Computers & Operations Research 65 (2016) 189–199

Contents lists available at ScienceDirect

Computers & Operations Research

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

Reducing airport gate blockage in passenger aviation: Models and analysis

Jeremy Castaing[a,*], Ishan Mukherjee[a], Amy Cohn[a], Lonny Hurwitz[b], Ann Nguyen[b], Johan J. Müller[b]

[a] Industrial and Operations Engineering Department, University of Michigan, Ann Arbor, MI 48109-2117, United States
[b] Southwest Airlines Co., P.O. Box 36611, 2702 Love Field Drive, Dallas, TX 75235, United States

ARTICLE INFO

Available online 4 March 2014

Keywords:
Airline scheduling
Gating
Robustness
Network flow

ABSTRACT

Commercial flights are typically assigned to an arrival gate at their destination *station* (airport) prior to their departure from their origin station. Although the gate is scheduled to be available when the flight arrives, this is not always the case in practice. Due to variability in departure and flight times, the arriving flight might arrive early, the previous flight departing from the gate might depart late, or both. When a flight arrives at its scheduled gate but has to wait because the preceding aircraft is still occupying that gate, we refer to this as *gate blockage*. Gate blockage can have many negative impacts, including passenger delays, missed connections, and increased fuel burn. Our research is focused on incorporating the inherent stochasticity of the system into the planning process to reduce the prevalence and impact of gate blockage. Specifically, we formulate an optimization problem to assign flights to gates so as to minimize the expected impact of gate blockage. We use historical data to predict delay distributions and conduct experiments to assess both the computational tractability of our approach and its potential for improvement in solution quality over existing approaches.

1. Introduction

Commercial flights are typically assigned to an arrival gate at their destination *station* (airport) well in advance of their actual departure. Although the gate is scheduled to be available when the flight arrives, this is not always the case in practice. Due to variability in departure and flight times, the arriving flight might arrive early, the previous flight departing from the gate might depart late, or both.

When a flight arrives at its scheduled gate but has to wait because the preceding aircraft is still occupying that gate, we refer to this as *gate blockage*. Gate blockage can have many negative impacts, including passenger delays, missed connections, and increased fuel burn. Our research is focused on incorporating the inherent stochasticity of the system into the planning process to reduce the prevalence and impact of gate blockage.

We begin by conducting an analysis of historical data from a major U.S. carrier, examining the frequency and extent of gate blockage in practice. We demonstrate how different gate assignments can lead to different degrees of gate blockage by incorporating information about the variability in flight arrival and departure times. To leverage this, we develop mixed integer programming (*MIP*)-based models to optimize the expected outcome of the gate assignment under stochastic conditions. We then conduct empirical analyses using real-world data to show both the computational tractability of our proposed approach and the potential benefits to be achieved through incorporating uncertainty in the planning process.

Our contributions are in advancing the literature on airline gate planning by assessing the impact of stochasticity on gate blockage and in proposing MIP-based approaches to reduce this impact. We conduct a historical analysis to highlight the frequency of gate blockage. We then present two optimization based approaches to reduce this blockage by incorporating system stochasticity, using a unique network design in which gates, rather than aircraft, flow through the system. The first approach assumes that all aircrafts are compatible with all gates; this is the model that motivated our research and on which most of our computational results are based. We also briefly discuss a second approach that consider the general case where not all aircrafts are compatible with all gates. In both cases, we approximate objective coefficients to represent the probability and severity of gate

* Corresponding author. Postal address: 2200 Fuller Court, Apt. 1212B, Ann Arbor, MI 48105, United States.
E-mail addresses: jctg@umich.edu (J. Castaing), imukherj@umich.edu (I. Mukherjee), amycohn@umich.edu (A. Cohn), lonny.hurwitz@wnco.com (L. Hurwitz), ann.nguyen@wnco.com (A. Nguyen), johan.muller@wnco.com (J.J. Müller).

http://dx.doi.org/10.1016/j.cor.2014.02.011
0305-0548

blockages as a function of gate turns. We provide computational experiments based on real-world data from a major U.S. carrier to show the tractability and effectiveness of our proposed approach.

The remainder of the paper is organized as follows: Section 2 describes our approach, as well as a survey of existing literature on the gate assignment problem, robust scheduling applied to passenger aviation, and other topics relevant to our study. In Section 3 we present a historical analysis of the frequency and patterns of gate blockage. In Section 4 we present two models for solving the gate assignment problem so as to minimize the potential for gate disruption. Section 5 describes the methods used to generate the objective coefficients of these two models and Section 6 is dedicated to various computational experiments. Section 7 presents our conclusions and some ideas for future research.

2. Motivation, problem statement and literature review

2.1. Motivation

In the U.S., the majority of commercial flights depart from and arrive at physical gates at the corresponding terminal. [This is in contrast to Europe, where flights frequently arrive at *hard stands* on the tarmac, from which passengers are bussed to the terminal.] Because two flights cannot occupy the same gate at the same time, a schedule is built to ensure available gates for all flights throughout the day. This gate assignment defines a sequence of *gate turns* for each gate.

A gate turn corresponds to an aircraft that leaves a gate (a departing flight) followed by an aircraft subsequently arriving at the same gate. In between, a minimum *gate buffer* or *sit time* (e.g. 5 min) must be allotted to allow the first aircraft to clear the area before the second aircraft can reach the gate.

Note that gate turns represent an outbound flight followed by an inbound flight, whereas an *aircraft turn* is an inbound flight followed by an outbound flight that uses the same aircraft. Fig. 1 illustrates the assignment of three aircraft turns, i.e. three pairs of inbound and outbound flights $([I_k, O_k])_{1 \le k \le 3}$, to a single gate. This assignment corresponds to two resulting gate turns. The gate is occupied when the timeline is bold.

In this paper, we focus on assigning aircraft turns (which have been pre-determined in an earlier stage of the planning process) to gates. Our primary objective is to minimize the potential for gate blockage and associated disruptions. Note that other metrics may be of concern as well when assigning flights to gates, such as distance for connecting passengers or effective utilization of ground resources, and we briefly touch on these extensions in our conclusions.

Flight departures are often delayed, for a number of reasons. These include mechanical or weather related problems, ground delay programs, and delays in passenger boarding. In addition, earlier delays in the system can propagate to delay downstream flights. Likewise, there are many reasons why a flight may be early in arriving. There is always buffer built into the system to accommodate variability in departure time, outbound taxi time, flight time, and inbound taxi time. When this buffer is not needed, flights may arrive early.

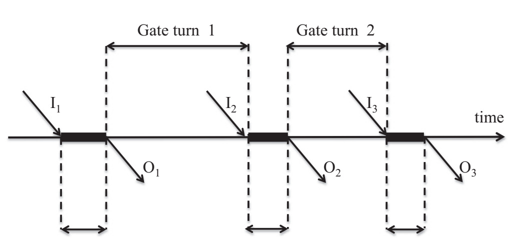

Fig. 1. Gate schedule with three aircraft turns and two gate turns.

Gate blockages can have many negative consequences. At a minimum, they inconvenience and frustrate the passengers on board the blocked aircraft. Blockages can also lead to passenger delays (which can lead to missed connections) and propagation of crew and aircraft delays. Furthermore, gate blockages lead to excess fuel burn (with both financial and environmental impacts), increased crew costs, and disruption to the planned utilization of ground resources. Finally, the presence of excess aircraft on the tarmac can lead to increased congestion, which in turn cannot only cause more ground delays, but has implications for passenger safety and aircraft damage as well.

Example: Consider two outbound flights O_1 and O_2 as well as two inbound flights I_1 and I_2. Suppose that scheduled departure times are 8:30 for O_1 and 8:40 for O_2 and that scheduled arrival times are 8:50 for I_1 and 9:00 for I_2.

One possible gate assignment is to pair O_1 and I_1 in one gate and O_2 and I_2 in another gate. This assignment is reasonable since it allows a 20 min gate turn time for both outbound-inbound pairs. Suppose, however, that based on historical data, we know that inbound flight I_1 often arrives late whereas inbound flight I_2 frequently arrives early. We further assume that outbound flights O_1 and O_2 leave consistently on time. In that case, a more robust gate assignment would be to pair O_1 and I_2 in one gate and O_2 and I_1 in another gate.

2.2. Problem statement

Given a set of aircraft turns (each defined by either an inbound flight followed by an outbound flight using the same aircraft, an outbound flight that is the day's first use of a given aircraft, or an inbound flight that is the day's last use of a given aircraft) and a set of gates for a single station, assign each aircraft turn to a gate, so as to maximize robustness, under the constraint that a gate can handle at most one aircraft at any given time.

We consider four different objective functions representing four different measures of robustness:

- *Expected number of blockages* (*objective P*): The first objective we consider is to minimize the *expected number of gate blockages*. To compute this, we have determined (based on historical averages) the probability of experiencing gate blockage associated with each possible gate turn. The objective is to then minimize the sum of these probabilities across all gate turns assigned in the optimal solution.
- *Expected total time of blockage* (*objective X*): One limitation of the first objective metric is that it ignores the duration of the blockages, weighting a short blockage no differently than a long blockage. Therefore, in our second metric, we minimize the *expected total time of blockage*.
- *Expected connecting passenger blockage minute* (*objective C*): The third objective is motivated by the fact that the impact of blockage time is not necessarily linear. A 20 min gate blockage imposed on a terminating passenger might have far less impact than a 10 min gate blockage imposed on a passenger with a very tight connection. As a first approximation to capture this, in our third measure of robustness we focus specifically on the gate blockage imposed on connecting passengers. Specifically, for each outbound/inbound flight pair forming a potential gate turn, we take the expected length of blockage for the flight pair and weigh this by the average number of connecting passengers on the inbound flight.
- *Worst case expected blockage* (*objective W*): Finally, as our fourth measure of robustness, we minimize the *worst case expected blockage* – the longest expected blockage that any of the assigned turns would experience. This effectively sets a cap on the longest

gate blockage, thereby recognizing the non-linear impact of gate blockages and striving to keep all blockages at a low level.

2.3. Literature review

Despite the potential benefit to be gained by making gate planning decisions more robust, there has been limited attention paid in the literature to this problem. In one example, [1], robust gating and propagation of delays for a multi-station network are considered. Another example, [2], takes a stochastic programming approach, focusing on a single station with multiple distributions for incoming flight delays. Perhaps closest to our research is the example of [3], which minimizes the expected number of gate blockages, given a formula to predict the distribution of gate blockage between a given outbound and inbound flight pair.

There has been a bit more research on other aspects of gating in passenger aviation. For example, [4] consider gate planning from the perspective of minimizing passenger walking distance and connection times. Chen et al. [5] compare the performance of three meta-heuristic algorithms for solving the gate assignment problem, focusing on resource utilization and passenger satisfaction, and [6] design another heuristic using stochastic optimization to solve the gate assignment problem. A stochastic optimization approach to the problem has also been proposed by [7].

Other areas of robust planning in passenger aviation have been studied more extensively, including flight scheduling [8–11]; scheduling and routing [12]; fleet assignment [13]; aircraft routing and maintenance planning [14–16] as well as crew scheduling [17–19].

Closely related to the issue of robust planning (which attempts to prevent disruptions before they occur, or to mitigate the impact of disruptions) is the issue of recovery and passenger re-accommodation (which addresses disruptions after they have occurred, to minimize their impact). There is an extensive literature in this area as well. Examples include [20–22], who consider the recovery of aircraft schedules during periods of disruption; [23,24], who look at the relationship between airline plans and the potential for delay propagation; [25,26], who consider crew recovery during irregular operations; [27,28] consider passenger recovery; [29], who consider maintenance schedule re-planning; and [30,31], who provide surveys of airline disruption management.

Finally, we conclude by suggesting a number of useful survey papers for the novice reader unfamiliar with passenger airline planning and operations: [27,28,32,33].

3. Case study for historical data analysis

To demonstrate the importance of our proposed approach to incorporating stochasticity in gate planning, we begin with a historical analysis of the frequency and patterns of gate blockage today. In particular, we want to answer the following questions:

- How often does gate blockage occur?
- When gates are blocked, how long is the blockage?
- Does it vary by station?
- Does it vary by time period?

The answers to these questions will serve as a motivation for the rest of the paper, where we seek to reduce gate blockage by incorporating stochasticity in the planning process.

3.1. Methodology

Our analysis focuses on the domestic operations of a single, major U.S. carrier. We summarize the data below, which has been loosely disguised per the request of the carrier. In our analysis, we consider four time periods corresponding to four different flight schedules (note that the length of these periods varies):

- Period 1 – January 2009 – 22 days.
- Period 2 – January 2010 – 31 days.
- Period 3 – December 2010 to February 2011 – 90 days.
- Period 4 – December 2011 to February 2012 – 91 days.

For each of these periods, we evaluate a panel of 15 of the largest airports in the carrier's network. These stations are described in Table 1.

The carrier also provided us with information for each flight including scheduled and actual departure from origin and arrival to destination, as well as the origin and destination gates that were used.

We do *not* have access, however, to explicit gate blockage because it is not readily available in the carrier database. Thus, to conduct our analysis, we have to reverse-engineer the available data to estimate gate blockages.

To do so, we first note that an upper bound on the gate blockage associated with a given inbound flight can be found by subtracting the time that the flight landed (*wheels on*) from the time that it reached the gate. Some of this time, however, will be necessary taxi time (i.e. the time that it takes to physically travel from the runway to the gate) and some of it may be delayed in taxi that are caused by something other than a blocked gate (e.g. congestion on the tarmac). We have therefore chosen to approximate gate blockage in the following way.

First, for each station during each time period, we calculate a nominal taxi time that we define as the median of all taxi times. We choose the median as a way of disregarding longer taxi times that are caused by external factors such as airport congestion, disruptive weather conditions, and in fact the presence of gate blockages themselves. At first glance, it seems that we should use the smallest observed taxi time as a nominal taxi time. However, we have chosen median instead of using a lower percentile in recognition of the fact that (a) different flights are going to different gates, which will introduce some variability into the nominal taxi time and (b) flights are also coming in from different runways and under different airport configurations.

Fig. 2 presents the distribution of taxi times for over 10,000 flights arriving at a single station. As expected, most of the flights have a taxi time of between 2 and 4 min – 3 min being the median – and the distribution has a tail containing a few flights with much longer taxi times, due to the reasons mentioned above.

Table 1
Main characteristics of the airports in our panel.

Station	Average number of flights per day	Average flights per gate per day
1	200–210	8.9
2	180–190	6.1
3	160–170	5.6
4	140–150	7.4
5	130–140	8.0
6	110–120	5.3
7	100–110	8.0
8	90–100	6.5
9	80–90	7.4
10	60–70	6.8
11	50–60	7.0
12	40–50	6.7
13	30–40	3.1
14	10–20	3.9
15	10–20	1.6

Next, for each flight on each day in the time period, we take the wheels on time and add to this the nominal taxi time. This is our estimated gate arrival time.

We then consider the flight departing from the same gate prior to the arrival and add to its actual departure time the minimum gate buffer time to determine when the gate became available.

If the estimated gate arrival time is earlier than the gate available time, then we record the difference as a gate blockage.

Note that we do not consider the *actual* arrival time of the flight to the gate, i.e. if the arrival time of the flight to the gate is later than the gate available time, we do not count that interval in the blockage. This is because we assume that other factors must have caused this delay.

Fig. 3 presents two possible outcomes of a gate turn, the first one without gate blockage and the second one with gate blockage.

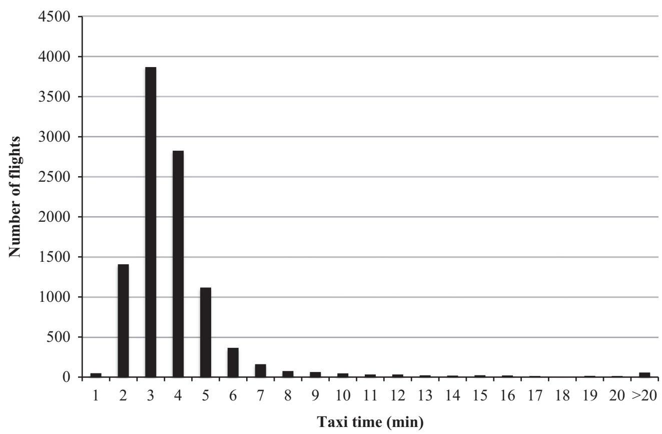

Fig. 2. Distribution of taxi time at one station.

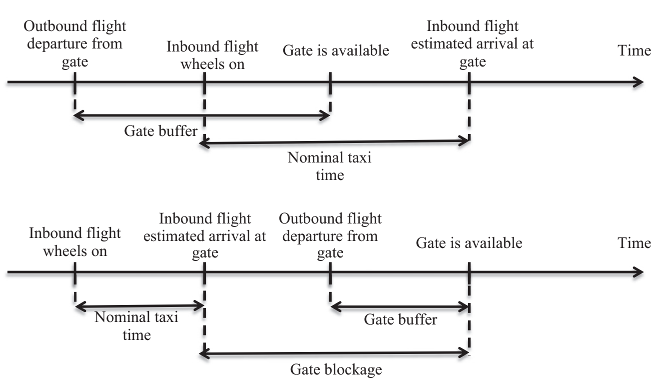

Fig. 3. Two possible outcomes of a gate turn.

3.2. Analysis

We study the frequency and the duration of all gate blockages (both those caused by late outbound departures and those caused by early inbound arrivals) for the four periods and 15 airports previously described. We begin with Table 2, a detailed summary of the results obtained for Period 3 which is the period we used to run our computational tests (see Section 6). The three last lines are the total across all stations for each column as well as the percentage of flights and the percentage of blocked flights in each bucket.

The fields in Table 2 have been computed as described below:

- The frequency of gate blockage: total number of gate blockages/ total number of arriving flights.
- The conditional average length of a gate blockage: total sum of gate blockage minutes/total number of gate blockages.
- The total number of flights whose blockage length is in the intervals (in minutes): [1,5], [6,10], [11,15], [16,20], and greater than 20 min.

We observe that although gate blockages are fairly rare on a percentage basis (averaging about 5% of all flights), in absolute terms this is a significant number, when you take into account the fact that there are roughly 25,000 domestic flights across the U.S. each day.

Note also that, although most gate blockages are of duration between 1 and 15 min, a significant number is of longer duration (about 20%). Furthermore, note that for connecting passengers a delay in arrival caused by a gate blockage of even 15 or 20 min may be sufficient to cause a missed connection.

Finally, we note a limitation in our analysis: in the case of extreme weather (e.g. thunderstorms impacting the airport), we may show a lengthy departure delay on an outbound flight that we compute to cause a lengthy gate blockage for the corresponding inbound flight. If ground operations are halted, however, then that inbound flight would have been delayed from reaching the gate even if the gate were unoccupied. In those extreme cases, our analysis may over-estimate the gate blockage.

Table 2
Detailed results for Period 3.

Station	Frequency of blockages (%)	Conditional average length of blockage	[1;5]	[8;10]	[11;15]	[16;20]	> 20
1	7.29	10.10	370	399	276	135	160
2	4.17	8.66	224	215	142	59	51
3	4.19	8.05	222	203	120	54	34
4	6.84	9.43	272	278	170	95	96
5	5.70	11.04	189	199	119	74	112
6	5.36	9.31	169	180	113	57	48
7	6.48	9.37	200	184	96	62	63
8	2.89	6.85	107	82	38	18	9
9	3.01	7.69	99	82	38	18	9
10	4.43	8.73	97	70	47	37	19
11	3.11	7.27	60	54	27	8	8
12	4.83	9.68	51	50	39	16	19
13	3.45	8.79	41	28	16	11	11
14	6.63	17.71	15	8	21	19	30
15	2.57	12.29	9	8	3	6	8
Total period	5.10	9.39	2125	2040	1262	662	681
Percentage of flights			1.60%	1.54%	0.95%	0.50%	0.51%
Percentage of blocked flights			31.39%	30.13%	18.64%	9.78%	10.06

Table 3 summarizes across all stations within a given time period. Observe that the results do not vary much from one time period to another.

In Fig. 4, we show the breakdown by length of all gate blockages, across all 15 stations for all four time periods. We break down blockages by 5 min intervals. The y-axis shows the percent of flights falling into each bucket, relative to the total number of flights flown; each column specifies the absolute number of gate blockages as well as the percent of gate blockages that fall into that bucket. Note that roughly one third of the gate blockages is more than 10 min in duration and almost 10% are of more than 20 min, corresponding to almost 1500 flights.

In Fig. 5, we break down the flights by time period. Observe that the percent of overall flights delayed appears to go up slightly between the first two periods and the second two periods; the distribution of gate blockages across their respective lengths remains roughly comparable across all time periods, as shown in Table 4.

Figs. 6 and 7 display the evolution over time of the probability and expected length of blockage for each station and over the four periods. In Fig. 5, each bar is the percentage of flights blocked for each of the 15 airports during each of the four considered periods of time. In Fig. 7 each bar is the average length of a blockage, conditioned on blockage occurring. Unlike the distribution of blockage length, the proportions of flights blocked in each airport have significant fluctuations over time (Fig. 6); for instance, the percentage of flights blocked in station 12 is two times larger during Period 1 than the other periods.

Finally, we look for possible correlations between airport characteristics (number of flights, gates, etc.) and the occurrence of gate blockage. For example, when there are more flights per gate, gate turns are tighter, and this suggests a higher likelihood of gate blockage. Figs. 8 and 9 present the probability of gate blockages at each station versus the daily ratio of flights per gate, in Periods 3 and 4 which represent the most flights.

We observe that stations at the tails with the lowest flight to gate ratio have the lowest probability of blockage, and similarly the highest ratios have the highest probabilities, which is not surprising. We do not observe a strong correlation in general, however, suggesting that many other factors beyond the amount of buffer in the gate turn impact potential for gate blockage.

This case study allows us to assess the frequency and the duration of gate blockages in a panel of US airports, over different time periods. The key results of this analysis are that (1) gate blockages affect roughly 5% of flights, (2) their duration is smaller than 10 min for 60% of them but around 9% of the blockages last more than 20 min, (3) the frequency and the length of gate blockages are highly variable depending on the stations, however they are similar from one period of time to another.

These observations show that gate blockages have a significant impact on daily airline operations, and motivate us to take into account gate blockages when building a gate assignment, which is the objective of the next sections.

4. Robust gate assignment

Motivated by the analysis presented in the previous section, we have developed a mathematical programming-based approach to the gate assignment problem, with the goal of improving solution robustness by incorporating variability in departure and arrival times. We first consider the case where all aircraft types (and thus all flights) are compatible with all gates; we refer to this as the *homogeneous* case. Then we generalize to the *heterogeneous* case, where certain gates are incompatible with certain aircraft types and thus the corresponding flights.

4.1. Robust Homogeneous Gate Assignment

To model the *Robust Homogeneous Gate Assignment* (*RHoGA*) *Problem*, we consider a network flow-based formulation, as is commonly used in airline planning. The key difference in our approach, however, is the perspective: *rather than flowing aircraft*

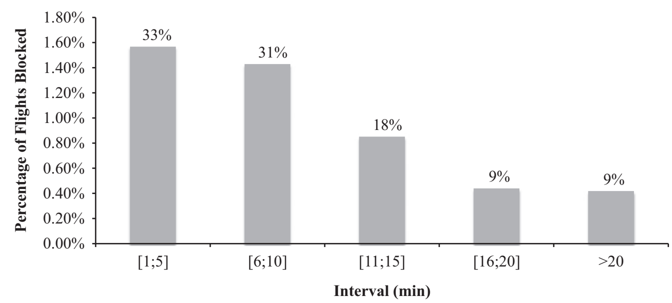

Fig. 4. Distribution of blockage length over 15 airports and four periods of time.

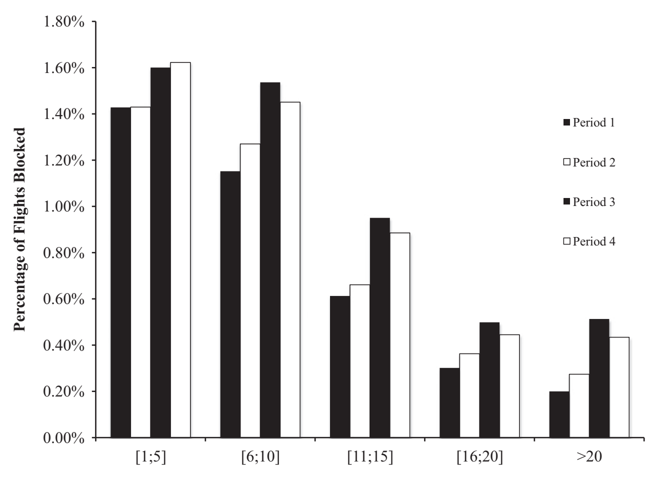

Fig. 5. Distribution of blockage length over 15 airports for each period of time.

Table 3
Results for all the periods.

Period	Frequency of blockages (%)	Conditional average length of blockage	[1;5]	[8;10]	[11;15]	[16;20]	> 20
Period 1	3.69	7.69	450	363	193	95	63
Period 2	4.00	8.28	772	686	357	196	148
Period 3	5.10	9.39	2125	2040	1262	662	681
Period 4	4.83	8.86	2192	1960	1196	601	586
All periods	4.71	8.92	5539	5049	3008	1554	1478
Percentage of flights			1.57%	1.43%	0.85%	0.44%	0.42%
Percentage of blocked flights			33.31%	30.36%	18.09%	9.35%	8.89%

Table 4
Average distribution of blockage times.

Blockage length (min)	[1,5]	[6,10]	[11,15]	[16,20]	> 20
Average percentage of blockage in that interval (%)	33	31	18	9	9

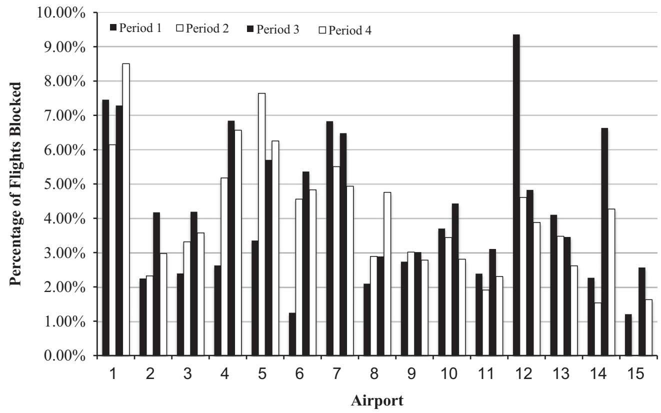

Fig. 6. Percentage of flights blocked in each station.

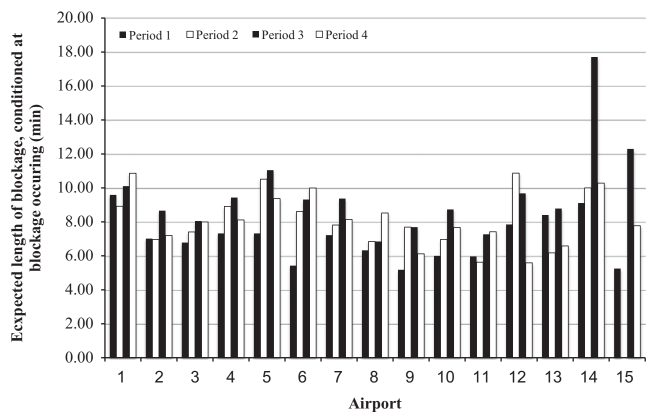

Fig. 7. Conditional expected length of blockage.

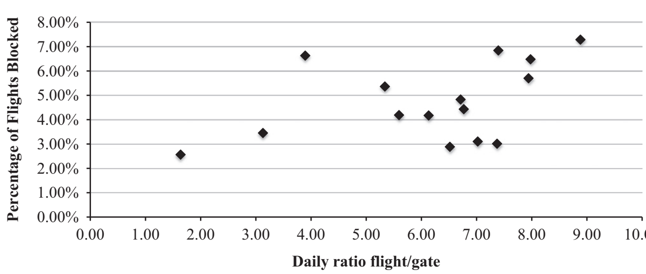

Fig. 8. Percentage of flights blocked in each station as a function of the daily number of flights per gate in Period 3.

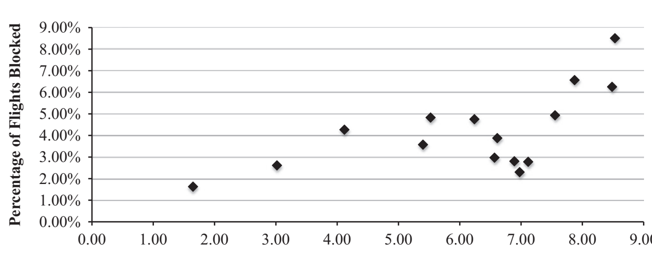

Fig. 9. Percentage of flights blocked in each station as a function of the daily number of flights per gate in Period 4.

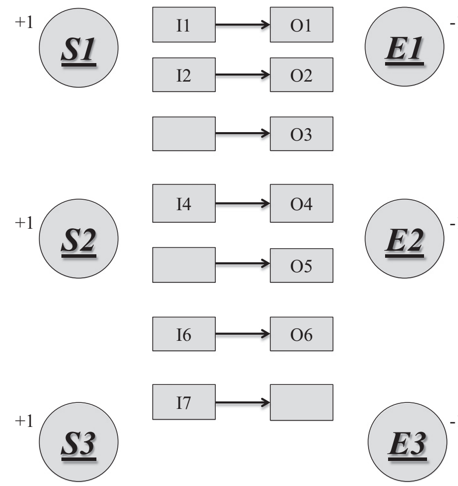

Fig. 10. Nodes of the network.

through gates or stations, as is commonly seen, we flow gates through aircraft turns. Figs. 10–13 represent a portion of a sample network.

Fig. 10 depicts the nodes in this network. There is one node (S_g) for each gate g representing that gate at the start of the day with a supply $d_g=1$ and one node (E_g) for each gate g with a demand $d_g=-1$ representing that gate at the end of the day. In addition, there is one pair of nodes for each aircraft turn a without any supply ($d_a=0$). Note that some turns consist of both an inbound and an outbound flight, while some represent just an outbound flight (where the aircraft would have overnighted at the station the preceding night) and some represent just an inbound flight (where the aircraft is intend to stay overnight at the station). We create an arc with lower and upper bounds of 1 between the inbound and outbound parts of the nodes, which ensures that each aircraft turn is assigned to exactly one gate.

Figs. 11–13 depict the arcs in this network. Fig. 11 shows arcs that originate from the gate start nodes. Generally, there is one arc from each gate to each aircraft turn; flow over this arc corresponds to assigning that turn as the gate's first activity of the day. If a specific flight is pre-determined to be the first flight of the day out of a given gate (for example, when using the model in an operational context, where last night's gate occupants are known), then there would only be one corresponding arc in the network to represent this – for example, in Fig. 10, the arc from S_2 to O_3 and from S_3 to O_5.

Fig. 12 depicts arcs between aircraft turn nodes. Generally, there is an arc from aircraft turn T_1 to aircraft turn T_2 so long as the departure time O_1 plus the minimum buffer time of the gate is earlier than the arrival time I_2. [Note that this can be relaxed in planning mode to help *determine* minimum buffer gate time – for example, if an outbound flight often leaves early and an inbound flight often arrives late, pairing their respective turns may be desirable, even if they are closer together in time than the system minimum.] If an aircraft turn is an outbound flight only, then it cannot have inbound arcs from other aircraft turns, as it is presumed to be the first flight of the day from a gate. Similarly, if an aircraft turn is an inbound flight only, then it cannot have outbound arcs to other aircraft turn nodes, as it is presumed that the aircraft will remain on the ground overnight.

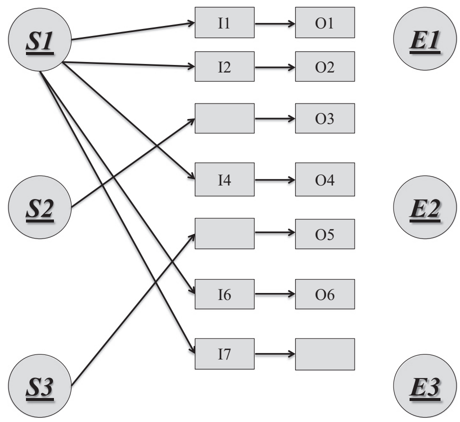

Fig. 11. Arcs leaving the gate start nodes.

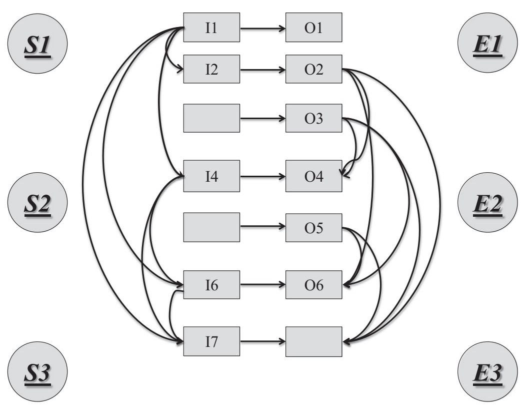

Fig. 12. Arcs between aircraft turns.

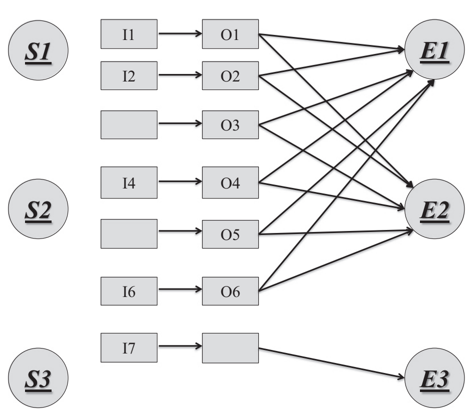

Fig. 13. Arcs arriving to gate end nodes.

Finally, Fig. 13 depicts arcs into the gate end nodes. There is generally one arc from each aircraft turn node to each gate end node; flow on this arc represents that aircraft turn being the last activity of the day at that gate. If a flight has been pre-assigned to a specific gate to overnight, then that would be the only arc into the gate end node, as illustrated by the arc from I_7 to E_3.

[Note that, in theory, we could also include arcs from S_g to E_g, representing the case in which gate g is unused throughout the day. In practice, such an occurrence rarely happens.]

Clearly, a path through this network, starting from gate S_g and ending at E_g, is a valid sequence of activities to be assigned to gate g.

The resulting model is a pure min-cost flow problem and therefore has the following structure:

$$\min_x \sum_{(a,b)\in A} c_{ab}x_{ab} \quad (1.1: \text{Objective})$$

$$\text{subject to} \quad \sum_{b:(a,b)\in A} x_{ab} - \sum_{b:(b,a)\in A} x_{ba}$$

$$= d_a \quad \forall a \in N \quad (1.2: \text{Flow balance})$$

$$x_{ab} \ge l_{ab} \quad \forall (a,b) \in A \quad (1.3: \text{Lower bound})$$

$$x_{ab} \le u_{ab} \quad \forall (a,b) \in A \quad (1.4: \text{Upper bound})$$

where N and A are the sets of nodes and arcs of the network described above.

The variable x_{ab} represents the flow along arc (a, b). Having a flow of 1 along an arc means that the two adjacent nodes belong to the same gate schedule.

The supply and demand d_a are set according to the description above (1 for the start of the day nodes, -1 for the end of the day nodes and 0 for all the aircraft turns nodes).

The bounds l_{ab} and u_{ab} are set to 0 and 1 on any arc with the exception of the arcs resulting from the splitting of the aircraft turns nodes whose lower and upper bounds are both 1 in order to ensure that each aircraft turn is assigned to exactly one gate.

The cost per unit of flow along arc (a, b) is set according to the estimated probabilities of blockage and the associated metrics (see Section 5).

Finally notice that we do not require a binary constraint on each variable x_{ab} since the arc incidence matrix is totally unimodular and therefore we will get an integral optimal solution.

The model above can be applied to the first three objective functions that we have outlined.

For the fourth (objective (W)), we must modify the problem slightly. Specifically, we define a new variable $z \in R$, which represents the maximum expected blockage. We then add one constraint for every aircraft turn arc x_{ab} of the form:

$$z \ge c_{ab}x_{ab}$$

Because we are minimizing z, the constraints effectively impose that $z = \max_{(a,b)\in A} c_{ab}x_{ab}$. Note that in this case we have now violated the total uni-modularity of the matrix and therefore we also have to impose integrality requirements on the flow variables.

4.2. Robust Heterogeneous Gate Assignment

In the previous subsection, we assumed that all flights (or, more precisely, all aircraft turns) could be assigned to all gates. In practice this is often not the case. For example, certain gates are not equipped to handle either very large or very small aircraft. Therefore, we generalize *RHoGA* to take this into account, defining the *Robust Heterogeneous Gate Assignment* (*RHeGA*) *Problem*.

Specifically, given a set of gates, a set of aircraft turns (as defined in the previous subsection), and the added component of a *compatibility matrix*, which defines for each gate and aircraft turn pairing whether the associated assignment is feasible, the objective is to assign each aircraft turn to a compatible gate, so as to maximize robustness.

We use a similar network structure, where gates rather than aircraft flow through the network, as in *RHoGA*. However, we can no longer model the problem as a pure minimum cost flow problem, because the gates are no longer fully interchangeable, i.e. they cannot be treated as commodities, because not all flights can be assigned to all gates.

Instead, we create one copy of the network *for each gate*, similar to the network from RHoGA (but without splitting the aircraft turn nodes). We remove from this network, however, all nodes corresponding to aircraft turns that are not compatible with the gate and all associated arcs going into or coming out of those nodes. Each gate-specific sub-network now again captures feasible paths (i.e. sequences of activity).

We then still need to ensure that all aircraft turns get assigned to exactly one gate. To do so, we add one constraint for each aircraft turn ensuring that the flow into the node representing that aircraft turn, across all arcs in the sub-networks for all gates, must equal one. Observe that we have, in the process, violated the pure minimum-cost flow structure. Therefore, we must now add integrality requirements – the flow on all arcs must be non-fractional to ensure feasible paths, i.e. gate-specific sequences of tasks.

The resulting RHeGA model is a network flow problem with side constraints, which can easily be viewed as a multi-commodity flow problem.

$$\min_{x} \sum_{g \in G} \sum_{(a,b) \in A} c_{ab} x_{ab}^{g} \quad (1.1: \text{Objective})$$

$$\text{subject to} \sum_{b:(a,b) \in A} x_{ab}^{g} - \sum_{b:(b,a) \in A} x_{ba}^{g}$$

$$= d_a \quad \forall a \in N \ \ \forall g \in G \quad (1.2: \text{Flow balance})$$

$$x_{ab}^{g} \geq l_{ab} \quad \forall (a,b) \in A \ \ \forall g \in G \quad (1.3: \text{Lower bound})$$

$$x_{ab}^{g} \leq u_{ab} \quad \forall (a,b) \in A \ \ \forall g \in G \quad (1.4: \text{Upper bound})$$

$$\sum_{g \in G} x_{ab}^{g} = 1 \quad \forall (a,b) \in A \quad (1.5: \text{Side constraint})$$

$$x_{ab}^{g} \text{binary} \quad \forall (a,b) \in A \ \ \forall g \in G \quad (1.6: \text{Integrality})$$

5. Computing coefficients

The two models presented in Section 4 require objective coefficients that capture the probability that a gate blockage occurs between two given aircraft turns.

As is the case with virtually all airline planning problems, identifying the appropriate data to populate our models is a non-trivial challenge. Historical data can be used but there may not be adequate data about past events to predict future occurrences. This is particularly true when making decisions about a future schedule for new flights that have not been included in prior schedules. Furthermore, historical data may not accurately reflect the future, with potential system changes having significant impact. Nonetheless, with these caveats in mind, approximations of objective parameters must be made. We do so by using historical data to predict how alternate schedules might have performed. We note that an important area of future research is to work towards improving these parameter estimates.

To predict the probability of aircraft turn T_1 imposing gate blockage on aircraft turn T_2 (for objective P), and the expected amount of this blockage (for objective X), we rely on historical data. Specifically, we consider all days during which both flights operated from a common scheduling period. We consider only day-specific flight pairs so that the correlation of weather impacts will be incorporated (for example, if T_1 is delayed in departing due to local inclement weather, it is more likely that T_2 will be delayed in arriving as well, and this should be recognized in our approximation).

Consider two possibilities. First suppose that T_1 and T_2 did in fact share a common gate on day d during our historical period. Was there gate blockage on this day and, if so, for how long? We know from the carrier-provided data what time T_1 pushed back from the gate (d_1) and therefore by adding the minimum buffer gate time b when the gate was available (d_1+b). We also know when T_2 landed (l_2), and we know when T_2 arrived at the gate (a_2). What we do not know, however, is how much of the window from l_2 to a_2 was taxi time and how much (if any) was gate blockage. To try to deconstruct this, we consider the default nominal taxi time $\tilde{T}$ defined in Section 3, such that $l_2+\tilde{T}$ is the *estimated* time of arrival at gate for aircraft turn T_2. Therefore, we assume that the remaining time $T_{2,1}^{blockage} = (d_1+b-l_2-\tilde{T})$ was gate blockage if positive and that there was not any gate blockage otherwise.

Second, suppose that aircraft turns T_1 and T_2 did *not* share the same gate on day d. We still need to know what would have happened if they had shared a gate since in our model we consider all possible assignments. Since we consider a nominal taxi time our approximation of the arrival time at the gate for an inbound flight does not depend on its gate, therefore we can apply the exact same reasoning as in the first case to obtain an approximation of an eventual gate blockage that could have happened if the two flights had shared the same gate.

Once we know how to compute an estimation of the gate blockage length between any pair of aircraft turns, we just need to loop through the flight data provided by the carrier, compute the estimated gate blockage length for each day on which those two aircraft turns occurred (even if they were not assigned to the same gate) and calculate the cost coefficient relative to this pair for each one of the four objectives described in Section 2.

6. Computational experiments

The purpose of our computational experiments is two-fold. First, we want to assess the tractability of our approaches, to assess if they are viable for use in practice, in planning as well as in operational contexts. Second, we want to analyze the potential benefit to be gained by using optimization-based techniques to build gating schedules.

In Section 6.1 we focus on the homogeneous problem, where all aircraft types (and thus all flight turns) are compatible with all gates. Section 6.2 addresses the heterogeneous problem, where certain flight turns are incompatible with certain gates.

6.1. Homogeneous experiments

For our homogeneous experiments, we focused on the aircraft turn data from one specific date, as provided by the carrier. We considered five stations (2, 3, 4, 5 and 6), which were among the largest in the network.

Using historical data to generate objective coefficients (as described in Section 5), we created four different schedules for each station, optimized under four different objective functions. Specifically, we created:

- (*optP*): the schedule that minimizes the sum of the probabilities of a gate blockage (i.e. the expected number of gate blockages),
- (*optX*): the schedule that minimizes the sum of the expected blockage minutes,
- (*optC*): the schedule that minimizes the sum of the expected connecting passenger blockage minutes,
- (*optW*): the schedule that minimizes the maximum expected blockage length.

In addition to these four schedules, we also constructed (FIFO), a schedule that assigns flights to gates in a first-in-first-out order. Fig. 14 provides the results. Each row corresponds to a different schedule. Each column corresponds to a different objective function. For example, for station 2, the schedule that minimizes the sum of the expected blockage minutes (schedule *optX*) has an objective of 2.071 under the objective W: maximum expected blockage.

For each column, it is of course true that the best value corresponds to the schedule which was optimized relative to that objective. It is interesting to note, however, that the optimal

2	P	X	C	MM
optP	0.505	33.714	959.999	11.464
optX	1.242	10.221	196.941	2.071
optC	3.332	85.875	140.019	25.043
optMM	1.888	20.921	635.745	1.25
FIFO	2.507	124.612	3597.291	15.142

5	P	X	C	MM
optP	1.198	58.779	1417.905	9.033
optX	1.655	28.772	751.391	4.033
optC	2.89	105.455	433.691	24.566
optMM	3.021	62.758	1607.543	4.033
FIFO	3.707	128.609	3589.717	12.034

3	P	X	C	MM
optP	0.535	19.748	535.71	4.866
optX	1.102	11.961	457.372	3.071
optC	2.641	64.605	303.571	18.068
optMM	1.44	21.566	698.03	1.689
FIFO	4.131	136.061	3795.56	15.892

6	P	X	C	MM
optP	1.736	90.327	1922.921	13
optX	2.427	57.017	1344.787	8.172
optC	4.098	113.491	622.915	13.851
optMM	4.323	101.23	2033.108	5.407
FIFO	4.345	184.431	4819.237	18.6

4	P	X	C	MM
optP	0.14	10.813	256.344	4.655
optX	0.185	3.343	17.994	2.48
optC	1.817	48.549	5.571	25.32
optMM	0.304	5.817	26.43	1.75
FIFO	1.668	82.61	1918.514	8.481

Schedules:
optP: optimal schedule under P objective
optX: optimal schedule under X objective
optC: optimal schedule under C objective
optMM: optimal schedule under MM objective
FIFO: First In First Out schedule

Fig. 14. Results of the homogeneous model.

schedule for the metric *X*: sum of the expected block time for each turn, performs very well under the other costs – in fact, in almost every case, for any given metric, the *optX* schedule is second only to the schedule optimized relative to that objective function. Our intuition to support this observation is that the *X* metric is a good trade-off between the other three objectives. The FIFO schedule is significantly worse than all other schedules for all metrics in almost all cases.

6.2. Heterogeneous range experiments

We have noted that the homogeneous model is a pure minimum cost flow formulation, which naturally has integrality properties, while the heterogeneous model has side constraints that can induce the need for branching.

To assess the pure impact of the formulation itself on computational performance, we consider the situation of no compatibility restrictions on the aircraft type (all flights can use all gates), and find the optimized schedule under the objective (*X*), on a specific date for station 5 using the two models. As expected since we do not have any gate restriction, we find the same optimal objective: 28.772 min. The run time for the homogenous model is 1 s and the run time for the heterogeneous model is 60 s. It is interesting to note that the difference in structure of the two models results in a much longer computational time for the heterogeneous model. All runs were conducted on a Intel Xeon E31230 computer clocked at 3.20 GHz and 8 GB DDR3 RAM clocked at 1333 MHz solved with CPLEX version 12.1.

We then seek to understand how the level of incompatibility between gates and aircraft types impacts performance (both for the run time and the optimal objective). To do so, we design the following *heterogeneous range experiment*: we consider three different aircraft types: types *S*, *M* and *L* corresponding to small, medium and large aircrafts and representing 20%, 4% and 76% of the aircraft turns. We assume that aircraft of type *M* can go to any gate, but that aircraft types *S* and *L* can be constrained. Under that assumption there are three possibilities for a gate:

- ([*S*,*M*]): it is compatible with only types *S* and *M*,
- ([*M*,*L*]): it is compatible with only types *M* and *L*,
- ([*S*,*M*,*L*]): it is compatible with all three aircraft types.

We consider station 2; this airport has 19 gates. The different possible gate constraints are represented in Fig. 15. An entry at coordinates (x, y) is equivalent to a scenario in which x gates are

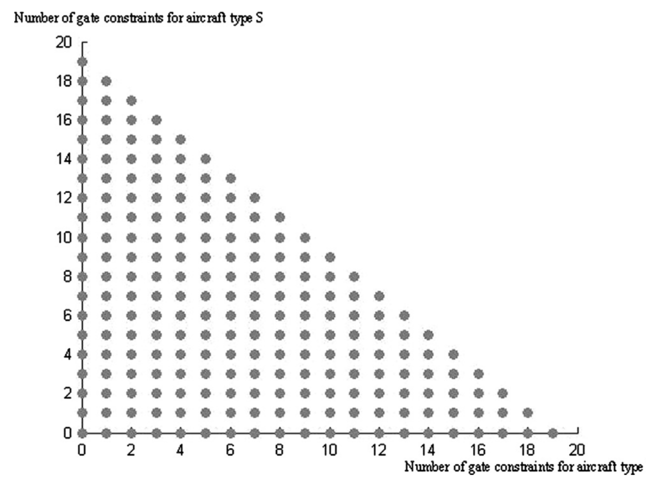

Fig. 15. Different scenarios considered in the heterogeneous range experiment.

([*S*,*M*]), y gates are ([*M*,*L*]) and the remaining $19-x-y$ gates are ([*S*,*M*,*L*]).

For each of these scenarios we run the heterogenous model on metric *X*: sum of the expected gate blockage lengths. Note that for some of these points (typically when all gate are restricted for one aircraft type) the problem will be infeasible which corresponds to an infinite cost. However, for the point (0,0), which represents the situation with no gate constraints at all, we will find the same optimal objective as the one obtained in the homogeneous model. The main purpose of this experiment is to study how the objective increases when we add gate constraints and also to study the impact on computational performance as a function of how constrained the model is.

To do so, we ran the heterogeneous code for each point of Fig. 15. We represent their optimal objective values in Fig. 16 and the computational time in Fig. 17. To compare qualitatively the different values, we use gray markers: lightest circles correspond to the lowest values and the darkest circles to the highest values. Star-shaped markers are used when the problem is infeasible.

The computational times (in seconds) are distributed as shown in Table 5.

The fact that the objective does not visibly change when moving in the vertical direction implies that constraining aircraft type *S* does not significantly impact the total cost. However constraining aircraft type *L* has a high price, even when aircraft

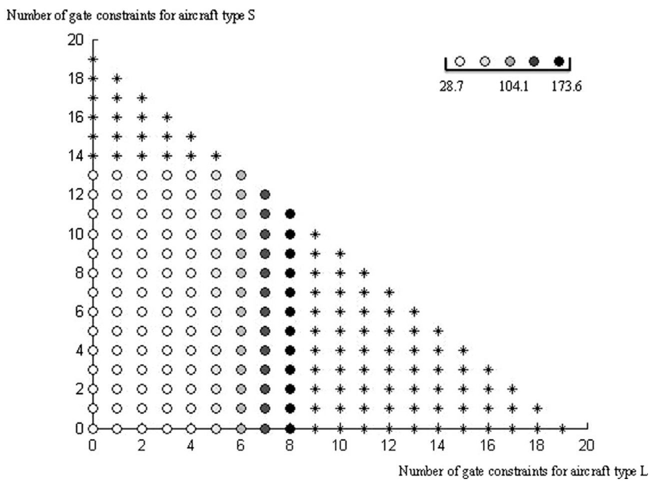

Fig. 16. Objectives of the heterogeneous range experiment.

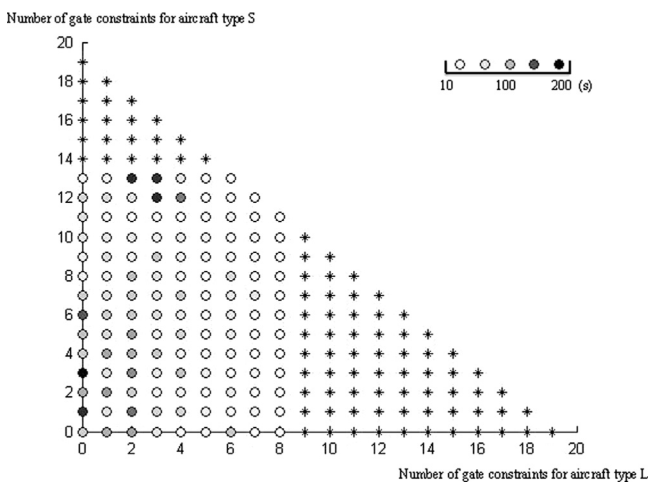

Fig. 17. Computational times of the heterogeneous range experiment.

Table 5
Distribution of the computational times of the heterogenous range experiment.

Min	25% percentile	Median	75% percentile	Max	Mean
8	17	31	54.5	237	45

type *S* is not constrained. This results from the fact that there are roughly four times more aircraft of type *L* than aircraft of type *S*.

In Fig. 16, we can see that constraining the problem typically reduces the computational time and that the lowest run times are obtained when the problem is almost infeasible in the sense that adding one gate constraint would make the problem infeasible. However we notice a zone of higher computational times when aircraft type *S* is very constrained. Furthermore, the time to prove infeasibility was very short, typically on the order of 1 s. As a whole, there does not seem to be any significant correlation between level of constraints and run time, with all problems solving quickly.

7. Future research and conclusions

7.1. Model extensions and future research

We present here some ideas for future projects to extend our research.

- *Objective based on missed connections*
  In our (*C*) metric: expected connecting passenger blockage minutes, we use the blockage time associated with connecting passengers as a surrogate for missed connections. This metric has two key limitations:
  1. It does not recognize the daily variability in passenger itineraries on any given flight.
  2. It treats the impact of blockages in a linear fashion. In fact, the impact is really binary: either the blockage is long enough to induce a missed connection or not.

  Ideally it would be better to estimate (and do so more accurately) the expected number of passengers missing their connections due to gate blockage. To do so, we recommend using a stochastic distribution for the connection time depending on influencing factors, such as the origin and destination of the flight and the hour of the day. The expected number of missed connections, given a delay of *d* minutes, would be the number of connecting passengers multiplied by the cumulative distribution function of the connection time evaluated in *d* (i.e. the probability that a connection time is less than *d* minutes).
- *Adjacency issues*
  An important constraint faced by airlines when building a gate assignment is the adjacency constraint which means that, in certain cases, two given aircraft types cannot be simultaneously at two adjacent gates: the orientation of adjacent gates may make it impossible to fit two wide-bodied aircrafts next to each other simultaneously, for instance. In order to take those adjacency issues into account in our model, we would need to add a constraint for each pair of adjacent gates and associated pairs of incompatible (in time and fleet type) aircraft turns to ensure that at most one of the two assignments is made. However this potentially represents a very large number of constraints and so alternative modeling and/or solution techniques might need to be developed.
- *Analysis of delay propagation in a multi-station network*
  Propagation of delays throughout the day is one of the consequences of gate blockage: a flight delayed due to gate blockage reaches its gate late and is consequently likely to leave the gate late, which increases the risk of generating a new gate blockage at the current station (as well as many other negative system impacts). Interestingly, this departure delay also reduces the chance that the flight will be blocked at its next destination. As such, a more effective gating approach would take into consideration the down stream effects caused by gate blockage-caused delay propagation.
- *Improved estimation of objective coefficients*
  We conducted our analyses by using historical data to estimate delay probabilities, and then using the probabilities to optimize gating assignments for the same time period. In reality, of course, we could not have the known data for the same time period that we are trying to plan. Therefore, a valuable area of research (not only for the purpose of the robust gating assignment problem but for a wide range of airline planning problems as well) is to better develop probability distributions for future flights.

7.2. Conclusion

A study of the current situation on a large sample of U.S. airports for a major domestic carrier shows that gate blockage occurs during 5% of commercial flights, with 2% of all flights being delayed by at least 10 min. Consequences of gate blockage such as missed connections and increased costs for airlines make it important to address this problem when building a gate assignment.

We propose network-based models for both the homogeneous and heterogeneous versions of the problem, show that these models are computationally tractable, and demonstrate that, for several different metrics of robustness, they significantly improve performance over a first-in-first-out assignment paradigm.

Not surprisingly, the heterogeneous model is slightly more computationally intensive than the homogenous model due to the pure minimum-cost-flow structure of the homogenous model. Nonetheless, for realistic instances the heterogeneous model solves quite quickly in practice, with runtimes on the order of a few minutes at most.

Our computational experiments show that these models give better results regarding the four tested objectives related to gate blockage than a standard first-in-first-out algorithm. Even if other criteria are taken into account by airlines when building their schedules, our research gives a useful tool which can be used to compare several possible schedules according to gate blockage metrics and will allow airlines to select more robust choices for their gate assignment.

Acknowledgment

The authors would like to thank Melinda Alberty, Ryan Chen, Peter Cohn, Melanie Faibish, Mark Grum, Zachary Rollin, and Sajan Shah (listed alphabetically) for their contribution to this research.

References

[1] Kim Sang Hyun, Feron Eric. Robust gate assignment. AIAA guidance, navigation, and control conference, August 2011.
[2] Lim Andrew, Wang Fan. Robust airport gate assignment. In: 17th IEEE international conference on tools with artificial intelligence, 2005. ICTAI 05, IEEE; 2005. 8 p.
[3] Li Chendong. Airport gate assignment: New model and implementation. 2008. arXiv preprint arXiv:0811.1618.
[4] Ding H, Lim Andrew, Rodrigues B, Zhu Yi. The over-constrained airport gate assignment problem. Comput Oper Res 2005;32(7):1867–80.
[5] Cheng Chun-Hung, Sin C. Ho, Cheuk-Lam Kwan. The use of meta-heuristics for airport gate assignment. Expert Syst Appl 2012;39(16):12430–7.
[6] Genç Hakkı Murat, Erol Osman Kaan, Eksin İbrahim, Berber Mehmet Fatih, Güleryüz Binnur Onaran. A stochastic neighborhood search approach for airport gate assignment problem. Expert Syst Appl 2012;39(1):316–27.
[7] Şeker Merve, Noyan Nilay. Stochastic optimization models for the airport gate assignment problem. Transp Res Part E: Logist Transp Rev 2012;48(2):438–59.
[8] Burke Edmund K, De Causmaecker Patrick, De Maere Geert, Mulder Jeroen, Paelinck Marc, Berghe Greet Vanden. A multi-objective approach for robust airline scheduling. Comput Oper Res 2010;37(5):822–32.
[9] Ahmadbeygi Shervin, Cohn Amy, Lapp Marcial. Decreasing airline delay propagation by re-allocating scheduled slack. IIE Trans 2010;42(7):478–89.
[10] Lapp Marcial, AhmadBeygi Shervin, Cohn Amy, Tsimhoni Omer. A recursion-based approach to simulating airline schedule robustness. In: Proceedings of the 40th conference on winter simulation, winter simulation conference; 2008. p. 2661–7.
[11] Teodorovic Dušan, Stojkovic Goran. Model to reduce airline schedule disturbances. J Transp Eng 1995;121(4):324–31.
[12] Lan Shan, Clarke John-Paul, Barnhart Cynthia. Planning for robust airline operations: optimizing aircraft routings and flight departure times to minimize passenger disruptions. Transp Sci 2006;40(1):15–28.
[13] Rosenberger Jay M, Johnson Ellis L, Nemhauser George L. A robust fleet-assignment model with hub isolation and short cycles. Transp Sci 2004;38 (3):357–68.
[14] Borndörfer Ralf, Dovica Ivan, Nowak Ivo, Schickinger Thomas. Robust tail assignment. In: Proceedings of the fiftieth annual symposium of AGIFORS; 2010.
[15] Lapp Marcial, Cohn Amy. Modifying lines-of-flight in the planning process for improved maintenance robustness. Comput Oper Res 2012;39(9):2051–62.
[16] Lapp Marcial. Methods for improving robustness and recovery in aviation planning [Ph.D. thesis]. The University of Michigan; 2012.
[17] Klabjan Diego, Johnson Ellis L, Nemhauser George L, Gelman Eric, Ramaswamy Srini. Airline crew scheduling with regularity. Transp Sci 2001;35(4):359–74.
[18] Schaefer Andrew J, Johnson Ellis L, Kleywegt Anton J, Nemhauser George L. Airline crew scheduling under uncertainty. Transp Sci 2005;39(3):340–8.
[19] Yen Joyce W, Birge John R. A stochastic programming approach to the airline crew scheduling problem. Transp Sci 2006;40(1):3–14.
[20] Thengvall Benjamin G, Bard Jonathan F, Yu Gang. Balancing user preferences for aircraft schedule recovery during irregular operations. IIE Trans 2000;32 (3):181–93.
[21] Eggenberg Niklaus, Salani Matteo, Bierlaire Michel. Constraint-specific recovery network for solving airline recovery problems. Comput Oper Res 2010;37 (6):1014–26.
[22] Yan Shangyao, Yang Dah-Hwei. A decision support framework for handling schedule perturbation. Transp Res Part B: Methodol 1996;30(6):405–19.
[23] Beatty Roger, Hsu Rose, Berry Lee, Rome James. Preliminary evaluation of flight delay propagation through an airline schedule. Air Traffic Control Q 1999;7(4):259–70.
[24] AhmadBeygi Shervin, Cohn Amy, Guan Yihan, Belobaba Peter. Analysis of the potential for delay propagation in passenger airline networks. J Air Transp Manag 2008;14(5):221–36.
[25] Abdelghany Ahmed, Ekollu Goutham, Narasimhan Ram, Abdelghany Khaled. A proactive crew recovery decision support tool for commercial airlines during irregular operations. Ann Oper Res 2004;127(1–4):309–31.
[26] Lettovsky` Ladislav, Johnson Ellis L, Nemhauser George L. Airline crew recovery. Transp Sci 2000;34(4):337–48.
[27] Barnhart Cynthia, Cohn AM, Johnson EL, Klabjan D, Nemhauser GL, Vance PH. Airline crew scheduling. New York: Springer; 2003 (Chapter: Airline crew scheduling, pp. 517–60).
[28] Cohn, Lapp M. Airline resource scheduling. John Wiley & Sons, Inc; 2010.
[29] Sriram Chellappan, Haghani Ali. An optimization model for aircraft maintenance scheduling and re-assignment. Transp Res Part A: Policy Pract 2003;37(1):29–48.
[30] Kohl Niklas, Larsen Allan, Larsen Jesper, Ross Alex, Tiourine Sergey. Airline disruption management—perspectives, experiences and outlook. J Air Transp Manag 2007;13(3):149–62.
[31] Filar Jerzy A, Manyem Prabhu, White Kevin. How airlines and airports recover from schedule perturbations: a survey. Ann Oper Res 2001;108(1–4):315–33.
[32] Barnhart Cynthia, Talluri Kalyan T. Airline operations research. Des Oper Civil Environ Eng Syst 1997:435–69.
[33] Gopalan Ram, Talluri Kalyan T. Mathematical models in airline schedule planning: a survey. Ann Oper Res 1998;76:155–85.