Automatic Design of Aircraft Arrival Routes with Limited Turning Angle*

Tobias Andersson Granberg[1], Tatiana Polishchuk[2], Valentin Polishchuk[3], and Christiane Schmidt[4]

1 **Communications and Transport Systems, ITN, Linköping University, Sweden**
2 **Communications and Transport Systems, ITN, Linköping University, Sweden**
3 **Communications and Transport Systems, ITN, Linköping University, Sweden**
4 **Communications and Transport Systems, ITN, Linköping University, Sweden**

Abstract

We present an application of Integer Programming to the design of arrival routes for aircraft in a Terminal Maneuvering Area (TMA). We generate operationally feasible merge trees of curvature-constrained routes, using two optimization criteria: (1) total length of the tree, and (2) distance flown along the tree paths. The output routes guarantee that the overall traffic pattern in the TMA can be monitored by air traffic controllers; in particular, we keep merge points for arriving aircraft well separated, and we exclude conflicts between arriving and departing aircraft. We demonstrate the feasibility of our method by experimenting with arrival routes for a runway at Arlanda airport in the Stockholm TMA. Our approach can easily be extended in several ways, e.g., to ensure that the routes avoid no-fly zones.

1998 ACM Subject Classification G.1.6 Optimization: Integer programming, J.2 Physical Sciences and Engineering: Aerospace

Keywords and phrases Air Traffic Management, Standard Terminal Arrival Routes, Standard Instrument Departures, Integer programming, Turn constraints

Digital Object Identifier 10.4230/OASIcs.ATMOS.2016.9

1 Introduction

Air transportation experienced significant growth over the last decades, and the International Air Transport Association (IATA) projected that the number of passengers will double to reach 7 billion/year by 2034 [1]. On the one hand, this reflects a healthy economic and technological development, on the other hand, the increased volume of air traffic poses many challenges. The Terminal Maneuvering Area (TMA), i.e., the area surrounding one or more neighboring aerodromes, is particularly affected by congestion. Thus, designing arrival and departure procedures in the TMA to allow for a high throughput is crucial for handling higher and higher volumes of air traffic.

One of the main challenges in route planning for air traffic management (ATM) is brought by the central role of humans-in-the-loop: the planes in the air are constantly monitored and guided by air traffic controllers (ATCOs) taking the highest level of responsibility for safe separation between aircraft (in contrast with other types of transportation—e.g., cars on roads are a fully distributed system). This imposes additional constraints on the route design—the aircraft following the flight paths should give rise to "low complexity" traffic

* This research is funded by the grant 2014-03476 (ODESTA: Optimal Design of Terminal Airspace) from the Sweden's innovation agency VINNOVA and in-kind participation of LFV.

patterns and avoid creating conflict points where human attention would be constantly needed to resolve the potential loss of separation between the aircraft.

At most airports predesigned standard routes for departure and arrival are established. Currently, these Standard Instrument Departures (SIDs) and Standard Terminal Arrival Routes (STARs) are designed manually. This design is based on the airspace layout and incorporates constraints like avoidance of no-fly zones, manageability by ATCOs, and others. However, the manual design will generally not result in optimal routes for any specific criteria.

1.1 Problem description and Results

In this paper, we present a mathematical programming framework for finding optimal STAR merge trees. As part of the input to the problem, we are given locations of the entry points to the TMA, and the location and direction of the airport runway. In the output we seek an arrival tree that merges traffic from the entries to the runway, i.e., a tree that has the entries as leaves and the runway as the root (contrary to the common convention, we assume that the edges of this arborescence are directed from leaves to root).

The novelty of the considered problem is that the merge tree must take into account a set of operational constraints imposed on the STAR:

1. *No more than two routes merge at a point:* Points where routes merge require elevated level of attention from the controllers, and thus traffic complexity around the merges should be kept at a minimum [12]. This translates to the requirement that every vertex of the tree must have in-degree less than or equal to 2.
2. *Merge point separation:* Having two merge points very close to one another (even if at each of the points only two paths merge) effectively creates a small zone with several routes merging together—which is, again, undesirable for control. Thus, it is required that the separation between any two merge points is larger than a given distance threshold L [12].
3. *No sharp turns:* Aircraft dynamics impose a limit on the angle at which the routes can turn (bank angle) [4, p. 61]. We thus require that the turn from a segment of a route to the consecutive segment is never smaller than a given angle threshold α. If we would allow arbitrarily short edges this would still allow to simulate a sharp turn with a sequence of many short edges. Hence, we obtain the limited turning angle by combining the parameter α with the limit, L, on the minimum length for any edge [8]. We assume that the runway is the last segment of every route: this way, the turn onto the runway must also be larger than α—the aircraft must align with the runway before the touchdown.
4. *Obstacle avoidance:* The routes should not pass over a specified set of regions (we do not digress into the specific nature of the obstacles—they may be no-fly zones, noise-sensitive areas, etc.).
5. *STAR–SID separation:* Since TMA traffic consists of aircraft arriving to and departing from the same airport, it is unavoidable that the STARs and SIDs cross; to alleviate the potential conflicts between in- and outbound flights (and relieve ATCOs from constantly solving the conflict-free scheduling problem by delaying or speeding up the planes), the STAR–SID crossings should happen far from the runway, where the arriving and departing planes are sufficiently separated *vertically* due to the difference of descend and climb slopes [4, p. 25]. We thus require that SIDs are not crossed by STARs closer than a given distance d from the runway. Note that we assume that the SIDs are given in the input: arriving traffic is slower and thus has more room for maneuverability; in addition, it was confirmed with the practitioners that, e.g., in Stockholm TMA (our guinea pig), current SIDs are satisfactory while the STARs are in need of improvement.

1.1.1 Objective functions

It is natural to seek STARs featuring short flight routes for aircraft. Thus, one objective function in our optimization problem is the *total length of the routes* from the entry points to the runway. At the same time, the STAR tree should "occupy little space"—both from the ATCO perspective (to minimize attention attraction area), and to avoid spreading the noise and other environmental impact of aviation over the larger region. This can be modeled by requiring that the produced tree has small *total length of the edges*.[1] We consider both objectives, and call them *paths length* and *tree weight* respectively (these shorter names better emphasize the difference between the objectives).

We explore the *Pareto frontier* of our multicriteria optimization problem: we output a set of *Pareto optimal* solutions – those that cannot be improved with respect to one of the objectives without sacrificing on the other. (In particular, from the Pareto frontier, it is easy to obtain the solution optimizing any linear combination of the two objectives.) Our development may thus be viewed as a decision support tool, presenting the airspace designers with a set of options to choose from, and helping the decision makers in quantifying tradeoffs between the route length and complexity of the solution (exploring such tradeoffs is standard in multi-criteria optimization, and has been studied also in ATM, e.g., with respect to airspace capacity estimation [7]).

Note that if we compute a tree with minimum tree weight that complies with operational constraints 1 and 3, we would compute a minimum Steiner tree, that guarantees degree 3 for Steiner nodes and an angle of 120° between edges incident to such a node. If we add the operational constraint 2—the initial version we are interested in—we obtain a generalized version of the minimum Steiner tree problem ($L = 0$ representing the original problem). Thus, as the minimum Steiner tree problem is NP-complete, the same holds for our problem.

1.2 Roadmap

In the remainder of this Section we review related work. Section 2 presents our main tool: a grid-based integer programming (IP) formulation for the most basic version of the STAR finding problem; it considers only the constraints 1, 2 and 3. In Section 3 we apply the IP to produce STAR trees for RWY19L of Arlanda airport in Stockholm TMA. We output the trees on the Pareto frontier, and also show how the obstacle avoidance constraint 4 is handled. In Section 3.2 we experiment with adding a large number of entry points, which highlights the difference between the two objective functions. Handling STAR–SID separation (constraint 5) is explored in Section 3.3. Section 4 concludes the paper.

1.3 Related work

Automatic design of STARs/SIDs has been studied earlier, but to the best of our knowledge, finding optimal trees, taking into account the turn constraints, has not been considered before. In prior work, the routes have been constructed iteratively (one-by-one) and/or did not adhere to the full set of our constraints and/or did not route the traffic all the way to the runway.

[1] Note the differences between the objectives: in the former, the length of each edge of the tree is counted as many times as it is used by the leaf-to-root paths; in the latter, each edge is counted only once. In a graph the optimal solution for the former objective is the *Shortest-Paths tree*; the optimizer of the latter is a *Steiner tree* (or Steiner *arborescence*, in a directed graph).

Pfeil [10] focusses on weather forecast and the redesign or design of TMAs pertaining to different weather scenarios. The author develops an IP model to optimally choose fix locations and corresponding routes in fixed sectors, and to renegotiate sector boundaries. For the TMA design from scratch a two-step solution is presented: first optimal outer fixes are selected with an IP, then 3D routes between fix-runway pairs are chosen with a modified version of the A* algorithm. This defines some first chosen routes as obstacles for later routes, that is, the construction is sequential. All algorithms are presented for the US model of a TMA: two circles of different radii around the runway, where all merges and maneuvers are assumed to be performed within the inner circle and are not considered. The same TMA model is used by Prete et al. [12].

Krozel et al. [8] considered turn-constrained route planning for a *single* path; trees and the merging of paths are not considered. Zhou et al. [16, 15] also construct single, individual routes (not arrival merge trees) through weather-impacted TMA. Similarly, Visser and Wijnen [13] construct single routes, the objective in their work is to minimize noise impact.

Choi et al. [2] present results on STAR merging, testing the impact of different merge topologies on scheduling of aircraft along the routes; however, the actual location of merge points is not of interest and turn constraints are not taken into account. In [11], circular arcs are used to iteratively construct a curvature-constrained tree; however, no optimality guarantee is given for the solution.

An example for mainly manual STAR construction is Micallef et al.'s [9] design for Malta international airport.

2 Grid-based IP formulation

We discretize the search space by laying out a square grid in the TMA, and snapping the locations of the entry points and the runway onto the grid; let $\mathcal{P}$ denote the set of (snapped) entry points, and r the runway. The side of the grid pixel is equal to our lower bound L on the distance between route vertices—this ensures that the merge point separation (constraint 2) is satisfied by any path in the grid. Every grid node is connected to its 8 neighbors, thus forming a graph $G = (V, E)$. The graph is bi-directed, i.e., for any two neighboring nodes i and j, both edges (i, j) and (j, i) exist in E; the only exceptions are the entry points (they do not have incoming edges) and r (it does not have outgoing edges). The length of an edge $(i, j) \in E$ is denoted by ℓ_{ij}.

Our model is an integer program, which in general is NP-hard, but we are able to solve relevant sizes of instances in Section 3. Our IP formulation is based on the flow IP formulation for Steiner trees [14, 5] (Min Cost Flow Steiner arborescence). We use decision variables x_e that indicate whether the edge e participates in the STAR. In addition, we have flow variables: f_e gives the flow on edge $e = (i, j)$ (i.e., the flow from i to j). The constraints are given in Equations (1)-(4):

$$\sum_{k:(k,i)\in E} f_{ki} - \sum_{j:(i,j)\in E} f_{ij} = \begin{cases} |\mathcal{P}| & i = r \\ -1 & i \in \mathcal{P} \\ 0 & i \in V \setminus \{\mathcal{P} \cup r\} \end{cases} \tag{1}$$

$$x_e \geq \frac{f_e}{N} \qquad \forall e \in E \tag{2}$$

$$f_e \geq 0 \qquad \forall e \in E \tag{3}$$

$$x_e \in \{0, 1\} \qquad \forall e \in E \tag{4}$$

where N is a large number (e.g., $N = |\mathcal{P}|$).

Equation (1) ensures that a flow of $|\mathcal{P}|$ reaches the runway r, a flow of 1 leaves every entry point, and in all other vertices of the graph the flow is conserved. Equation (2) enforces edges with a positive flow to participate in the STAR. The flow variables are non-negative (Equation (3)), the edge variables are binary (Equation (4)).

Our two objective functions—paths length and tree weight—are given in Equations (5) and (6), respectively:

$$\min \quad \sum_{e \in E} \ell_e f_e \tag{5}$$

$$\min \quad \sum_{e \in E} \ell_e x_e \tag{6}$$

2.1 Degree constraints

We extend the above vanilla MinCostFlow Steiner tree IP to handle the constraints defined in Section 1. First, we ensure that the outdegree of every node is at most 1 and that the maximum indegree is 2 (operational constraint 1):

$$\sum_{k:(k,i) \in E} x_{ki} \leq 2 \quad \forall i \in V \setminus \{\mathcal{P} \cup r\} \tag{7}$$

$$\sum_{j:(i,j) \in E} x_{ij} \leq 1 \quad \forall i \in V \setminus \{\mathcal{P} \cup r\} \tag{8}$$

$$\sum_{k:(k,r) \in E} x_{kr} = 1 \tag{9}$$

$$\sum_{j:(r,j) \in E} x_{rj} \leq 0 \tag{10}$$

$$\sum_{k:(k,i) \in E} x_{ki} \leq 0 \quad \forall i \in \mathcal{P} \tag{11}$$

$$\sum_{j:(i,j) \in E} x_{ij} = 1 \quad \forall i \in \mathcal{P} \tag{12}$$

Equations (9) and (10) ensure that the runway r has one ingoing and no outgoing edges, respectively; Equations (12) and (11) make sure that each entry point has one outgoing and not ingoing edge, respectively; the maximum indegree of 2 for all other vertices is given by Equation (7), the maximum outdegree of 1 by Equation (8).

2.2 Turn angle constraints

If an edge $e = (i, j)$ is used, all outgoing edges at j must form an angle of at least α with e (operational constraint 3). Let A_e be the set of all outgoing edges from j that form an angle $\leq \alpha$ with e, i.e., $A_e = \{(j, k) : \measuredangle ijk \leq \alpha, (j, k) \in E\}$, and let $a_e = |A_e|$, see Figure 1. We add the constraint:

$$a_e x_e + \sum_{f \in A_e} x_f \leq a_e \quad \forall e \in E \tag{13}$$

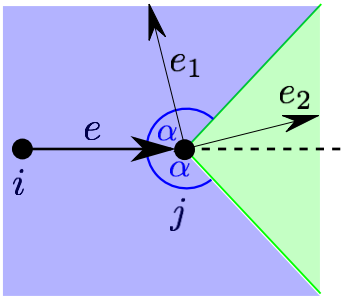
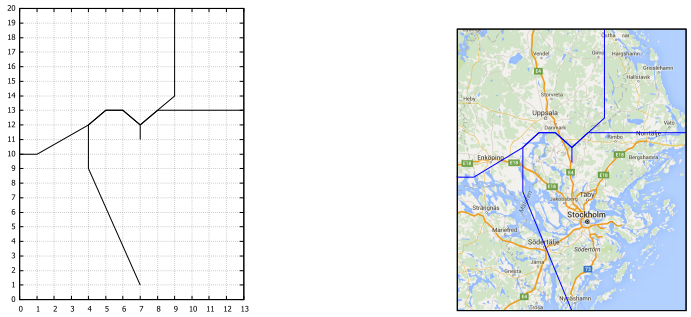


Figure 1 Limited turn: if edge $e = (i, j)$ is used, only edges within the light green region are allowed, that is, edges with an angle of at least α with e. If edges in the light blue region, A_e, are used, x_e must be set to zero. Here: $e_1 \in A_e, e_2 \notin A_e$.



(a) (b)

Figure 2 (a) A STAR in the grid with $L = 6$nmi and turns sharper than a threshold angle α of 135 degrees not allowed. (b) The same STAR underlaid with a map of the Stockholm region.

2.3 SID constraints

In Section 1 (constraint 5) we described the necessity to exclude conflicts between arriving aircraft on the constructed STAR and departing aircraft on the given SID. Specifically, we disallow STAR edges to intersect SID edges within distance d from the runway. That is, we consider the set of all points on SID edges that along the SID have a distance of at most d to the runway, and delete all edges from E that intersect with this set. Formally: let $dist_{\text{SID}}(x, y)$ denote the distance of two points x, y along the SID, $\text{SID}(d) = \{p \in \text{SID} : dist_{\text{SID}}(p, v_{STAR}) \leq d\}$, $p(\{i, j\}) = \{i, p\}$ for $p \in \{i, j\}$ a subset of edge $\{i, j\}$ up to a given point p, $E_{\text{SID}} = \{\text{edges } \{i, j\} \in \text{SID} : i, j \in \text{SID}(d)\} \cup \{p(\{i, j\}) : \{i, j\} \in \text{SID}, i \in \text{SID}(d), j \notin \text{SID}(d), dist_{\text{SID}}(p, v_{STAR}) = d\}$, $F = \{e \in E : e \cap E_{\text{SID}} \neq \emptyset\}$, and $E' = E \setminus F$. If we integrate the SID constraints, we simply use the edge set E' instead of E.

3 Experimental Study: Arlanda Airport

In this section we present solutions to our IP for the STAR design for the Arlanda airport in Stockholm TMA. The TMA is managed by LFV (Swedish Air Navigation Service Provider), and is manually designed based on expert opinion. In 2012 LFV ordered an initial study that confirmed the need to investigate possibilities of improving the TMA design with the help of advanced optimization tools. Currently, a collaboration between LFV and Linköping University (LiU) researchers this topic within the ODESTA (Optimal Design of

Figure 3 The path through the grid (left) is shortcut (right).

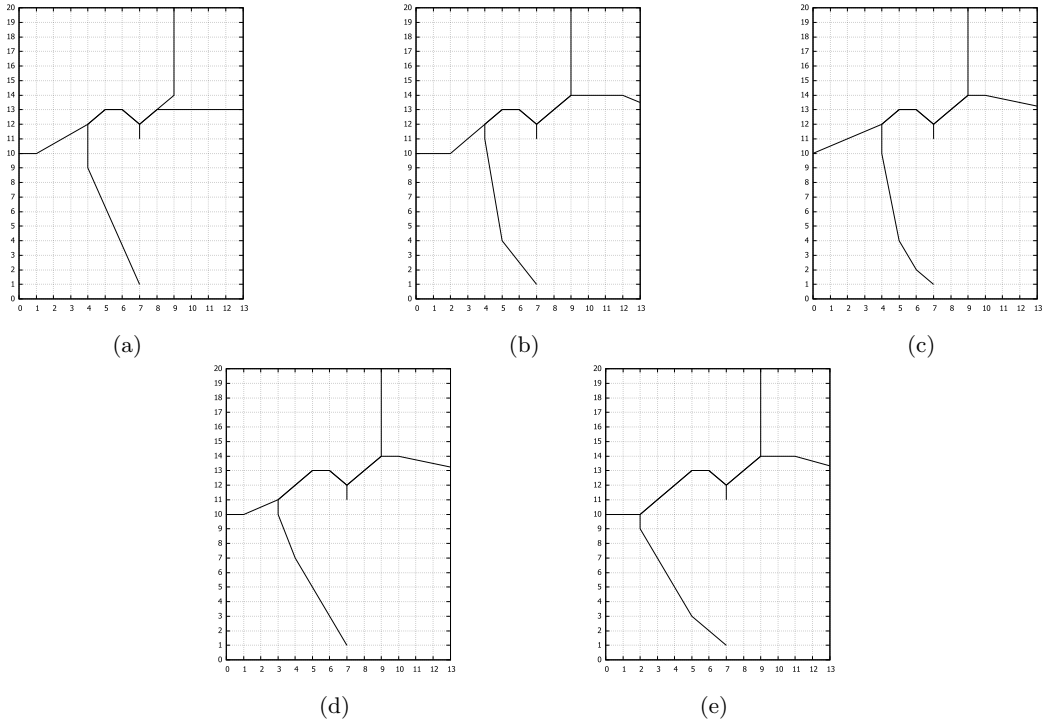


Figure 4 Total paths length: (a) 286, (b) 292.5, (c) 295.75, (d) 299, (e) 302.25. Tree weight: (a) 232.7, (b) 230.1, (c) 227.37, (d) 222.95, (e) 222.17.

Terminal Airspace) project. A reference group with members from LFV, EUROCONTROL, Trafikverket (Swedish Traffic Agency), Swedavia (a company that owns and operates the major airports of Sweden), and Transportstyrelsen (Swedish Transportation Authority) is a vital part of this project.

We consider Arlanda's runway 19L, and the four main entry points NILUG, XILAN, HMR, and ARS. We solve an IP with constraints (1)-(4), (7)-(12), (13), and objective functions (5) and (6) on square grids of size 14×20 and 25×30. The entry points and the runway are snapped to the closest grid vertices. Figure 2 shows the STAR in the grid and overlaid on a map for Stockholm's TMA. We use 8 edge directions: horizontal and vertical grid edges, and grid diagonals. After the solution is found, we postprocess it in order to have smoother paths (not restricted to the grid): we do shortcuts by removing vertices as long as the turn angle constraint is not violated (Figure 3).

Figure 4 shows Pareto optimal solutions, the Pareto frontier is shown in Figure 5. Table 1 presents the associated CPU times and number of branch and bound nodes.

The IP was solved using AMPL and Gurobi [6] on a single server with 24GB RAM, processor Intel(R) Xeon(R) CPU E5-2650 0 @ 2.00GHz and 64-bit operating system.

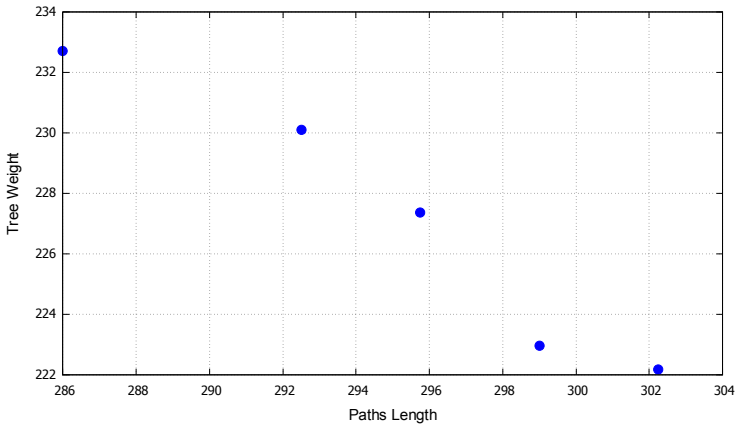


Figure 5 Pareto optimal solutions, the corresponding STARs are shown in Figure 4. The x-axis shows the total paths length, the y-axis the tree weight.

Table 1 The CPU time and number of branch and bound nodes to find the Pareto-optimal trees, from heaviest to lightest.

CPU time, s	993	2950	6949	7793	9447
# B&B nodes	8912	24342	33314	45244	52740

3.1 Obstacle avoidance

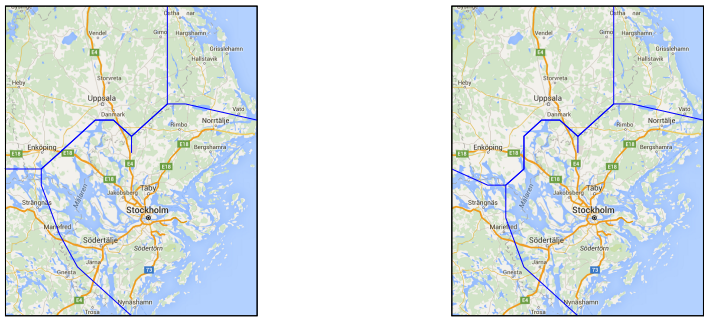
It can be seen from Figure 6(a) that for the Pareto optimal solution from Figure 4(e) our route from the west flies directly over Enköping. We can easily mitigate noise impact and other environmental consequences for populated areas, and remove all edges that intersect those areas from the edge set E. Figure 6(b) shows the resulting output STAR that circumnavigates Enköping.

Note, that our approach based on solving an IP on a graph easily allows to incorporate arbitrary obstacles, e.g. no-fly zones, as edges crossing these can simply be forbidden for a solution.

3.2 Increased Number of Entry Points

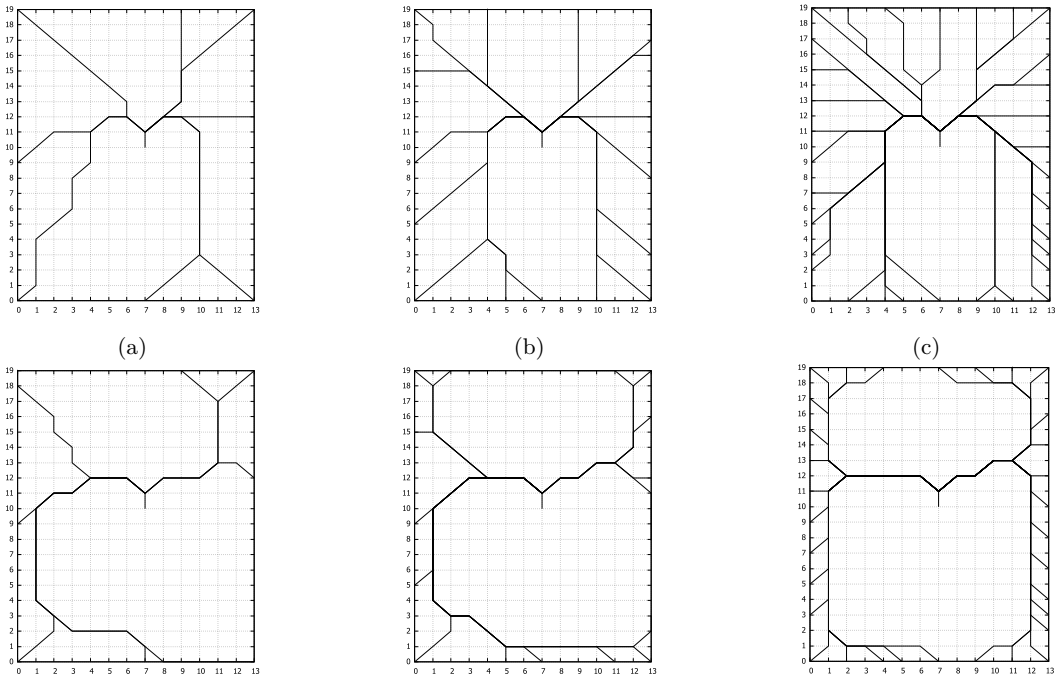
One might argue that our trees on the Pareto frontier (refer to Figure 4) do not differ too much between themselves. To emphasize the difference between our two objective functions, we ran experiments with additional (artificial) entry points. From the practical perspective, adding more entries may be appreciated by airlines, as this would give them the flexibility to plan straighter routes, and thus save time and fuel; however, the traffic situation might become more complex and difficult to control. To show how the model can be used to provide different suggestions for the TMA design, we created scenarios with 8, 16, and 32 entry points. Figure 7 (a)-(c) shows STARs of minimum paths length, Figure 7 (d)-(f) shows STARs of minimum tree weight.

For optimal total length, the trees merge paths from different entry points as soon as allowed. This implies that most merge points are located in close proximity to the TMA boundary—an effect ATCOs would not appreciate, since they would rather have some time to get a good "hold" on the planes before merging them (in fact, conflicts close to the boundary with adjacent sectors is one of the top 5 operational safety priorities identified



(a) (b)

Figure 6 (a) Routes without obstacle avoidance, same routes as in Figure 4(e), (b) routes avoiding Enköping.



(a) (b) (c)

(d) (e) (f)

Figure 7 Solutions for increased number of entry points: eight entries (a),(d), 16 entries (b),(e), and 32 entries (c),(f). In (a)-(c) we minimize the paths length (objective function (5)), in (d)-(f) we minimize the tree weight (objective function (6)). Paths length: (a) 458.60, (b) 893.61, (c) 1751.68, (d) 567.69, (e)1181.48, (f) 2108.86. Tree weight: (a) 311.33, (b) 451.47, (c) 671.06, (d) 268.64, (e) 465.74.

by EUROCONTORL [3]). On the other hand, the solutions with an optimal paths length, though they clearly serve the airlines' request for short trajectories best under the given constraints, are trees that produce a quite dense network of routes within most of the TMA, which might make it hard to control the traffic. Thus, it might be helpful to use linear combination of these two functions. As we mentioned before, the optimizer of any linear combination is found among the Pareto optimal solutions.

In addition, solutions for a large number of entry points, for example, the 32 entry point solutions, could be used to suggest the number and location of entry points for a design from scratch. If we assume that the grid covers an area larger than the TMA, the minimum tree weight solution, Figure 7(f), suggests two entry points, based on the minimum paths length solution, Figure 7(c), we might advocate for 16 entry points.

3.3 SID constraints

In this section, we experiment with keeping the STARs vertically separated from SIDs at the points where the arrival and departure routes cross; i.e., we experiment with constraint 5 outlined in Section 1—STARs are not allowed to intersect a given SID closer than d from the runway (the necessary modifications to the IP are described in Section 2.3). Figure 8 shows minimum route length STARs for increasing values of d. Each tree was obtained within approximately 2 CPU hours ($\sim 10^5$ B&B nodes); the numbers did not differ much for different values of d. The whole SID is light gray, and the parts that are not allowed to be intersected by the STAR are red. Figure 9(a) shows that, as expected, the objective function (total length of the routes in the tree) grows with increasing d. To explain the apparently non-linear growth, we model the STAR–SID interaction as follows (Fig. 9(b)): Assume the STAR needs to connect vertices of a regular polygon (the entry points) to its center (the runway), but there is some shape (for simplicity—a set of radius-d circular sectors disjoint other than at the common apex at the center) that needs to be avoided. The length of each route in the STAR depends on d as $d + \sqrt{(1-d)^2 + 1/2}$ (assuming the entry points lie on unit circle), which is similar to what we observed in the experiments (Fig. 9(c)).

4 Conclusion and Discussion

In this paper we present a proof of concept for our grid-based IP approach for finding aircraft arrival routes with limited turning angle. The approach incorporates constraints on the merge points in the STAR that facilitate handling for aircraft controllers: separation of merge points and a limit on the number of routes that merge. In addition, to allow for handling of both arriving and departing aircraft, we show that we can easily integrate constraints from the departure routes. We present arrival routes for the Stockholm TMA.

Static obstacles, e.g., no-fly zones, can be added to our method: all edges that intersect the obstacle are deleted from the edge set E (actually, reducing the problem instance size). This gives also the potential to integrate flexible obstacles like weather, when we recompute based on weather forecast. Our STARs might still cross noise-sensitive areas; in Section 3.1 we chose to completely forbid these areas, in future work we may integrate noise impact by increasing the cost for edges that intersect noise-sensitive areas.

Moreover, the number of aircraft that enter the TMA via different entry points varies. Thus, we might choose to not only minimize the length of paths from entry points to the runway, but consider a weighted version that minimizes the sum of trajectory lengths flown by all arriving aircraft. That is, each path is counted as many times as it is used by aircraft. Hence, we minimize the demand-weighted distance. We can easily integrate this by changing

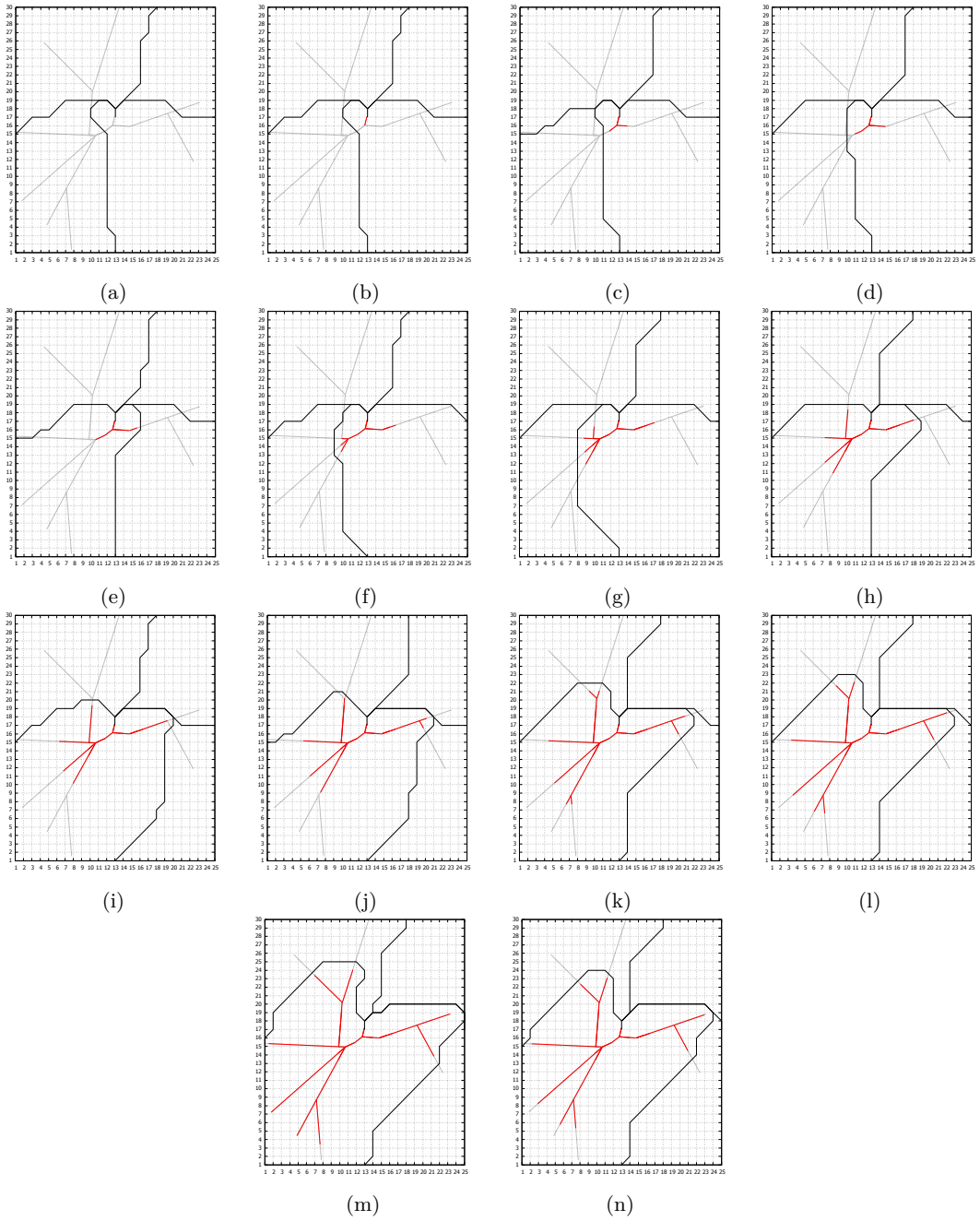
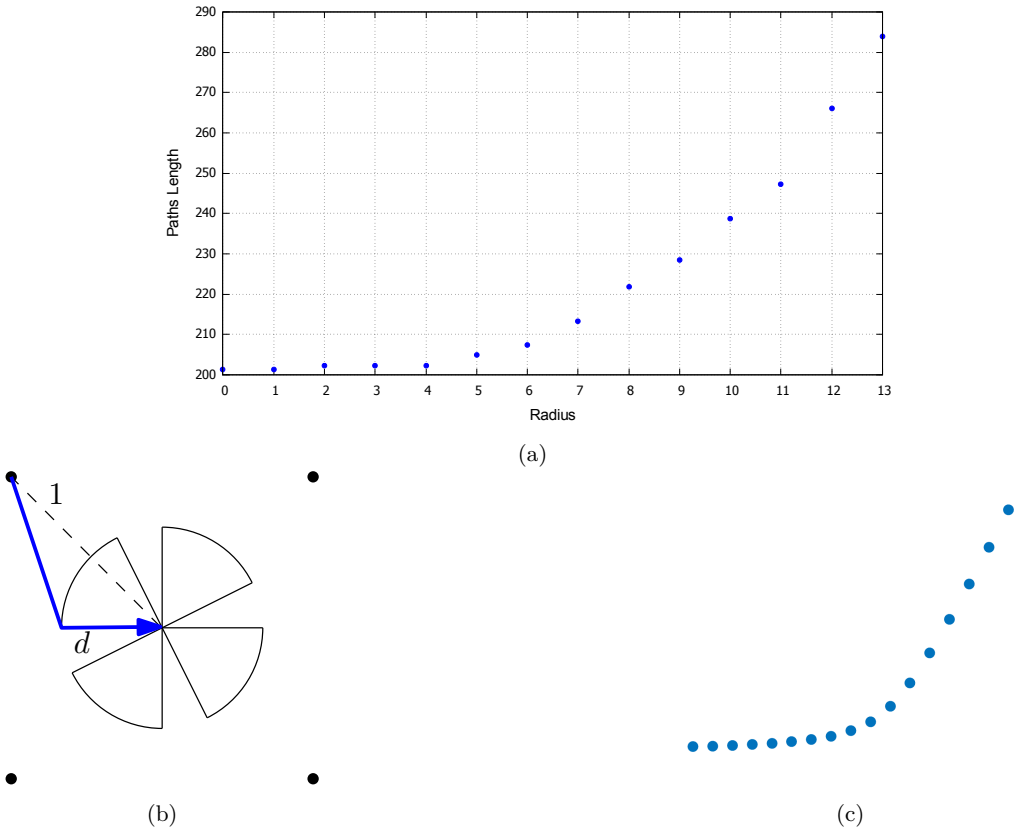


Figure 8 Minimum paths length STARs for increasing values of d. The SID is shown in light gray, edges in E_{SID} are shown in red. Paths lengths: (a) 201.34, (b) 201.34, (c) 202.37, (d) 202.37, (e) 202.37, (f) 204.85, (g) 207.34, (h) 213.34, (i) 221.82, (j) 228.55, (k) 238.79, (l) 247.28, (m) 266.01, (n) 284.01. Tree weight: (a) 176.88, (b) 176.88, (c) 177.64, (d) 181.88, (e) 181.88, (f) 184.37, (g) 186.85, (h) 186.85, (i) 185.09, (j) 188.82, (k) 200.31, (l) 205.79, (m) 216.04, (n) 228.04.



(a) (b) (c)

Figure 9 (a) Paths length (y-axis) over the radius d (x-axis) for the STARs from Figure 8. (b) Black dots are the entries and the blue is a STAR route avoiding the forbidden sectors. (c) $d + \sqrt{(1-d)^2 + 1/2}$ as a function of d.

the right-hand side of Equation (1): If w_k aircraft enter the TMA via entry $k \in \mathcal{P}$, we substitute the -1 for $i \in P$ by $-w_i$, and the $|\mathcal{P}|$ for $i = r$ by $\sum_{k\in\mathcal{P}} w_k$ (and increase N accordingly).

Our approach can be embedded into a tool for simultaneous optimization of routes and the TMA control sectors (within which the ATCOs control the traffic)—a topic for future research. Further possible extensions include simultaneous design of both SIDs and STARs, and 3D routes.

Acknowledgments. We thank Billy Josefsson (project co-PI, LFV) and the members of the project reference group—Eric Hoffman (Eurocontrol), Hakan Svensson (LFV/NUAC), Anne-Marie Ragnarsson (Transportstyrelsen, the Swedish Traffic Administration), Anette Näs (Swedavia, the Swedish Airports Operator)—for discussions of the STAR and SID structures and design flexibility.

References

1 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 S. Choi, J. E. Robinson, D. G. Mulfinger, and B. J. Capozzi. Design of an optimal route structure using heuristics-based stochastic schedulers. In *Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th*, pages 2.A.5–1–2.A.5–17, Oct 2010.

3 EUROCONTROL. SAFMAP analysis. `https://docs.google.com/presentation/d/1Clq33HyI_PNoeRuzoCmPtfLqtoK45DFrZqOvWXKXXEk/edit#slide=id.g13444e53d7_0_0`.

4 EUROCONTROL. European airspace concept handbook for PBN implementation edition 3.0. http://www.eurocontrol.int/publications/airspace-concept-handbook-implementation-performance-based-navigation-pbn, 2013.

5 Michel X. Goemans and Young-Soo Myung. A catalog of Steiner tree formulations. *Networks*, 23(1):19–28, 1993.

6 Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2015. URL: `http://www.gurobi.com`.

7 J Krozel, JSB Mitchell, V Polishchuk, and J Prete. Airspace capacity estimation with convective weather constraints. *AIAA Guidance, Navigation, and Control Conference*, 2007.

8 Jimmy Krozel, Changkil Lee, and Joseph S.B. Mitchell. Turn-constrained route planning for avoiding hazardous weather. *Air Traffic Control Quarterly*, 14(2), 2006.

9 Matthew Micallef, David Zammit-Mangion, Kenneth Chircop, and Alan Muscat. A proposal for revised approaches and procedures to Malta international airport. In *28th Congress of the International Council of the Aeronautical Sciences, 23 - 28 September 2012, Brisbane, Australia*, 2012.

10 Diana Michalek Pfeil. *Optimization of airport terminal-area air traffic operations under uncertain weather conditions*. PhD thesis, Massachusetts Institute of Technology. Operations Research Center, 2011.

11 Valentin Polishchuk. Generating arrival routes with radius-to-fix functionalities. In *ICRAT 2016*, 2016.

12 Joseph Prete, Jimmy Krozel, Joseph Mitchell, Joondong Kim, and Jason Zou. Flexible, Performance-Based Route Planning for Super-Dense Operations. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, Guidance, Navigation, and Control and Co-located Conferences. American Institute of Aeronautics and Astronautics, aug 2008.

13 H G Visser and R A A. Wijnen. Optimization of Noise Abatement Departure Trajectories. *Journal of Aircraft*, 38(4):620–627, jul 2001.

14 Richard T. Wong. A dual ascent approach for Steiner tree problems on a directed graph. *Mathematical Programming*, 28(3):271–287, 1984.

15 Jun Zhou, Sonia Cafieri, Daniel Delahaye, and Mohammed Sbihi. Optimization of Arrival and Departure Routes in Terminal Maneuvering Area. In *ICRAT 2014, 6th International Conference on Research in Air Transportation*, Istanbul, Turkey, May 2014.

16 Jun Zhou, Sonia Cafieri, Daniel Delahaye, and Mohammed Sbihi. Optimizing the design of a route in Terminal Maneuvering Area using Branch and Bound. In *EIWAC 2015, ENRI International Workshop on ATM/CNS*, Tokyo, Japan, November 2015. ENRI.