Improving the Tractability of a MILP Model for Regional Aviation Network Routing

Harald Bjurulf

Spring 2026
Master’s thesis, 30 hp
Department of Mathematics and Mathematical Statistics

Abstract

This thesis studies a two-layer mixed-integer linear programming model for regional electric aviation under Public Service Obligation (PSO) requirements. The aircraft layer represents aircraft movement in a time-space network. The service layer represents the flow of passenger groups in a time-space network. Flows in the aircraft layer determine capacity in the service layer. This thesis develops a fast, exact aggregation of PSO-requirements by reducing a subproblem to the bipartite minimum vertex cover. It also studies alternative capacity constraints and summed passenger-flow constraints. Numerical experiments use synthetic data and show that the summed passenger-flow constraints provide significant performance improvements.

Förbättrad beräkningsbarhet av en heltalsprogrammeringsmodell för regional flygplanering

Sammanfattning

Examensarbetet studerar en blandad heltalsprogrammeringsmodell (MILP) med två lager för regionalt elflyg under allmän trafikplikt (PSO). Flygplanslagret representerar flygplans rörelser i ett tids-rumsnätverk. Tjänstelagret representerar flödet av passagerargrupper i ett tids-rumsnätverk. Kapaciteten i tjänstelagret bestäms av flödena i flygplanslagret. Detta examensarbete utvecklar en snabb, exakt aggregering av PSO-krav genom en reduktion till bipartit minsta hörntäckning (bipartite minimum vertex cover). Vidare studeras alternativa kapacitetsrestriktioner och summerade passagerarflödesrestriktioner. Numeriska experiment utförs med syntetisk data och visar att de summerade passagerarflödesrestriktionerna ger betydande prestandaförbättringar.

Contents

1 Introduction

Air transport planning is a challenging application area for operations research. Airlines and public authorities decide which routes to serve, when flights should depart, and which aircraft types should service a route, which specific aircraft should traverse a flight, and which personnel should crew a flight. These decisions are discrete and therefore frequently modeled as mixed-integer linear programs, or MILPS. Time-space networks are a common choice of structure. In this paradigm each node represents an airport at a specific time, an arc represents a flight, waiting time, or another activity(Barnhart, Belobaba, et al. 2003; Hane et al. 1995; Lohatepanont and Barnhart 2004; Xu et al. 2024).

Classical airline planning models are developed for commercial airlines. They have a clear focus on maximizing profit and minimizing operating costs. However, in many remote regions, air travel is imperative for access to healthcare, education, labor markets, and public services. Despite this, small populations often mean that demand is too low to support commercial flights. Within the European Union, service in these remote regions can be subsidized through the use of Public Service Obligations, or PSOs. The legal framework allows for their use on routes that are important for a peripheral or developing region, or on thin routes where the market does not provide adequate service(European Parliament and Council of the European Union 2008). The obligations can take the form of minimum frequency, capacity, travel time, and fare requirements (Williams and Pagliari 2004; Bråthen and Eriksen 2018).

The planning problem under a PSO setting differs from a pure commercial schedule design. There is a dual goal of not only reducing costs but also maximizing public good and satisfying social accessibility requirements. There exists a growing literature regarding social air transport planning. João Pedro Pita et al. (2013) and João P. Pita et al. (2014) developed integrated flight scheduling and fleet assignment models for subsidized air networks. Leandro et al. (2021) used a related approach for the Greek PSO network. Kinene, Granberg, et al. (2020) studied route selection for subsidized air services in Sweden. Kinene and Birolini (2024) later developed an optimization framework for subsidized air transport networks using electric aircraft.

The possible introduction of electric aircraft adds further questions to the air transport planning problem. Their range is limited by battery capacity. Their turnaround time may include charging. Airport charging capacity may become a bounded resource with limited availability. Battery swapping has been proposed as a method to reduce turnaround time, but this requires multiple batteries and charging logistics (Justin, Payan, Briceno, et al. 2020; Mitici et al. 2022; Oosterom and Mitici 2023). As a result, electric regional aviation planning must connect network design, fleet assignment, charging time, and passenger service.

This thesis builds on the model by Westin et al. (2024). The model studies regional electric aviation in northern Scandinavia under PSO-like service requirements. They use two linked time-space networks. The first layer represents aircraft movement. The second layer represents passenger movement. The movement of aircraft in the first layer induces capacity for passengers in the second layer. The model disassociates the turnaround time of the aircraft from the turnaround time of the passengers.

The model lacks fixed itineraries and instead allows free passenger movement. Passengers are modeled as commodities in a multi-commodity flow problem. This formulation is more flexible at the cost of increased computational cost. The model increases in size as the number of airports, aircraft types, and passenger groups increases.

The computational cost is not only dependent on model size. Different MILP formulations can describe the same integer-feasible solutions, yet vary widely in practical performance. A formulation with a tighter linear relaxation can reduce the branch-and-bound search. Extra variables and constraints can also make each relaxation more expensive to solve. The effect of a formulation change must therefore be tested on instances. Therefore, this thesis has a computational focus.

1.1 Aim

This thesis aims to study how the two-layer MILP model for regional electric aviation can be reformulated to improve tractability. The focus is on formulation structure, exact aggregation of PSO requirements, and numerical performance. The thesis does not develop a new aircraft performance model. It also does not estimate demand. Travel times, aircraft capacities, charging times, and PSO requirements are treated as input data.

Three research questions are considered.

1. How can PSO requirements be represented with fewer passenger groups without changing the service requirements of the model?
2. How do alternative capacity formulations, linking inequalities, and redundant aggregate constraints affect the performance of the MILP model?
3. How does solver performance change when the number of airports and PSO requirements increases?

1.2 Contributions

The thesis makes three methodological and computational contributions. They focus on model formulation and computation.

First, it provides a Python implementation of the model with Gurobi as the MILP solver (Gurobi Optimization, LLC 2026). The implementation is modular, allowing for controlled changes in network size, aircraft types, PSO requirements, and formulation choices. Therefore, model variants can easily be compared on the same generated instances.

Second, it studies the exact aggregation of PSO requirements. Several PSO requirements can be represented by the same passenger group when they share a relevant origin or destination condition. The aggregation problem is reduced to a minimum vertex cover in bipartite graphs. Thus, it reduces the number of commodities and the size of the MILP formulation.

Third, it compares the base model with selected alternative formulations. The reported numerical runs test a disaggregated capacity formulation, an auxiliary capacity formulation, and redundant aggregate flow constraints inspired by cut-set reasoning in network flow models (Chouman et al. 2017). The numerical study reports solution time, final optimality gap, and runtime variation on a fixed instance. The aim is to identify which tested changes are useful in practice.

This thesis is positioned between three areas of literature. The first area is socially oriented airline planning under PSO requirements. The second area is electric regional aviation and charging-aware scheduling. The third area is MILP formulations for network flow and vehicle routing problems.

1.3 Scope

The scope is limited to deterministic planning instances. The model is studied as a planning model for scheduled regional services. It is not a real-time disruption management model. Demand uncertainty, ticket prices, crew scheduling, maintenance scheduling, airport slot allocation, charger siting, and charger power sizing are outside the scope.

The experiments are based on generated instances rather than a full policy case study. Allowing for a controlled comparison between formulations. The same network generation rules and PSO generation rules are used for each formulation. The results support conclusions about formulation behavior under the tested assumptions.

1.4 Outline

Section 2 gives the background. It reviews airline planning models, PSO air transport, electric regional aviation, MILP formulations, and multi-commodity flow, and provides an informal description of the model. Section 3 states the model and the formulation variants. Section 4 describes the computational experiments. Section 5 gives the results. Sections 6 and 7 discuss the results and summarize the work.

2 Background

This section gives the background for the model and the numerical study. It starts with operations research in airline planning. It then discusses PSO air transport and electric regional aviation. It then introduces the mathematical tools used in the thesis. These are mixed-integer linear programming and multi-commodity flow. The final parts describe the base model and the gap addressed by this thesis.

2.1 Airline planning and operations research

Airline planning contains strategic, tactical, and operational decisions. Each level is modeled as a separate problem that is often solved in sequence. Long-term problems include network design and fleet planning. Short-term problems include aircraft routing, crew assignment, and disruption management. The division is necessary as full planning models are intractable (Barnhart, Belobaba, et al. 2003; Sherali, Bish, et al. 2006; Xu et al. 2024).

Fleet assignment is the problem of assigning an aircraft type to a scheduled flight leg. Fleet assignment models must cover each flight and maintain aircraft balance at each station. Hane et al. (1995) used a time-space network formulation for a large daily fleet assignment problem. Sherali, Bish, et al. (2006) gives a survey of fleet assignment concepts, models, and algorithms. Later models include more operational detail, such as time windows, maintenance, passenger spill, and itinerary choice (Rexing et al. 2000; Sherali, Bae, et al. 2010; Wei et al. 2020; Yan et al. 2022).

An integrated schedule design and fleet assignment model simultaneously designs the schedule and assigns aircraft. This integration finds better solutions at the cost of increased computational time. Lohatepanont and Barnhart (2004) presents an integrated model for schedule design and fleet assignment. Desaulniers et al. (1997) uses aircraft routing and scheduling formulations based on set partitioning and time-constrained multi-commodity flow. More recent work includes passenger choice and demand effects (Birolini et al. 2021; Wei et al. 2020; Yan et al. 2022).

The model used in this thesis belongs to this broad family of time-space network models. It differs from many commercial airline models because it is written for PSO service requirements and electric aircraft. The objective is not airline profit. The model must provide required transport opportunities while respecting aircraft capacity, timing, and charging-related turnaround constraints.

2.2 Public Service Obligations and regional aviation

Airline deregulation enabled airlines to choose routes based on commercial value. This increased efficiency in many markets, but it also created a risk that thin and remote routes would lose service. PSO systems were introduced to support socially important routes that are not commercially attractive. In the European system, a route can be subject to a PSO when it is considered vital for the economic and social development of the region served (European Parliament and Council of the European Union 2008; Williams and Pagliari 2004).

PSO requirements may specify minimum frequency, capacity, maximum travel time, maximum fares, and aircraft properties. Bråthen and Eriksen (2018) discusses how service level requirements affect social efficiency in Norway. Fageda et al. (2018) provides a wider review of public policies used to maintain air connectivity in remote regions. These studies show that PSO design is both a transport planning problem and a public policy problem.

Optimization models for PSO networks usually have a social objective. They may minimize total social cost or balance passenger benefits and subsidy costs. João Pedro Pita et al. (2013) studied the setting of PSO standards in the Azores. João P. Pita et al. (2014) developed a socially oriented flight scheduling and fleet assignment model for Norway. Leandro et al. (2021) extended this line of work in a Greek case study. These papers move from describing PSO policy to optimizing service design.

A common modeling choice in several PSO models is to generate passenger itineraries before the optimization model is solved. The model then chooses among a given set of routes. This limitation works when the route set is known, but it can miss new transfer patterns. The model by Westin et al. (2024) instead lets passenger routes arise from the service layer. The increased flexibility comes at the cost of computational complexity.

2.3 Electric regional aviation

Electric and hybrid-electric aircraft are often discussed as possible tools for reducing aviation emissions on short routes. All-electric aircraft remove direct combustion emissions during flight. Their full climate effect still depends on the source of electricity, battery production, and aircraft use. Technical studies show that battery energy density is the main limiting factor for larger aircraft and longer routes (Schäfer et al. 2019; Adu-Gyamfi and Good 2022).

The first likely applications are short regional routes with low or moderate passenger demand. Such routes better match the expected range and capacity of early electric aircraft than long-haul markets do. Regional aviation and PSO networks are relevant early applications. The Nordic policy report by the Nordic Network for Electric Aviation also argues that the Nordic countries have conditions that support the early development of electric regional aviation, including climate targets, regional route structures, and existing sectoral cooperation (Nordic Innovation 2024).

The feasibility and cost of first-generation electric aircraft remain uncertain. Bærheim et al. (2023) studies the potential and limits of battery-powered regional flights in Norway. Avogadro and Redondi (2024) compares the costs of first-generation electric aircraft with conventional aircraft. These studies support the scope of this thesis. Aircraft performance and cost are treated as input data, not as fixed conclusions.

Electric aircraft planning differs from conventional aircraft planning in several ways. Range limits can make some direct legs infeasible. Charging time can make short turnarounds impossible. Airport charging capacity can become a binding resource. Battery swapping can reduce delays, but it requires spare batteries and charging logistics. These issues link the aircraft schedule to infrastructure planning (Justin, Payan, Briceno, et al. 2020; Mitici et al. 2022; Oosterom and Mitici 2023).

Recent studies examine these operational problems. Justin, Payan, Briceno, et al. (2020)

optimize battery swap and recharge strategies for electric aircraft operations. Mitici et al. (2022) formulate electric flight scheduling with battery charging and swapping. Oosterom and Mitici (2023) optimize battery charging and swapping infrastructure for electric short-haul aircraft in Norway. Vehlhaber and Salazar (2023) combines aircraft assignment, routing, and charging while accounting for renewable energy availability. Tang et al. (2026) study mixed operations with electric and fuel aircraft and battery charging and swapping.

Other studies consider network and infrastructure design. Kinene, Birolini, et al. (2023) studies the design of electric aircraft charging networks for regional routes. Kinene and Birolini (2024) study subsidized air transport networks with electric aircraft and compare conventional, electric, and mixed fleets. Justin, Payan, and Mavris (2022) optimize fleet assignment and scheduling for electrified regional air mobility. Together, these studies show that electric aviation planning combines network design, fleet assignment, charging, and passenger service.

The present thesis focuses on the computational formulation. Charging infrastructure and detailed energy use are treated as input assumptions or future extensions. The thesis studies the computational behavior of a time-space MILP model that can represent regional electric aircraft with charging-related turnaround times. This step is needed because detailed electric aviation models can become too large before they are useful for planning.

2.4 Mixed-integer linear programming

A mixed-integer linear program is an optimization problem with a linear objective function and linear constraints. However, some of the variables are limited to only assume integer values. A general minimization form is

$$\begin{aligned} \min \quad & c^T x + d^T y && (1)\\ \text{s.t.} \quad & Ax + By \geq b, && (2)\\ & x \in \mathbb{Z}^p, && (3)\\ & y \in \mathbb{R}^q. && (4) \end{aligned}$$

Where x is the vector of integer variables and y is the vector of continuous variables. Common variations of MILP are the integer programs, where every variable is in $\mathbb{Z}$, and binary integer programs, where every variable is in $\{0, 1\}$. MILPs are hard to solve because the feasible set is not convex (Wolsey 2020; Clautiaux and Ljubić 2025). It is unlikely that polynomial solution algorithms exist since MILPs are NP-hard. Despite this, MILPs are of great practical use, as many problems can be easily reduced to MILP.

The most basic method of solving MILPs, barring enumeration, is branch-and-bound. The linear programming (LP) relation is obtained by removing any integrality constraints. The LP-solution is then a lower bound of the optimal function value. If the LP-solution respects the integrality constraints, then the search can terminate. If the LP-solution does not respect the integrality constraints, then select an integer variable x_j whose value in the LP-relaxation x_j^{LP} is not an integer. That variable is branched by creating and recursively solving the two subproblems with constraints $x_j \leq \lfloor x_j^{\mathrm{LP}} \rfloor$ or $x_j \geq \lceil x_j^{\mathrm{LP}} \rceil$ added. Any integer solution found is a global upper bound of the MILP solution. This leads to a recursive algorithm where any branch can terminate early if the LP relaxation is infeasible or if the local lower bound becomes higher than the global upper bound.

Branch-and-cut is an extension of the branch-and-bound algorithm. It dynamically generates additional valid inequalities, called cutting planes, which remove the current LP-solution. These cuts reduce the search space and increase the lower bound. Another common extension is row-and-column generation. In row-generation, some of the constraints are added dynamically as needed instead of generating every constraint simultaneously. Similarly, in column-generation, additional variables are generated dynamically. A common column-generation algorithm is the branch-and-price algorithm, which is used for very large MILPs (Barnhart, Johnson, et al. 1998).

Generating a strong initial, valid solution allows the branch-and-bound algorithm to start with a stronger upper bound and will therefore reduce the number of nodes searched. Similarly, a stronger formulation yields a higher LP solution, which also reduces the number of nodes searched. However, stronger formulations may use more variables or more constraints. This can increase the time needed to solve each relaxation. Therefore, formulation choice is a practical issue with significant performance implications. Recent reviews of practical integer programming stress that solver performance depends strongly on the chosen formulation (Clautiaux and Ljubić 2025).

In practice, computational experiments are necessary for comparing formulations. Solver time can vary across instances, and failures at a time limit are common. Performance profiles are one standard way to compare optimization software or model variants across a test set (Dolan and Moré 2002). This thesis uses simpler performance measures, but the same idea applies. A model variant is only useful if it improves performance on relevant instances.

2.5 Multicommodity flow

The multi-commodity flow problem generalizes the single-commodity network flow problem. Several commodities must be routed through a shared network with shared capacities. Each commodity has its own source and sink. The multi-commodity flow problem is a natural model for passenger flow, freight flow, and communications networks(Magnanti and Wong 1984; Crainic 2000; Wang 2018a). Surveys by Wang (2018a) and Wang (2018b) give a more detailed account of applications, formulations, and solution methods.

Let $G = (V, A)$ be a directed graph. Let K be a set of commodities. For each commodity $k \in K$, let b_v^k be the net supply at node v. A positive value means that v is a source. A negative value means that v is a sink. Let C_a be the capacity of an arc $a \in A$. An arc-based formulation uses explicit variables x_a^k for the flow of good k using arc a. The basic formulation is

$$\sum_{a \in \delta^+(v)} x_a^k - \sum_{a \in \delta^-(v)} x_a^k = b_v^k, \qquad v \in V,\ k \in K, \tag{5}$$

$$\sum_{k \in K} x_a^k \leq u_a, \qquad a \in A, \tag{6}$$

$$x_a^k \geq 0, \qquad a \in A,\ k \in K, \tag{7}$$

where $\delta^+(v)$ is the set of arcs leaving node v and $\delta^-(v)$ is the set of arc entering v. If the commodities must be moved as integers, then $x_a^k \in \mathbb{Z}$ is added. If a commodity must use only one path, then $x_a^k \in \{0,1\}$ and any consumption of capacity or flow can be replaced with $b_k x_a^k$, where b_k is the total flow of good k. However, a single-path formulation requires a single source and a single sink per commodity.

Arc-based formulations are easy to understand and implement. However, it can scale poorly for larger networks as the number of constraints grows linearly in the number of commodities. An alternative is path based formulations, which eliminate the need for flow-conservation constraints at the price of an exponential number of variables. Path-based formulations require column generation and specialized solvers.

Cut-sets are a common method for strengthening multi-commodity flow formulations. Several valid inequalities can be found by considering a subset S of the vertices V. Any commodity with a source in S but the sink in $\overline{S}$ will add to a minimum flow from S to $\overline{S}$. For example, this induces a demand on the minimum capacity across the cut. Chouman et al. (2017) studies commodity representations and cut-set-based inequalities for a capacitated fixed-charge network. Ideas adapted from their work are used in this thesis.

2.6 The base model for regional electric aviation

Westin et al. (2024) proposed a network design optimization model for regional electric aviation in northern Scandinavia under PSO constraints. The model was motivated by the possibility of using electric aircraft to improve regional connectivity and the logistical challenges this would represent. The original case study concerned the Kvarken region and was implemented in Matlab and Gurobi. The model is described as a pickup-and-delivery vehicle routing problem with time windows, a heterogeneous fleet, split load, and transshipment. They note that proving optimality becomes challenging for large instances.

The model consists of two coupled multi-commodity flow problems. The first problem is referred to as the flight layer. This problem represents the movement of aircraft between airports. A flight arc for an aircraft type r can only be used if the distance, travel time, and turnaround time are feasible for that aircraft type. The second problem, referred to as the service layer, represents passenger movements. The service layer has sources and sinks decided by the PSO requirements. The capacity of an arc in the service layer is decided by the number of aircraft on the corresponding arcs in the flight layer. See a visualization in Figure 1. Note that if the turnaround time of aircraft and passengers is different, the corresponding arcs in the layers can arrive at different time points. A detailed mathematical formulation can be found in Section 3.1.

The model proposed by Westin et al. (2024) differs from itinerary-based models. In an itinerary-based model, the possible passenger paths are predetermined, whereas this model allows for any possible passenger flow to be considered. However, this extra freedom comes at the cost of computational complexity.

The model rapidly grows in size as the instance size increases. Let N be a set of airports and let T be a set of time points. The problem is defined on the node set $V = N \times T$. In other words, a node is an airport at a specific point in time. The number of nodes is $|V| = |N| \cdot |T|$. The number of arcs has a more complex relationship, but an upper bound is $|N|^2 \cdot |T|$. This rapid growth is consistent with the general behavior of integrated airline planning models, which often lead to large MILPs with many network-flow variables and linking constraints (Barnhart, Belobaba, et al. 2003; Lohatepanont and Barnhart 2004; Sherali, Bae, et al. 2010). This thesis primarily aims at improving the feasibility of the model by optimizing the selection of passenger groups and by introducing valid inequalities.

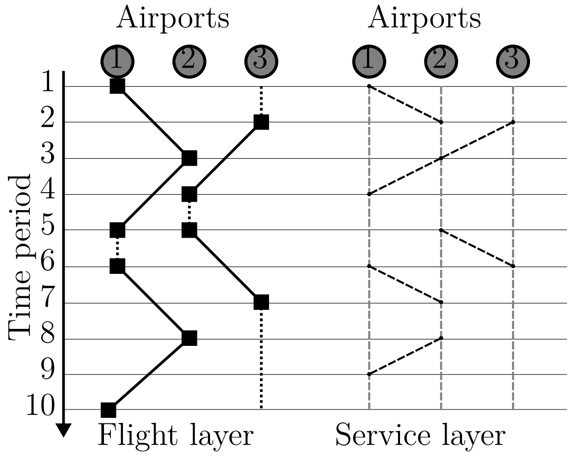


Figure 1: An illustration of dual layer model introduced by Westin et al. (2024). The image is sourced from Westin et al. 2024.

3 Model Formulation

This section presents the mathematical model used in the computational study. It first states the base model from Westin et al. (2024) in a compact form. It then describes the PSO aggregation method and the formulation variants studied in this thesis. The purpose is to keep the planning problem unchanged while providing the solver with a MILP with a clearer structure.

3.1 Basic model

The basic model consists of two linked network flow problems. The first problem, referred to as the flight layer, describes aircraft movements. The second problem, referred to as the service layer, describes the flow of passengers. The two layers use the same node-sets but have different arcs. The difference enables models with different turnaround times for passengers and aircraft.

The notation and sets used for the model are,

$i \in N$	the set of airports,
$t \in T$	the set of time points,
$p \in P$	the set of passenger groups,
$v \in V = N \times T$	the nodes of the graph,
$r \in R$	the set of aircraft types,
$s \in S$	the arcs in the service layer,
$f \in F_r$	the flight layer arcs associated with aircraft r,
$F_r \supseteq F_{sr}$	the flight arcs associated with service arc s,
G	the set of ground arcs.

A node $v = (i, t)$ represents airport i at time t. The service arcs S represent possible routes a passenger could take. The flight arcs F_r represent the possible routes an aircraft of type r can use. The arcs F_{sr} are the flight arcs of aircraft r contributing to the capacity of service arc s. Ground arcs are any arcs representing waiting at an airport. For any arc set, the superscripts IN and OUT denote arcs entering and leaving a vertex. For example, F_{rv}^{IN} is the set of aircraft arcs of type r entering vertex v, and S_v^{OUT} is the set of service arcs leaving vertex v.

The constants used to define the model are

b_{vp}	the number of passengers of group p added to node v,
P_r^M	the passenger capacity of aircraft r,
C_{fr}	the maximum number of r-type aircraft which can use arc f,
C_v	the landing and take-off capacity node v,
C_r	the maximum number of r-type aircraft,
c_{ir}	the cost of owning and operating r-type aircraft based airport i,
c_{fr}	the cost of using aircraft r on arc f,
t_0	the start of the planning period,
t_M	the end of the planing period,
I_{vi}^0	assumes the value 1 if $v = (i, t_0)$, 0 otherwise,
I_{vi}^M	assumes the value 1 if $v = (i, t_M)$, 0 otherwise.

The decision variables are

$y_{fr} \in \mathbb{Z}_{\geq 0}$	number of aircraft of type r using aircraft arc f,
$x_{sp} \in \mathbb{Z}_{\geq 0}$	passenger flow of group p using service arc s,
$z_{ir} \in \mathbb{Z}_{\geq 0}$	number of aircraft of type r based at airport i.

The passenger flow variables are integers, as the existence of continuous flow does not imply the existence of integer flows, even if arcs have integer capacity.

The base MILP is

$$\min \quad \sum_{i \in N} \sum_{r \in R} a_{ir} z_{ir} + \sum_{r \in R} \sum_{f \in F_r} c_{fr} y_{fr} \tag{8}$$

subject to

$$\sum_{f \in F_{rv}^{\mathrm{IN}}} y_{fr} + \sum_{i \in N} I_{vi}^0 z_{ir} - \sum_{f \in F_{rv}^{\mathrm{OUT}}} y_{fr} - \sum_{i \in N} I_{vi}^M z_{ir} = 0, \qquad v \in V,\ r \in R, \tag{9}$$

$$b_{vp} + \sum_{s \in S_v^{\mathrm{IN}}} x_{sp} - \sum_{s \in S_v^{\mathrm{OUT}}} x_{sp} = 0, \qquad v \in V,\ p \in P, \tag{10}$$

$$\sum_{p \in P} x_{sp} \leq \sum_{r \in R} \sum_{f \in F_{sr}} P_r^M y_{fr}, \quad s \in S \setminus G, \tag{11}$$

$$y_{fr} \leq C_{fr}, \qquad r \in R,\ f \in F_r, \tag{12}$$

$$\sum_{r \in R} \sum_{f \in F_{rv}^{\mathrm{OUT}} \setminus G} y_{fr} \leq C_v, \qquad v \in V, \tag{13}$$

$$\sum_{r \in R} \sum_{f \in F_{rv}^{\mathrm{IN}} \setminus G} y_{fr} \leq C_v, \qquad v \in V, \tag{14}$$

$$\sum_{i \in N} z_{ir} \leq C_r, \qquad r \in R. \tag{15}$$

The objective function contains fixed aircraft costs and aircraft arc operating costs. Constraint (9) conserves aircraft flow. Note that the use of z_{ir} is equivalent to adding an arc for each airport from the last time point t_M to the first time point t_0. Therefore, the flow of aircraft can be understood as a conservative flow in a directed cyclic graph. Constraint (10) conserves passenger flows. The constants b_{vp} defines sources and sinks for each passenger group. Constraint (11) links the two layers. It states that the total flow across a service arc must be less than the capacity of the aircraft that services it. Note that this condition does not apply to ground arcs $g \in G$ as they are assumed to have infinite capacity. Constraints (13) and (14) limit the number of take-offs and landings. The remaining constraints bound flows across specific arcs and the total number of aircraft in the model.

The model allows for transshipment. Passengers can start with one aircraft, wait at an airport, and then continue with another aircraft. It also allows split flow, meaning that a passenger group may use several different paths to reach the same destination.

3.2 Efficient modeling of Public Service Obligations

In this thesis, a PSO requirement is represented by the required amount of passenger flow between two airports and time points. A requirement q has a demand D_q, origin airport o_q, earliest departure time α_q, destination airport d_q, and latest arrival time β_q. A simple way to model this is to treat each PSO requirement as a passenger group. However, this creates a large number of commodities in the service layer and, therefore, a large model with a slower LP-relaxation.

Consider multiple PSO requirements with a shared origin airport o and time α. Then they can be combined into a single passenger group with one source and multiple sinks. Similarly, if multiple PSO requirements share the same destination conditions so that they can be combined into one group with multiple sources and one sink. These combinations are valid because the model allows for split flow and has no requirement-specific costs on passenger flows. In fact, there is no cost assigned to passenger flows at all.

Finding the smallest number of passenger groups under this paradigm is tractable. Let O be the set of all destination conditions. Let D be the set of all destination conditions. Then the PSO requirements is the set $E \subset O \times D$. Finding the minimum number of passenger groups is then just selecting the smallest set C from O and D such that every $e \in E$ has at least one end in C. However, this is just a vertex cover in the bipartite graph

$$H = (O, D, E).$$

A minimum vertex cover gives the smallest number of origin and destination classes needed within this aggregation scheme. Since H is bipartite, a minimum vertex cover can be found from a maximum matching. This follows from König's theorem. The matching can be computed in polynomial time, for example by the Hopcroft–Karp algorithm (Hopcroft and Karp 1973).

3.3 Alternative capacity formulations and additional constraints

The base model uses the aggregate capacity constraint (11). This subsection gives alternative formulations and valid inequalities that interact with this constraint. None of the variations are strengthening. Instead, they are intended to either extend the expressiveness of the model or to expose more of the problem structure in the hope that this will enable the solver to find stronger cuts.

3.3.1 Summed passenger flow constraints

The first set of constraints is derived by summing passenger conservation over a subset of nodes for a passenger group. The constraint was found due to a mistake during the implementation of a cut-set style constraint. Fortunately, early testing showed strong performance.

Let $W \subset V$ be a subset of nodes. For each passenger group $p \in P$, summing (10) for all $v \in W$ gives the valid constraint

$$\sum_{v \in W} b_{vp} + \sum_{s \in \delta_S^-(W)} x_{sp} - \sum_{s \in \delta_S^+(W)} x_{sp} = 0, \qquad p \in P, \tag{16}$$

where $\delta_S^-(W)$ is the set of service arcs entering W and $\delta_S^+(W)$ is the set of service arcs leaving W. The effect of internal flows is canceled when constraint (10) is summed. The constraints are redundant in the sense that they are solely sums of existing constraints. However, they can impact the computation through presolve, cuts, or branching. The experiments limits the subsets W to the sets $W_t = \{(i, t') \in V \mid t' \leq t\}$. In practice, this means the cuts are across the time axis. This also means that there are no entering flows, and the constraint can be rewritten as

$$\sum_{s \in \delta_S^+(W)} x_{sp} = \sum_{v \in W} b_{vp}, \qquad p \in P.$$

3.3.2 Explicit capacity variables

The second variation introduces an explicit capacity variable $c_s \in \mathbb{Z}$ for each service arc. The base capacity constraint is replaced by

$$c_s = \sum_{r \in R} \sum_{f \in F_{sr}} P_r^M y_{fr}, \qquad s \in S \setminus G, \tag{17}$$

$$\sum_{p \in P} x_{sp} \leq c_s, \qquad s \in S \setminus G. \tag{18}$$

The variable c_s is clearly redundant as it does not change the feasible set. It is hoped that the simpler structure of constraint (17) and (18) will enable stronger cuts.

3.3.3 Aircraft-type-disaggregated capacity

The second variation disaggregates passenger flow by aircraft type. This is done by introducing variables $x_{spr} \in \mathbb{Z}_{\geq 0}$. Leading to changes to constraints (10) and (11). The passenger conservation constraint is

$$b_{vp} + \sum_{r \in R} \sum_{s \in S_v^{\mathrm{IN}} \setminus G} x_{spr} - \sum_{r \in R} \sum_{s \in S_v^{\mathrm{OUT}} \setminus G} x_{spr} + \sum_{s \in S_v^{\mathrm{IN}} \cap G} x_{sp} - \sum_{s \in S_v^{\mathrm{OUT}} \cap G} x_{sp} = 0. \tag{19}$$

The new passenger capacity constraint is

$$\sum_{p \in P} x_{spr} \leq \sum_{f \in F_{sr}} P_r^M y_{fr}, s \in S \setminus G,\ r \in R. \tag{20}$$

This formulation gives an explicit link between aircraft type and the passenger capacity. The main goal of this formulation is to make the model more expressive by enabling the pricing of passenger flow differently by mode of transportation.

3.4 Implementation

The model is implemented in Python and solved with Gurobi (Gurobi Optimization, LLC 2026). The implementation builds an explicit graph structure, which simplifies the interpretation of results. Each variation is enabled or disabled by passing a flag to the constructor. In particular, the combination of constraints (18) and (20) results in the constraint

$$\sum_{p \in P} x_{spr} \leq c_{sr} = \sum_{f \in F_{sr}} P_r^M y_{fr}. \tag{21}$$

Charging is represented by constructing feasible aircraft arcs. If an aircraft type requires a longer turnaround time after a flight, the corresponding aircraft arc ends later. It also follows the structure of the original two-layer model, in which aircraft and passenger timing are allowed to differ (Westin et al. 2024). This representation captures charging time. It does not optimize charger capacity, grid limits, spare batteries, or the location of charging infrastructure.

4 Numerical Experiments

This section describes the numerical experiments. For each experiment, the base model B in Section 3.1 is compared with variants C, U, T, and C+U+T. C denotes the summed passenger flow constraints from Section 3.3.1. U denotes the explicit capacity variables from Section 3.3.2. T denotes the Aircraft-type-disaggregated flows from Section 3.3.3. C+U+T denotes the combination of all three variants. The impact of PSO aggregation from Section 3.2 was not tested due to limitations with the instance generation. For further implementation details, refer to Section 3.4.

The experiments compare the formulations on synthetic instances. The formulations were always run on the same instances. This design aims to separate general formulation effects from the assumptions needed in a policy case study. The runs use simplified operational assumptions. Airport takeoff and landing limits are inactive, aircraft ownership limits are inactive, and flight arc capacities are inactive. These bounds are left inactive so that generated PSO requirements can be combined to create larger, still-feasible instances. Without these simplifications, it becomes harder to generate instances that are guaranteed to be feasible.

The experiments use a 20-minute time limit. If a run proves optimality within the time limit, the reported value is the solver's runtime. If optimality is not proven, the reported value is the solver's relative MIP gap. Table 1 gives the labels used in the result figures.

Table 1: Formulation labels used in the numerical figures.

Label	Meaning
B or base	Base formulation
C or 1	Summed passenger flow constraints
U or 2	Explicit capacity variables
T or 3	Type-disaggregated capacity
C+U+T or 1+2+3	Combined formulation with C, U, and T

4.1 Instance generation

The generated instances use a fixed half-hour time grid

$$T = \left\{ 6 + \frac{n}{2} \,\middle|\, n \in \mathbb{Z},\ 0 \leq n \leq 34 \right\}. \tag{22}$$

The planning period runs from 06.00 to 23.00. Under the arc-generation rules used, the constant time step means that a feasible passenger trip can be displaced in time, so long as all departure and arrival times remain in the planning period.

The first step is to choose a connected set of airports. Let A be the set of candidate airports. Let $R^{\max}$ be the largest aircraft range in the test data, and let $d(i, j)$ be the distance between airports i and j. A connected set A_c with N airports is generated as follows.

1. Draw one airport a from A and set $A_c = \{a\}$.

2. Draw a new airport a_n from

$$\left\{a \in A \setminus A_c \mid \min_{a_c \in A_c} d(a_c, a) \leq R^{\max}\right\}.$$

3. Set $A_c := A_c \cup \{a_n\}$.
4. Repeat steps 2 and 3 until $|A_c| = N$.

If the candidate set in step 2 is empty, the draw is restarted. The procedure gives a connected airport set under the aircraft range graph. It does not sample all connected airport sets with equal probability.

The second step is to generate PSO requirements. A PSO requirement consists of an origin airport, a destination airport, an earliest departure time, a latest arrival time, and a passenger count. To generate a PSO requirement, a pair of airports (i, j) is drawn, subject to $i \neq j$ and to the possibility of traveling in the service layer from i to j in the planning period. Let τ_{ij} represent the shortest time needed to travel from i to j. A random time windows (t_s, t_e) is drawn subject to

$$t_e - t_s \geq \tau_{ij}.$$

This ensures that the generated PSO requirement is feasible. The process is repeated until sufficiently many requirements are generated.

4.2 Experiment 1. Performance across instance sizes

The first experiment studies how the formulations behave as the number of airports and PSO requirements change. The airport counts are

$$N_A = \{5, 10, 15, 19\},$$

and the number of PSO requirements are

$$N_R = \{5, 10, 15, 20, 30\}.$$

One instance is generated for each pair in $N_A \times N_R$. Each PSO requirement has one passenger. This setting separates the effects of network size and the number of passenger groups from those of passenger volume.

The first experiment is an exploratory size test. One instance is generated for each size, so the results are used to identify patterns and to decide which variants should be tested more carefully.

4.3 Experiment 2. Runtime distribution on a fixed instance

The second experiment uses one fixed instance with 6 airports and 7 PSO requirements. The number of passengers in each PSO requirement is drawn uniformly from $\{1, 2, \ldots, 20\}$. Each formulation is solved 100 times on the same instance. Each solve uses the same model data. The experiment, therefore, measures the repeated wall-clock runtime for a single mathematical instance. The order of the runs is randomized to reduce the effect of external machine load.

The recorded value is the runtime to prove optimality. A 95% confidence interval for the mean runtime is computed for each formulation from the sample standard deviation. These intervals describe run-to-run timing variation for this fixed instance on this machine.

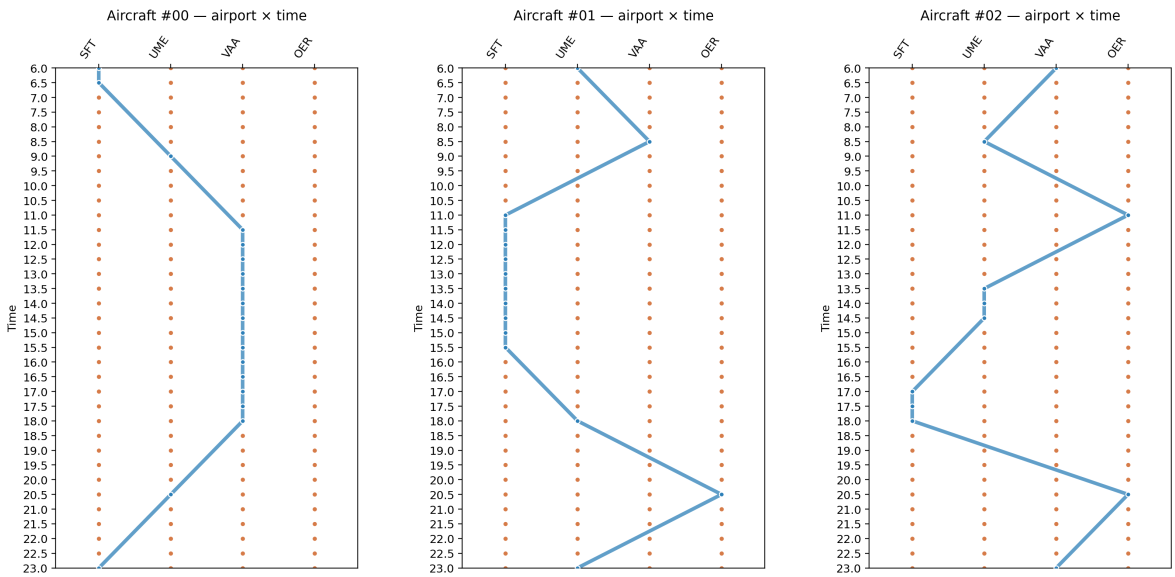
5 Results

This section reports the results from the two experiments. It also presents a visualization of an example solution. The result figures use the labels in Table 1.

5.1 Example solution

Figures 2 and 3 show one optimal solution for a small instance. The plots show how the model uses the time-space network and how aircraft capacity is created over time. They are separate from the runtime comparison.

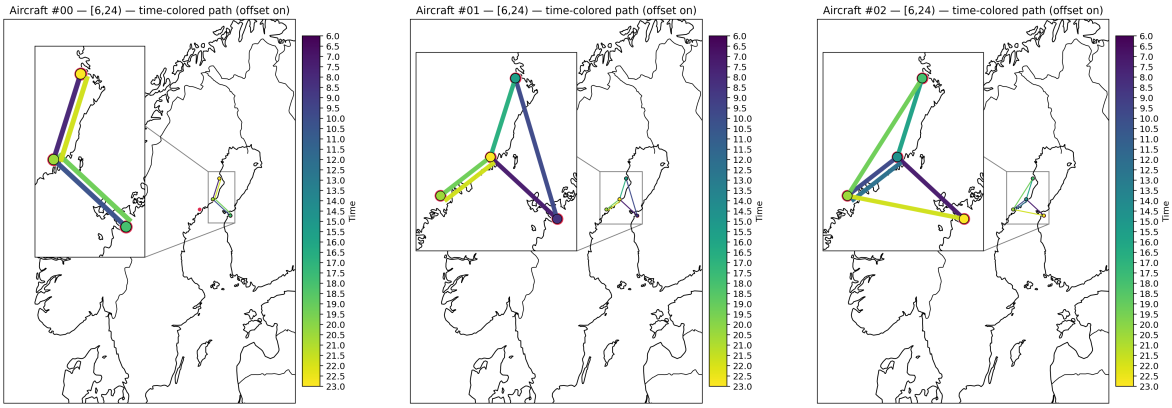
In Figure 2, each subfigure shows one aircraft. The horizontal axis is the airport. The vertical axis is time, with later times shown lower in the plot. A slanted line is a flight arc. A vertical line is a waiting or turnaround arc at the same airport. The solution uses three aircraft. Aircraft 0 starts at SFT, flies to UME and VAA, waits at VAA for much of the day, and returns through UME to SFT. Aircraft 1 starts at UME and visits VAA, SFT, OER, and UME. Aircraft 2 starts at VAA and visits UME, OER, UME, SFT, OER, and VAA. Aircraft 1 and aircraft 2 are both at OER at time 20.5. This creates a feasible meeting point in the aircraft layer. If the corresponding service arcs and passenger groups use this point, it can support a passenger transfer. Since passenger flow is not plotted, the figure is interpreted as aircraft timing and capacity rather than as an itinerary plot. The figure shows why the two-layer formulation can represent transfers without fixing passenger itineraries in advance.



(a) Aircraft 0 (b) Aircraft 1 (c) Aircraft 2

Figure 2: Aircraft movement in the airport-time graph for a small optimal solution.

Figure 3 shows the same aircraft movements on a map. The colors represent time. Repeated visits are slightly offset in the plot to keep them visible. The map shows that the solution uses short regional legs between the selected airports. Figure 2 is better for reading exact times, because the printed color scale in the map is small. The map should be interpreted as aircraft movement. Passenger paths are determined in the service layer and may use one or more aircraft arcs.



(a) Aircraft 0 (b) Aircraft 1 (c) Aircraft 2

Figure 3: Aircraft movement in physical space for the same small optimal solution. The color scale gives time.

5.2 Experiment 1. Performance across instance sizes

Table 2 gives the results of the first experiment. An entry ending in seconds indicates that the run proved optimality within the 20-minute time limit. An entry ending in percent means that the run reached the time limit, and the value is the final MIP gap. The best result in each row is shown in bold. A solved run is better than an unsolved run. If no model is solved, the lowest final gap is best.

Figure 4 gives the same results as bar plots. The vertical axis is the final MIP gap. When a run is solved to optimality, the bar represents a zero final gap, and the runtime is printed on the plot. The printed runtimes are annotations, not values on the MIP-gap axis.

The results show two main patterns. First, C+U+T is often the best variant for small- and medium-sized generated instances with one passenger per PSO requirement. It is best in 12 of the 20 instance sizes. Second, C alone is also strong. It is best in 7 of the 20 instance sizes. The base model is best only once, and that case is very small. U and T are not the best in any instance. Because there is only one random instance per size, the table is an exploratory comparison of sizes rather than a monotonic scaling curve. Topology, time windows, and feasible transfer structure also affect difficulty. This explains why some larger airport counts can be easier than smaller ones.

Table 2: Results for Experiment 1. Time values mean proven optimality within 20 minutes. Percent values mean final MIP gap after 20 minutes.

Airports	PSO	B	C	C+U+T	U	T
5	5	12.2s	**2.3s**	2.7s	5.7s	9.1s
5	10	**1.4s**	2.3s	4.1s	1.9s	3.6s
5	15	59.2s	86.7s	**21.0s**	113.3s	105.6s
5	20	704.5s	739.2s	**98.1s**	659.7s	158.1s
5	30	133.6s	143.5s	**56.2s**	108.5s	181.7s
10	5	66.2s	**5.9s**	6.8s	66.9s	76.7s
10	10	17%	4%	**210.1s**	17%	19%
10	15	26%	21%	**522.4s**	29%	22%
10	20	46%	31%	**11%**	43%	51%
10	30	28%	30%	**13%**	28%	25%
15	5	58%	30%	**786.4s**	59%	61%
15	10	19%	24%	**5%**	22%	26%
15	15	44%	40%	**21%**	47%	61%
15	20	38%	30%	**13%**	32%	45%
15	30	57%	**31%**	36%	62%	70%
19	5	13.0s	**5.5s**	8.0s	12.5s	16.6s
19	10	10%	**109.6s**	314.1s	11%	16%
19	15	64%	42%	**35%**	65%	73%
19	20	83%	**65%**	71%	83%	97%
19	30	74%	**51%**	57%	74%	79%

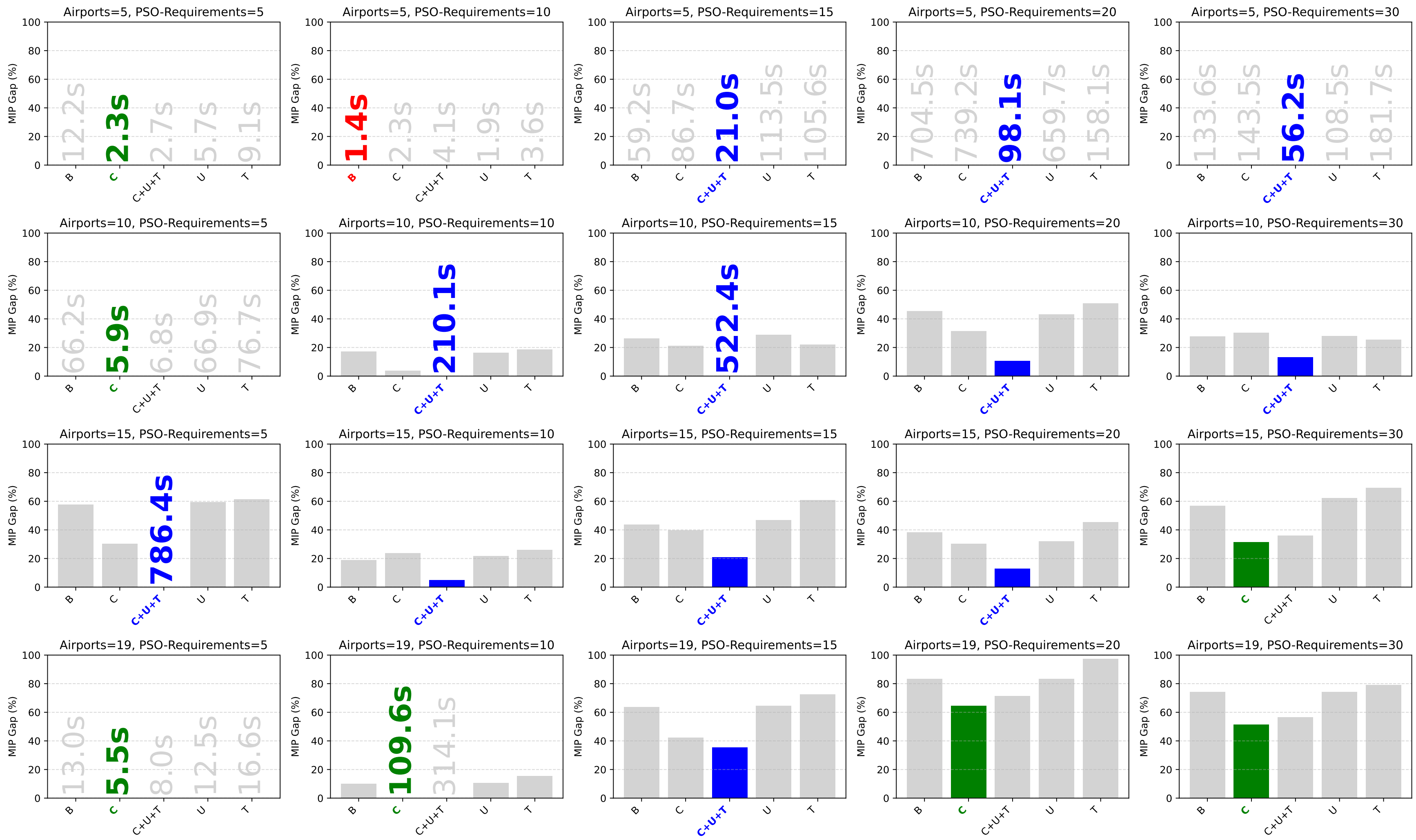


Figure 4: Results for Experiment 1. The vertical axis is final MIP gap. If a run solved to optimality, the runtime is printed instead. Model labels are defined in Table 1. The subplot titles includes the number of airports and PSO requirements in an instance.

These results identify C+U+T as the best-performing variant in many one-passenger instances. The interpretation is limited by the experiment design. In the unsolved cases, the best means the lowest final gap at the time limit. In larger unsolved cases, the final gaps remain high. The remaining gaps indicate that the main computational difficulty persists, even though the formulation variants alter the relative performance of the branch-and-cut search.

5.3 Experiment 2. Runtime distribution on a fixed instance

Figure 5 shows the runtime distribution for the fixed instance. The vertical axis is a density scale. The C formulation is separated from the other variants. Its runs are concentrated around 100 seconds. The base model and C+U+T are mostly around 210-230 seconds. The U and T variants are slower. The T variant also has the widest spread and visible outliers. The spread reflects repeated solves of the same mathematical instance.

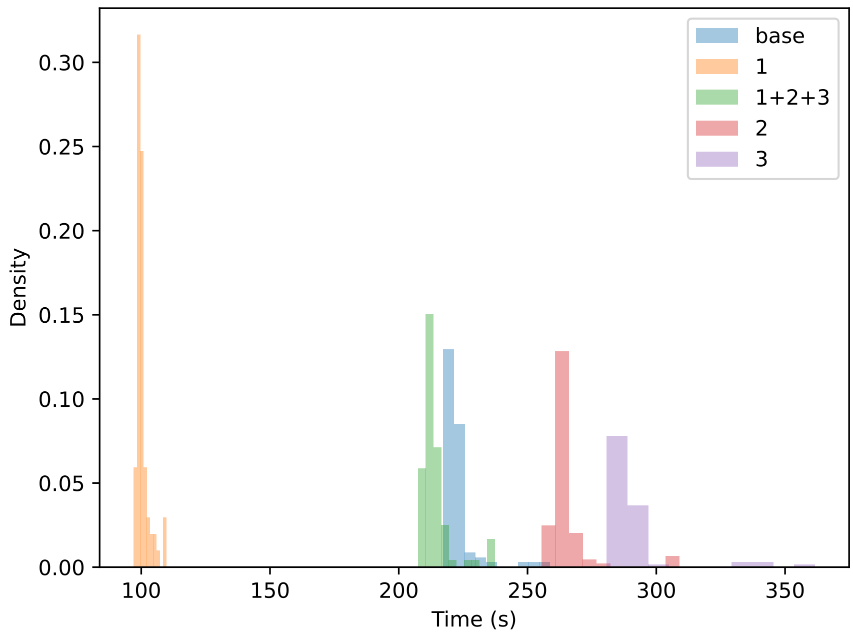


Figure 5: Runtime density distributions from 100 runs on the fixed instance. Numeric model labels are defined in Table 1.

Figure 6 gives the corresponding 95% confidence intervals for the mean runtime. The intervals support the same conclusion as the histogram for this fixed instance. They describe repeated wall-clock runtimes on a single instance. C is the fastest formulation. C+U+T is faster than the base model on average, but the difference is much smaller than the difference between C and the base model. U and T are slower than the base model.

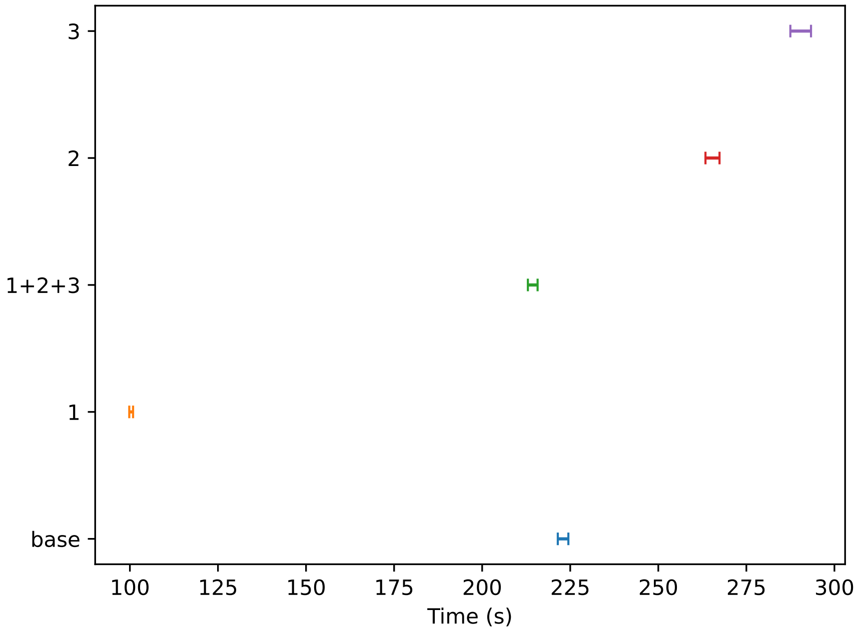


Figure 6: 95% confidence intervals for mean runtime on the fixed instance. Numeric model labels are defined in Table 1. The intervals describe repeated solves of one instance.

Taken together, the two experiments identify the summed passenger flow constraints C as the most effective tested change among the reported variants. This is due to solver performance. C keeps the same feasible set. The constraints in C are redundant, but they expose aggregate flow relations that can help presolve, cut generation, or branching. The explicit capacity variables U and the type-disaggregated formulation T add variables and constraints. In the fixed instance, this extra model size is not offset by a better relaxation.

6 Discussion

The first part of this section will discuss interpretations and possible explanations of the numerical results. At the same time, possible refinements of the experiment will be discussed. The second part will discuss possible future research questions and underline why they are believed to be important.

6.1 Interpretation of Results

Both experiment 1 and experiment 2 support that the summed passenger flow formulation C performs better than the base formulation B. The difference is statistically significant when considering the confidence intervals of experiment 2. The cause of the improvement is unclear. A possibility arises from considering the structure of the constraint. Since the cut W is chosen along the time axis, all flows entering and exiting the area are unidirectional. Therefore, the constraints assume the form

$$\sum_{s \in \delta_s^+(W)} x_{sp} = D_{Wp},$$

where D_{Wp} is the demand of good p across the boundary of W. Positive flow-variables instantly give upper bounds of the form

$$x_{sp} \leq D_{Wp},$$

which might be useful for deriving cuts or for pruning variables in the pre-solve stage. However, these gains might be gained by simply using graph-search algorithms to statically determine the maximum flow of each service edge before invoking the solver. This improvement should be included in future experiments to isolate the benefits of tighter bounds in the presolve stage from any cuts enabled by the constraint.

Both experiments agree that formulations U and T provide either no benefit or are actively detrimental. However, they disagree on formulation C+U+T. Experiment 1 favors C+U+T, whereas experiment 2 only places it slightly better than the base formulation. Additionally, the combination of U and T given by (21) is very likely to render the joint capacity variable useless as the right-hand side of the equation is guaranteed only to have one term. Therefore, future work should use C+U and C+T instead of C+U+T. Additionally, a larger set of instances closer to real-world data, with bounds on flight arc and airport capacities, should be used to determine whether C, C+U, or C+T actually performs better on more realistic instances.

More detail should also be given to the input data. Aircraft energy use, charging time, turnaround time, range, cost, and passenger demand are treated as a given. This is appropriate for a formulation study. However, the future of electric aircraft is unclear, and future work should include a variety of scenarios to determine the sensitivity to these data.

Finally, experiment 1 shows that solution times and MIP gaps generally increase with instance size. However, it is also clear that the details will vary strongly depending on the structure of the Airports and PSO requirements. This is expected, as a particularly tight time requirement would leave few possible passenger routes and thus constrain aircraft routing. Conversely, a wide time window would provide greater flexibility in aircraft and passenger routing, thereby expanding the feasible set.

6.2 Future directions of research

There is significant work to do to solve this model for practical instance sizes. However, conducting this research is worthwhile only if the mode offers actual advantages over alternatives. Therefore, I suggest three research directions to determine whether the model is a worthwhile investment.

First, the model should be compared as is to existing vehicle routing formulations and flight logistics models. If at all possible, this should be done in collaboration with industry to check the feasibility of model assumptions. However, acquiring real-world data to construct realistic instances is necessary. The main focus will be on whether the model can produce solutions that are quantitatively better, in other words, cheaper, than existing alternatives.

Second, the model should be expanded to reflect energy and charging capacity better. Presently, it assumes that a vehicle will always charge after using an arc and that every node has charging infrastructure. Dropping the first assumption would give the model greater freedom and might yield quantitatively better solutions. Dropping the second assumption would allow the model to more accurately reflect reality, as the infrastructure to handle electric aircraft or other electric vehicles may not be universally available. This would connect the model with charging and battery scheduling models (Justin, Payan, Briceno, et al. 2020; Mitici et al. 2022; Oosterom and Mitici 2023; Vehlhaber and Salazar 2023).

Third, the expanded model from the second step should be compared with other models, particularly those for electric aircraft. Particular focus should be given to the quality of the solutions.

7 Conclusion

This thesis studies a two-layer MILP model for regional electric aviation under PSO requirements. The aircraft layer represents feasible aircraft movements and capacity. The passenger layer routes passenger groups through the service network. The formulation is suitable for electric aircraft because aircraft timing and passenger timing can differ.

The thesis shows that PSO requirements can be represented by fewer passenger groups by aggregating requirements with shared departure or arrival conditions. Efficient aggregation can be reduced to bipartite vertex cover, which has polynomial algorithms.

The thesis states several alternative capacity formulations, valid inequalities, and redundant constraints. The reported numerical runs test C, U, T, and C+U+T. These variants keep the planning problem unchanged, but they change the size and structure of the MILP.

The numerical experiments identify the summed passenger flow constraints C as the clearest tested improvement in these runs. They improve performance in the fixed-instance experiment and perform well in experiment 1. The combined formulation C+U+T is often best in experiment 1, while C is best in the repeated fixed-instance experiment. The explicit capacity variables and the type-disaggregated formulation do not improve performance when used alone in the tested cases.

Solver performance decreases as the number of airports and PSO requirements increases. However, this general trend is sensitive to the structure of individual instances.

The numerical evidence provides an initial benchmark and a direction for further experiments. The next step is to test C, C+U, and C+T on more instances and to add finite aircraft fleets, active airport limits, PSO aggregation, demand-link constraints, several passenger volumes, and more detailed electric charging constraints. However, future work should focus on comparing this model with competing models to determine whether further research into solving it is motivated.

8 Use of AI in this work

This work makes minor use of generative AI. The main use case has been as an alternative to search engines in order to identify initial papers and keywords to read. From there, the standard techniques of mapping sources and identifying key papers and authors within the field were used.

An attempt was made to use generative AI to perform part of the project's programming. However, the generated code was of poor quality and difficult to maintain. Therefore, the generated code was limited in scope, primarily for figure generation. Fortunately, generative AI proved very useful for quickly identifying the names of functions in the libraries used.

The use of AI tools in the writing process has been limited. An LLM was used for an initial title suggestion. LLMs were also used to create an initial Swedish abstract. Finally, a grammar-checking tool was used for a final pass through the report.

Acknowledgment

I thank my supervisor, Jonas Westin, for his advice and guidance during the project. I would also like to thank Per Åhag for his valuable criticisms. Finally, I thank my family and friends for their patience and support during the intense final weeks of the writing process.

References

Adu-Gyamfi, Bright Appiah and Clara Good (2022). "Electric aviation: A review of concepts and enabling technologies". In: *Transportation Engineering* 9, p. 100134. DOI: 10.1016/j.treng.2022.100134.

Avogadro, Nicolò and Renato Redondi (2024). "Demystifying electric aircraft's role in aviation decarbonization: Are first-generation electric aircraft cost-effective?" In: *Transportation Research Part D: Transport and Environment* 130, p. 104191. DOI: 10.1016/j.trd.2024.104191.

Bærheim, Trym, Jacob J. Lamb, Jonas Kristiansen Nøland, and Odne S. Burheim (2023). "Potential and limitations of battery-powered all-electric regional flights: A Norwegian case study". In: *IEEE Transactions on Transportation Electrification* 9.1, pp. 1809–1825. DOI: 10.1109/TTE.2022.3200089.

Barnhart, Cynthia, Peter Belobaba, and Amedeo R. Odoni (2003). “Applications of operations research in the air transport industry”. In: *Transportation Science* 37.4, pp. 368–391. DOI: 10.1287/trsc.37.4.368.23276.

Barnhart, Cynthia, Ellis L. Johnson, George L. Nemhauser, Martin W. P. Savelsbergh, and Pamela H. Vance (1998). “Branch-and-price: Column generation for solving huge integer programs”. In: *Operations Research* 46.3, pp. 316–329. DOI: 10.1287/opre.46.3.316.

Birolini, Sebastian, António Pais Antunes, Mattia Cattaneo, Paolo Malighetti, and Stefano Paleari (2021). “Integrated flight scheduling and fleet assignment with improved supply-demand interactions”. In: *Transportation Research Part B: Methodological* 149, pp. 162–180. DOI: 10.1016/j.trb.2021.05.001.

Bråthen, Svein and Knut Sandberg Eriksen (2018). “Regional aviation and the PSO system: Level of service and social efficiency”. In: *Journal of Air Transport Management* 69, pp. 248–256. DOI: 10.1016/j.jairtraman.2016.10.002.

Chouman, Mervat, Teodor Gabriel Crainic, and Bernard Gendron (2017). “Commodity representations and cut-set-based inequalities for multicommodity capacitated fixed-charge network design”. In: *Transportation Science* 51.2, pp. 650–667. DOI: 10.1287/trsc.2015.0665.

Clautiaux, François and Ivana Ljubić (2025). “Last fifty years of integer linear programming: A focus on recent practical advances”. In: *European Journal of Operational Research* 324.3, pp. 707–731. DOI: 10.1016/j.ejor.2024.11.018.

Crainic, Teodor Gabriel (2000). “Service network design in freight transportation”. In: *European Journal of Operational Research* 122.2, pp. 272–288. DOI: 10.1016/S0377-2217(99)00233-7.

Desaulniers, Guy, Jacques Desrosiers, Yvan Dumas, Marius M. Solomon, and François Soumis (1997). “Daily aircraft routing and scheduling”. In: *Management Science* 43.6, pp. 841–855. DOI: 10.1287/mnsc.43.6.841.

Dolan, Elizabeth D. and Jorge J. Moré (2002). “Benchmarking optimization software with performance profiles”. In: *Mathematical Programming* 91, pp. 201–213. DOI: 10.1007/s101070100263.

European Parliament and Council of the European Union (2008). *Regulation (EC) No 1008/2008 of the European Parliament and of the Council of 24 September 2008 on common rules for the operation of air services in the Community (Recast) (Text with EEA relevance)*. Official Journal of the European Union, L 293, 31 October 2008, pp. 3–20. URL: https://eur-lex.europa.eu/eli/reg/2008/1008/oj.

Fageda, Xavier, Ancor Suárez-Alemán, Tomás Serebrisky, and Reinaldo Fioravanti (2018). “Air connectivity in remote regions: A comprehensive review of existing transport policies worldwide”. In: *Journal of Air Transport Management* 66, pp. 65–75. DOI: 10.1016/j.jairtraman.2017.10.008.

Gurobi Optimization, LLC (2026). *Gurobi Optimizer Reference Manual, Version 13.0*. URL: https://docs.gurobi.com/projects/optimizer/en/current/index.html.

Hane, Christopher A., Cynthia Barnhart, Ellis L. Johnson, Roy E. Marsten, George L. Nemhauser, and Gabriele Sigismondi (1995). “The fleet assignment problem: Solving a large-scale integer program”. In: *Mathematical Programming* 70, pp. 211–232. DOI: 10.1007/BF01585938.

Hopcroft, John E. and Richard M. Karp (1973). “An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs”. In: *SIAM Journal on Computing* 2.4, pp. 225–231. DOI: 10.1137/0202019.

Justin, Cedric Y., Alexia P. Payan, Simon I. Briceno, Brian J. German, and Dimitri N. Mavris (2020). “Power optimized battery swap and recharge strategies for electric aircraft operations”. In: *Transportation Research Part C: Emerging Technologies* 115, p. 102605. DOI: 10.1016/j.trc.2020.02.027.

Justin, Cedric Y., Alexia P. Payan, and Dimitri N. Mavris (2022). “Integrated fleet assignment and scheduling for environmentally friendly electrified regional air mobility”. In: *Transportation Research Part C: Emerging Technologies* 138, p. 103567. DOI: 10.1016/j.trc.2022.103567.

Kinene, Alan and Sebastian Birolini (2024). “Optimization of subsidized air transport networks using electric aircraft”. In: *Transportation Research Part B: Methodological* 190, p. 103065. DOI: 10.1016/j.trb.2024.103065.

Kinene, Alan, Sebastian Birolini, Mattia Cattaneo, and Tobias Andersson Granberg (2023). “Electric aircraft charging network design for regional routes: A novel mathematical formulation and kernel search heuristic”. In: *European Journal of Operational Research* 309.3, pp. 1300–1315. DOI: 10.1016/j.ejor.2023.02.006.

Kinene, Alan, Tobias Andersson Granberg, Valentin Polishchuk, and Clas Rydergren (2020). “Decision support for an optimal choice of subsidised routes in air transportation”. In: *Journal of Air Transport Management* 82, p. 101724. DOI: 10.1016/j.jairtraman.2019.101724.

Leandro, Francisco, Antonio R. Andrade, and Sofia Kalakou (2021). “Designing aviation networks under public service obligations (PSO): A case study in Greece”. In: *Journal of Air Transport Management* 93, p. 102042. DOI: 10.1016/j.jairtraman.2021.102042.

Lohatepanont, Manoj and Cynthia Barnhart (2004). “Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment”. In: *Transportation Science* 38.1, pp. 19–32. DOI: 10.1287/trsc.1030.0026.

Magnanti, Thomas L. and Richard T. Wong (1984). “Network design and transportation planning: Models and algorithms”. In: *Transportation Science* 18.1, pp. 1–55. DOI: 10.1287/trsc.18.1.1.

Mitici, Mihaela, Madalena Pereira, and Fabrizio Oliviero (2022). “Electric flight scheduling with battery-charging and battery-swapping opportunities”. In: *EURO Journal on Transportation and Logistics* 11, p. 100074. DOI: 10.1016/j.ejtl.2022.100074.

Nordic Innovation (2024). *Reinventing aviation in the Nordics: The state of electric aviation and the case for enhanced industry-government collaboration.* Tech. rep. Nordic Innovation. URL: https://norden.diva-portal.org/smash/get/diva2:1912532/FULLTEXT02.pdf.

Oosterom, Simon van and Mihaela Mitici (2023). “Optimizing the battery charging and swapping infrastructure for electric short-haul aircraft: The case of electric flight in Norway”. In: *Transportation Research Part C: Emerging Technologies* 155, p. 104313. DOI: 10.1016/j.trc.2023.104313.

Pita, João P., Nicole Adler, and António P. Antunes (2014). “Socially-oriented flight scheduling and fleet assignment model with an application to Norway”. In: *Transportation Research Part B: Methodological* 61, pp. 17–32. DOI: 10.1016/j.trb.2013.12.006.

Pita, João Pedro, António Pais Antunes, Cynthia Barnhart, and António Gomes de Menezes (2013). "Setting public service obligations in low-demand air transportation networks: Application to the Azores". In: *Transportation Research Part A: Policy and Practice* 54, pp. 35–48. DOI: 10.1016/j.tra.2013.07.003.

Rexing, Brian, Cynthia Barnhart, Tim Kniker, Ahmad Jarrah, and Nirup Krishnamurthy (2000). "Airline fleet assignment with time windows". In: *Transportation Science* 34.1, pp. 1–20. DOI: 10.1287/trsc.34.1.1.12277.

Schäfer, Andreas W., Steven R. H. Barrett, Khan Doyme, Lynnette M. Dray, Albert R. Gnadt, Rod Self, Aidan O'Sullivan, Athanasios P. Synodinos, and Antonio J. Torija (2019). "Technological, economic and environmental prospects of all-electric aircraft". In: *Nature Energy* 4, pp. 160–166. DOI: 10.1038/s41560-018-0294-x.

Sherali, Hanif D., Ki-Hwan Bae, and Mohamed Haouari (2010). "Integrated airline schedule design and fleet assignment: Polyhedral analysis and Benders' decomposition approach". In: *INFORMS Journal on Computing* 22.4, pp. 500–513. DOI: 10.1287/ijoc.1090.0368.

Sherali, Hanif D., Ebru K. Bish, and Xiaomei Zhu (2006). "Airline fleet assignment concepts, models, and algorithms". In: *European Journal of Operational Research* 172.1, pp. 1–30. DOI: 10.1016/j.ejor.2005.01.056.

Tang, Yuxiang, Li Wang, Wenxuan Kang, Wenchao Liu, and Yufeng Zhuang (2026). "Electric and fuel aircraft schedule optimizing problem with battery charging and swapping: The case of flight in China". In: *Journal of Air Transport Management* 130, p. 102890. DOI: 10.1016/j.jairtraman.2025.102890.

Vehlhaber, Finn and Mauro Salazar (2023). "Electric aircraft assignment, routing, and charge scheduling considering the availability of renewable energy". In: *IEEE Control Systems Letters* 7, pp. 3669–3674. DOI: 10.1109/LCSYS.2023.3339998.

Wang, I-Lin (2018a). "Multicommodity network flows: A survey, Part I: Applications and formulations". In: *International Journal of Operations Research* 15.4, pp. 145–153. DOI: 10.6886/IJOR.201812_15(4).0001.

– (2018b). "Multicommodity network flows: A survey, Part II: Solution methods". In: *International Journal of Operations Research* 15.4, pp. 155–173. DOI: 10.6886/IJOR.201812_15(4).0002.

Wei, Keji, Vikrant Vaze, and Alexandre Jacquillat (2020). "Airline timetable development and fleet assignment incorporating passenger choice". In: *Transportation Science* 54.1, pp. 139–163. DOI: 10.1287/trsc.2019.0924.

Westin, Jonas, Leif Olsson, and Per Åhag (2024). "Network design optimization for regional electric aviation in northern Scandinavia". In: *2024 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*. Bangkok, Thailand: IEEE, pp. 400–404. ISBN: 979-8-3503-8609-7. DOI: 10.1109/IEEM62345.2024.10856984.

Williams, George and Romano Pagliari (2004). "A comparative analysis of the application and use of public service obligations in air transport within the EU". In: *Transport Policy* 11.1, pp. 55–66. DOI: 10.1016/S0967-070X(03)00040-4.

Wolsey, Laurence A. (2020). *Integer Programming*. 2nd ed. John Wiley & Sons. ISBN: 978-1-119-60653-6. DOI: 10.1002/9781119606475.

Xu, Yifan, Sebastian Wandelt, and Xiaoqian Sun (2024). "Airline scheduling optimization: Literature review and a discussion of modelling methodologies". In: *Intelligent Transportation Infrastructure* 3, liad026. DOI: 10.1093/iti/liad026.

Yan, Chiwei, Cynthia Barnhart, and Vikrant Vaze (2022). “Choice-based airline schedule design and fleet assignment: A decomposition approach”. In: *Transportation Science* 56.6, pp. 1410–1431. DOI: 10.1287/trsc.2022.1141.