Journal of Air Transport Management xxx (2016) 1–9

Contents lists available at ScienceDirect

Journal of Air Transport Management

journal homepage: www.elsevier.com/locate/jairtraman

Pre-tactical optimization of runway utilization under uncertainty

Manu Kapolke [a, *], Norbert Fürstenau [b], Andreas Heidt [a], Frauke Liers [a], Monika Mittendorf [b], Christian Weiß [b]

[a] Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) Economics, Discrete Optimization, Mathematics, Cauerstr. 11, 91058 Erlangen, Germany
[b] German Aerospace Center (DLR), Institute of Flight Guidance, Lilienthalplatz 7, 38108 Braunschweig, Germany

ARTICLE INFO

Article history:
Received 30 July 2015
Received in revised form
13 November 2015
Accepted 4 February 2016
Available online xxx

Keywords:
Pre-tactical planning
Runway
Optimization
Uncertainty
Robustness
Data analysis

ABSTRACT

Efficient planning of runway utilization is one of the main challenges in Air Traffic Management (ATM). It is important because runway is the combining element between airside and groundside. Furthermore, it is a bottleneck in many cases. In this paper, we develop a specific optimization approach for the pre-tactical planning phase that reduces complexity by omitting unnecessary information. Instead of determining arrival/departure times to the minute in this phase yet, we assign *several* aircraft to the same time window of a given size. The exact orders within those time windows can be decided later in tactical planning. Mathematically, we solve a generalized assignment problem on a bipartite graph. To know how many aircraft can be assigned to one time window, we consider separation requirements for consecutive aircraft types. In reality, however, uncertainty and inaccuracy almost always lead to deviations from the actual plan or schedule. Thus, we present approaches to incorporate uncertainty directly in our model in order to achieve a stabilization with respect to changes in the data. Namely, we use techniques from robust optimization and stochastic optimization. Further, we analyze real-world data from a large German airport to obtain realistic delay distributions, which turn out to be two-parametric Γ-distributions. Finally, we describe a simulation environment to test our new solution methods.

1. Introduction

ATM systems are driven by economic interests of the participating stakeholders and, therefore, are performance oriented. As possibilities of enlarging airport capacities are limited, one has to enhance the utilization of existing capacities to meet the continuous growth of traffic demand. The runway system is the main element that combines airside and groundside of the ATM System. Therefore, it is crucial for the performance of the whole ATM System that the traffic on a runway is planned efficiently. Such planning is one of the main challenges in ATM. Uncertainty, inaccuracy and non-determinism almost always lead to deviations from the actual plan or schedule. A typical strategy to deal with these changes is a regular re-computation or update of the schedule. These adjustments are performed in hindsight, i.e. after the actual change in the data occurred. The challenge is to incorporate uncertainty into the initial computation of the plans so that these plans are robust with respect to changes in the data, leading to a better utilization of resources, more stable plans and a more efficient support for ATM controllers and stakeholders. Incorporating uncertainty into the ATM planning procedures further makes the total ATM System more resilient, because the impact of disturbances and the propagation of this impact through the system is reduced.

In the present paper, we investigate the problem of optimizing runway utilization under uncertainty. The goal is to incorporate uncertainties into the initial plan in order to retain its feasibility despite changes in the data. We focus on the pre-tactical planning phase, i.e. we assume the actual planning time to be several hours, or at least 30 min, prior to actual arrival/departure times. We develop an appropriate mathematical optimization model for this particular planning phase. The basic idea is that in pre-tactical planning we can reduce the complexity of the problem by not determining an exact arrival/departure sequence in terms of exact landing/take-off times for each aircraft, as we do later in tactical planning. Instead, we answer the question of how many aircraft can be scheduled to one time window of a given size without violating distance requirements. (For example, it is definitely possible to

* Corresponding author.
E-mail addresses: manu.kapolke@fau.de (M. Kapolke), norbert.fuerstenau@dlr.de (N. Fürstenau), andreas.heidt@fau.de (A. Heidt), frauke.liers@fau.de (F. Liers), monika.mittendorf@dlr.de (M. Mittendorf), christian.weiss@dlr.de (C. Weiß).

http://dx.doi.org/10.1016/j.jairtraman.2016.02.004
0969-6997/

assign more than one aircraft to a time window from 12:00 pm to 12:10 pm.) Then, we consider a discretized time horizon consisting of such time windows and assign each aircraft to one of them.

This paper is an extension of Fürstenau et al. (2014), where the authors set up a mixed integer program (MIP) for the pre-tactical optimization of runway utilization. Afterwards, the impact of disturbances on the deterministic solutions was investigated. The results showed that it is crucial to enrich the optimization approach by protection against uncertainties, in order to produce less necessary replanning. In the current paper, we thus incorporate uncertainties directly into the model by using techniques from robust and stochastic optimization. The remainder of this paper is organized as follows: In Section 2, we give an overview over the literature related to runway optimization and explain why our approach is different. We develop the pre-tactical runway optimization model in Section 3. In Sections 4 and 5 we describe our approaches to incorporate uncertainties into this model, and present some computational results in Section 4. In order to be able to test our approaches in a more realistic setting, we analyze real-world delay data from a large German airport in Section 7 (extending the descriptions in Fürstenau et al. (2014)), where we also describe our simulation environment to test current and future solution methods. Finally, we conclude in Section 8.

2. Related work

There are many different approaches that deal with the optimization of runway utilization in the literature. Most of them treat the runway scheduling problem in the tactical planning phase.

2.1. Deterministic approaches

The most cited MIP model in this context is probably the one introduced by Beasley et al. (2000). Their linear objective function minimizes delay, the constraints come from the aircraft dependent separation times. They also present an integer program (IP) formulation where time is discretized, but they don't explore it further because of disappointing computational experiences. Soomer and Franx (2008) consider the problem from an airline point of view. They use Beasley's MIP but allowing airlines to define their own cost functions for each flight. Bertsimas et al. (2011) develop a comprehensive IP for Air Traffic Flow Management which integrates all phases of a flight, different costs for ground and air delays, rerouting, continued flights and cancellations. Kjenstad et al. (2013) state a time-discretized model. They assign an aircraft to a time window and claim that a number of subsequent time windows (dependent on the aircraft type) remains unassigned. In their model, they also consider minimal taxiways and the possibility to drop departures. Their linear objective function minimizes delay and the number of dropped departures.

Many authors use heuristic methods aiming to provide solutions in close to real-time. To schedule aircraft in a first-come first-served order (FCFS) seems to be fair and also reduces the work of traffic controllers. However, such an approach doesn't provide maximal throughput or minimal delay in general (Bennell et al., 2011). Dear (1976) developed the concept of Constrained Position Shifting, were each aircraft can be scheduled only a limited number of steps away from the FCFS sequence. Balakrishnan and Chandran (2010) solved this problem as a shortest path problem on a special network.

Anagnostakis and Clarke (2003) formulate a two-stage heuristic algorithm for the outbound runway scheduling problem. In the first stage, candidate weight class sequences are determined w.r.t. distance requirements, ordered by the corresponding throughput. In the second stage individual aircraft are assigned using operational constraints (e.g. earliest and latest departure times of aircraft).

As mentioned, in our optimization model (described below and in Fürstenau et al. (2014)) we allocate time windows to aircraft. However, though many papers about runway optimization deal with ”slot allocation”, this term is used to describe different problems. Often, it is associated with the Ground-Holding Problem (GHP), where ”slot” means a certain departure time which is assigned to an aircraft. Ball and Dahl Vossen (2009) also address the GHP, but they assign arrival slots to aircraft which provide the corresponding departure delay in hindsight. They consider matchings in a bipartite graph which they call the “flight allocation graph”. The main focus in this paper lies on the graph structure and matching algorithms.

None of the approaches above deal with “slots” as time windows to which *several* aircraft can be assigned. Thus, to the best of our knowledge there is no approach similar to ours in which the pre-tactical planning phase is modelled by assigning such time windows to aircraft.

2.2. Approaches that incorporate uncertainties

All runway optimization approaches presented above assume that all parameters are known with certainty. We found few works where uncertainties are incorporated. However, none of them are using robustness concepts similar to those described in this paper. Chandran and Balakrishnan (2007), e.g., develop an algorithm with Constrained Position Shifting that handles uncertainty in the estimated time of arrival. Hu and Di Paolo (2008) formulate a genetic algorithm and compute solutions disturbing the estimated arrival time of 20% of the aircraft. Sölveling (2012) presents a two-stage stochastic program for solving the mixed-mode runway scheduling problem with uncertain earliest times. In the first stage he determines the weight class sequence. An exact sequence of individual aircraft follows in the second stage.

3. The modeling

As mentioned above, we model the problem of optimizing runway utilization in the pre-tactical planning phase by assigning time windows to aircraft. Throughout this paper, we consider single-mode runways with only arriving aircraft. In the future, we will adjust our models to mixed-mode runways. But since the single-mode problem is already quite complex from a mathematical point of view, we decided to focus on arrivals for now. In our modeling approach we claim that each aircraft has to receive exactly one time window as each aircraft has to be scheduled. On the other hand, the number of aircraft that can be assigned to one time window depends on its size and the weight classes of the aircraft. The underlying idea is that, contrary to tactical planning, we don't need to determine arrival times to the minute yet, because we are several hours (or at least 30 min) prior to the first scheduled time. Thus, the exact arrival sequences within the time windows can be decided later.

In this section, we develop a MIP for the described problem. The objective is the maximization of punctuality. In other words, the deviation from scheduled times in both directions shall be minimized. The MIP constraints consist of general assignment constraints and the modeling of minimal time distance requirement. Those minimum separation times between two consecutive aircraft depend on their corresponding weight classes. Hereof, we consider three different aircraft categories (*Light*, *Medium* and *Heavy*) and use Table 1 (ICAO Document 4444, 2007).

Before we can state our model, we have to analyze the underlying problem structure more precisely.

For each aircraft, we consider several corresponding times:

Please cite this article in press as: Kapolke, M., et al., Pre-tactical optimization of runway utilization under uncertainty, Journal of Air Transport Management (2016), http://dx.doi.org/10.1016/j.jairtraman.2016.02.004

Table 1
Minimum separation times (in seconds).

Predecessor \ successor	Heavy	Medium	Light
Heavy	100	125	150
Medium	75	75	125
Light	75	75	75

- *Scheduled time of arrival* (ST): a fix time that yields a benchmark to identify delay and earliness of the aircraft. This may be the time the passenger finds on his flight ticket.
- *Earliest time of arrival* (ET): depends on operational conditions (and on the impact of disturbances).
- *Latest time of arrival* (LT): latest time the aircraft can land without holdings. It depends on the earliest time ET and on the actual planning time (or take-off time, respectively, if the aircraft is still on the ground).
- *Maximal latest time of arrival* (maxLT): a hard condition for landing which is calculated with respect to physical, operational and other relevant conditions (for instance, amount of fuel, prioritization, etc.).

Those times further determine the corresponding time windows ST_W, ET_W, LT_W and $maxLT_W$ for each aircraft.

3.1. Assignment graph

To model the problem of assigning aircraft to time windows, we consider a bipartite graph $G=(A\cup W,E)$ consisting of a vertex set A of aircraft and a vertex set W of time windows of a given size in a given time period (ordered chronologically). An edge $(i,j)\in E$ corresponds to a possible assignment of aircraft i to time window j. Possible assignments concerning a certain aircraft are all time windows from ET_W to $maxLT_W$.

Now, a feasible solution for our assignment problem is a set of edges such that.

- every aircraft vertex is linked with exactly one edge from this set, i.e. every aircraft is assigned to exactly one time window,
- every time window vertex is linked with a number of edges from this set, so that no separation time constraints are violated.

In Fig. 1 we see a small example of a bipartite graph with a possible assignment of aircraft $a_1,\ldots,a_4\in A$ to time windows $w_1,w_2,w_3\in W$.

3.2. Decision variables

To solve our assignment problem, we have to decide whether to choose a certain edge or not. To model this decision in our MIP, we introduce a binary variables x_{ij} for each edge $(i,j)\in E$:

$$x_{ij}=\begin{cases}1, & \text{if aircraft } i \text{ is assigned to time window } j\\ 0, & \text{otherwise}\end{cases}$$

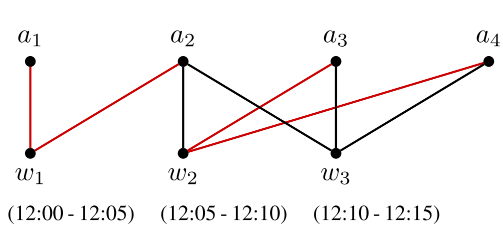

Fig. 1. Assignment graph. Red edges show a possible assignment: aircraft a_1 and a_2 are assigned to time window w_1, a_3 and a_4 are assigned to w_2. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

3.3. Objective function

Our objective is the minimization of delay and earliness, respectively. We model delay/earliness as edge weights. The weight c_{ij} of an edge $(i,j)\in E$ results from the distance of time window j to the ST_W of aircraft i (counted in number of time windows). Delay is penalized quadratically for reasons of fairness (e.g., a solution in which one aircraft has a delay of six time windows is worse than a solution in which two aircraft have a delay of three time windows each). Earliness is penalized linearly. If the assigned time window is after the LT_W (i.e. between LT and maxLT), we add an extra penalization term, namely the squared distance from LT_W. Assume an aircraft i with ST_W w_5, ET_W w_1, LT_W w_{10} and $maxLT_W$ w_{13}. Then we'd have, e.g., $c_{i2}=3$, $c_{i8}=3^2$ and $c_{i12}=7^2+2^2$.

Now the objective function of our optimization model is the following[1]

$$\min \sum_{(i,j)\in E} c_{ij}x_{ij} \tag{1}$$

3.4. Aircraft constraints

First of all, we have to assert that each aircraft is assigned to exactly one time window. So we claim

$$\sum_{j\in W_i} x_{ij}=1 \tag{2}$$

for each $i\in A$, where $W_i=\{j\in W:(i,j)\in E\}$ describes the set of time windows that aircraft i can be assigned to.

3.5. Time window constraints

Further, we have to determine the number of aircraft that can be assigned to one time window. In order to do so, we need to consider the distance requirements, dependent on the weight classes of consecutive aircraft. We use the minimum separation times shown in Table 1.[2] Clearly, the maximum number of aircraft that fit in one time window is reached when a sequence from Light to Heavy is assumed. In more detail, to avoid separation times of 125 and 150 s, such a sequence contains sub-sequences of aircraft of the same type. First, all Lights are scheduled, followed by all Mediums, and finally by all Heavies. According to Table 1, we therefore need a separation time of 75 s after each Light and each Medium, whereas we need 100 s after each Heavy except the last one (the needed separation time after the last aircraft in a time window models the distance requirements at the window boundary and is analyzed later in this section).

For each time window we get upper bounds on the number of aircraft by assuming such a sequence from Light to Heavy. Mathematically, it yields the following two constraints for each $j\in W$:

[1] Since all penalization terms are modelled within the c_{ij}-coefficients, other penalization strategies can also be applied without changing the structure of our MIP.

[2] An adaptation of the results in this paper to other minimum separation time tables is possible as well.

Please cite this article in press as: Kapolke, M., et al., Pre-tactical optimization of runway utilization under uncertainty, Journal of Air Transport Management (2016), http://dx.doi.org/10.1016/j.jairtraman.2016.02.004

$$75\sum_{i\in L_j} x_{ij} + 75\sum_{i\in M_j} x_{ij} + 100\sum_{i\in H_j} x_{ij} \leq s + 100 \quad (3)$$

$$75\sum_{i\in L_j} x_{ij} + 75\sum_{i\in M_j} x_{ij} \leq s + 75 \quad (4)$$

Here, s is the size of the time windows (in seconds). Further, $L_j=\{i\in L:(i,j)\in E\}$ describes the set of Lights that may be assigned to time window j (and thus, the corresponding sum yields the number of Lights that *are* assigned to it). M_j and H_j are defined analogously.

If we assign aircraft to a time window without exceeding these bounds in (3) and (4), we know that there *exists* a sequence of those aircraft that fits in the time window. However, we do not determine how that sequence looks like exactly in terms of concrete predecessors and successors. We are still flexible in (re)arranging different aircraft of the same type. And if the time window contains enough "empty space", we can even deviate from the Light-Medium-Heavy order without changing the assignment.

Further, we extend (3) and (4), because they do not assert security distances at the time window boundaries yet. This means that the last aircraft in one time window and the first aircraft in the subsequent time window may be planned to land at the very same time. In order to obtain feasible solutions, we can generally claim 150 extra seconds as buffers in every time window. But of course, this approach only provides a heuristic procedure for solving the problem because those buffers will be unnecessarily large for some time windows. In Fürstenau et al. (2014), we describe a way to model distance requirements at the window boundaries precisely. For this purpose, we introduce additional variables for each time window which model the situation at the boundaries (dependent on corresponding aircraft types). Afterwards, we modify our constraints to assure suitable minimum separation times at the end of each time window. For instance, if we have a Heavy at the end of a time window j and a Medium at the beginning of the subsequent one (assuming a sequence from Light to Heavy in both windows), we assure 125 extra seconds of separation time at the end of j (according to Table 1). If we have a Medium at the end of j instead, we assure 75 s and so on.

4. Incorporating uncertainties

In this section, we want to incorporate uncertainty into the model to receive a robustification of our solution plan. In general, robustification means to ensure that deviations in the input data do not have a large impact on the solution. Considering the optimal solution of the nominal problem, i.e. the problem where uncertainties are ignored, small deviations in the input data could have the effect that the nominal optimum becomes infeasible for the disturbed problem, i.e. the problem where the input data suffers from deviations.

In mathematics, there exist different approaches to handle uncertainty in optimization. In *stochastic optimization* (e.g. Kall and Mayer, 2013) the goal is to describe the uncertainty by probability distributions. Knowing these distributions, one can then optimize the expected values. A second approach to the problem of modelling uncertainty is located in *robust optimization* (e.g. Ben-Tal et al., 2009; Bertsimas and Sim, 2003), where the goal is to immunize against predefined worst-case scenarios. In contrast to stochastic optimization, the probability distributions of the uncertainties do not need to be known. However, one has to predefine uncertainty sets that determine the values of the uncertain parameters against which the optimization problem has to be protected. The task is to find robust feasible solutions, i.e. solutions that are feasible for all parameter values in the uncertainty set. Among all robust feasible solutions, the robust optimal solutions are those with the best guaranteed objective function values.

4.1. Robust optimization approach

In the setting for our model described in Section 3, the uncertain parameters are the ET windows ET_W and, dependent on those, LT_W and $maxLT_W$. Hence, we have to predefine an uncertainty set for each aircraft. Therefore, we have to chose deviations of the earliest time we want to be protected against. For each aircraft this yields an interval of possible earliest times and thus a set of possible ET_W's. These ET_W's also determine the possible $maxLT_W$'s.

Now, we actually solve our optimization model from Section 3. But in the robust approach we assume an assignment graph that only contains edges which are feasible for every scenario according to our chosen uncertainty set. A *scenario* is defined by exactly one interval of possible assignments for each aircraft. An example of feasible assignments for an aircraft in the robust model is illustrated in Fig. 2. As mentioned, the robust model assumes the worst-case, i.e. the extreme cases for earliest time (w_4) and maximal latest time (w_7) in the predefined uncertainty set are taken into account, whereas the other time windows which lay within the uncertainty set for both times (w_2, w_3, w_8, w_9) are forbidden.

4.2. Stochastic optimization approach

We follow a single-stage stochastic optimization approach in which we optimize over all assignments which are "expected to be possible" dependent on the underlying probability distribution. Therefore, we consider the expected values for ET and maxLT for each aircraft, or the corresponding time windows, respectively. Afterwards, we optimize the obtained "expected scenario", i.e. we solve our mathematical model described above with decision variables (edges in the assignment graph) that correspond to the feasible assignments in this scenario. In Fig. 3 we show an example of feasible assignments in the expected scenario for one aircraft.

5. Advanced robustness concepts

The robustification approach in the previous section was a strict one, which potentially can be too conservative and produce large delay values. In this section we take a look at less conservative robust approaches to avoid the over-conservatism of strict robustness.

5.1. Recoverable robustness

In the context of railway, recoverable robustness (Liebchen et al., 2009) has been established. We now apply these approaches to our optimization model for pre-tactical planning.

Since a (nominal) solution might become infeasible by a realized scenario, the concept of recoverable robustness, introduced by Liebchen et al. (2009), simultaneously considers the optimization

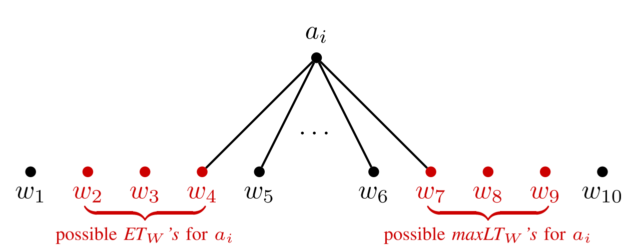

Fig. 2. Possible assignments for an aircraft a_i in the robust model.

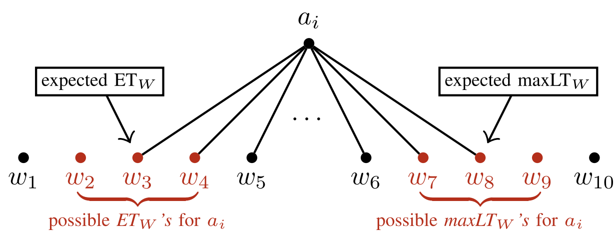

Fig. 3. Possible assignments for an aircraft a_i in the stochastic model.

of the problem and an algorithm, which repairs the solution to a feasible one, in cases of infeasibility. This algorithm is called recovery action. In contrast, the strict robust approach postulates that a strict robust solution is feasible for all possible uncertainties in the uncertainty set, which of course is a more strict assumption. However, a recoverable robust solution can be recovered by limited actions to a feasible one for all occurring scenarios (not necessarily the same feasible solution in each scenario).

We consider our nominal model and introduce first stage variables x_{ij}, which denote the assignment variables of the nominal solution (as defined in Section 3.2). Furthermore, we have to define additional recovery variables y_{ij}^s for each aircraft $i \in A$, time window $j \in W_i^s$ and scenario $s \in S$ (each scenario s is determined by $\cup_{i \in A} W_i^s$, where W_i^s describes the time horizon to which aircraft i can be assigned in this scenario; note that the number of time windows contained in W_i^s for a certain i is independent of s). These second stage variables denote second-stage assignments for each scenario (equal to 1 if aircraft i is assigned to time window j in scenario s).

Thus we want to minimize the delay costs of the nominal solution and the worst-case costs for the recovery action for each scenario. Since we want to plan a nominal solution x as close as possible to a feasible solution y^s for all scenarios, the recovery costs for each scenario $s \in S$ would be the difference between the computed nominal solution of the x-variables and the solution of the y^s-variables. As we require fairness for the recoverable robust solution in general, we square this difference and optimize the following objective function

$$\min_x \sum_{i \in A} \sum_{j \in W_i} c_{ij} x_{ij} \quad + \tag{5}$$

$$\max_{s \in S} \min_{y^s} \sum_{i \in A} \left(\sum_{j \in W_i} j \cdot x_{ij} - \sum_{l \in W_i^s} l \cdot y_{il}^s \right)^2 \tag{6}$$

whereby the first summand (5) denotes the nominal optimization problem with delay costs c_{ij} and the second summand (6) describes the recovery costs.

Now we require a feasible assignment on both first stage and second stage. Thus, besides the constraints defined in Sections 3.4 and 3.5 for the x-variables, we further need the same constraints for the y^s-variables *for each scenario* $s \in S$ (using the corresponding assignment graphs). Although x- and y-variables decompose in the constraints, this problem is mathematically challenging. Beside the quadratic objective function we obtain a min-max-min-structure. Furthermore, it is a problem with $1 + |S|$ many assignments with quadratic objective function. Therefore we propose a simplification.

5.2. Recovery to strict robust solution

In order to handle the problem of the min-max-min-structure, we can recover the nominal solution to a strict robust one. By that, we do not have y^s-variables for each scenario, but only consider the worst-case. Again, using the structure of uncertainty, the strict robust time horizon set can be defined as $W_i^R = \cap_{s \in S} W_i^s$, which is feasible for all scenarios $s \in S$. Thus we consider decision variables y, which are defined in the set W_i^R. We then can simplify the objective function, namely (6), by cutting off the $\max_{s \in S}$ part and minimizing over those y-variables instead of the y^s-variables.

We still have to consider the same constraints, but regarding the second stage, they reduce to constraints for only *one scenario* (the strict robust one). Thus, we have to solve two assignment problems, which are decomposed in the constraints and only coupled in the quadratic objective function. Furthermore, the min-max-min-structure is reduced to a minimization problem. However, this approach requires the existence of a feasible strict robust solution. Further, we do not get the optimal recovery costs for every realized scenario, but we get *one* recovery action which is always feasible.

Since the objective function above is still quadratic in the variables, we linearize it. Therefore, we introduce new binary u-variables for substitution $u = xy$ and use the fact, that for every binary variable $x \in \{0,1\}$, $x^2 = x$ holds. We then achieve two assignment problems with a linear objective function. Furthermore the variables x_{ij}, y_{il} and u_{ijl} are linearly coupled within the constraints (modeling the fact $u = xy$).

6. Computational results for the robust and stochastic optimization approaches

The following results were obtained by the integer programming solver *Gurobi* (version 5.6). For the experiments we used a laptop with Intel i7 CPU, 4 cores (2.70 GHz) and 8 GB RAM. We tested instances of 200 aircraft which have to be assigned to 36 time windows of 600 s (10 min). The distribution of the weight classes was always 82% Medium, 11% Heavy and 7% Light. The ST_W's for all aircraft are chosen randomly, i.e. uniformly distributed. The ET_W's are assumed to be the predecessors of the ST_W's. We further assumed LT_W and maxLT_W to be 40 min (4 time windows) and 60 min (6 time windows), respectively, after ET_W. The disturbances on our ET windows were Γ-distributed with $\tau = 1.82$ windows (yields mean delay $\mu = 0.73$) and $\sigma = 1.19$ (which is reasoned in Section 7).

In this computational study we compare four approaches: *nominal*, *stochastic*, *strict robust*, and *recovery to strict robust solution*. The uncertainty set for the strict robust solutions is determined by shifting ET/maxLT by $\mu \pm k \cdot \sigma$ time windows, with $k = 1$ (note that $k = 0$ yields the stochastic approach). For each approach, we generated 100 random instances. In Table 2 we see the averaged results.

The first observation considering Table 2 is that most runtimes are very low. In fact, the value shown for the recovery to strict robust approach is influenced by 7 instances which reached the time limit of 15 min. Thus, 93 instances have significantly lower runtimes than shown in the table. The parameter *infeasible assignments* shows whether the optimal solution of the corresponding approach is still feasible after the disturbances occurred. It describes the number of aircraft that have been assigned to time windows to which they cannot be assigned in the disturbed situation. We see that the robust approaches have the least infeasible assignments. In fact, these approaches are significantly better than the stochastic approach which provides about three times the number of infeasible assignments of the strict robust approach. However, the stochastic approach still is better than the nominal one. This satisfies our expectations, because it shows that optimizing with robust or stochastic models provides more stable plans that become “less infeasible” facing disturbances. Further, the result for the ‘recovery to strict’ approach is what we expect from recoverable robustness, i.e. providing a trade-off between nominal

Table 2
Experimental results.

Approach	Runtime (s)	Average delay (windows)	Delayed aircraft (#)	Infeasible assignments (#)
nominal	3.45	0.02	23.78	12.22
stochastic	7.59	0.31	56.76	9.11
strict robust	12.76	1.29	200	3.24
recovery to strict	136.10	0.66	127.13	6.51

Instances of 200 aircraft on 10min-windows, with Γ-distributed disturbances.

delay and strict robust stability. (To actually recover the computed solutions into the strict robust scenario in case of infeasibility, we would replan 129.49 aircraft with a maximum replan distance of 1.22 time windows.) However, we also notice that even the strict robust approach still has some infeasible assignments, i.e. it is not totally robust. This results from our chosen uncertainty set, i.e. the relatively small chosen $k = 1$. Anyway, because being more robust means deleting more possible assignments, we have to choose uncertainty sets carefully.

Concerning the *average delay* value (counted in time windows) and the number of *delayed aircraft*, we observe inverse relations: these values are smaller for the nominal approach than for the stochastic one, and even larger for the robust approaches. Considering the strict robust approach, we see that all aircraft are delayed. However, this is not surprising at all due to our setting. We chose ST_W to follow the nominal ET_W. Hence, if we choose to protect us against a deviation of more than one time window (which we did), we can't assign any aircraft to its ST_W anymore. Considering the 'recovery to strict' approach, its reduced conservatism yields the possibility to choose solutions with lower delay costs (in comparison with the strict robust approach), but the recovery condition still has a price (compared to the nominal approach).

So far, in this paper we have described a mathematical approach for optimizing runway utilization in the pre-tactical planning phase. Further we have enhanced our developed optimization model by incorporating uncertainties in different ways (robust and stochastic) and discussed some computational results. In the following section, we now analyze real-world disturbances from our database from a large German airport. Finally, we describe a simulation environment to test our current and future approaches with those realistic disturbances.

7. Empirical delay data analysis and baseline simulation

For validating the new scheduling models by means of Monte Carlo (MC) simulations we start in the following Subsection 7.1 with the brief description of an appropriate stochastic delay model (see (Fürstenau et al., 2015) for additional discussion). It is first used for quantifying the statistics of empirical delay data (Subsection 7.2) and for deriving departure delay distributions, which in turn are used as the dominating stochastic disturbance for the MC-simulations. In Subsection 7.3, MC-results of baseline scheduling-simulations are presented using continuous time models and the nominal version of the new discrete MIP-based one. Initial results obtained with the stochastic and robust version are presented in a follow-up paper (Fürstenau et al., 2015).

7.1. Stochastic delay model

Understanding and modeling the statistics, dynamics, and propagation of air-traffic arrival and departure delays is a prerequisite of any attempt to optimize the punctuality of schedules and airport capacity, and minimizing necessary buffer times for required robustness of performance (e.g. Tu et al., 2008; Wong and Tsai, 2012). In the present section we provide a simple stochastic arrival and departure delay model that is tested by means of empirical delay data from a large German airport.

Empirical histograms of delay data exhibit a pronounced nonsymmetry (e.g. Tu et al., 2008) that was modeled by Wu (2010) by means of the two-parametric Beta-probability density function (limited to the open (0,1) interval).

For our purpose the family of two-parametric Gamma (Γ)-PDF's (limited to $\mathbb{R}^+$, with shape and scaling parameters a, b) appears more appropriate as analytical model because it contains the one parametric Poisson process (sometimes used for modelling interarrival delays) as a special case (Dodson and Scharcanski, 2003).

A realistic model of arrival delays, in addition to the asymmetry has to include a significant amount of early arrivals, i.e. delay $t^D<0$. Furthermore, besides the statistics of the sequence of all different arrivals a_i (different flights) during single days of operation (single day statistics) also single flight (=airline) statistics (e.g. all arrivals j of the same flight a_{ij} over a time interval of e.g. half a year) have to be modeled (Abdel-Aty et al., 2007). The delay statistics naturally exhibits daily, weekly, and seasonal periodicities and trends, i.e. nonstationary behavior. Consequently any realistic model has to be a combination of deterministic and random components (Abdel-Aty et al., 2007; Tu et al., 2008) which is one reason for the inappropriateness of the Poisson model. For taking into account early arrivals ($t^D < 0$) each histogram data set has to be transferred into $\mathbb{R}^+$ by subtracting the minimum delay (minimum earliness $t^D_{\min}$) before data fitting with the Γ-model. The Γ-PDF as a generalization of the Poisson process of inter-arrival times (t) and delays may be parametrized by the shape parameter a and mean $\tau = ab$, with standard deviation $\sigma=\tau/a$.

$$f(t;\tau,a) = \left(\frac{a}{\tau}\right)^a \frac{t^{a-1}}{\Gamma(a)} e^{-\frac{at}{\tau}} \tag{7}$$

with normalized time scale t/τ, and $\Gamma(a) = n!$ for $a \in \mathbb{N}$. For $a = 1$, (7) reduces to the Poisson case of maximum randomness, i.e. exponential t-distribution. For $a < 1$, (7) models a process with larger variance than the random process due to clustering, i.e. nonindependent clustered events. For large $a > 1$, with the Γ-PDF approaches a (τ,σ)-Normal distribution.

The analysis of empirical arrival and departure delay histograms in the following Section 7.2 together with MC computer experiments in Section 7.3 in fact indicate Γ-models to provide reasonable approximations for the arrival and departure delay statistics as one usable metric for the optimizer performance differences.

7.2. Analysis of empirical arrival- and departure delays and derivation of disturbance statistics

In this section we model the empirical arrival and departure delays of flights a_{ij} $(i = 1-m)$ with a stochastic Γ-process according to (7), with delays = random deviations from scheduled arrival times (STA, flight plan), and we derive an empirical disturbance statistics for use with the MC-simulations. As proposed by Abdel-Aty et al. (2007). we analyze and model daily delays observed within the time series of all flights a_{ij} $(i = 1 - m > 200)$ during full

Please cite this article in press as: Kapolke, M., et al., Pre-tactical optimization of runway utilization under uncertainty, Journal of Air Transport Management (2016), http://dx.doi.org/10.1016/j.jairtraman.2016.02.004

days of operation, as well as delay data from a selection of single flights a_j over a couple of months (with $j = 1 - n \geq 150$ monitored arrivals or departures).

Fig. 4 shows an example of arrival delay distribution f(ATA−STA) for a single full day of operation (ATA = AIBT). We also analysed a sample of 33 flights (different callsigns) with ≥150 arrivals each/half year (out of 1384 within 7–12/2013). The χ^2-tests of the maximum likelihood (ML) Γ-fits to the empirical delay histograms differ significantly between single days as well as between single flights. This is no surprise, of course, due to the neglection of any deterministic effect (correlations between consecutive flight arrival times or delays depending on traffic density, ATC-sequencing etc.).

For the simulations in Section 7.3 we will use departure delays as the only disturbance during the flight. This is motivated by the fact that according to Eurocontrol statistics (see Performance Review Commission, 2013) departure delays represent the main source of arrival delays. Fig. 5 depicts a summary plot of departure delay Γ-fits with normalized delay axis $\frac{t^D}{\tau}$, obtained from 46 single flights a_{ij} with ≥150 departures (j) out of 1579 analysable flights a_i from altogether 32604 departures during a 6-months time span.

The figure legend provides the fit results for the parameter estimates a,b with Γ-mean τ (identical for empirical histogram and ML-estimate), a-b correlation coefficient, and χ^2-test of Γ-hypothesis (0-hypothesis rejection for $p < 5\%$). The fit example in this case in fact formally should be rejected at the $p = 5\%$ level, basically due to the deviations around zero delay (AIBT - STA = $t^D = 0$). Besides the necessity of considering the above mentioned deterministic effects, this deviation around $t^D = 0$ can be explained by active ATC interventions to minimize delays. Nevertheless we obtained many examples without 0-hypothesis rejection, i.e. $p(\chi^2) > 5\%$. The average fit parameter estimates for the 33 single flights a_i are (±1 stddev): $\langle a \rangle = 3.5(1.3)$, $\langle b \rangle = 8.7(3.4)$, $\langle \tau \rangle = 27.5(7)$ min, with average minimum earliness $\langle t^D_{\min} \rangle = -23.9(8.8)$ min (transformation into $\mathbb{R}^+$ by $t^D_{\min}(a_i)$ for each single fit), yielding an average arrival delay of $\langle \mu^D \rangle^{\text{Arr}} := \langle \tau \rangle - \langle t^D_{\min} \rangle \approx 3.6(11)$ min, with stderror of mean $\varepsilon = 2$ min.

The corresponding average departure delay parameters (±1 stddev) are $\langle a \rangle = 2.5(0.8)$, $\langle b \rangle = 8(3.4)$, $\langle \tau \rangle = 18.2(5.4)$ min, $\langle t^D_{\min} \rangle = -10.9(4.1)$ min, yielding an average departure delay $\langle \mu^D \rangle^{\text{Dpt}} := \langle \tau \rangle + \langle t^D_{\min} \rangle \approx 7.3(6.6)$ min, with stderror of mean $\varepsilon = 1$ min. Comparing this value with the average of the 33 mean arrival delays yields the departure delays about 4 min larger. This

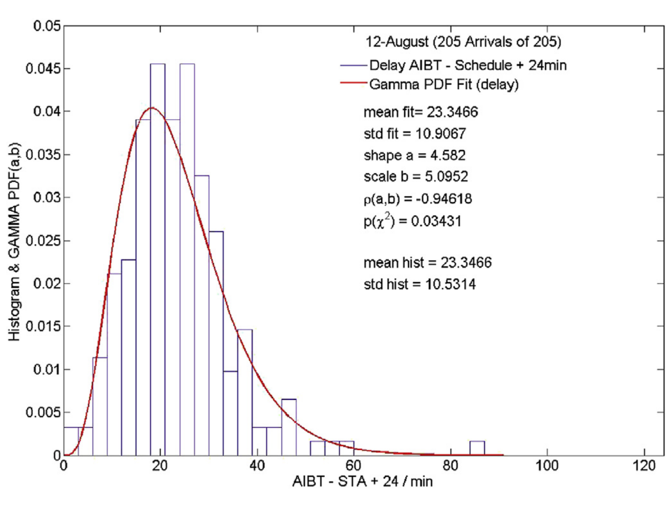

Fig. 4. Example of empirical arrival-delay histogram (AIBT - STA - $t^D_{\min}$ (earliness <0: $t^D_{\min} = -24$ min)) from the data base at a large German airport (shifted into $\mathbb{R}^+$) with Γ-PDF fits yielding a,b estimates. Full-day (17 h) traffic with 205 evaluated arrivals.

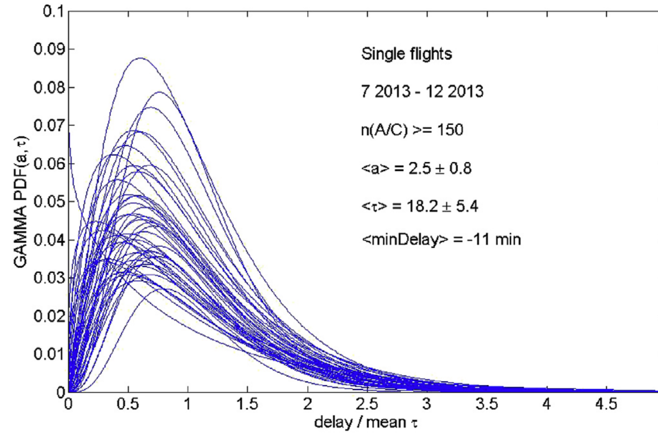

Fig. 5. Normalized (by individual Γ-mean τ) departure delay Γ-PDF fits for all departures of 46 flights with >150 departures over 6 months). For average delays the averages $\langle \tau \rangle$ of Γ-fit means (representing the empirical histograms) have to be reduced by the average earliness: $\langle \mu(\text{departure}) \rangle = 18.2 - 10.9 = 7.3$ min, $\langle \sigma \rangle = 11.9$ min.

difference compares well with statistics reported in Performance Review Commission (2013). Also the larger variation of the mean arrival delays $\sigma(\mu^D)^{\text{Arr}} \approx 7$ min as compared to the mean departure delay variation $\sigma(\mu^D)^{\text{Dpt}} \approx 5$ min compares well with PRR-results, although this may be explained partly by the different sample size. Because no sufficient empirical data for departure delays from the airports of flight origin were available we use the departure delays of the destination airport as representative departure disturbance value for the MC-simulations.

Derived from an empirical data set as used for Fig. 4, the tuple (take-off time TOT, ET, STA, LT, LTmax)$_i$ from a well defined series of 209 flights of a full single-day of traffic (17 h time span) was used for the MC-simulations of the standard traffic scenario (S1). Because the corresponding average traffic density of ca. 12 flights/h was low compared to the published capacity of 27 arrivals/h (plus 27 departures/h) we created in addition a dense scenario (S2) for testing the optimizers. The whole traffic of 209 A/C of the empirical standard scenario in this case is compressed to a reduced time span (8 h from originally 17 h, starting at 6:00) yielding a traffic density of 26 arrivals/h near the capacity limit. This was realized in such a way that all flights with arrival times $< t_0 + 8$ h remain unchanged and the rest up $t_0 + 17$ h is put in between these flights with correspondingly shifted (ET, STA, LT, LTmax)- times.

Furthermore each flight a_i is characterized by its individual weight class that determines its minimum separation distance from the previous flight a_{i-1} according to Table 1 (Section 3). Because the original scenario contained only 8 H-class A/C we increased the number (and traffic complexity) to 24 by changing those M-class with long flight distance (> 1500 km) into H-class. The modified empirical scenario (S6.2, S7.2) contained 24 (11.5%) H-class A/C, 14 (6.7%) L-class, and 171 (81.8%) M-class A/C.

7.3. Monte Carlo simulations

In this section we describe Monte-Carlo baseline simulations as foundation for the validation of the new stochastic and robust scheduling models by means of corresponding ongoing computer experiments (published in (Fürstenau et al. (2015)).

7.3.1. General aspects

For calculating and updating the individual target times TT_i for each flight a_i ($i = 1,\ldots,m$) of the full day schedule (with ET ≤ TT ≤ LT

< LTmax), the computer experiments used a simplified time-based trajectory model defined by the individual earliest and latest times of arrival (ET, LT, LTmax). For the pre-tactical phase before departure ET_i = constant, $LT_i = LTmax_i$. After the departure ET_i converges to TT_i according to $\sim \Delta t^{Sim}(TT - ET)/(TT - t^{Sim})_i$, and the interval (LT - ET_i) as function of simulation time t^{Sim} decreases linearly according to $(ET - t^{Sim})_i/2$, with some modifications during final approach <30 min before arrival which however, are not of interest within the present work.

Target Time TT_i for each simulation time step Δt^{Sim} (=4 min) is the optimization result with regard to minimizing for the whole daily arrival sequence the deviations from the individual schedules STA_i, or alternatively from ET_i. An update of optimized a_i-sequences is calculated for each Δt^{Sim}, and the daily sequence will undergo changes as long as new flights are starting from their respective departure airports, with the individual departure delay drawn from the same average Γ-PDF ($a = 2.5$, $b = 8$, $\tau = 18.2$ min; see previous section) and shifted back to the delay scale μ^D. Typically, for 17 h of daily operation of our empirical dataset we have ca. 260 simulation steps per MC-run. Runtime depends on the traffic density, time of operation and sequencing algorithm. With 200 MC-runs per experiment we typically have up to several hours of simulation time for a specific model and scenario. The simulations run on a high performance PC with 2xIntel 64 Bit E5645 12 core processors (24 cores with hyperthreading "on"), 2.4 GHz, 24 GB RAM.

7.3.2. Baseline simulations

In order to establish a baseline, the MC-simulations as a first step were performed without considering a-priori knowledge of disturbance. The simulations used the First-Come-First-Serve rule (FCFS) and a (continuous time) standard optimizer (Take Select 8–2 (Helmke, 2011)) requiring a monotonous version of the objective function with zero cost for early arrivals, and we compare it with the nominal model using the discrete (MIP) Gurobi optimizer.

Fig. 6 depicts an MC-simulation (MC057: S7.2) with the FCFS rule (i.e. no optimization) as an example for a single day (= single run) delay statistics for all flights of 8 h of operations. The figure shows the delay histogram with Γ-PDF fit that may be compared with the empirical PDF of Fig. 4. In most cases the Γ-PDF fits to the delay histograms exhibit good χ^2-test results ($p(\chi^2) > 5\%$-rejection threshold of 0-hypothesis).

A corresponding result is obtained for the single flights a_i-analysis with 200 repeated arrivals each. The runtime (± stddev) of this experiment for S6/S7 respectively was 1.7(0.6)/1.1(0.4) s. The 209 individual histograms with Γ-PDF fits for each single flight and the summary plot exhibits results similar to the single days case. The latter numbers (averages $\langle\cdot\rangle$ of fit-parameters (a,b;τ,σ), with standard deviations $(\cdot)$, times in minutes) for the 200 MC-runs (MC062(S6.2)/MC057(S7.2)) are: $\langle a\rangle = 1.7(0.2)/2.8(0.4)$; $\langle\tau\rangle = 23.2(1.2)/34(20)$; $\langle t^D_{min}\rangle = -30.2(0.5)/-29(2)$, yielding for the delay $\langle\mu^D\rangle = -7.0(1.4)/5.3(5.0)$. The average number of sequence position changes (re-scheduling) per flight a_i is $\langle rs\#\rangle = 2.2(0.9)/12.7(5.6)$. Table 3 summarizes the corresponding simulations with the (continuous time) Take Select TS8-2 optimizer and the nominal discrete Gurobi-MIP optimizer for the high density scenario S7.2.

Although for the low traffic density (S6.2) the FCFS rule yields the best results with regard to re-scheduling as well as delay statistics, for the high traffic density (S7.2) in contrast the use of an optimization algorithm appears advantageous for all three metrics: number of re-schedulings ($rs\#$), mean delay μ and (as expected) runtime RT. The general agreement on average of mean delays (μ), as obtained from single day and single flight delays proves the consistency of the analysis of the 200•209 ≈ 40000 entries MC-data tables although, for the TS8-2 optimizer, the single flight analysis (in contrast to the single day evaluation) exhibits significant inter-individual scattering. We also observe a tendency towards more symmetric PDF's (smaller skewness $2/a$) with increasing traffic density.

In our future investigations, we will test our new developed optimization methods from the previous sections within the described simulation environment and compare them with FCFS and TS8-2. The above mentioned results of the empirical data analysis and baseline simulations indicate the two-parametric Γ-PDF to be a reasonable approach for modeling the random disturbances for the validation of our approaches.

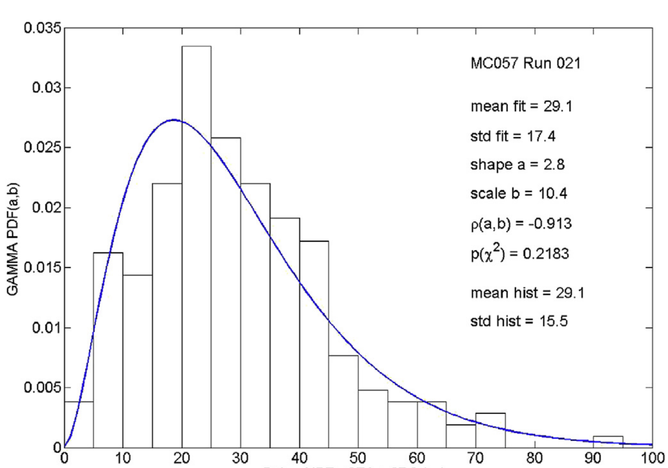

Fig. 6. Example histogram and Γ-PDF fit of baseline MC-simulations with FCFS method. Single MC-run = single day of operation (8 h, 209 arrivals) depicting statistics of the dense arrival sequence (scenario S7.2). $p > 5\%$ and low covariance $\rho(a,b)$ suggest acceptance of Γ-fit.

8. Conclusion

We have developed a mathematical optimization model for the pre-tactical optimization of assigning time windows for runway utilization. In this model, several aircraft can be assigned to the same time window which reduces the complexity of the problem. Further, we enriched the model by protection against uncertainties using techniques from robust and stochastic optimization. Our computational study showed that such an incorporation of a priori knowledge on uncertainties has a large effect on the resulting solutions. The stochastic approach optimizes the expected scenario and, therefore, is more likely to remain feasible in the face of disturbances than the nominal approach. Thus, on average it provides more stable plans and less necessary replanning. However, robust optimization methods provide even more stable solutions. Using the strict robust approach, we definitely know that a solution (if one exists) will be feasible for all scenarios within the predetermined uncertainty set. Thus, it is the approach with the highest possible stability. However, this may come at the price of increased delay. Recoverable robustness on the other hand takes into account that a time window assignment might become

Table 3
Comparison Of Baseline Results (for simulation with high traffic density).

Model(MC#)	$\langle RT(\text{std})\rangle/s$	$\langle a\rangle$	$\langle\mu(\text{std})\rangle$/min	$\langle rs\#(\text{std})\rangle$
FCFS (57)	1.1 (0.4)	2.8	5.3 (5.0)	12.7 (5.6)
TS8-2 (56)	203 (63)	3.5	3.5 (6.0)	8.1 (3.2)
Nominal MIP (87)	15.8 (1.5)	3.2	1.7 (3.9)	0.45 (0.27)

infeasible in some scenario. In that case, it applies a recovery action, i.e. a replanning step, that makes the assignment feasible again. This potentially necessary recovery is already incorporated in the computation of the initial solution. Hence, recoverable robustness provides a promising trade-off between little delay (as nominal) and high stability (as strict robust). This is also true, if we consider the 'recovery to strict robust solution' approach (under the assumption that a strict robust solution exists). This approach is way easier to solve than the general recoverable robust approach, since we don't have an exponentially large scenario set to consider and further we have an easier objective function. Our corresponding computed solutions remain quite stable while producing less delay than the strict robust approach.

We performed statistical analyses of real-world traffic data from a large German airport for deriving a departure delay model based on the two-parametric Γ-PDF to generate realistic disturbances for Monte Carlo computer experiments. Furthermore, we described a simulation environment for these experiments to test current and future optimization approaches. Results of simulations are presented with two continuous time and the nominal model using the discrete Gurobi optimizer. Results for the new stochastic and robust models are presented in a follow-up paper (Fürstenau et al., 2015).

Acknowledgment

This work is co-financed by EUROCONTROL acting on behalf of the SESAR Joint Undertaking (the SJU) and the EUROPEAN UNION as part of Work Package E in the SESAR Programme. Opinions expressed in this work reflect the authors' views only and EUROCONTROL and/or the SJU shall not be considered liable for them or for any use that may be made of the information contained herein.

We are grateful to A. Martin for many stimulating scientific discussions about mathematical optimization and for sharing valuable insights with us. Further, we are indebted to Olga Gluchshenko who prepared the initial dataset for the Monte Carlo simulations. Many thanks are due to S. Loth for paving the way to the airport data sources and to M. Helms for initial preparation of the large amount of empirical data from that source. We would like to thank H. Helmke for valuable advice throughout this work concerning the optimization aspects and to J. Rataj for continuous support.

References

Abdel-Aty, M., Lee, C., Bai, Y., Li, X., Michalak, M., 2007. Detecting periodic patterns of arrival delay. J. Air Transp. Manag. 13, 355–361. S.
Anagnostakis, I., Clarke, J.-P., 2003. Runway operations planning: a two-stage solution methodology. In: Proceedings of the 36th Hawaii International Conference on System Sciences, Honolulu, HI.
Balakrishnan, H., Chandran, B., 2010. Algorithms for scheduling runway operations under constrained position shifting. Oper. Res. 58 (6), 16501665.
Ball, M., Dahl Vossen, T., 2009. Matchings in connection with ground delay program planning. Networks 53 (3), 293306.
Beasley, J., Krishnamoorthy, M., Sharaiha, Y., Abramson, D., 2000. Scheduling aircraft landings – the static case. Transp. Sci. 34 (2), 180197.
Bennell, J., Mesgarpour, M., Potts, C., 2011. Airport runway scheduling. 4OR: a quarterly. J. Oper. Res. 9 (2), 115138.
Ben-Tal, A., El Ghaoui, L., Nemirovski, A., 2009. Robust Optimization. Princeton University Press, Princeton.
Bertsimas, D., Sim, M., 2003. Robust discrete optimization and network flows. Math. Program. 98, 49–71.
Bertsimas, D., Lulli, G., Odoni, A., 2011. An integer optimization approach to large-scale air traffic flow management. Operations Res. 59 (1), 211227.
Chandran, B., Balakrishnan, H., 2007. A dynamic programming algorithm for robust runway scheduling. In: Proceedings of the 2007 American Control Conference, New York.
Dear, R., 1976. The Dynamic Scheduling of Aircraft in the Near Terminal Area. Technical report. Flight Transportation Laboratory, Massachusetts Institute of Technology, Cambridge, Mass.
Dodson, C., Scharcanski, J., 2003. Information Geometric Similarity measurement for near-random stochastic processes. IEEE Trans. Syst., Man, Cybern. A Syst. Humans 33 (no.4), 435–440. S.
Fürstenau, N., Heidt, A., Helmke, H., Kapolke, M., Liers, F., 2014. Pre-tactical time window assignment: runway utilization and the impact of uncertainties. In: Schaefer, Dirk (Ed.), Proceedings of the SESAR Innovation Days (2014) EUROCONTROL, ISBN 978-2-87497-077-1.
Fürstenau, N., Heidt, A., Kapolke, M., Liers, F., Mittendorf, M., Weiß, C., 2015. Pre-tactical planning of runway utilization under uncertaintiy: optimization and validation. In: Schaefer, Dirk (Ed.), Proceedings of the SESAR Innovation Days (2015) EUROCONTROL. ISSN: 0770-1268.
Helmke, H., 2011. Scheduling algorithms for ATM applications – tools and toys. In: Proceeding 30th Digital Avionics Systems Conference, Seattle.
Hu, X.-B., Di Paolo, E., 2008. Binary-representation-based genetic algorithm for aircraft arrival sequencing and scheduling. IEEE Trans. Intell. Transp. Syst. 9, 301310.
ICAO Document 4444, 2007. Procedures for Air Trafic Management, fifteenth ed., pp. 69–126
Kall, P., Mayer, J., 2013. Stochastic Linear Programming: Models, Theory and Application. Springer, Wiesbaden.
Kjenstad, D., Mannino, C., Nordlander, T.E., Schittekat, P., Smedsrud, M., 2013. Optimizing AMAN-SMAN-DMAN at hamburg and arlanda airport. In: Preceedings of the SID, Stockholm.
Liebchen, C., Lübbecke, M.E., Möhring, R.H., Stiller, S., 2009. of LNCS. The Concept of Recoverable Robustness, Linear Programming Recovery, and Railway Applications. Robust and Online Large-Scale Optimization, vol. 5868. Springer, Berlin, p. 127.
Performance Review Commission, 2013. Performance Review Report PRR2012. Eurocontrol, Brussels.
Sölveling, G., 2012. Stochastic Programming Methods for Scheduling of Airport Runway Operations under Uncertainty. PhD thesis. Georgia Institute of Technology.
Soomer, M., Franx, G., 2008. Scheduling aircraft landings using airlines' preferences. Eur. J. Oper. Res. 190 (1), 277291.
Tu, Y., Ball, M., Jank, W., 2008. Estimating flight departure delay distributions – a statistical approach with long-term trend and short-term pattern. J. Am. Stat. Assoc. 103, 112–125. S.
Wong, J.-T., Tsai, S.-C., 2012. A survival model for flight delay propagation. J. Air Transp. Manag. 23, 5–11. S.
Wu, C.-L., 2010. Airline Operations and Delay Management. Farnham/GB. Ashgate.

Please cite this article in press as: Kapolke, M., et al., Pre-tactical optimization of runway utilization under uncertainty, Journal of Air Transport Management (2016), http://dx.doi.org/10.1016/j.jairtraman.2016.02.004