FLEET & TAIL ASSIGNMENT UNDER UNCERTAINTY

GLOMB, LIERS, AND RÖSEL

All authors: FAU Erlangen-Nuremberg, Department of Mathematics, Cauerstraße 11, 91058 Erlangen, Germany

Abstract. Airlines solve many different optimization problems and combine the resulting solutions to ensure smooth, minimum-cost operations. Crucial problems are the Fleet Assignment, which assigns aircraft types to flights of a given schedule, and the Tail Assignment, which determines individual flight sequences to be performed by single aircraft. In order to find a cost-optimal solution, many airlines use mathematical optimization models. For these to be effective, the available data and forecasts must reflect the situation as accurately as possible. However, especially in times of a pandemic, the underlying plan is subject to severe uncertainties: Staff and demand uncertainties can even lead to flight cancellations or result in entire aircraft having to be grounded. Therefore, it is advantageous for airlines to protect their mathematical models against uncertainties in the input parameters. In this work, two computational tractable and cost-efficient robust models and solution approaches are developed: First, we set up a novel mixed integer model for the integrated fleet and tail assignment, which ensures that as few subsequent flights as possible have to be canceled in the event of initial flight cancellations. We then solve this model using a procedure that ensures that the costs of the solution remain considerably low. Our second model is an extended fleet assignment model that allows us to compensate for entire aircraft cancellations in the best possible way, taking into account rescheduling options. We demonstrate the effectiveness of both approaches by conducting an extensive computational study based on real flight schedules of a major German airline. It turns out that both models deliver stable, cost-efficient solutions within less than ten minutes, which significantly reduce follow-up costs in the case uncertainties arise.

Keywords. Mixed Integer Programming, Robust Optimization, Fleet Assignment, Tail Assignment

1. Introduction

Airlines have relied on mathematical optimization methods for many years to manage the multitude of processes they have to coordinate. The range of these processes is very diverse: strategic decisions include the purchase of aircraft and the design of the network structure. In the medium term, flight plans are created that are based on forecasts. This is accompanied by planning the assignment of aircraft to individual flights. Airliners usually divide this process into two steps due to its complexity: The assignment of aircraft types is referred to as *fleet assignment*, since aircraft types are referred to as *fleets* in the airline context. The *tail assignment*, in turn, describes the assignment of individual aircraft (identified by their *tail* number) to flights. Both processes are of particular importance in the context of airline planning, as they have to be planned well in advance and are influenced by highly dynamic uncertainties. For this reason, airlines repeatedly reoptimize these plans until the day of the actual flight execution. These re-planning measures can include the exchange of different aircraft types (so-called *equipment changes*) as well as type-internal *tail swaps*.

E-mail address: lukas.glomb@fau.de (CA's email), frauke.liers@fau.de, florian.roesel@fau.de.

Optimization models for the above problems that do not hedge against uncertainty are called nominal models. There are different methods to safeguard nominal models against uncertainties. Three of the most prominent approaches are robust optimization (see (5)), stochastic optimization (see (7)) and the use of heuristic application-specific models (see (29)). Robust optimization safeguards models against all parameter realizations that are within a specified uncertainty set. Thereby, the feasibility of the solution can be guaranteed for all realizations from the uncertainty set. With the help of stochastic optimization, models can be safeguarded whenever stochastic information about the realization, such as its distribution, is available. Stochastic models secure the model with respect to the expected value of parameters and can be applied if the probability of occurrence of individual uncertainty realizations - so-called scenarios - is known. Application-specific models make use of special problem structures to ensure that the solutions of a model remain feasible or near-optimal if disruptions occur. The nominal model is modified in such a way that solutions that work well in practice are selected with preference.

As uncertainties often strongly impact aviation, airlines are increasingly using these hedging optimization concepts to safeguard the solutions of their mathematical models. Planners often use application-specific heuristics to simplify the model complexity and to generate more stable solutions. Speaking about the assignment of aircraft, the number of potential flight connections is reduced by general rules to make the routes of aircraft fit to a given general scheme. This has the disadvantage that many possibly better solutions that do not fit this scheme are excluded from the set of feasible solutions. In addition, stochastic optimization methods that work by means of certain probability distributions are implemented. However, many of these models are designed to compensate for comparatively harmless disruptions such as airplane delays. Others deal with severe uncertainties such as potential flight cancellations, which can directly influence the structure of the problem, but in doing so they drastically reduce the number of possible flight connections and thus the set of possible feasible solutions.

Complex algorithm techniques like the L-Shaped method, which can in fact deal with such crucial structural uncertainties, are often limited by their effectiveness whenever the number of considered scenarios grows.

This paper takes a new look on fleet assignment and tail assignment using robust optimization methods to come by this challenges. This is done with two novel models: Large variations in demand, newly created schedules and unplanned maintenance often lead to flight cancellations. We thus develop a model which deals with structural uncertainties in the flight schedule if some flights are at risk of cancellation. We introduce an application-specific modeling approach and embed it in a procedure which is often used in robust optimization to ensure the costs of the robust solution to remain reasonably low. Instead of reducing the number of possible solutions to fit a given scheme, the model aims at minimizing consequential cancellations in case of flight cancellations without the possibility of replanning.

A second relevant kind of major disruptions are entire aircraft failures, which occur if major unplanned technical disruptions or personnel shortages cause an aircraft to stay on ground for a longer time-period. These usually lead to a whole series of flight cancellations, which are expensive for airlines. The potential occurrence of such major disruptions is usually not robustly considered in fleet or tail assignment plans, as the resulting robust plans are deemed too expensive and intractable to calculate. Thus, we propose a new model, which optimizes the fleet assignment in such a way that it can be ensured that flight cancellations can be optimally compensated by means of tail swaps. We prove that it can capture the costs of these flight cancellations accurately and is thus equivalent to scenario-expanded models typically used in robust or stochastic optimization.

The remainder of this paper is as follows: In Section 2, we provide an extensive overview of the existing literature on fleet and tail assignment with a particular focus on uncertainty concepts. In Section 3, a modelling approach for the integrated fleet and tail assignment based on flight connections is introduced. This model is then extended to reflect the costs incurred by flight cancellations resulting in further schedule changes. The model is embedded in a procedure that ensures that the cost of the robust solution remains reasonably low compared to the nominal solution. In Section 4 we introduce a basic modelling approach for the fleet assignment problem using the so-called time-space graph. This model is modified so that it can capture and minimize the costs associated with aircraft failures, taking tail-swaps into account. We prove its equivalence to a two-stage robust scenario-expanded model using acceptable scenario assumptions. In Section 5 we describe computational studies regarding both introduced robustness concepts, benchmark them against established models in literature and discuss the results. For both models cost-efficient solutions are found within only ten minutes, which assign aircraft in such a way that in case of occurring uncertainties their follow-up costs can be significantly reduced compared to conventional models. Section 6 concludes with a summary and gives an outlook on future promising research topics in this field.

2. Literature

Both fleet assignment and tail assignment are problems that have been extensively addressed in the literature for a long time. Already in (1) an integer linear program was described, which solves the fleet assignment problem by means of connection variables. Therefore, binary variables for each aircraft type and each valid flight connection are required, which lead to the number of variables increasing quadratically in the number of flights for each additional flight considered.

To reduce the complexity of the problem, it was broken down into fleet and tail assignment. First models for the assignment of aircraft types to flights can be found in (6) and (18). Both publications use a graph structure that will later be known as the time-space graph. It is an aggregation of the connection-based graph resulting in an integer multi-commodity flow problem with a reduced number of variables. In turn, however, the flight sequences of individual aircraft cannot be uniquely identified.

Due to the structure of the problem and the large number of variables, (4) decided to solve the problem using column generation. In this case, individual variables do not represent an assignment to flights or connections, but the assignment of entire flight sequences. However, since there are exponentially many ways to combine individual flights into flight sequences, a full enumeration would require too many variables. Therefore, favorable flight sequences are successively added to the model using column generation until the optimal solution is found.

Since the considered problem becomes very large in practice, there exist other solution approaches in the literature, such as in (15). The authors use a rolling horizon approach with provable solution quality guarantee in order to solve the fleet and tail assignment problem taking into account the maintenance planning.

In more recent literature, the solution approaches have not changed significantly. Instead, more and more planning processes have been incorporated into the basic models: Often, the tail assignment problem is solved in combination with maintenance planning, as in (16) and (9), for example. In (10) and (24), the fleet assignment is solved in combination with scheduling concepts, which determine when and how frequently flights have to be executed. (26) and (23) incorporate crew assignment decisions while providing solution to aircraft assignment problems. In (14), the integrated fleet and tail assignment with turnaround decisions is considered and solved using a decomposition procedure.

Both fleet and tail assignment are subject to many uncertainties and the solutions often have to be adjusted due to weather conditions, crew considerations, flight cancellations and much more. Therefore, many publications deal with the generation of stable, secured solutions.

A very popular method to ensure stable tail assignment solutions is to avoid delay propagation. This concept is based on a distinction between two types of delays: Primary delay denotes delay that is independent of the aircraft's flight sequences. It includes, for example, delay caused by trajectory adjustments due to in-flight conflicts (see (20)) or re-prioritization of the landing sequence, which is determined by runway scheduling. Secondary delay denotes the consequential delay of subsequent flights caused by primary delay on previous flights, which is passed on along flight connections due to lack of buffer times. Instead of reducing the primary delay as for example in (19), the following papers aim at finding suitable fleet and tail assignment solutions that minimize the secondary delay: One of the first papers on this subject was presented (21): A stochastic program is set up, which minimizes the absolute expected propagated delay. In (3) so-called propagation trees for the analysis of propagated delay are established, which can measure the network's propagated delay based not only on delayed aircraft but also on other subsequent resources, like crews or passengers. (8) developed a most robust rotation algorithm, which minimizes the probability of delay propagation by avoiding too tightly planned flight sequences using column generation. (11) minimize the propagated delay by solving an integrated tail assignment and crew pairing problem. The resulting model is capable of capturing and minimizing the occurring delay based on both decision dimensions. In (31) robust optimization is used to construct an uncertainty set regarding the delay propagation against which the tail assignment solution is hedged. The solutions do not get too expensive since the robust uncertainty set is calculated by using stochastic parameters to avoid hedging against unrealistic delay situations.

Less common in the literature are approaches that hedge against entire flight cancellations or aircraft failures, which we address in this paper: (2) evaluates the robustness of different flight sequences by means of *swap* opportunities, which means possibilities for interchanging different aircraft during the plan. The author compares multiple solutions of the problem and ranks them by the number of possible swap opportunities. (25) safeguard the fleet assignment problem using a heuristic which maximizes the number of short flight cycles. A flight cycle describes a flight sequence between stopovers at a central main airport, which is referred to as *hub airport*. If the entire cycle has to be canceled due to a single flight cancellation, as few flights as possible are affected. In addition, indirect connections from different hub airports are reduced in order to prevent delays and cancellations from spreading throughout the entire network. (13) use a special Benders Decomposition approach called L-Shaped method to solve a model which stabilizes the plan in the case of flight cancellations. Therefore they expand the nominal problem by additional variables, which model the tail assignment again for each flight cancellation scenario. Thus, one is able to obtain the best possible initial tail assignment up to the point in time when failures occur.

Other robust approaches try to safeguard related problems which are often combined with the task of fleet and tail assignment: (28) maximize the possibilities for crew swaps, (32) try to safeguard the crew scheduling using a two-stage stochastic program and in (11) both aircraft routing and crew pairing are hedged to cause minimal propagated delays. Nevertheless, since using Benders Decomposition with continuous subproblems, the costs calculated per scenario only approximate the real costs arising.

3. Light Robustness for Aircraft Routing: Flight Cancellations

In this section, a model is introduced which can be used to perform the fleet assignment as well as the tail assignment. Afterwards this model is modified to be able to deal with flight cancellations. This adjusted model is then embedded in a procedure which enables to compute robust solutions while keeping the costs for the robustification within acceptable borders.

3.1. **Integrated Fleet and Tail Assignment.** The model's purpose is to identify unique optimal flight sequences for each individual aircraft. It can either be formulated just to compute a feasible tail assignment based on a given fleet assignment or to optimize across fleets and thus make the fleet assignment decisions as well. We describe the cross-fleet model, which we refer to as *aircraft assignment* in the remainder of this paper.

The model presented here is based on (1), using graph structures called connection network graphs.

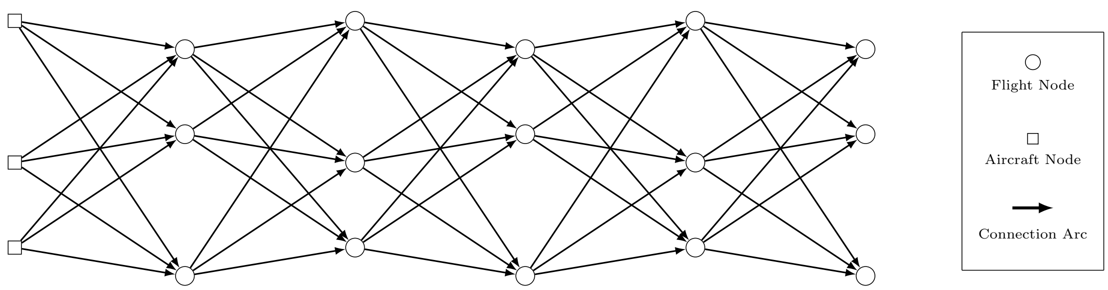

Figure 1. Connection network graph for one fleet $k \in K$ with three aircraft and 15 flight nodes

To understand the idea of this graph and to be able to set up the aircraft assignment model, all necessary sets, parameters and variables are summarized in Table 1. The corresponding graph structure is depicted in Figure 1.

The set K corresponds to the set of fleets, while all flights are included in the set L. The set of all aircraft is denoted with Q. For each aircraft there exists an additional node included in the connection network graph in Figure 1. The outgoing arcs state connections to first flights which can be executed by the aircraft corresponding to the start node. All connections together are called event connections and form the set A. L_k, Q_k and A_k denote the set of flights, aircaft and connections which are part of the graph for fleet $k \in K$. The Coefficient c_i^k denotes the costs for assigning an aircraft of fleet $k \in K$ to flight $i \in L_k$.

The model includes binary variables: $x^k_{(i,j)}$ is set to 1 whenever flight j is executed directly after flight i by the same aircraft of fleet k. The variable y_i^k indicates whether a single flight $i \in L_k$ is executed by fleet $k \in K$. Note, that the y-variables do not have to be part of the model and can be substituted by sums of x-variables. They are introduced to simplify further model adjustments. The variable z_i indicates whether flight i is canceled or not.

Set	**Description**
K	set of fleets
L	set of flights
Q	set of aircraft
$A \subset (L \cup Q) \times L$	set of event connections
$L_k \subset L$	set of flights $l \in L$ executable by fleet $k \in K$
$Q_k \subset Q$	set of aircraft $q \in Q$ in fleet $k \in K$
$A_k \subset A$	set of flight connections $a \in A$ executable by fleet $k \in K$
$K_i \subset K$	set of fleets capable of executing flight $i \in F$
Parameter	**Description**
c_i^k	costs for assigning fleet $k \in K$ to flight leg $i \in L$
c_i	costs for the cancellation of flight $i \in L$
Variable	**Description**
$x_{(i,j)}^k$	binary variable indicating if an aircraft of fleet $k \in K$ is assigned to connection $(i,j) \in A_k$
y_i^k	binary variable indicating if fleet $k \in K$ is assigned to flight $i \in L_k$
z_i	binary variable indicating if flight leg $i \in L$ is canceled or not

TABLE 1. Sets, parameters and variables for the basic tail assignment model

An optimal solution of the following model corresponds to a cost optimal assignment of aircraft fleets to flight connections:

$$\min \sum_{k\in K}\sum_{i\in L_k} c_i^k y_i^k + \sum_{i\in L} c_i z_i \tag{1a}$$

$$\text{s.t.} \quad \sum_{k\in K_i} y_i^k + z_i = 1 \qquad \forall i \in L_k \tag{1b}$$

$$\sum_{h:(h,i)\in A_k} x_{(h,i)}^k = y_i^k \qquad \forall i \in L_k, k \in K \tag{1c}$$

$$\sum_{j:(i,j)\in A_k} x_{(i,j)}^k \leq y_i^k \qquad \forall i \in L_k, k \in K \tag{1d}$$

$$\sum_{i:(q,i)\in A_k} x_{(q,i)}^k \leq 1 \qquad \forall q \in Q_k, k \in K \tag{1e}$$

$$x, y, z \quad \text{binary} \tag{1f}$$

The Objective (1a) is to minimize the costs of assigning flights to specific fleets and cancellation costs which arise whenever no fleet is assigned to a flight. Constraint (1b) makes sure, that every flight is either covered by exactly one aircraft or it is canceled. Constraint (1c) and Constraint (1d) together guarantee the flow conservation on each flight node. Feasible flows are disjoint paths, since x and y are binary, since the flow into and out of each node has to equal 0 or 1.

Constraints (1e) make sure that not more than the number of available aircraft is used. In order to characterize solutions of Problem (1), a *flight string* is defined as follows:

Definition 1. *Let x^*, y^*, z^* be a solution vector of Problem* (1)*. Then we define the flight string L_k^q as the set of all flights $l \in L_k$ which are executed by aircraft $q \in Q_k$:*

$$L_k^q := \{l \in L_k | \exists \mathcal{A} = \{(q,i_1),(i_1,i_2),...,(i_{n-1},i_n),(i_n,l)\} \subseteq A_k : \sum_{(i,j)\in\mathcal{A}} {x_{(i,j)}^k}^* = |\mathcal{A}|\}$$

Simply speaking, Definition 1 introduces subsets L_k^q of L_k for each $q \in Q_k$ and all $k \in K$ containing all flights that are assigned to aircraft q of type k. We note that it depends explicitly on a given feasible solution of the Problem (1).

3.2. **Robust model hedging against flight cancellations.**

Especially in times of pandemics, airlines have to face strongly varying demand caused by inaccurate forecasts. If the passenger number on a flight is very low, it may be more lucrative to re-book passengers than to let the plane take off. Flight cancellations can also be triggered by many other factors: Technical difficulties, temporary airspace closures, or severely accumulated delays.

Flight cancellations usually lead to entire sequences of flights having to be suspended because the aircraft is no longer available at the destination airport of the original canceled flight. Deadhead flights, i.e. flights without passengers in order to get the aircraft to the right location, are sometimes an option, but often too costly or simply impossible due to time restrictions.

Hence, airlines are constantly rescheduling to compensate for such flight cancellations: The Airline Operations Center tries to reschedule flight routes in order to cancel as few subsequent flights as possible. However, this is not always possible due to time factors such as tight schedules, a lack of aircraft or a shortage of qualified crews and pilots. For this reason, the initial flight sequences is designed to ensure that even in this case the plan remains as stable as possible. This stabilization approach without re-planning has already been addressed in the publication of (25). The focus there is on so-called short cycles, i.e. flight sequences that serve only a few flights between two stops at a central hub airport. Empirically, this method is efficient, but has some drawbacks: In practice, the concept works mainly for flight plans of hub-and-spoke networks, which imply that aircraft return frequently to main airports (called hubs), but is less effective for point-to-point networks, which range of connections is more diverse. In theory, only the set of possible solutions of the model was reduced. The approach is hence a heuristic, and it comes with no provable solution quality.

In order to build on the great potential of this idea, we enhance it by modeling the short cycles explicitly on the basis of scenarios without restricting the solution space. In doing so, we do not depend on any particular network structure. The novel model will then be embedded in a procedure, which is designed to adjust the cost difference between the nominal and hedged solutions according to user preference. Thus, this approach limits the additional costs incurred for hedging solutions against uncertainties. The idea of this approach is in the spirit of the concept of *light robustness* introduced in in (12).

The remainder of this section is organized as follows: First, the auxiliary Model (2) is introduced to explain how flight cancellations can be recorded under the assumption that no re-planning is allowed. This model thus represents a generalization of the evaluation of flight cancellation costs, which is specified in (25). Based on this evaluation method, we present the novel Model (3) as an extension of Model (5), which can capture the costs of flight cancellations directly when optimizing the aircraft assignment. We then prove that this model does not constrain the set of feasible flight routes (Lemma 1) and generally overestimates the cost of the original evaluation model (Theorem 1). We then give sufficient conditions under which it accurately represents the costs of the reference model in Theorem 2.

First we explain in which flight cancellation scenarios how many flights are affected. To do this, we want to investigate how canceled flights affect the possibility to execute other flights, if the paradigm of "no re-planning" is considered:

Definition 2. *Let x^*, y^*, z^* be a solution of Problem* (1) *and let L_k^q for all $q \in Q_k$ and $k \in K$ be the resulting flight strings introduced in Definition 1. Let furthermore $\tilde{L} \subseteq L$ be the set of flights which have to be canceled and $\tilde{L}_k^q$ be a subset of L_k^q including all flights which touch-down before the departure time of all flights in $L_k^q \cap \tilde{L}$. The Re-optimization model without re-planning of Model* (1) *is equivalent to*

$$\min \sum_{i \in L} c_i z_i$$

$$s.t. \qquad (1b) - (1f)$$

$$(2a) \qquad z_i = 1 \qquad \forall i \in \tilde{L}$$

$$(2b) \qquad y_i^k \leq {y_i^k}^* \qquad \forall i \in L_k, k \in K$$

$$(2c) \qquad x_{(i,j)}^k = {x_{(i,j)}^k}^* \qquad \forall (i,j) \in A_k \cap \{L_k^q \times \tilde{L}_k^q\}, q \in Q_k, k \in K$$

$$(2d) \qquad x_{(i,j)}^k = 0 \qquad \forall (i,j) \in A_k \cap \{L_k^{q_1} \times L_k^{q_2}\}, q_1, q_2 \in Q_k, q_1 \neq q_2, k \in K$$

This means that all flights executed still have to be assigned to the same individual aircraft. The Constraint 2c models non-anticipativity as it is not possible to cancel flights in advance. It is equivalent to state that one knows about the flight cancellation after all previous flights of the corresponding flight string are already executed. As a result, aircraft on whose flight string a flight is canceled remain at the departure airport of the canceled flight until the next scheduled flight on their flight route departs from that airport. Therefore, all flights that should have taken place in the meantime are also canceled. This includes at least one additional flight, since this flight must transport the aircraft back to the original airport. Although the model does not consider rescheduling, it is realistic in practice whenever aircraft often return to the same airports, which is what we target in the following model. If this is the case, rescheduling does not necessarily lead to fewer flight cancellations, since aircraft and crews are limited and would thus in turn be missing on other flights.
In order to determine the number of additional flights that are canceled in this way and the corresponding cancellation costs, we modify an already established modeling technique that occurs in so-called Aircraft Maintenance Routing (27) : Depending on the selected flight sequence per aircraft, the take-offs performed are counted for each flight to ensure that maintenance has to be performed again after a certain number of take-offs. However, we have to count remaining flights instead of flights that have already been executed. In addition, the relevant number of subsequent flights depends on an airport as a reference point. Thus, the value of subsequent flight cancellations at a flight connection may vary, depending on whether the flight itself is the initial cancellation or a previous flight is initially canceled. The situation can be modeled with the help of the sets, parameters and additional variables listed in Table 2.
The set $\tilde{L}$ denotes the flights which will possibly get canceled initially. S denotes the set of airports at which flights can depart or arrive. The parameter M describes the maximal amount of costs which can arise if all flights of a single flight string are canceled. Thus, M can be approximated using the maximal cancellation costs per flight and the maximal number of flights a single aircraft of a certain type can execute. Our model includes additional continuous variable sets: The variable $\omega_i^{s,k}$ equals the costs which arise if all flights which are executed after flight $i \in L_k$ with fleet $k \in K$ are canceled until the next arrival at airport $s \in S$. ω_i^s aggregates this value over all fleets and thus represents the additional costs if flight i is canceled. The model

Sets	Description
$\tilde{L}$	set of flights which may possibly be canceled
S	set of airports
Parameter	**Description**
$\text{dep}(j)$	airport in S from which flight j departs
M	sufficiently large number (maximal cancellation costs on a single flight string)
$s(i,j)$	airport on which an aircraft is grounded between the execution of flight i and j
Variable	**Description**
$\omega^{s,k}_{(i,j)}$	continuous variable indicating additional costs that arise if all flights executed after connection (i,j) with fleet k got canceled until the next arrival at airport s
$\omega^{s,k}_{i}$	continuous variable indicating additional costs that arise if all flights executed after flight i with fleet k got canceled until the next arrival at airport s
ω^{s}_{i}	continuous variable indicating additional costs that arise if flight i got canceled and all following flights are canceled until airport s is reached

TABLE 2. Sets, parameters and variables for the flight cancellation extension

can be formulated as follows:

$$\min \sum_{j\in\tilde{L}} \omega_j^{\text{dep}(j)} \tag{3a}$$

$$\text{s.t.} \qquad (1b) - (1f)$$

$$0 \le \omega^{s,k}_{(i,j)} \le \omega^{s,k}_{j} \qquad (i,j)\in A_k, k\in K, s\in S \tag{3b}$$

$$\omega^{s,k}_{j} - M(1-x^k_{(i,j)}) \le \omega^{s,k}_{(i,j)} \le M x^k_{(i,j)} \qquad (i,j)\in A_k, k\in K, s\in S \tag{3c}$$

$$c_i y^k_i + \sum_{(i,j)\in A^k: s(i,j)\neq s} \omega^{s,k}_{(i,j)} \le \omega^{s,k}_{i} \qquad \forall i\in L, k\in K, s\in S \tag{3d}$$

$$\sum_{k\in K} \omega^{s,k}_{i} = \omega^{s}_{i} \qquad \forall i\in L, s\in S \tag{3e}$$

$$x,y,z \quad \text{binary} \tag{3f}$$

$$\omega \quad \text{continuous} \tag{3g}$$

Model (3) is an extension of Model (1) and thus it contains Constraints (1b) - (1f). The objective function has been changed in order to minimize the follow-up cost ω-variables in the case that all flights $l\in\tilde{L}$ are canceled. Constraints (3b) and (3c) are a linearization of the constraint $\omega^{s,k}_{(i,j)} = \omega^{s,k}_{j} x^k_{(i,j)}$ using a sufficiently large number M, which exceeds the maximal value $\omega^{s,k}_{j}$ can take. Having this in mind, Constraint (3d) ensures, that each variable $\omega^{s,k}_{i}$ equals the cancellation costs of flight $i\in L_k$ if executed with fleet $k\in K$ and the propagated cancellation costs from subsequent flights of $j\in L_k$, whenever $x^k_{(i,j)}=1$ and the aircraft does not stand at airport s in between the two flights. The single values $\omega^{s,k}_{i}$ are summed up over all fleet $k\in K$ in Constraint (3e).

First, we compare the newly set up model to the basic Tail Assignment Model (1). The following Lemma shows that the constraints do not further restrict the possible solution space of the flight strings. This means that all feasible solutions of Model (1) remain feasible in Model (3) since one is able to find values for the ω-variables, which yield a feasible solution of Model (3) combined with the nominal solution x^*, y^*, z^*.

Lemma 1. *Let x^*, y^*, z^* denote a feasible solution of Model* (1)*. Then, there exist ω^*, such that x^*, y^*, z^*, ω^* is a feasible solution of Problem* (3)*.*

Proof 1. *We show that the solution can be extended by explicitly giving a feasible solution for* (3) *and thus an assignment of the* ω*-variables. Like stated in Definition 1, the solution of Problem* (1) *decomposes into single flight strings for each aircraft denoted with* $L_{k^*}^{q^*}$. *W.l.o.g we assume* $L_{k^*}^{q^*} = \{l_1, l_2, ..., l_m\}$ *to be ordered regarding the time of execution from earliest flight to the last flight. With respect to Constraint* (3c) *we set*

$$\begin{aligned}
&\omega_{l_i}^{s,k} = 0 && && \forall i \in \{1,...,m\}, k \in K \setminus \{k^*\}\\
&\omega_{l_m}^{s,k^*} = c_{l_m} && && \\
&\omega_{l_i}^{s,k^*} = c_{l_i} && \text{if } s(l_i, l_{i+1}) = s && \forall i \in \{1,...,m-1\}\\
&\omega_{l_i}^{s,k^*} = c_{l_i} + \omega_{l_{i+1}}^{s,k^*} && \text{if } s(l_i, l_{i+1}) \neq s && \forall i \in \{1,...,m-1\}\\
&\omega_{(l_i,l_{i+1})}^{s,k^*} = \omega_{l_{i+1}}^{s,k^*} && && \forall i \in \{1,...,m-1\}\\
&\omega_{l_i}^{s} = \omega_{l_i}^{s,k^*} && && \forall i \in \{1,...,m\}
\end{aligned}$$

for all airports $s \in S$. *It is straight-forward to verify that this variable assignment fulfills all constraints of Model* (3) *and can be used for every single flight string.* □

Even though the solutions are not changed by the additional constraints, the assignment of certain additional ω-variables can be used to read how the objective function value behaves when solving the re-planning Problem (2) based on a given solution:

Theorem 1. *Let* x^*, y^*, z^*, ω^* *denote an optimal solution of Model* (3) *with an objective value* C_{rob}^*. *If all flights* $\tilde{L} \subseteq L$ *are canceled, the optimal objective value* C_{rep}^* *of Problem* (2) *yields an objective value for which following inequality holds:*

$$C_{rep}^* \leq C_{rob}^*$$

Proof 2. *We show that the* ω*-extension of Proof* (1) *fulfills the inequality. Afterwards we prove the optimality of the extension for given values of* x^*, y^*, z^*.

Regardless of the choice of ω*-variables, the solution of Problem* (3) *decomposes into single flight strings* $L_k^q = \{l_1, l_2, ..., l_m\}$ *for all* $q \in Q_k$ *and* $k \in K$. *Let* $\tilde{l}$ *be an element of* $\tilde{L} \cap L_{k^*}^q$:

- *Case 1* ($\tilde{l} = l_m$)*:*
  Since the last flight of the flight string has to be canceled in Problem (3), $\omega_{l_m}^{dep(l_m)^*} = \omega_{l_m}^{dep(l_m),k^*} = c_{l_m}$. *Regarding Problem* (2), *the objective value equals* c_{l_m}, *too, because* $l_m \in \tilde{L}$ *and thus* $z_{l_m} = 1$.
- *Case 2* ($\tilde{l} = l_i, i \in \{1, ..., m-1\}$)*:*
  Regarding Model (3) $\omega_{l_i}^{dep(l_i)^*} = \omega_{l_i}^{dep(l_i),k^*} = c_{l_i} + \omega_{l_{i+1}}^{dep(l_{i+1}),k^*} = ... = c_{l_i} + ... + c_{l_j}$ *holds. If* $l_j = l_m$, *then there exists no flight in* $\{l_i, ..., l_j\}$, *which departs from airport* s *and thus there exists no* $(l_{i-1}, l_h) \in A_k$ *with* $h \in \{i, ..., j\}$. *All flights* $\{l_i, ..., l_j\}$ *have to be canceled. If* $l_j \neq l_m$, *then* $s(l_j, l_{j+1}) = dep(l_i)$ *and thus* $(l_{i-1}, l_{j+1}) \in A_k \cap \{L_{k^*}^q \times L_{k^*}^q\}$. *All flights* $\{l_i, ..., l_j\}$ *have to be canceled here as well. Hence,* $z_i = ... = z_j = 1$ *in* (2) *and the objective increases by* $\sum_{l=1}^{j} c_l$, *which equals* $\omega_{\tilde{l}}^{dep(\tilde{l})^*}$. *If one does this for all* $l \in \tilde{L}$, *additional costs of at most* $\sum_{l \in \tilde{L}} \omega_l^{dep(l)^*}$ *arise.*

We have to show that $\sum_{l \in \tilde{L}} \omega_l^{dep(l)^*} = C_{rob}^*$. *Assume* $\sum_{l \in \tilde{L}} \omega_l^{dep(l)^*} \neq C_{rob}^*$. *Then, for an optimal solution there exists an* $l \in \tilde{L} \cap L_q^k (= \{l_1, l_2, ..., l_m\}$ *w.l.o.g.), such that* $\omega_l^{dep(l)} \leq \omega_l^{dep(l)^*}$ *and therefore a tuple* (k, l) *with* $\omega_l^{dep(l),k} \leq \omega_l^{dep(l),k^*}$ *exists. This is a contradiction, since with* $y_{l_i}^k = y_{l_i}^{k^*}$ *the inequality* $\omega_{l_i}^{dep(l_i),k} \geq c_{l_i} + \omega_{l_{i+1}}^{s,k^*} = \omega_{l_i}^{dep(l_i),k^*}$ *holds for* l_i *because of* (3d) *whenever* $s(l_i, l_{i+1}) \neq s$ *and* $\omega_{l_i}^{dep(l_i),k} \geq c_{l_i} = \omega_{l_i}^{dep(l_i),k^*}$ *whenever* $s(l_i, l_{i+1}) = s$. *All other values are implied by this variable assignments.* □

Nevertheless, the assignment of the ω-variables in Proof 1 is not unique. For example, the ω-variables of each flight per flight string can be increased by $M - \max_{l \in L} c_l$ with the solution remaining feasible. Therefore, the ω-variables are included in the objective function to keep them as small as possible, since they represent the cost of additional follow-up flight cancellations. In the following, it is stated under which conditions the solution of Problem (3) equals the solution of Model (2).

Theorem 2. *Assume the same notation as in Theorem 1. If $i \in L_k^q \Rightarrow j \notin L_k^q$ for all $i, j \in \tilde{L}$ with $i \neq j$, then the following equality holds:*

$$C^*_{rep} = C^*_{rob}$$

Proof 3. *Due to Theorem 1, it remains to show, that $C^*_{rep} \geq C^*_{rob}$ applies. Since flights have to be executed by the same aircraft in the replanning solution, the solution of Problem (3) decomposes into single flight strings $L_k^q = \{l_1, l_2, ..., l_m\}$. Since there is at most one flight in $L_k^q \cap \tilde{L}$ and no flight can be canceled in advance due to (2c), $l_1 \in \tilde{L} \cap L_{k^*}^q$ w.l.o.g. if at least one flight in $L_{k^*}^q$ is canceled:*

- *Case 1 ($m = 1$):*
  Since the last flight of the flight string has to be canceled in Problem (3), $\omega_{l_m}^{dep(l_m)^*} = \omega_{l_1}^{dep(l_1),k^*} = c_{l_1}$. Regarding Problem (2), the objective value equals c_{l_1}, too, because $l_1 \in \tilde{L}$ and thus $z_{l_1} = 1$.
- *Case 2 ($m > 1$):*
  Since the solution mentioned in Lemma 1 is optimal, $\omega_{l_1}^{dep(l_1)^*} = \omega_{l_1}^{dep(l_1),k^*} = c_{l_1} + \omega_{l_2}^{dep(l_2),k^*} = ... = c_{l_1} + ... + c_{l_j}$ holds for Problem (3). This value equals the cancellation costs of all flights in the flight sequence $L_{k^*}^q$ until one of the included flights land at the start airport of flight l_1. This also equals the objective value of Problem (2) regarding flight string $L_{k^*}^q$, since the next flight in $L_{k^*}^q$ is the first flight (with respect to the original order in the flight string), which can be executed by the repective aircraft q in k^*.

Next, it will be shown how Model (3) can be embedded in a procedure in order to safeguard solutions of the aircraft assignment against flight failures. Model (3) only minimizes the cost of flight cancellations. Solving the model in isolation would lead to arbitrarily high assignment costs, since these costs are neglected in the objective function. Even if the actual assignment costs of the nominal problem were included in the objective function of Model (3), these costs would be relatively small compared to the cancellation costs of entire flights and would thus hardly influence the solution. This can be circumvented with the use of Procedure 1, which is based on the idea of *light robustness* introduced in (12): Instead of minimizing the objective of the nominal model along with the cost of flight cancellations in one model, one solves the nominal optimization Problem (1) first. Then, the safeguarding Model (3) is solved where the nominal objective function is bounded using an additional constraint. The bound is set depending on the nominal optimal objective function value. Procedure 1 requires all information about Model (1) and Model (3) as well as the maximal allowed factor $\delta \in \mathbb{R}_{\geq 0}$ which indicates how much the robust solution costs may deviate from the nominal solution costs. In Line 4 the nominal tail assignment model is solved to obtain the optimal objective value z_{nom}. We use this information to bound the maximal nominal objective term by adding a constraint to Model (3) in Line 5. This model is solved afterwards and produces a feasible tail assignment, which costs at most $\delta \cdot z_{\text{nom}}$ monetary units more than the optimal assignment and costs of flight cancellations in $\tilde{L}$ do not exceed $z_{\text{rob}} = \sum_{j \in \tilde{L}} \omega_j^{\text{dep}(j)^*}$. We have shown that modeling methods can also handle uncertainties that affect the problem structurally and influence multiple constraints. The robustness of the model was ensured by modeling new constraints

Algorithm 1 Light Robustness Cancellation Algorithm

1: **INPUT**: Model (1), Model (3), factor bounding maximal costs of robust solution δ
2: **OUTPUT**: Robust Tail Assignment *sol* which at most costs $\delta \cdot z_{\text{nom}}$ more than the optimal nominal assignment and produces at most z_{rob} extra costs in case all flights $\tilde{L}$ were canceled
3: **procedure** LIGHT_ROBUSTNESS((1), (3), δ)
4: $z_{\text{nom}} \leftarrow$ Solve Model (1)
5: Add Constraint (1a) $\leq z_{\text{nom}}(1+\delta)$ to Model (3)
6: $z_{\text{rob}}, sol \leftarrow$ Solve Model (3)
7: **return** (sol)

and variables which represent the cancellation costs of subsequently canceled flights. However, since these costs are much higher than the usual operating costs that have to be optimized, the modeling technique was embedded into a procedure to keep the costs for the robust solution compared to the non-safeguarding solution within reasonable limits. Nevertheless, this is still one-stage robustness: The initial plan is adjusted to generate only small additional costs in case of uncertainties - but without re-planning measures calculated in a second stage as a consequence of the occurrence of the uncertainty. Since the possibility of re-planning is used by airlines in many cases, we will deal with aircraft failures that lead to changes in aircraft routing in the next section. In contrast to the established approaches in the literature, we do not rely on a classical scenario-expanded model, but derive a compact model: Using this formulation, the number of variables in the robust model do not anymore depend on the number of considered scenarios. We prove that this model is equivalent to a scenario expansion under certain conditions.

4. Robust Aircraft Routing with Replanning: Aircraft Failures

In this section we not only deal with the cancellation of individual flights, but also with the failure of entire aircraft in the course of the schedule. For this purpose another model is introduced, which is designed to perform the task of fleet assignment. Afterwards this model is modified to be capable of tracking flight cancellation costs in the case that one aircraft is not available at one point in time of the plan. In contrast to Section 3 it is capable of implicitly calculating the best possible tail-swap combination to compensate the missing aircraft, which means the costs are calculated while incorporating possible re-planning decisions.

4.1. **Fleet Assignment.** The classic fleet assignment model was developed to reduce the complexity of aircraft-flight assignment by first deciding exclusively which aircraft types are used for which part of the flight plan. In the case where multiple aircraft of the same type exist, solving this problem does not necessarily lead to a unique assignment of flights to individual aircraft.

The model presented here is based on a set of directed graphs, which are called time-space graphs, one for each aircraft type. All sets, coefficients and variables needed for the model are listed in Table 3. The corresponding acyclic time-space graph is depicted in Figure 2.

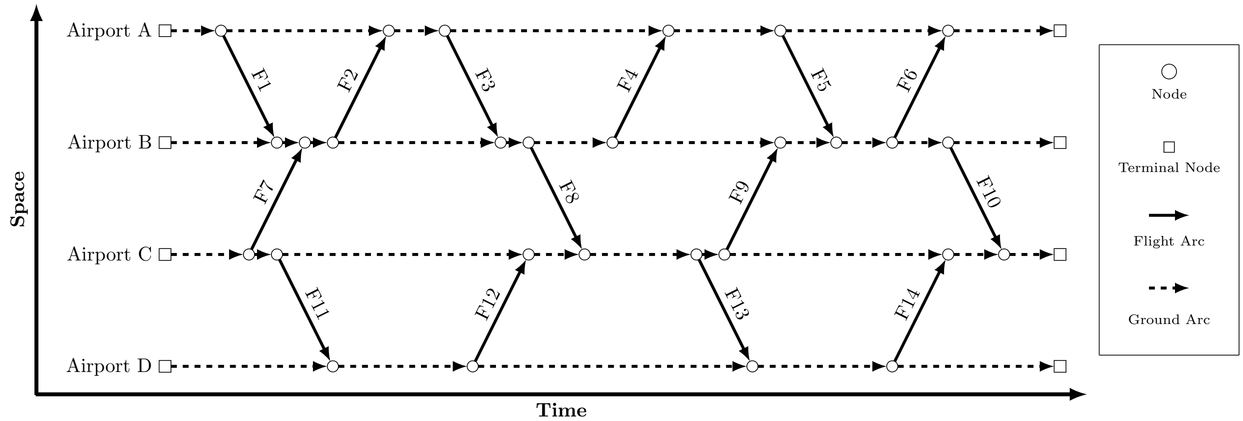

FIGURE 2. Acyclic time-space graph for one fleet $k \in K$ with four airports and 14 flight arcs

The set of fleets is described by K, whereas the set of flights is referred to as F. Each fleet $k \in K$ consists of $|Q_k|$ single aircraft and is associated with one directed time-space graph. Taken by itself, the graph of k consists of nodes N_k and arcs E_k, which can be further subdivided into ground arcs and flight arcs. All nodes in the graph can be uniquely assigned to an airport. Each arc connecting two nodes of the same airport is called ground arc, otherwise flight arc. The set of flight arcs is denoted as F_k. Nodes which are connected by a flight arc correspond to take-off events at the origin airport or landing events at the destination airport. The ground arcs of one and the same airport drawn in Figure 2 represent the time axis at this airport. The nodes for each flight arc are inserted according to the departure or arrival time of the flight at the respective airport. All other nodes do not have both incoming and outgoing arcs. Thus, these nodes are called terminal nodes and are part of the set T_k. Arcs starting from a terminal node are called start arcs and are part of the set S_k, whereas all arcs ending at a terminal node are called end arcs and are included in the set H_k.

In order to set up the objective of the model, one also needs coefficients that quantify the costs of an assignment. $c^k_{(i,j)}$ describes the cost of an assignment of aircraft type $k \in K$ to execute a flight $(i,j) \in F_k$, which is characterized by its takeoff node $i \in N_k$ and landing node $j \in N_k$ in the graph belonging to k.

There are three sets of variables in Model (5): $x^k_{(i,j)}$ denotes a binary variable which equals 1 whenever flight $(i,j) \in F_k$ is executed by fleet $k \in K$ and 0 otherwise. $y^k_{(i,j)}$ is a continuous variable that represents the number of aircraft of type k on a given ground arc $(i,j) \in E_k \setminus F_k$. Due to the structure of the graph and the integrality of the x-variables, integrality does not have to be explicitly required for the y-variables in the following model. The z-variables do not depend on the fleet k, since they indicate whether a flight is canceled or not. $z_{(i,j)}$ equals 1 if the corresponding flight (i,j) is canceled, otherwise 0.

The goal of the model is to generate an integer flow on each of the individual graphs such that, across all graphs, each flight edge is passed through by at most one aircraft type. Based on the introduced sets,

Set	Description
K	set of fleets
F	set of flights
$K_{(i,j)}$	set of fleets capable of executing flight $(i,j) \in F$
Q_k	set of available aircraft per fleet $k \in K$
N_k	set of nodes in the graph of $k \in K$
E_k	set of arcs in the graph of $k \in K$
$T_k \subseteq N_k$	set of terminal nodes in the graph of $k \in K$
$F_k \subseteq E_k$	set of flight arcs executable by fleet $k \in K$
$S_k \subseteq E_k$	set of start arcs in the graph of $k \in K$
$H_k \subseteq E_k$	set of end arcs in the graph of $k \in K$
Parameter	**Description**
$c^k_{(i,j)}$	costs of assigning fleet $k \in K$ to flight leg $(i,j) \in F_k \subset E_k$
$c_{(i,j)}$	costs in the case that flight $(i,j) \in F$ is canceled
Variable	**Description**
$x^k_{(i,j)}$	binary flight arc flow variable indicating whether flight $(i,j) \in F_k$ is assigned to fleet $k \in K$
$y^k_{(i,j)}$	continuous ground arc flow variable representing the number of aircraft of fleet $k \in K$ on ground during $(i,j) \in E_k \setminus F_k$
$z_{(i,j)}$	binary variable indicating whether flight (i,j) is canceled

Table 3. Sets, parameters and variables for the fleet assignment model based on the underlying acyclic directed time-space graphs.

coefficients, and variables, the fleet assignment model can be written as follows:

$$\text{(5a)} \qquad \min \sum_{k \in K} \sum_{(i,j) \in F_k} c^k_{(i,j)} x^k_{(i,j)} + \sum_{(i,j) \in F} c_{(i,j)} z_{(i,j)}$$

$$\text{(5b)} \qquad \text{s.t.} \quad \sum_{k \in K_{(i,j)}} x^k_{(i,j)} + z_{(i,j)} = 1 \qquad \forall (i,j) \in F$$

$$\text{(5c)} \qquad \sum_{i:(i,j) \in F_k} x^k_{(i,j)} + \sum_{i:(i,j) \in E_k \setminus F_k} y^k_{(i,j)} - \sum_{l:(j,l) \in F_k} x^k_{(j,l)} - \sum_{l:(j,l) \in E_k \setminus F_k} y^k_{(j,l)} \geq 0 \qquad \forall j \in N_k \setminus T_k, k \in K$$

$$\text{(5d)} \qquad \sum_{(i,j) \in S_k} y^k_{(i,j)} \leq |Q_k| \qquad \forall k \in K$$

$$\text{(5e)} \qquad x, z \quad \text{binary}$$

$$\text{(5f)} \qquad y \in \mathbb{R}^+_0$$

The objective term (5a) minimizes the assignment costs over all fleet-flight combinations and the cancellation costs. Constraint (5b) is called *cover constraint* and ensures that each flight is executed by exactly one fleet ($\sum_{k \in K_{(i,j)}} x^k_{(i,j)} = 1$) if it is not canceled ($z_{(i,j)} = 1$). It should be noted that this is the only sort of constraint that connects the individual graphs per fleet. For every node, which is not a terminal node, Constraint (5c) makes sure that the sum of all variables on ingoing arcs is greater than or equal to the sum of all variables on outgoing arcs. If the in ingoing and outgoing flow deviate, this means, that the missing aircraft represented by the outgoing flow is not needed anymore for the execution of flights. Combined with the integrality constraints for the binary variables Constraint (5c) implies, that all y-variables take integral values. To guarantee that the number of used aircraft does not exceed the number of available aircraft per fleet k, (5d) bounds the sum of variables on the start arcs of each graph by $|Q_k|$ from above. The initial airport of a single aircraft is implicitly determined by the possible connections included in the set S_k.

4.2. **Fleet Assignment: Extension.**

In this section we deal aircraft failures instead of flight cancellations. Aircraft failures occur when technical difficulties arise or errors are identified during maintenance events. In such cases, it can be assumed that the aircraft will need to be thoroughly inspected and repaired by specialist personnel, which can take a long time. Thus, in this case, it is unrealistic to assume that the airline will keep up the further tail assignment without re-planning.

Such re-planning can be done in two possible ways: First, aircraft of the same fleet can be interchanged. This procedure is called *tail swap* and is relatively easy to implement because the aircraft can carry the same number of passengers. Furthermore, the corresponding assigned crew does not necessarily have to be changed in order to assign a crew qualified for another aircraft type to the flight. The second option consists of swapping aircraft from different fleets, which is called *equipment change*. This is usually very expensive and extensive, since in addition to deviating seat capacity and crew rescheduling, time changes in ground processes and costs in using other aircraft for other flights must also be factored in.

For these reasons, we want to focus on *tail swaps* here. Therefore, we modify the fleet assignment problem in such a way that possible aircraft failures can be well compensated by means of *tail swaps*. Compensation here also means, as in Section 3, that one wants to avoid additional costs due to resulting flight cancellations. Thus, unlike the previous sections, we focus on Fleet Assignment Model (5), and include only the relevant information about consequences for tail assignment.

It is not enough to hedge against the situation where a reduced number of aircraft is available only. In this case, the aircraft that potentially fail are no longer considered in the solution at all. The actual goal, however, is to use the aircraft and schedule them in such a way that a failure at any time or location results in comparably low costs. Thus, before the uncertainty is realized, one wants to make decisions that favor the reoptimization in the case of an aircraft being unable to take off at some point.

For this purpose, both robust and stochastic optimization use the concept of multistage optimization models, which can be modeled by means of scenario expansion. In this, the planning phase in the form of nominal Model (5) is combined with sub-problems for each individual scenario involving a combination of aircraft failures at predefined connections. Each subproblem itself is in turn a fleet assignment model that starts from the time of scenario occurrence and operates with the reduced number of aircraft. The scenario expanded model is composed as follows: The starting points of all scenario subproblems must coincide with the respective end states of the planning phase, and the objective functions of the subproblems are incorporated into the overall objective function weighted by the probability of scenario occurrence. Note that new scenario variables must be added for each scenario. Thus, the number of variables in the scenario-expanded model increases with each additional scenario almost by the number of variables in the nominal model, which leads to the fact that it is difficult to solve.

In the following, we will present a compact formulation of the scenario-expanded model. Compact here means that the number of variables does not depend on the number of scenarios. We will also show that under certain circumstances the formulation we have established behaves equivalently to the scenario-expanded model.

Therefore, we focus on a modeling technique that is capable of modeling cancellation costs properly and that has a fixed number of variables independent of the number of scenarios in the case that the scenarios fulfill certain conditions. To do this, we will first look at two examples that illustrate how the aircraft failure costs can be captured whenever an aircraft can not be used any more.

Example 1. *We consider a flight schedule as shown in Figure 3, in which five flights are executed by the same fleet. Due to the timing of the flights, two aircraft are required for this, as can be seen from the assignment of the y-flow variable in the figure. At each airport there must be exactly one aircraft at the beginning of the flight plan. The cost of a flight cancellation for flight* F_1*-*F_4 *is* 10.000€ *each while the cancellation of flight* F_5 *costs only* 9.000€*. Assume that at the very beginning of the plan, the aircraft located at Airport A is out of service for the rest of the day. Now we want to calculate the resulting costs, neglecting the assignment costs. There are two possible routes that the failed aircraft can choose in the nominal plan: Either* F_1*,* F_3 *and* F_4*, or* F_1*,* F_3 *and* F_5*. In this example, the minimal costs for the aircraft failure consist of the potential cheapest aircraft route regarding cancellation costs of the flights included, and thus equal* 29.000€ *(*F_1*,* F_3*,* F_5*).*

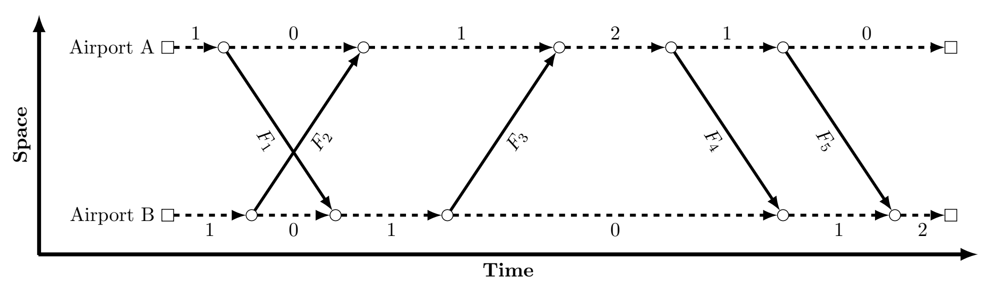

FIGURE 3. Time-Space graph for one fleet drawn with arc labels that denote the y-variable values of a feasible assignment. The values represent the number of aircraft on ground on each ground arc.

Using Example 1, it can be seen that the costs can be calculated by following the possible paths of the potentially failing aircraft, which results from the flow at the ground arcs and over the individual flight arcs. If one finds a path that minimizes the sum of the cancellation costs of all included flights, one has already found an upper bound for the potential costs of the failure. However, it is also possible to use other aircraft and thus other flight strings. This can be illustrated by the following Example 2.

Example 2. *The initial situation is exactly the same as in Example 1. There is only one difference: The failure of flight* F_2 *now costs only* 8.000€*. This does not change the calculation method from Example 1. However, the aircraft failure now costs less: Since the cancellation of* F_2 *is comparatively cheap, the optimal solution results when the unaffected aircraft (initially located at Airport B) is no longer executing flight* F_2 *and instead is assigned to* F_3 *and* F_4*. The resulting costs are* 27.000€ *and can no longer be calculated based on the potential flight strings of the aircraft that is located at Airport A.*

The slightly modified Example 2 shows that the solution by tail swaps can also be more complicated: In order to determine the minimum costs for the outage at Airport A, it had to be taken into account that flights

executed by another aircraft may be canceled in order to make the aircraft available for the flight string of the failing aircraft. This is equivalent to cancelling a flight that cannot be on the flight string of the failing aircraft. This implicitly changes the flow at the ground arcs and can thus not be detected by simply looking at the existing ground flow at the ground arcs.
However, one can still map a path through the time-space graph that reproduces the costs from Example 2. This path is illustrated in Figure 4.

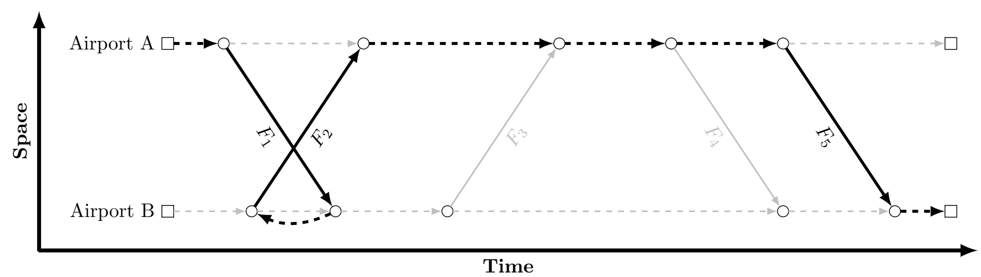

FIGURE 4. Path in the time-space graph of Example 2, which includes all flights which have to be canceled in order to receive the minimal cancellation costs when the aircraft at airport A may not be used.

The path, however, uses a ground arc on which there is actually no flow and includes it against its original direction. Going backwards on the path can be explained as follows: Whenever another aircraft does not execute flights on its own flight string, the flow at the subsequent ground arc after the scheduled start of this flight effectively increases. This means that the number of aircraft available on ground effectivly increases by one. The additional aircraft can be used to replace other missing aicraft at this airport. At the same time, the parts of the original flight string of this aircraft have to be canceled in turn.
We now extend the existing Model (5) to track the cost of this path and propagate it to the original ground arc on which an aircraft may fail. In order to model this behavior we need additional sets, parameters and variables compared to Model (5). These are described in Table 4. The set Σ describes the scenario set. An element contained in it is a scenario σ, which in turn represents a set of tuples (k, i, j). This tuple describes a failure of an aircraft of type $k \in K$ on the ground arc $(i, j) \in E_k \setminus (F_k \cup S_k \cup H_k)$ of the correponding time-space graph. The parameter p_σ describes the probability of occurrence of scenario σ. In order to be able to implement the described tracking of costs in a model, one has to distinguish the set of ground arcs more granularly: Due to the possible events corresponding to nodes before and after a ground arc, which can be either a takeoff or a landing of an aircraft, there are four cases, which can be seen in Figures 5-8.

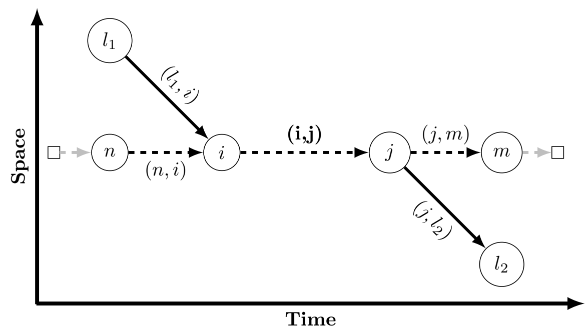

FIGURE 5. Ground arc in the set AD between arrival and departure

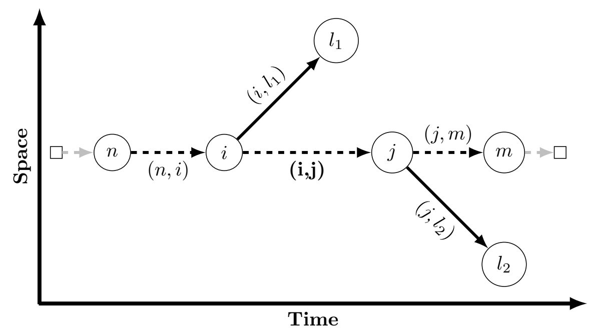

FIGURE 6. Ground arc in the set DD between two departures

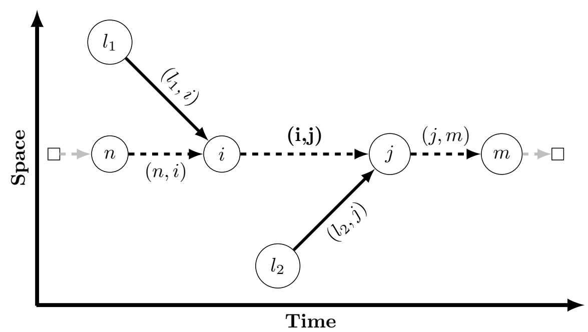

FIGURE 7. Ground arc in the set AA between two arrivals

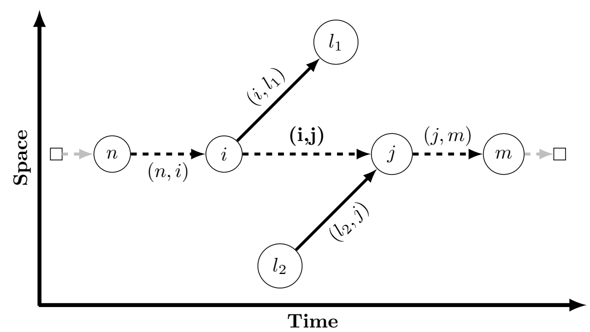

FIGURE 8. Ground arc in the set DA between departure and arrival

Depending on which case applies to the arc of the time-space graph, we divide all ground arcs into the sets AD, DD, AA and DA. It should be noted that the nodes in the time-space graph can always be partitioned by separating simultaneous takeoff and landing events so that four cases can be distinguished. For all ground and flights arcs in the graph, an additional binary variable $u^k_{(i,j)}$ and a continuous variable $v^k_{(i,j)}$ are needed. The value of $v^k_{(i,j)}$ represents the cumulated cancellation costs, which arise if an aircraft of fleet k fails during the time interval and on the airport which correspond to arc (i, j). The binary variable $u^k_{(i,j)}$ indicates from which adjacent arc the value $v^k_{(i,j)}$ is inherited. Since every node in the graph has at most degree three and the path may not go twice over an arc one binary is sufficient to decide which arc will be the next one contained in the path. The path is thus uniquely encoded by the values of the u-variables. Based on all values of u, it is thus possible to specify the path on which the flights responsible for the cost v of an arc are located. For the modeling of the paths that occur in Example 2, one needs for each ground arc an additional set of binary and continuous variables $u^{\text{for},k}_{(i,j)}$ and $v^{\text{for},k}_{(i,j)}$, which serve the same purpose, but the costs are inherited forward in time against the actual direction of the ground arcs in the time-space graph. For all cost variables $v^k_{(i,j)}$ and $v^{\text{for},k}_{(i,j)}$ one can construct lower bounds $\underline{v}^k_{(i,j)}, \underline{v}^{\text{for},k}_{(i,j)}$ and upper bounds $\overline{v}^k_{(i,j)}, \overline{v}^{\text{for},k}_{(i,j)}$ easily by estimating the longest possible flight sequence. These bounds can be used to linearize the model. The interval R includes all points in time in the given schedule after the realization of the scenario is revealed. To identify whether

Set	Description
Σ	set of aircraft failure scenarios
σ	aircraft failure scenario consisting of a set of tuples (k,i,j), which denotes that one aircraft of fleet $k \in K$ will break during ground arc $(i,j) \in A_k$
$AD \subseteq E_k$	set of ground arcs connecting an arrival node with a departure node
$DD \subseteq E_k$	set of ground arcs connecting two departure nodes
$AA \subseteq E_k$	set of ground arcs connecting two arrival nodes
$DA \subseteq E_k$	set of ground arcs connecting a departure node with an arrival node
R	reaction time interval starting when the occuring scenario is known and ending with the last landing event of the schedule
Parameter	**Description**
p_σ	propability that scenario σ occurs
$\underline{v}^k_{(i,j)}, \overline{v}^k_{(i,j)}$	lower, upper bound for $v^k_{(i,j)}$
$\underline{v}^{\text{for},k}_{(i,j)}, \overline{v}^{\text{for},k}_{(i,j)}$	lower, upper bound for $v^{\text{for},k}_{(i,j)}$
$t(i)$	event time of event i
Variable	**Description**
$u^k_{(i,j)}$	binary variable of fleet $k \in K$ on the arc $(i,j) \in E_k$, which indicates from which subsequent arc costs are inherited
$v^k_{(i,j)}$	continuous variable of fleet $k \in K$ on the arc $(i,j) \in E_k$, which represents cumulated cancellation costs (backward)
$u^{\text{for},k}_{(i,j)}$	binary variable of fleet $k \in K$ on the arc $(i,j) \in E_k \setminus F_k$, which, indicates depending on the arc type from which arc costs are inherited
$v^{\text{for},k}_{(i,j)}$	continuous variable of fleet $k \in K$ on the arc $(i,j) \in E_k \setminus F_k$, which represents cumulated cancellation costs (forward)

Table 4. Sets, parameters and variables for the robust fleet assignment model regarding aircraft failures

variables connected to events can be adjusted according to R, $t(i)$ corresponds to the point in time when an event (take-off, landing) takes place. We assume that the fleet assignment is calculated some weeks in advance and the uncertainty for the day is revealed at the night before the day of operations. Therefore only adjustments after this point are allowed: Using Example 2 for illustration, one may not go back in time further on the drawn path, since one assumes that all flights, which depart before included in this path would have to be canceled to compensate the scenario which is not known at this point in time. For the sake of simplicity and to keep the model readable, all indices in Model 6 correspond exactly to the indices used in

Figure 5-8, which lead to the following model:

$$
\begin{aligned}
&\text{(6a)} && \min \text{(5a)} + \textstyle\sum_{\sigma\in\Sigma} p_\sigma \sum_{(k,i,j)\in\sigma} v^k_{(i,j)} && \\
& && \text{s.t. (5b)} - \text{(5f)} && \\
&\text{(6b)} && v^k_{(i,j)} \geq v^k_{(j,m)} u^k_{(i,j)} + v^{\text{for},k}_{(n,j)}(1-u^k_{(i,j)}) + (c_{(i,j)} - c^k_{(i,j)}) && \forall k \in K, (i,j) \in F_k \\
&\text{(6c)} && v^k_{(i,j)} \geq v^k_{(j,l_2)} u^k_{(i,j)} + v^k_{(j,m)}(1-u^k_{(i,j)}) && \forall k \in K, (i,j) \in AD_k, DD_k \\
&\text{(6d)} && u^k_{(i,j)} \geq x^k_{(j,l_2)} - y^k_{(j,m)} && \forall k \in K, (i,j) \in AD_k, DD_k \\
&\text{(6e)} && u^k_{(i,j)} \leq x^k_{(j,l_2)} && \forall k \in K, (i,j) \in AD_k, DD_k \\
&\text{(6f)} && v^k_{(i,j)} = v^k_{(j,m)} && \forall k \in K, (i,j) \in AA_k, AD_k \\
&\text{(6g)} && v^{\text{for},k}_{(i,j)} \geq v^{\text{for},k}_{(n,i)}(1-u^{\text{for},k}_{(i,j)}) + v^k_{(i,j)} u^{\text{for},k}_{(i,j)} && \forall k \in K, (i,j) \in AD_k, AA_k \\
&\text{(6h)} && u^{\text{for},k}_{(i,j)} \leq y^k_{(i,j)} && \forall k \in K, (i,j) \in AD_k, AA_k \\
&\text{(6i)} && v^{\text{for},k}_{(i,j)} \geq v^{\text{for},k}_{(n,i)}(1-u^{\text{for},k}_{(i,j)}) + v^k_{(i,l_1)} u^{\text{for},k}_{(i,j)} && \forall k \in K, (i,j) \in DD_k, AD_k \\
&\text{(6j)} && u^{\text{for},k}_{(i,j)} \leq y^k_{(n,i)} && \forall k \in K, (i,j) \in DD_k, AD_k \\
&\text{(6k)} && u^k_{(i,j)} \in \{0,1\} && \forall k \in K, (i,j) \in E_k \\
&\text{(6l)} && v^k_{(i,j)} \in [\underline{v}^k_{(i,j)}, \overline{v}^k_{(i,j)}] \subset \mathbb{R}^+_0 && \forall k \in K, (i,j) \in E_k \\
&\text{(6m)} && u^{\text{for},k}_{(i,j)} \in \{0, \mathbb{1}_{t(i)\in R}\} && \forall k \in K, (i,j) \in E_k \setminus F_k \\
&\text{(6n)} && v^{\text{for},k}_{(i,j)} \in [\underline{v}^{\text{for},k}_{(i,j)}, \overline{v}^{\text{for},k}_{(i,j)}] \subset \mathbb{R}^+_0 && \forall k \in K, (i,j) \in E_k \setminus F_k
\end{aligned}
$$

The objective (6a) minimizes the nominal and the scenario-related costs. The nominal costs include the costs of assigning certain aircraft types to flights and the costs of scheduled flight cancellations. Each scenario σ is weighted depending on its probability of occurrence p_σ and takes into account the costs that arise from flight cancellations caused by tail-swaps. These costs correspond to the cancellation costs $c_{(i,j)}$ reduced by the assignment costs $c^k_{(i,j)}$. We assume that executing a flight is always cheaper than cancelling it and thus $c_{(i,j)} > c^k_{(i,j)}\ \forall (i,j) \in F_k, k \in K$.

To construct the path described in examples 1 and 2, each arc has to have exactly one subsequent arc on the path. Since all nodes in the graph have at most degree three, one binary u variable per arc is sufficient to decide which arc comes next on the path. The v-variables represent the aggregated cancellation costs of all subsequent flights which are included in this path.

Constraint (6b) propagates the v-values along a flight arc $(i,j) \in F_k$. The value of $v^k_{(i,j)}$ consists of the costs of canceling of the respective flight (i,j) (cancellation minus assignment costs) and additionally the v-value on the next arc of the path. This is illustrated in Figure 9: If $u^k_{(i,j)} = 1$, $v^k_{(i,j)}$ is raised by the value $v^k_{(j,\cdot)}$ of the subsequent ground arc $(j,\cdot)$, which corresponds to the right red arrow. Otherwise, if $u^k_{(i,j)} = 0$, $v^k_{(i,j)}$ is raised by the v^{for}-value of the ground arc $(\cdot,j)$, which ends at the arrival node j, which is illustrated by the left red arrow. The corresponding path follows exactly the opposite direction of the cost flow. This same mechanism applies for the Constraints 6c, 6g, 6i, which is why we omit to describe it in detail for each constraint.

The following constraints (6c)-(6f) propagate the cancellation costs backwards in time according to Example 1: Constraint (6c) ensures, that the v-variable on each ground arc, which is followed by a departure, correspond to the minimal v-value of both outgoing arcs, which is sketched in Figure 10. Since one outgoing arc is a flight arc, Constraint (6e) makes sure, that the value can only be propagated from the flight if the flight is executed by the corresponding fleet. Constraint (6d) does the same for the outgoing ground arc and Constraint (6f) ensures that the value of $v^k_{(i,j)}$ equals the v-value on the outgoing arc, whenever the connecting node is an arrival node.

All constraints (6g)-(6j) define how v-values and therefore cancellation costs can propagated forward in the time-space graph, which is necessary to cover the case of Example 2. For all arcs which start with a landing event, Constraints (6g) and (6h) guarantee, that $v^{\text{for},k}_{(i,j)}$ corresponds to the forwarded costs on the previous arc or to the value of $v^k_{(i,j)}$ on the same arc. The value on the same arc can only be chosen in the case that $y^k_{(i,j)} \neq 0$. Constraint (6g) does not influence the path, since the next arc on the path is the previous ground arc independent of the value of $u^{\text{for},k}_{(i,j)}$. This behavior is depicted graphically in Figure 11. Constraints (6i) and (6j) are equivalent for ground arcs which start with a departure node. The difference is, that $v^{\text{for},k}_{(i,j)}$ can either take the value of $v^{\text{for},k}_{(n,i)}$ from the corresponding variables on the ingoing ground arc or the value of $v^k_{(i,l_1)}$ on the flight arc which departs from i. This cost propagation is sketched in 12.

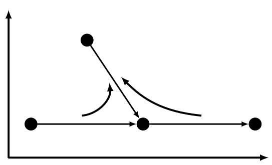

Figure 9. (6b)

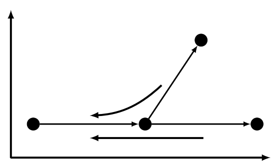

Figure 10. (6c)

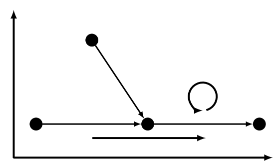

Figure 11. (6g)

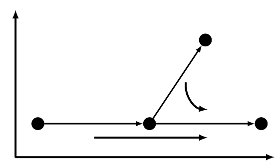

Figure 12. (6i)

Constraints (6k)-(6n) establish the different variable spaces whereby all v-variables can be bounded: $v^{\text{for},k}_{(i,j)}$ has to be bounded for all $(i,j) \in S_k$ from below by a number bigger than the highest possible costs, which may arise if one aircraft of type k fails to execute a flight string. This number can be estimated by the biggest possible number of flights one aircraft can possibly serve and the most expensive cancellations of this aircraft. All other $v^{\text{for},k}_{(i,j)}$ $\forall (i,j) \in E_k \setminus S_k$ can be bounded by the minimal cancellation cost of a flight, when it is executed by fleet k. All variables $v^k_{(i,j)}$ with $(i,j) \in H_k$ can be bounded by 0 from above (and thus be set to 0), whereas one can find a specific upper bound for all remaining $(i,j) \in E_k \setminus H_k$ by approximating the number of possible flights which can be executed after the end of ground arc (i,j) and the maximal cancellation costs of one flight executed by fleet k. All variables $u^{\text{for},k}_{(i,j)}$ are bounded from above by 0 if the event i at the point in time $t(i)$ takes place before the ocurring scenario is known. Thus no flights can be canceled in advance to stand by for potential tail swaps, which is indicated by $u^{\text{for},k}_{(i,j)} = 1$ in Constraint (6i).

Keep in mind that Model (6) is not linear but can be linearized using the same technique as in Constraint (3b) and (3c). It can be applied to linearize the product of continuous and binary variables by using information about the upper and lower bounds of the continuous variable. In order not to go beyond the scope of the paper and since the linearization can be done straight-forward, it has been omitted here to elaborate it in more detail.

Instead, we now want to prove that the costs that occur in the case of reoptimization are accurately represented by the v-variables of the model. Assume x^*, y^*, z^* to be an feasible solution of Model (5) and let σ_k be the subset of tuples in σ which includes fleet $k \in K$. The calculation of the costs of a single scenario can be done by adjusting the nominal Model (5) as follows:

Model (7) $(\sigma_k, (x^*, y^*, z^*))$:

$$\text{(7a)} \quad \min \ 5a$$

$$\text{(7b)} \quad \text{s.t.} \quad 5b - 5f$$

$$\text{(7c)} \quad y^k_{(i^*,j^*)} = y^{k\,*}_{(i^*,j^*)} - 1 \qquad \forall (i^*, j^*) \in \sigma_k$$

$$\text{(7d)} \quad x^k_{(i,j)} \mathbb{1}_{(t(i)\notin R)} = x^{k\,*}_{(i,j)} \mathbb{1}_{(t(i)\notin R)} \qquad \forall (i,j) \in A_k, k \in K$$

$$\text{(7e)} \quad x^k_{(i,j)} \leq x^{k\,*}_{(i,j)} \qquad \forall (i,j) \in A_k, k \in K$$

Based on the solution of Model (5) the scenario dependent ground arcs are reduced by an aircraft in (7c), all flights before the time interval R may not be changed or canceled (7d) and all flights taking place within R have to be executed by the same fleet, which is stated in (7e). We now present Theorem 3, that states under which conditions Model (6) accurately represents the cancellation costs and is thus equivalent to a scenario-expanded model.

Theorem 3. *Let σ_k be the subset of tuples in σ which includes fleet $k \in K$ and x^*, y^*, z^* be a feasible solution of Model (5). Let furthermore x^{**}, y^{**}, z^{**} be an optimal solution of Model (7)$(\sigma_k, (x^*, y^*, z^*))$. If $|\sigma_k| \leq 1 \ \forall k \in K$, i.e., $\sigma_k = \emptyset$ or $\sigma_k = \{(i^*, j^*)\}$ for an occuring scenario σ', then values u^*, v^* exist which minimize the objective of Model (6) when x, y, z equal x^*, y^*, z^* and the following equality holds:*

$$\sum_{k\in K} \sum_{(i,j)\in F_k} c^k_{(i,j)} x^{k\,*}_{(i,j)} + \sum_{(i,j)\in F} c_{(i,j)} z^*_{(i,j)} + \sum_{(k,i,j)\in\sigma'} v^{k\,*}_{(i,j)} = \sum_{k\in K} \sum_{(i,j)\in F_k} c^k_{(i,j)} x^{k\,**}_{(i,j)} + \sum_{(i,j)\in F} c_{(i,j)} z^{**}_{(i,j)}$$

Proof 4. *Since the solution approach decomposes the problem into individual graphs per fleet and only tail-swaps can be made, it is sufficient to estimate the costs separately for each fleet. We neglect the costs of a priori flight cancellations since Constraint* (7e) *implies that all flights, which are canceled in* (6) *are also canceled in Model* (7) *and thus it is equivalent to show*

$$\sum_{(i,j)\in F_k} c^k_{(i,j)} x^{k\,*}_{(i,j)} + v^{k\,*}_{(i^*,j^*)} = \sum_{(i,j)\in F_k} c^k_{(i,j)} x^{k\,**}_{(i,j)} + \sum_{(i,j)\in F_k} c_{(i,j)} z^{**}_{(i,j)}.$$

The proof is conducted in two steps. First, we prove that based on the solution of Model (7) *one is able to construct an equivalent solution in Model* (6)*, for which the equality holds. In the second step, we prove this solution of Model* (6) *is optimal.*
The solution x^{**}, y^{**}, z^{**} can be characterized by the additional z-variables, which have to be set to 1 *to ensure that considering* (7c) *the flow conservation constraint* (5c) *remains feasible: Therefore one has to look at all nodes, where the flow constraint is violated and has to correct the flow on the subsequent arcs. Starting with arc $(i, j) = (i^*, j^*)$ and an arc list $\overline{A}_k = [(i, j)]$ one can distinguish four cases, whereby each arc on which flow variables were adjusted is added to the arc list:*

(1) The flow into a departure node is decreased
  (7)*: $x^{k\,**}_{(j,\cdot)} = 0 \Leftrightarrow z^{k\,**}_{(j,\cdot)} = 1$. If $z^{k\,**}_{(j,\cdot)} = 1$ objective increases by $c_{(j,\cdot)} - c^k_{(j,\cdot)}$.*
  (6)*: $x^{k\,**}_{(j,\cdot)} = 0 \Leftrightarrow u^{k\,*}_{(i,j)} = 1$. If $u^{k\,*}_{(i,j)} = 1$, $v^{k\,*}_{(h,l)}$ increases by $c_{(j,\cdot)} - c^k_{(j,\cdot)} \forall (h,l) \in \overline{A}_k$.*

(2) The flow into a departure node is increased
  (7)*: $x^{k\,**}_{(i,\cdot)} = 0 \Leftrightarrow z^{k\,**}_{(i,\cdot)} = 1$. If $z^{k\,**}_{(i,\cdot)} = 1$ objective increases by $c_{(i,\cdot)} - c^k_{(i,\cdot)}$.*
  (6)*: $x^{k\,**}_{(i,\cdot)} = 0 \Leftrightarrow u^{for,k\,*}_{(i,j)} = 1$. If $u^{for,k\,*}_{(i,j)} = 1$, $v^{k\,*}_{(h,l)}$ increases by $c_{(j,\cdot)} - c^k_{(j,\cdot)} \forall (h,l) \in \overline{A}_k$.*

(3) The flow into an arrival node is decreased

(7)*: If* $(i,j) \in F_k$*:* $y^{k\,**}_{(\cdot,j)} = y^{k\,*}_{(\cdot,j)} + 1$ *or* $y^{k\,**}_{(j,\cdot)} = y^{k\,*}_{(j,\cdot)} - 1$*, otherwise* $y^{k\,**}_{(j,\cdot)} = y^{k\,*}_{(j,\cdot)} - 1$.

(6)*: If* $y^{k\,**}_{(\cdot,j)} = y^{k\,*}_{(\cdot,j)} + 1 \Rightarrow u^{k\,*}_{(i,j)} = 0$*, otherwise* $u^{k\,*}_{(i,j)} = 1$.

(4) The flow into an arrival node is increased

(7)*:* $y^{k\,**}_{(\cdot,i)} = y^{k\,*}_{(\cdot,i)} + 1$.

(6)*: If* $y^{k\,**}_{(\cdot,i)} = y^{k\,*}_{(\cdot,i)} + 1 \Rightarrow u^{for,k\,*}_{(i,j)} = 1$*, otherwise* $u^{for,k\,*}_{(i,j)} = 0$.

All remaining variables of Model (6) *can be set according to the following scheme to produce a feasible solution:* $u^{for,k\,*}_{(i,j)} = 1, v^{for,k\,*}_{(i,j)} = \overline{v}^{for,k}_{(i,j)}$, $v^{k\,*}_{(i,j)} = v^{k\,*}_{(j,\cdot)}$, $u^{k\,*}_{(i,j)} = 0\ \forall (i,j) \in F_k$, $u^{k\,*}_{(i,j)} = x^{k\,*}_{(j,\cdot)}\ \forall (i,j) \in E_k \setminus F_k$. *The objective changes imply*

$$v^{k\,*}_{(i^*,j^*)} = \sum_{(i,j)\in F_k} c^k_{(i,j)}(x^{k\,**}_{(i,j)} - x^{k\,*}_{(i,j)}) + \sum_{(i,j)\in F_k} c_{(i,j)} z^{**}_{(i,j)}$$

and thus the prove of the first direction is finalized.

Now we prove the optimality of the obtained solution. Assume, that a feasible $v^{k\,\#}_{(i^*,j^*)} < v^{k\,*}_{(i^*,j^*)}$ *exists, which implies that there exists an optimal solution* $x^*, y^*, z^*, u^\#, v^\#$ *which yields a lower objective value than* x^*, y^*, z^*, u^*, v^**. Then we construct a valid path according to* $u^\#, v^\#$ *as follows: Starting with the initial ground arc* (i^*, j^*) *in the path one can again distinguish three cases depending on the current arc visited:*

(1) (Case 1) Ground arc visited against direction
if $u^{k\,\#}_{(i,j)} = 1$ *the next arc is subsequent flight arc (Case 3),*
otherwise subsequent ground arc (Case 1).

(2) (Case 2) Ground arc visited in actual direction
if $u^{for,k\,\#}_{(i,j)} = 1$ *the next arc is preceding flight arc (Case 3),*
otherwise preceding ground arc (Case 2).

(3) (Case 3) Flight arc
if $u^{k\,\#}_{(i,j)} = 1$ *the next arc is subsequent ground arc (Case 1),*
otherwise the preceding ground arc of the arrival node (Case 2).

This procedure terminates, since there can be no cycles on just one arc by construction and cycles over more than one arc are prevented due to $c_{(i,j)} - c^k_{(i,j)} > 0\ \forall k \in K$*. Also, the last arc of path has to be in* H_k*. Otherwise there exists a flight arc in the path after which only ground arcs exist that run backward in time. Then the solution* $(x^*, y^*, z^*, u^\#, v^\#)$ *cannot be optimal due to the lower bounds of all* $v^{for,k\,\#}_{\cdot}$*, which leads to a smaller objective value whenever the u-variable on the corresponding flight is set to* 1*. The remaining schedule can be executed by the initially used aircraft minus one failing aircraft if the flow variables along the path are modified: For every arc in Case 1 the corresponding y is increased by one, whereas in Case 2 it has to be increased. In Case 3 the x-variables are set to* 0*. Since the flow conservation is feasible, this solution is also feasible for Model* (7)*. Hence and since* $v^{k\,\#}_{(i^*,j^*)}$ *corresponds to the sum of all flight cancellation costs minus assignment costs and all flights of the path and* $v^{k\,\#}_{(i^*,j^*)} < v^{k\,*}_{(i^*,j^*)}$*, we have also found a solution of Model* (7) *which yields a better objective value than* x^{**}, y^{**}, z^{**}*. This contradicts the assumption.* □

Theorem 3 states that Model (6) can correctly capture the cost of flight cancellations with subsequent re-planning if only one aircraft from the same fleet fails in each scenario. This assumption is not too restrictive, since aircraft failures of the same type are rather unlikely as long as no general warning has been issued by the manufacturer. If this is the case, one must generally plan without this fleet and tail swaps become unnecessary anyway. However, if there are scenarios in which two aircraft of the same fleet may fail, the relevant fleet can be divided into two sub-fleets, which solves this issue. Whenever the condition is fulfilled the model captures the costs regardless of the number of disjoint scenarios. Thus, any number of disjoint scenarios can be added and no additional scenario variables are needed. Hence, we found a compact formulation of a two-stage robust approach, which optimizes a reduced tail assignment on every stage and for every possible scenario and considers only the cancellation costs. A similiar model has already been described in (13), where a scenario expansion of the model has been performed: For each scenario the variables of a further (time reduced) tail assignment have to be added. This is a major drawback when it comes to the computational tractability of the problem.

5. Computations

In this section, all previously listed models and procedures are tested. It is examined whether the generated solutions produce reasonable results and are practicable. To answer these questions, the deviation from the nominal objective function value will be recorded for all models and the benefit in case of flight cancellations or aircraft failures will be evaluated. In addition, the runtime of all calculations is recorded.

In order to state representative results, a mean value computational study is performed for the flight cancellation Procedure 1 and the robust fleet assignment computations using Model (6).

To generate instances that are as realistic as possible, the following procedure was applied: Smaller schedules are randomly cut out from a real summer flight schedule of a large German airline before pandemic times. Care was taken to ensure that the schedules generated in this way contain dense flight sequences, that means flight sequences with not to much time in between two consecutive flights. For each instance, it was possible to specify how many flights should be included and how many airports may be used for take-off and landing. A file was also created for the airports used, which contains all necessary specifications of each airport. When a schedule was created, a rudimentary tail assignment was already performed in order to provide an appropriate selection of aircraft in addition to the flights of the instance. The data for the aircraft also comes from the large German airline already mentioned. It should be emphasized that this tail assignment was not used further and only serves to select an appropriate number of aircraft.

Of course, problem-specific adjustments had to be made for the individual computational studies. These are described in the individual subsections.

All algorithms are implemented using Python 3.7.1 and all mixed-integer problems are solved with Gurobi 9.1.1, see (17). All parts of the computational study were performed on a 4-core machine with a Xeon E3-1240 v6 CPU (3.7GHz base frequency) and 32 GB RAM.

5.1. **Flight Cancellations.**

In this subsection, Procedure 1 is tested. For this purpose, an instance set was generated, which contains information about potential flight cancellations. All important data of the instance set can be viewed in Table 5.

The instance set consists of 50 individual instances. The instances do not contain many airports, because we need a connection network graph with as many connections as possible to make efficient tail swaps possible. Otherwise, the tail assignment would be trivial and since the solution space is reduced, the robustification as well. The number of aircraft was again adjusted to the number of flights and aircraft of different fleets

Instance Set	
# Instances	50
# Flights	90 - 120
# Airports	4 - 5
# Aircraft	13 - 18
Aircraft Types	multiple
Flight Cancellations	1 - 12

TABLE 5. Characteristics of the used instance set for both computational evaluations

are used. The flight cancellations were chosen randomly and there are between one and 12 per instance. It should be mentioned that only one instance has more than 10 cancellations.

First, the costs of the solution produced by Procedure 1 are to be evaluated. For this purpose, the procedure was started with a time limit of 10 minutes. It must be noted that there are costs that arise independently of the scenario occurrence through the implementation of the solution and the so-called scenario costs that occur if the flights endangered by the cancellation really fail. The nominal costs of the solution amount to 1.176.704€ on average over all instances, while the scenario costs are almost twice as high with 2.280.929€. The reason for this is that both the assignment costs and the cancellation costs are calculated on the basis of the deviation between demand and supply per flight and thus take on very large values in the event of flight cancellations.

Figure 13 shows how much of the scenario costs can be saved proportionally over the course of the computational runtime if the relative upper bound δ (in Procedure 1) of the additional assignment costs for the robust solution compared to the optimal nominal assignment costs varies between 0.00 and 0.03. The

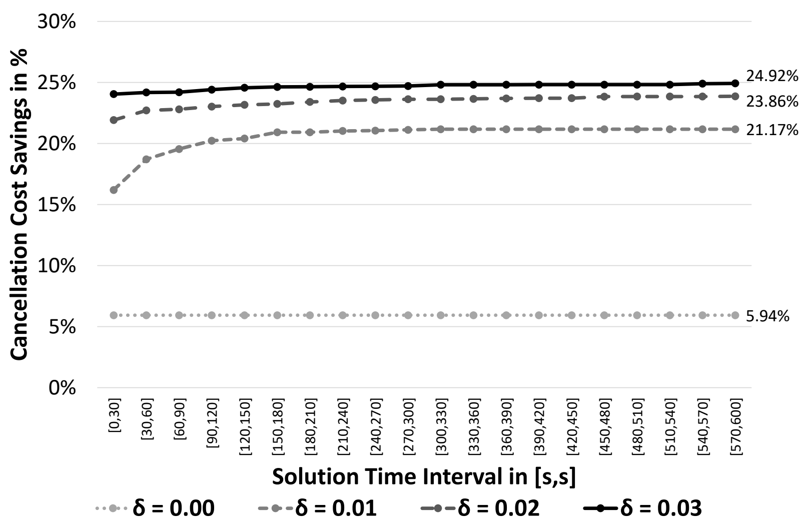

FIGURE 13. Different curves of the cancellation costs when applying Procedure 1 depending on different deviation values δ.

Figure shows that the application of Procedure 1 results in high-quality solutions regarding assignment costs while reducing the cost of consequential flight cancellations significantly on average. Even if there are no costs for the robustness of the solution ($\delta = 0.00$), which means that the fleet assignment is not changed, on average 5.94% of the costs can be saved by simply adjusting the flight strings of the aircraft in the case of scenario occurrence. Here, the calculation of the robust solution takes less than 30 seconds for all instances. Allowing for a 1% deviation from the nominal cost of the robust solution ($\delta = 0.01$), the scenario cost can be reduced by 20% on average after only two minutes of computational runtime. Investing even more in a secured solution ($\delta \in \{0.02, 0.03\}$) can save an average of almost a quarter of the cost of flight cancellations. The runtime indicates that usable robust solutions could already be generated within 30 seconds and from five minutes onwards hardly any improvement occurs. Most of the instances could be solved to optimality. In some calculations, the solver fails in proving optimality, which explains the longer runtimes.

To emphasize the effectiveness of the computational results, we have plotted the development of the flight sequences of a representative instance in Figure 14. Here, five different solutions (rows) and the subsequent flight failures per initial flight failure (columns) are plotted. It can be seen immediately that already the change to the robust objective function resulted in three flights less being canceled without paying a single Euro ($\delta = 0.00$) to robustify the solution. If the price of robustness is allowed to increase, one can see that the number of canceled flights could be further reduced. There are no restrictions on the feasibility space and cycles of any length are feasible as long as they lead to the optimal solution when they are combined.

In summary, Model (3) embedded into Procedure 1 produces very good results. Even without additional costs, safer tail assignment solutions could be found, saving almost 6% of the costs incurred when the scenario flights fail. If the airline is risk-averse and willing to invest more money, up to 25% of the cancellation costs can be saved. Furthermore, almost all solutions could be calculated within 10min.

5.2. **Aircraft Failures.** In this subsection we evaluate the results of the secured fleet assignment, which is described in Model (6). Again, the instance set described in Table 5 was used for the calculations, but of course the individual flight failure data is not taken into account. Instead, each calculation started with solving the nominal fleet assignment problem. This involved identifying and fixing the takeoff positions of each aircraft type. Subsequently, scenarios described in Section 4 were randomly selected, each of which had two aircraft of different types fail at the start of the plan. By selecting only one aircraft per type and scenario, we satisfy the criterion of Theorem 3 in this case. The number of scenarios equals 10 for each instance, and each scenario is weighted equally, taking it for granted that at least one scenario occurs.

Using the 50 instances, we performed a mean value study. The aim of this study is to demonstrate the efficiency of the compact model presented. Since, according to Theorem 3, Model (6) is equivalent to a full scenario expansion, we naturally want to use it for benchmarking. In order to solve a scenario-expanded model as efficiently as possible, one uses the so-called L-Shaped method (see (30), (22)) in practice. Therefore, all instances are also solved using the L-shaped method. In the following, we present the results of the comparison of both solution approaches.

In the initial computational studies with Model (6), it was particularly noticeable that it took a relatively long time to reduce the incumbent value. Experiments showed that in the course of the solution process, reassignments of x-variables were made only rarely and that the u and v-variables added for the robustification were changed much more frequently. This forced us to develop a heuristic procedure to find good initial solutions, which we explain in the following: For the first 120 seconds of the solution process, we added the following constraints to Model (6):

$$u^k_{(i,j)} = 1 \qquad \forall k \in K, (i,j) \in F_k \tag{8a}$$

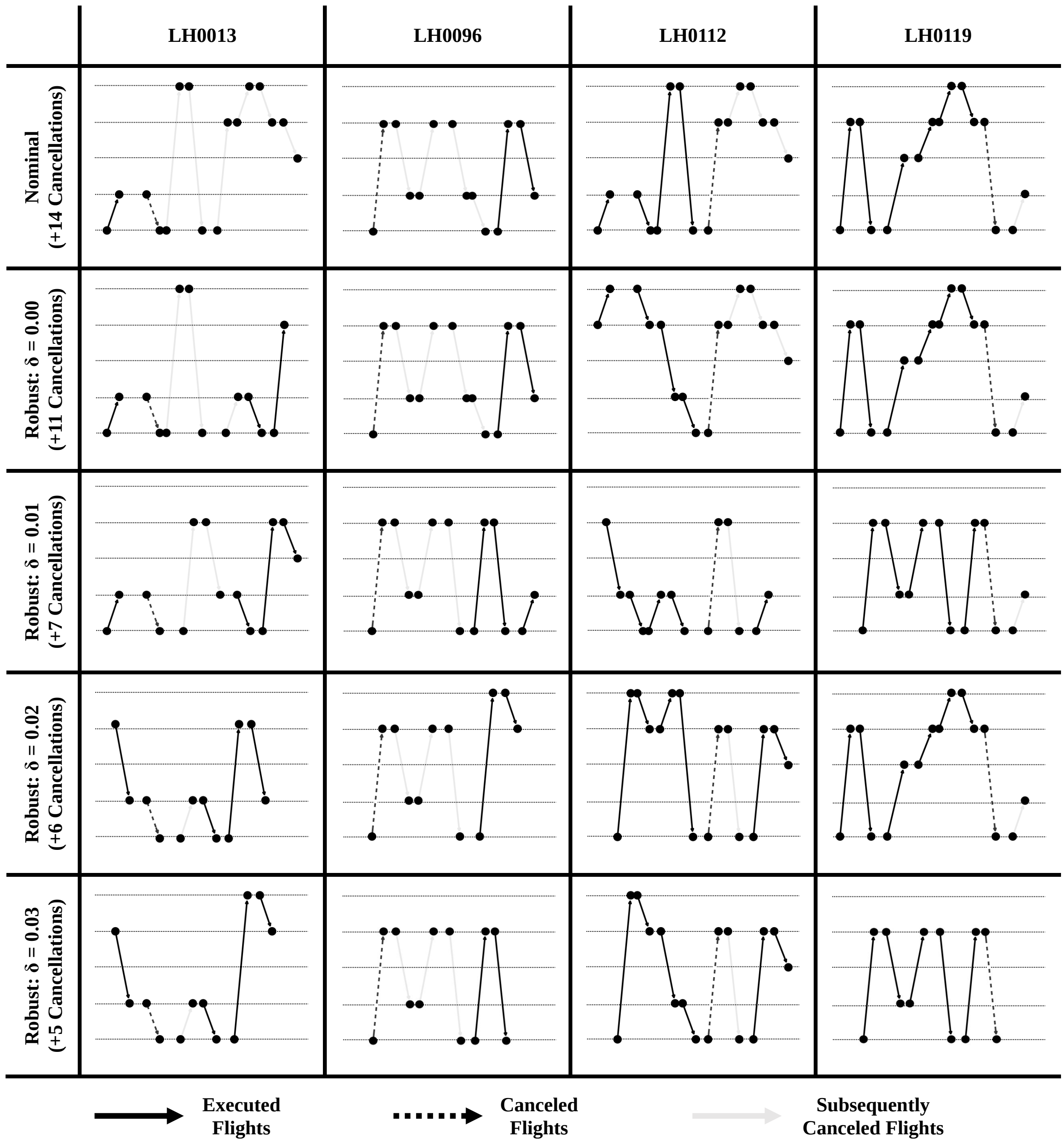

FIGURE 14. The development of flight sequences per flight failure in instance 13 with increasing values for δ in Procedure 1.

This effectively results in aircraft failure costs being calculated as in Example 1 and simplifies the calculations significantly, since the constraints (6g) to (6j) no longer have to be observed. After 120 seconds, the fixing Constraints (8a) were deleted again. The solution obtained in this way served as the starting solution for the complete Model (6). Thus, each run of our new model consists of two steps: First, the heuristic is worked with for 120 seconds, then it is improved by the entire model for 480 seconds and validated for its actual optimality. The L-Shaped method starts directly from the solution obtained by the nominal solution run as mentioned before.

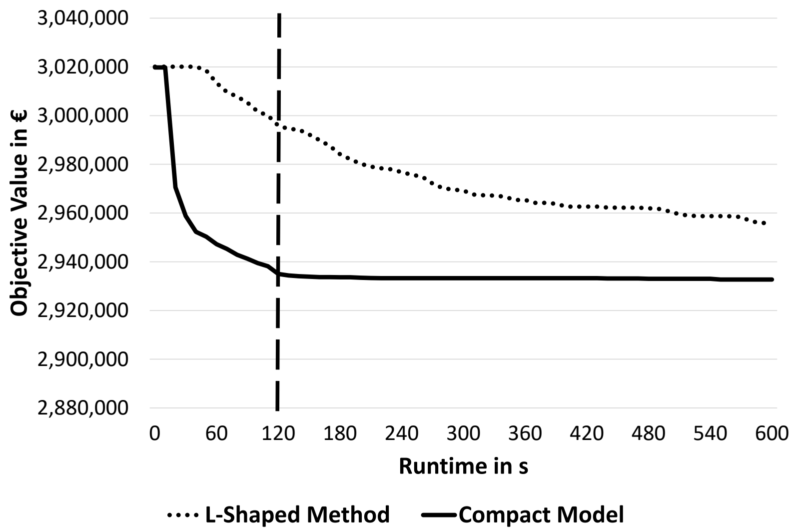

FIGURE 15. Incumbent values of Model (6) plotted over runtime compared to the objective values generated by the L-Shaped method.

In Figure 15 one can see the course of the incumbent value of both solution methods: Both start on a plateau at the level of the value that the objective function corresponds to when the nominal optimal tail assignment is assumed. After only a few seconds, a much better incumbent can be found by our model, while the L-Shaped method rather leads to a slow descent of the objective function value. After 120 seconds, our approach switches to the accurate model: The incumbent decreases only very slightly at some points. The L-Shaped method can still improve the objective function value from this point on, but does not come close to the objective function value that could be found using Model (6).

To see in more detail what impact the solutions we found have on our problem, we broke down the parts of the objective function in more detail: Figure 16 shows the additional assignment costs of the robust solutions generated by the compact Model (6) compared to the L-Shaped method plotted over time. This corresponds to the cost of implementing the robust solution reduced by the cost of implementing the nominal solution. The costs incurred by the realization of the uncertainty are not yet included. These are plotted in Figure 17: Here, the average costs over all scenarios are considered, which additionally arise if the scenarios actually occur. Thus, the value plotted in Figure 15 corresponds to the sum of the values in Figure 16 and Figure 17 plus the additional constant difference to the average nominal costs over all instances.

In Figure 16, it can be clearly seen that the additional assignment costs increase much faster for the compact model and then remains at a lower level than for the L-Shaped method. At the same time, this fast increase can be explained by Figure 17, where it can be seen that the mean costs of the realization of a scenario from the scenario set decreases more than the additional assignment costs increase. After 10 minutes, Model (6) calculates on average solutions that have lower assignment costs and lower costs in the realization of scenarios

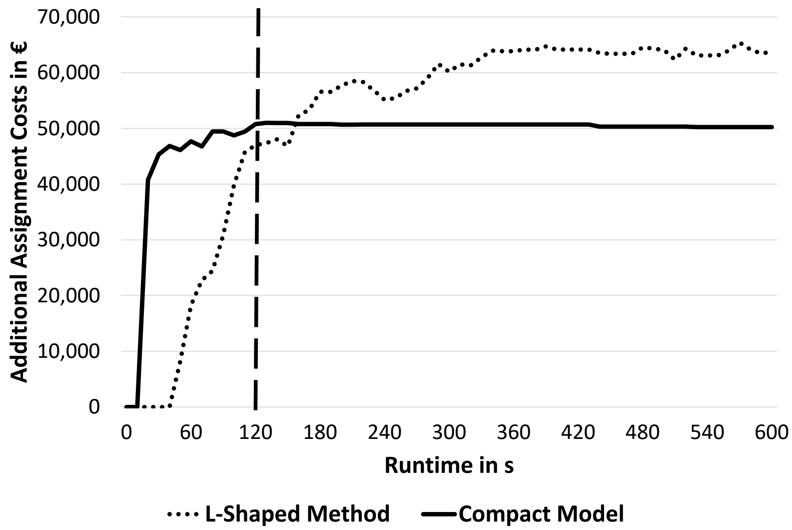

FIGURE 16. The additional assignment costs of Model (6) plotted over runtime compared to the additional assignment costs of the solutions found by the L-Shaped method.

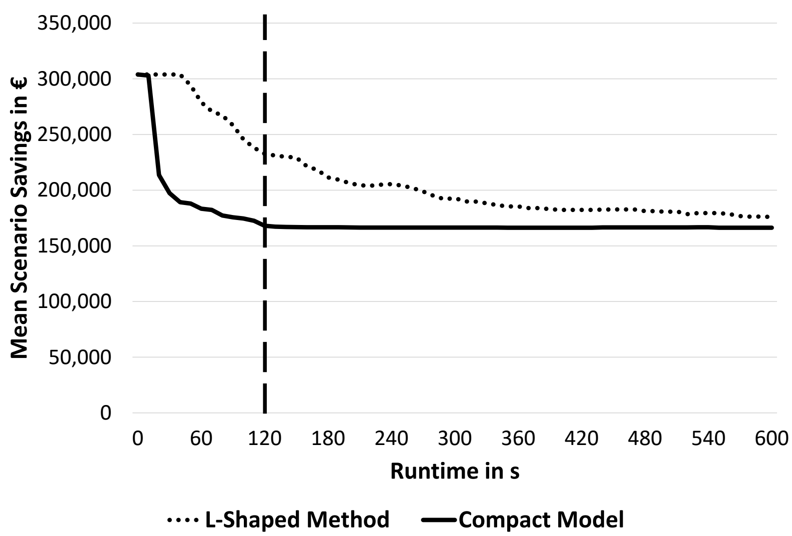

FIGURE 17. The mean additional costs over all scenarios of Model (6) plotted over runtime compared to the costs which occur using the L-Shaped method.

than the solutions calculated by the L-Shaped method. Thus, the solutions found are cheaper and also more robust than the solutions found by the L-Shaped method.

Finally, we want to take a look at the optimality gaps of both models: These are plotted over time in Figure 18. Here, too, we can initially see the same trend as in all the other figures. The optimality gap of the compact Model (6) falls very quickly and then stagnates, while the gap of the L-Shaped method only descends slowly. Note that the gap of the compact model is not monotonically descending. This is due to the fact that during the first 120 seconds the models are solved with the additional constraint (8a). When the optimal solution of this problem relaxations are found, the constraint is neglected and the solution process

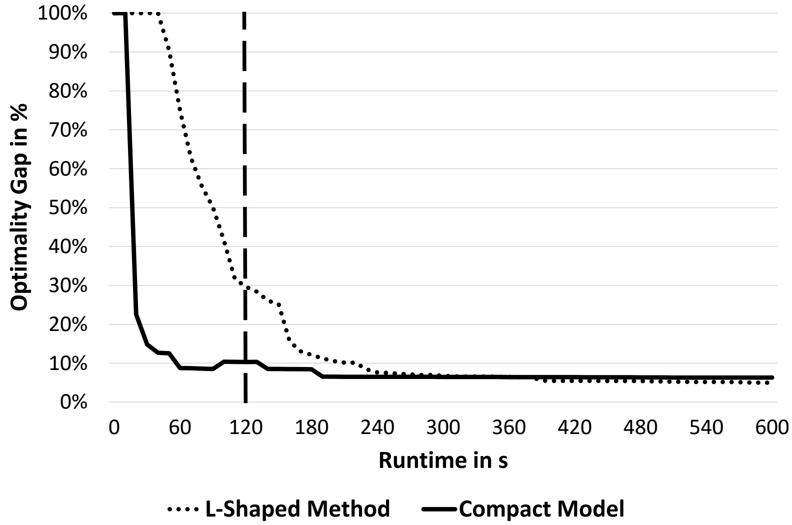

FIGURE 18. The developement of the optimality gap over runtime when using Model (6) compared to optimality gaps of the L-Shaped method.

proceeds. The evaluation of the solution in the unrestricted model leads to higher optimality gaps which cause the mean increase in Figure 18. From 120 seconds on, however, the course is monotonously descending and reflects the actual gap of the problem. However, the mean gap of the L-Shaped method falls below the gap of the compact model over time. This means that the lower bounds found with the L-Shaped method are tighter and thus larger. This has to be considered under the aspect that the incumbent value of the L-Shaped method at this time is still much higher than that of the compact model. This in turn indicates that the solutions found by our model are very close to the optimal solution, but the model is unable to verify their optimality since the LP-relaxation of the problem is not tight.

In summary, it can be said that Model (6) leads to very good results in the test computations: The model leads much faster to high-quality solution values than the comparable L-Shaped method. On average, the obtained solutions are not only less expensive, but also quantifiably superior in terms of additional costs when a scenario occurs. In addition, all results were calculated within less than 10 minutes and the steepest decline of the incumbent value occurred within the first 120 second. This fast solution finding makes it possible to use the approach profitably in practice, where solution times may be very limited.

6. Summary and Outlook

In summary, this paper has successfully demonstrated the potential of fleet and tail assignment protected against uncertainty. New robust models could be found for severe disruptions of the nominal problem: In the case of flight cancellations, an efficient new model and solution procedure could be developed, which resulted in significant savings for flight cancellations while limiting the costs for hedging the solution. Regarding whole aircraft failures, a model could be established that can correctly anticipate and minimize the consequential costs. All models led to near-optimal solutions within 10 minutes and can therefore be used in practice.

Starting from these high-quality results in the isolated area of aircraft assignment optimization, the next step is to find safeguarding optimization models and procedures for overarching problem structures. In this context, schedule design, maintenance planning and crew assignment can serve as starting points, which can be robustified together with the problems already considered.

7. Acknowledgements

This research has been conducted in the framework of the research project OPs-TIMAL, financed by the German Federal Ministry of Economic Affairs and Energy (BMWi) [grant number 20X1711L].

References

[1] Jeph Abara. Applying integer linear programming to the fleet assignment problem. *Interfaces*, 19(4):20–28, 1989.

[2] Yana Ageeva. *Approaches to incorporating robustness into airline scheduling*. PhD thesis, Massachusetts Institute of Technology, 2000.

[3] Shervin Ahmadbeygi, Amy Cohn, and Marcial Lapp. Decreasing airline delay propagation by re-allocating scheduled slack. *IIE transactions*, 42(7):478–489, 2010.

[4] Cynthia Barnhart, Natashia L Boland, Lloyd W Clarke, Ellis L Johnson, George L Nemhauser, and Rajesh G Shenoi. Flight string models for aircraft fleeting and routing. *Transportation science*, 32(3):208–220, 1998.

[5] Aharon Ben-Tal and Arkadi Nemirovski. Robust optimization–methodology and applications. *Mathematical programming*, 92(3):453–480, 2002.

[6] Matthew E Berge and Craig A Hopperstad. Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. *Operations research*, 41(1):153–168, 1993.

[7] John R Birge and Francois Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

[8] Ralf Borndörfer, Ivan Dovica, Ivo Nowak, and Thomas Schickinger. Robust tail assignment. 2010.

[9] Lloyd W Clarke, Christopher A Hane, Ellis L Johnson, and George L Nemhauser. Maintenance and crew considerations in fleet assignment. *Transportation science*, 30(3):249–260, 1996.

[10] Guy Desaulniers, Jacques Desrosiers, Yvan Dumas, Marius M Solomon, and François Soumis. Daily aircraft routing and scheduling. *Management Science*, 43(6):841–855, 1997.

[11] Michelle Dunbar, Gary Froyland, and Cheng-Lung Wu. Robust airline schedule planning: Minimizing propagated delay in an integrated routing and crewing framework. *Transportation Science*, 46(2):204–216, 2012.

[12] Matteo Fischetti and Michele Monaci. Light robustness. In *Robust and online large-scale optimization*, pages 61–84. Springer, 2009.

[13] Gary Froyland, Stephen J Maher, and Cheng-Lung Wu. The recoverable robust tail assignment problem. *Transportation Science*, 48(3):351–372, 2014.

[14] Lukas Glomb, Frauke Liers, and Florian Roesel. A novel decomposition approach for holistic airline optimization. 2021.

[15] Lukas Glomb, Frauke Liers, and Florian Rösel. A rolling-horizon approach for multi-period optimization. *European Journal of Operational Research*, 2021.

[16] Ram Gopalan and Kalyan T Talluri. The aircraft maintenance routing problem. *Operations Research*, 46(2):260–271, 1998.

[17] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2020.

[18] Christopher A Hane, Cynthia Barnhart, Ellis L Johnson, Roy E Marsten, George L Nemhauser, and Gabriele Sigismondi. The fleet assignment problem: Solving a large-scale integer program. *Mathematical Programming*, 70(1-3):211–232, 1995.

[19] Andreas Heidt, Hartmut Helmke, Manu Kapolke, Frauke Liers, and Alexander Martin. Robust runway scheduling under uncertain conditions. *Journal of Air Transport Management*, 56:28–37, 2016.

[20] Benno Hoch, Frauke Liers, Sarah Neumann, and Francisco Javier Zaragoza Martınez. The non-stop disjoint trajectories problem, 2020.

[21] Shan Lan, John-Paul Clarke, and Cynthia Barnhart. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation science*, 40(1):15–28, 2006.

[22] Gilbert Laporte and François V Louveaux. The integer l-shaped method for stochastic integer programs with complete recourse. *Operations research letters*, 13(3):133–142, 1993.

[23] Nikolaos Papadakos. Integrated airline scheduling. *Computers & Operations Research*, 36(1):176–195, 2009.

[24] Brian Rexing, Cynthia Barnhart, Tim Kniker, Ahmad Jarrah, and Nirup Krishnamurthy. Airline fleet assignment with time windows. *Transportation science*, 34(1):1–20, 2000.

[25] Jay M Rosenberger, Ellis L Johnson, and George L Nemhauser. A robust fleet-assignment model with hub isolation and short cycles. *Transportation science*, 38(3):357–368, 2004.

[26] Rivi Sandhu and Diego Klabjan. Integrated airline fleeting and crew-pairing decisions. *Operations Research*, 55(3):439–456, 2007.

[27] Shengzhi Shao, Hanif D Sherali, and Mohamed Haouari. A novel model and decomposition approach for the integrated airline fleet assignment, aircraft routing, and crew pairing problem. *Transportation Science*, 51(1):233–249, 2017.

[28] Sergey Shebalov and Diego Klabjan. Robust airline crew pairing: Move-up crews. *Transportation science*, 40(3):300–312, 2006.

[29] Barry C Smith and Ellis L Johnson. Robust airline fleet assignment: Imposing station purity using station decomposition. *Transportation Science*, 40(4):497–516, 2006.

[30] Richard M Van Slyke and Roger Wets. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM journal on applied mathematics*, 17(4):638–663, 1969.

[31] Chiwei Yan, Cynthia Barnhart, and Vikrant Vaze. Choice-based airline schedule design and fleet assignment: A decomposition approach. *Available at SSRN 3513164*, 2020.

[32] Joyce W Yen and John R Birge. A stochastic programming approach to the airline crew scheduling problem. *Transportation Science*, 40(1):3–14, 2006.