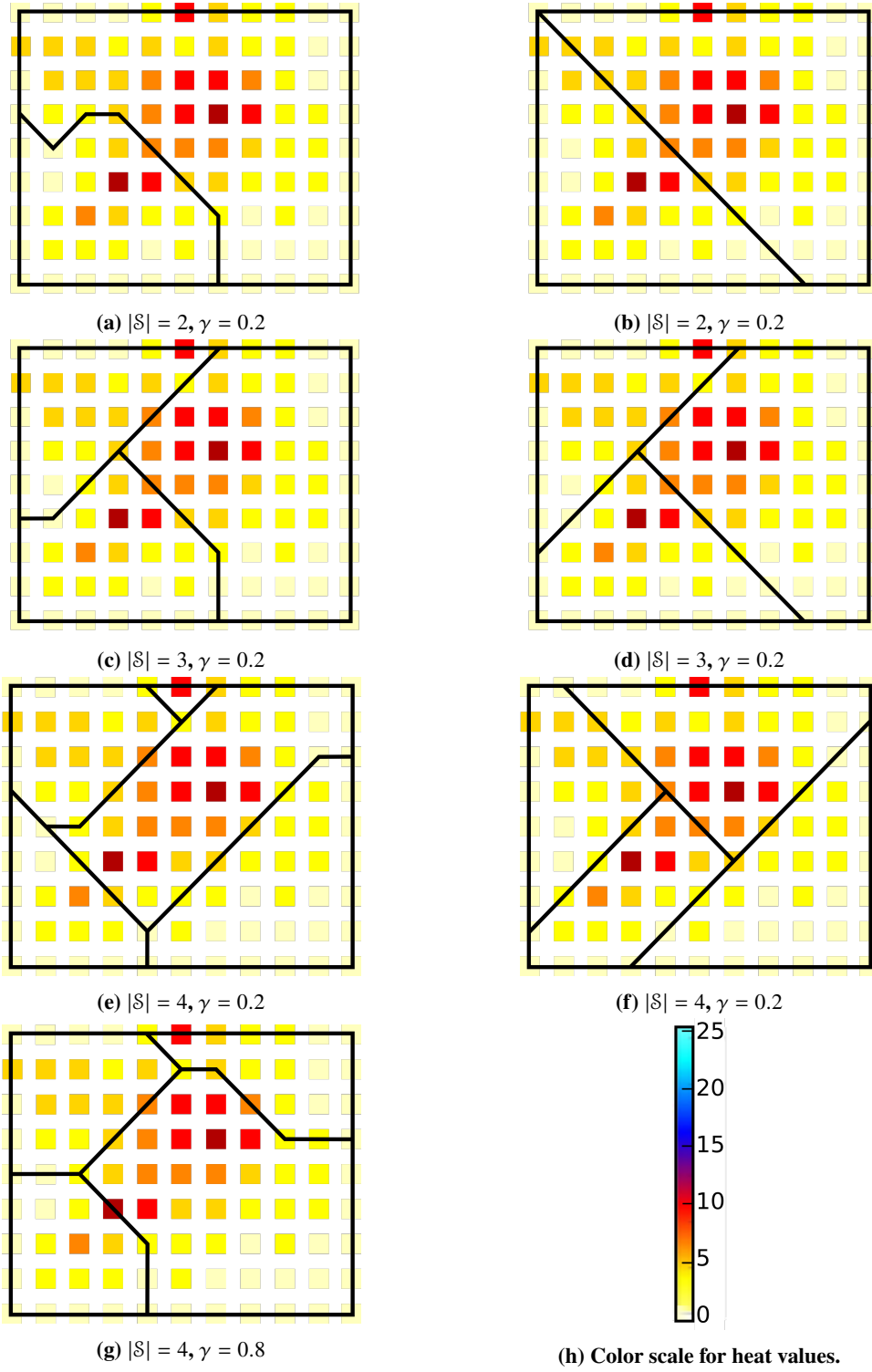


Fig. 15 Sectorizations obtained by MIP. Left column: Sectorizations without convexity constraints. Right column: Convex sectorizations. Left column: $\mathcal{C}_2 = \{c, d, e, f\}$. Right column: $\mathcal{C}_3 = \{c, d, e, f, g\}$. All with $c_2 = 0.6$ and $w_{i,j} = h_i + h_j$.

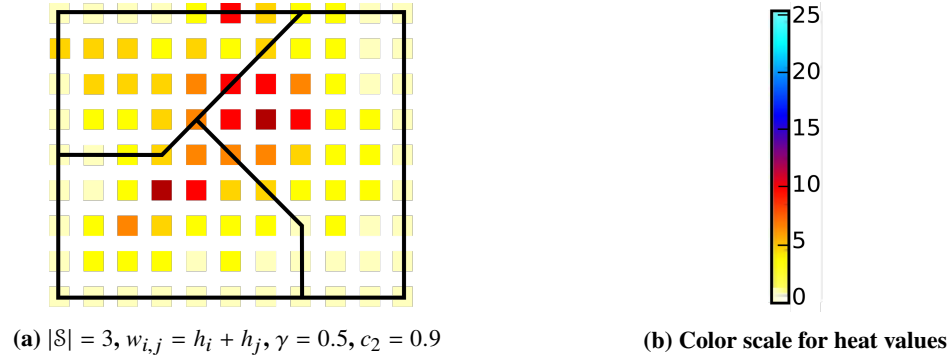


Fig. 16 Limited interior angle of reflex vertices: maximum deviation of 45° .

We apply our techniques to sectorize Stockholm Terminal Maneuvering Area (TMA), and highlight the options for interior conflict points with a tailored instance. We used heat maps for weighted controller clicks ([27]) as input for our sectorization. However, our model does not depend on these specific maps, any other available taskload or complexity measure can be used as input. As in a TMA only a few sectors are needed, and adding the convexity constraint is computationally expensive, we compare our results to convex sectorizations obtained by enumerating all possible topologies for a given number of sectors.

Funding Sources

Acknowledgments

We thank Billy Josefsson (project co-PI, LFV) and the members of the project reference group—Eric Hoffman (Eurocontrol), Håkan Svensson (LFV/NUAC), Anne-Marie Ragnarsson (Transportstyrelsen, the Swedish Traffic Administration), Anette Näs (Swedavia, the Swedish Airports Operator), Patrik Bergviken (LFV), Håkan Fahlgren (LFV)—for discussions of sectorization structure and design flexibility.

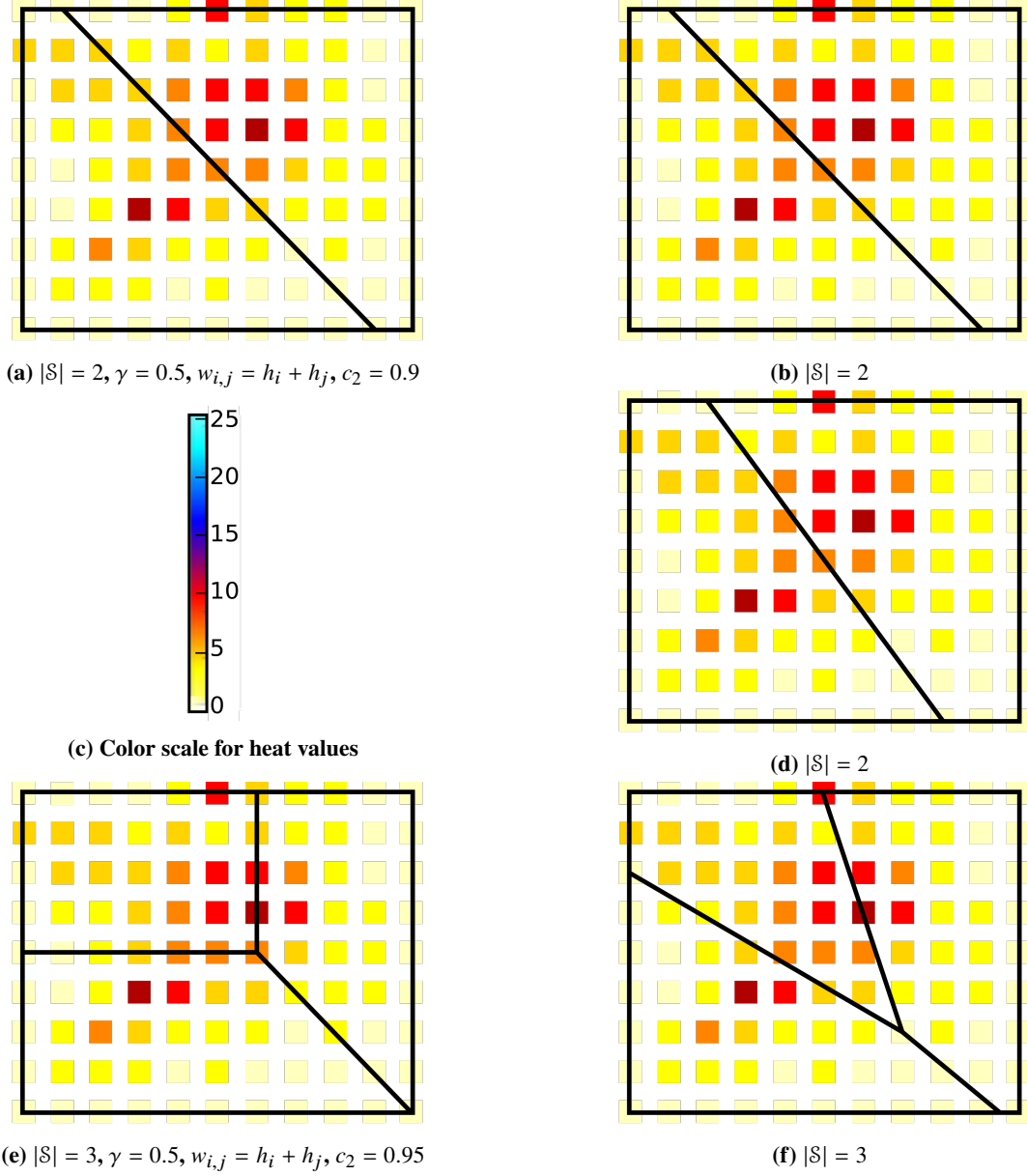


Fig. 17 Convex sectorizations. Left column: Convex sectorizations obtained by MIP ($\mathcal{C}_3 = \{a, b, c, d, e\}$). Right column: Convex sectorizations obtained by enumerating the topologies.

References

- [1] IATA, "IATA Air Passenger Forecast Shows Dip in Long-Term Demand," <http://www.iata.org/pressroom/pr/Pages/2015-11-26-01.aspx>, November 2015. Accessed on June 2, 2016.
- [2] EUROCONTROL, "EUROCONTROL Long-Term Forecast Flight Movements 2010 - 2030," 2010.
- [3] Loft, S., Sanderson, P., Neal, A., and Mooij, M., "Modeling and Predicting Mental Workload in En Route Air Traffic Control: Critical Review and Broader Implications," *Human Factors*, Vol. 49, No. 3, 2007, pp. 376–399. doi:10.1518/001872007X197017.
- [4] Flener, P., and Pearson, J., "Automatic Airspace Sectorisation: A Survey," *CoRR*, Vol. abs/1311.0653, 2013.
- [5] Sabhnani, G. R., Yousefi, A., and Mitchell, J. S., "Flow Conforming Operational Airspace Sector Design," *Proceedings of the 10th AIAA Aviation Technology, Integration and Operations (ATIO) Forum*, American Institute of Aeronautics and Astronautics, 2010. doi:<https://doi.org/10.2514/6.2010-9377>.
- [6] Trandac, H., Baptiste, P., and Duong, V., "Airspace sectorization with constraints," *RAIRO - Operations Research*, Vol. 39, No. 2, 2005, pp. 105–122. doi:<https://doi.org/10.1051/ro:2005005>.
- [7] Josefsson, B., Polishchuk, V., and Sedov, L., "Towards Simplified Optimal Sectorization," *7th SESAR Innovation Days*, 2017. doi:10.2829/233548.
- [8] Gerdes, I., Temme, A., and Schultz, M., "Dynamic airspace sectorization using controller task load," *6th SESAR Innovation Days*, 2016.
- [9] Granberg, T. A., Polishchuk, T., Polishchuk, V., and Schmidt, C., "A Framework for Integrated Terminal Airspace Design," *Proceedings of the Flight Simulation Conference: Modelling and Simulation in Air Traffic Management*, Royal Aeronautical Society, 2017.
- [10] Kostitsyna, I., "Balanced Partitioning of Polygonal Domains," Ph.D. thesis, Stony Brook University, August 2013.
- [11] Xue, M., "Three dimensional sector design with optimal number of sectors," *Proceedings of the AIAA Conference on Guidance, Navigation, and Control (GNC) and Modeling and Simulation Technologies (MST)*, American Institute for Aeronautics and Astronautics, 2010. doi:<https://doi.org/10.2514/1.51979>.
- [12] Brinton, C. R., Leiden, K., and Henkey, J., "Airspace sectorization by dynamic density," *Proceedings of the 9th AIAA Aviation Technology, Integration and Operations (ATIO) Forum*, American Institute of Aeronautics and Astronautics, 2009. doi:<https://doi.org/10.2514/6.2009-7102>.
- [13] Kopardekar, P., and Magyarits, S., "Dynamic Density: Measuring and predicting sector complexity," *Proceedings of the 21st Digital Avionics Systems Conference (DASC)*, Vol. 1, 2002, pp. 2.C.4–1–2.C.4–9. doi:<https://doi.org/10.1109/DASC.2002.1067920>.

- [14] Conker, R. S., Moch-Mooney, D. A., Niedringhaus, W. P., and Simmons, B. T., "New process for "clean" sheet airspace design and evaluation," *Proceedings of ATM 2007, the 7th USA/ Europe Air Traffic Management Research and Development Seminar*, 2007.
- [15] Gianazza, D., "Forecasting workload and airspace configuration with neural networks and tree search methods," *Artificial Intelligence*, Vol. 174, No. 7, 2010, pp. 530 – 549. doi:<https://doi.org/10.1016/j.artint.2010.03.001>.
- [16] Leiden, K., Peters, S., and Quesada, S., "Flight Level-based dynamic airspace configuration," *Proceedings of the 9th AIAA Aviation Technology, Integration and Operations (ATIO) Forum*, American Institute of Aeronautics and Astronautics, 2009. doi:<https://doi.org/10.2514/6.2009-7104>.
- [17] Bloen, M., and Gupta, P., "Configuring airspace sectors with approximate dynamic programming," *Proceedings of the 27th International Congress of the aeronautical sciences*, 2010.
- [18] Kulkarni, S., Ganesan, R., and Sherry, L., "Static sectorization approach to dynamic airspace configuration using approximate dynamic programming," *Integrated Communications, Navigation and Surveillance Conference (ICNS)*, 2011, 2011, pp. J2–1–J2–9. doi:<https://doi.org/10.1109/ICNSURV.2011.5935295>.
- [19] Drew, M. C., "A method of optimally combining sectors," *Proceedings of the 9th AIAA Aviation Technology, Integration and Operations (ATIO) Forum*, American Institute of Aeronautics and Astronautics, 2009.
- [20] Jägare, P., "Airspace sectorisation using constraint programming," , 2011. Report IT 11 021, Faculty of Science and Technology.
- [21] Allignol, C., Barnier, N., Flener, P., and Pearson, J., "Constraint programming for air traffic management: a survey," *Knowledge Engineering Review*, Vol. 27, No. 3, 2012, pp. pp 361–392. doi:<https://doi.org/10.1017/S0269888912000215>, URL <https://hal-enac.archives-ouvertes.fr/hal-00934670>.
- [22] Delahaye, D., Schoenauer, M., and Alliot, J.-M., "Airspace sectoring by evolutionary computation," *IEEE 1998, International Conference on Evolutionary Computation*, Anchorage, United States, 1998, pp. pp 218 – 223. doi:<https://doi.org/10.1109/ICEC.1998.699504>.
- [23] Martinez, S. A., Chatterji, G. B., Sun, D., and Bayen, A. M., "A weighted-graph approach for dynamic airspace configuration," *Proceedings of the AIAA Aviation Technology, Integration and Operations (ATIO) Forum*, American Institute of Aeronautics and Astronautics, 2007. doi:<https://doi.org/10.2514/6.2007-6448>.
- [24] Zelinski, S., and Lai, C. F., "Comparing methods for dynamic airspace configuration," *Proceedings of the 30th Digital Avionics Systems Conference (DASC)*, 2011. doi:<https://doi.org/10.1109/DASC.2011.6096043>.
- [25] Meckiff, C., Chone, R., and Nicolaon, J.-P., "The tactical load smoother for multi-sector planning," *Proceedings of the 2nd USA/ Europe Air Traffic Management Research and Development Seminar*, 1998.

- [26] Yousefi, A., Donohue, G. L., and Qureshi, K. M., “Investigation of en route metrics for model validation and airspace design using the total airport and airspace modeler (TAAM),” *Proceedings of the fifth USA/Europe Air Traffic Management R&D Seminar*, 2003.
- [27] Zohrevandi, E., Polishchuk, V., Lundberg, J., Svensson, Å., Johansson, J., and Josefsson, B., “Modeling and analysis of controller’s taskload in different predictability conditions,” *6th SESAR Innovation Days*, 2016.
- [28] Djokic, J., Lorenz, B., and Fricke, H., “Air traffic control complexity as workload driver,” *Transp. Res. Part C: Emerging Technologies*, Vol. 18, No. 6, 2010, pp. 930 – 936. doi:<https://doi.org/10.1016/j.trc.2010.03.005>.
- [29] Fekete, S. P., Friedrichs, S., Hemmer, M., Papenberg, M., Schmidt, A., and Troegel, J., “Area- and Boundary-Optimal Polygonalization of Planar Point Sets,” *EuroCG 2015*, 2015, pp. 133–136.
- [30] “Area Chart (Stockholm TMA),” https://www.aro.lfv.se/Editorial/View/5695/ES_AD_2_ESSA_4-1_en, 2019. Accessed on 02/05/2019.
- [31] Dervic, A., and Rank, A., “ATC complexity measures: Formulas measuring workload and complexity at Stockholm TMA,” , 2015.
- [32] Nilsson, J., Ehliar, J., Nilsson, J., Lundin, R., and Wagner, T., “Final report SISTMA,” <https://drive.google.com/file/d/0B6yfkI4NYP3LdEdvMmJ4VGcyQUE/view>, 2015.
- [33] “Tetralith server, NSC, Linköping University,” <https://www.nsc.liu.se/systems/tetralith/>, 2019. Accessed on 28/01/2019.
- [34] Granberg, T. A., Polishchuk, T., Polishchuk, V., and Schmidt, C., “A Novel MIP-based Airspace Sectorization for TMAs,” *12th International seminar for Air Traffic Management Research and Development (ATM R&D)*, 2017.
- [35] Granberg, T. A., Polishchuk, T., Polishchuk, V., and Schmidt, C., “Convex Sectorization—a Novel Integer Programming Approach,” *2017 Integrated Communications Navigation and Surveillance (ICNS) Conference*, 2017. doi:<https://doi.org/10.1109/ICNSURV.2017.8011883>.