Optimal Gate Assignment Under Consideration of the Ground Infrastructure

Philipp Zeunert
Institute of Flight System Dynamics
Technische Universität München
Munich, Germany
philipp.zeunert@tum.de

Eric Möbius, Markus Herrich
Institute of Numerical Mathematics
Technische Universität Dresden
Dresden, Germany
markus.herrich@tu-dresden.de

***Abstract*—Besides safety, the main philosophy of a five-star airport is to ensure the highest achievable level of quality in terms of comfort and passenger transfer efficiency, measured by short paths and low connecting times. In the event of critical connecting times, an optimal assignment of gates may decide whether passengers reach their onward flights or not. In this paper, we address the modeling and solving of the Airport Gate Assignment Problem with a focus on the properties of a five-star hub airport. Besides general ground infrastructure properties, we also discuss the gate assignment under consideration of the aircraft wingspan. Finally, we will use a model of the Terminal 2 at Munich Airport as a basis for case studies. The underlying problem is a binary quadratic optimization problem, and we will discuss techniques for solving the problem by means of fast solution procedures. In the near future, an extension will be introduced that will consider the selection of seats depending on the passenger's preference, the estimated connecting time and the yields generated by reservation fees.**

Airport Gate Assignment Problem; Quadratic Assignment Problem; Five-Star Airport; Wingspan

I. INTRODUCTION

As an international air transport rating organization, Skytrax audits airports to classify them by the quality product, facilities and staff product. The highest achievable rating is the five-star airport certification, which is held exclusively by 12 airports worldwide[1]. The ease of the transfer process, including, but not limited to, the walking distances and ease of wayfinding are among many audit aspects [1]. Achieving high quality at all times is a particular challenge for large, international hub airports with a complex hub-and-spoke network system, which is susceptible to disruption.

The largest hub airport worldwide in terms of passenger traffic, and also the largest airline hub of Delta Air Lines, is Hartsfield-Jackson Atlanta International Airport (ATL). More than 110 million passengers and 903,818 aircraft operations were counted in a 12-month period[2], with about 300,000 passengers using ATL every day [2]. Delta Air Lines is the largest carrier at ATL and has an average market share of 72.9% [3]. The largest airport with respect to passenger traffic in Europe is London Heathrow. In 2018, the airport counted 80.1 million passengers, 30 % of whom were transfer passengers [4]. We will focus on Munich Airport (MUC), where 46.3 million passengers were counted in the same year; 37% of these passengers used the airport to transfer [5].

As we can see from the statistics, the challenge for hub airport operators and hub airlines is to control a large volume of transferring passengers efficiently, even at peak times. One key element in this is the optimal assignment of airport gates, so that preferably both arrival and departure gates are within a short walking distance for all passengers – this, however, is the point of view of customers and airlines. On the other hand, infrastructure owners[3] would prefer to minimize the number of gates and increase utilization [6]. Fulfilling all objectives at once is not possible without compromise. Achieving the utmost passenger comfort is the objective of our work. For this reason, we minimize walking distance.

An optimal assignment of the gates to a set of aircraft is the solution to the Airport Gate Assignment Problem (AGAP). The challenge of the underlying (in most cases large-scale) mathematical program is to determine a vector of (binary) decision variables that minimizes an objective function without violating given constraints in due time. This problem has been

[1] In January 2020, subject to changes [1]
[2] Period: November 2018 to October 2019
[3] Airports and terminals are usually owned by states and counties. In the case of MUC, Terminal 2 is a joint venture between Flughafen München GmbH and the Lufthansa Group.

well discussed in many variants over the last 45 years, but is still relevant nowadays [6].

II. Related Work

The two major modernization projects NextGen and Single European Sky (SES) have once again made the AGAP topical, as the infrastructural base is provided for advanced data processing in combination with the involvement of stakeholders. This makes it possible to revise the AGAP while taking into account other linked optimization problems: the robust gate assignment under consideration of decisions made by the Departure Manager (DMAN) and the stakeholders involved in the Airport Collaborative Decision Making (A-CDM) program is addressed in [7]. The authors introduced methods of decreasing the number of conflicts involved in longer gate occupancy times when departure delays occur. In contrast to [7], former research has focused more on the modeling and the identification of solution techniques. However, these topics are still up-to-date, as new objectives and constraints are joining the AGAP, which may simultaneously also require new solution techniques or adaptions to existing ones.

The AGAP can be modeled as a linear or non-linear optimization problem with (binary) integer or mixed integer decision variables. In combinatorial optimization, the AGAP has been modeled as a quadratic assignment problem (QAP) or click portioning problem (CPP) [6]. Classic objectives are to minimize passenger walking distance/time or the costly usage of aircraft tugs for towing [8]. Furthermore, it is important not to violate operational side constraints, such as the consideration of the aircraft wingspan. Objectives related to the occupancy of gates may also be formulated as constraints [9]. In some cases, more than one objective may be present, which may lead to multi-criteria optimization or pareto optimization [10]. Another challenge is to cope with NP-hard problems, which arise from QAP. The problem cannot be solved accurately in polynomial time [11]. For this reason as well as operational ones, a nearly optimal or satisfactory solution must be determined using heuristics and metaheuristics when it is not viable to determine an exact solution to a large-scale problem [6].

The objective of our work is to focus on the modeling of a tailored real-case AGAP, however, without considering interfaces with related problems at this stage. We want to propose some extensions to the existing classic AGAP that will address the ground infrastructure at an airport in general, and especially the situation at the Terminal 2 at MUC. Further, we will propose a novel way to consider wingspan. We will approximately solve the underlying QAP using a heuristic.

The structure of this paper is as follows: First, we want to deduce the classic AGAP with the aim of minimizing total walking time. Second, we extend the model by enabling bus and luxury car transportation from the terminal to remote stands and vice versa under the consideration of the vehicle capacity. Furthermore, we address the situation in which two or even more taxi guidance lines occur at a terminal stand, enabling a more flexible assignment of stands to aircraft, taking into account the wingspan. Third, we will introduce and test a heuristic to approximately solve the problem.

III. The Classic AGAP

Before we start modeling, we have to collect the required information available to us. Since we are aiming to minimize the total walking time for all passengers, we have to know the passenger flow and the terminal layout. In addition, we will consider the rotation of m aircraft during a fixed period of time, and the mean walking time a passenger needs to connect from gate k to gate l. Each gate is associated with either a *terminal stand,* allowing passenger (de)boarding using passenger bridges, or a *remote stand,* which requires bus transportation. The total number of gates is denoted by n. Throughout the paper, we assume $m \leq n$. Moreover, we denote the set of all aircraft and the set of all gates as $A := \{1, \dots, m\}$ and $P := \{1, \dots, n\}$, respectively.

A. The Input and Output Data

Let us start with the basic shape of the underlying problem: we consider a passenger flow matrix $Y \in \mathbb{N}_0^{m \times m}$, which provides us with the number of passengers y_{ij} that arrive on aircraft i and will depart on aircraft j, $i, j \in \{1, \dots, m\}$. In this context, we focus on the rotations of m aircraft, instead of scheduled flights, to keep the matrix size of $m \times m$ fixed. If an aircraft i commences or terminates its service in the given period of time, no passengers will arrive on or depart by aircraft i, respectively. Furthermore, we define the matrix of the walking times between all gates using $T \in \mathbb{R}^{n \times n}$. As we consider remote stands, we also assume a remote stand to be a gate. For an arriving flight, the time it takes to get by bus from the remote stand to a central drop-off point at the terminal, as well as the walking time to the departure gate of the onward flight, is added to obtain the corresponding entry of the matrix. However, the time it takes to get to the remote stand of an onward flight is not considered because passengers have to be at the gate only at a certain gate closure time.

The solution of the QAP is the decision as to which aircraft i blocks a certain stand k. Let us introduce binary decision variables $x_{ik} \in \{0,1\}$, $i \in \{1, \dots, m\}$, $k \in \{1, \dots, n\}$, which can be interpreted as follows:

$$x_{ik} := \begin{cases} 1 & if\ aircraft\ i\ blocks\ position\ k \\ 0 & else \end{cases} \quad (1)$$

With $x \in \{0,1\}^{mn}$, we denote the vector of all decision variables, i.e.

$$x := (x_{11} \ldots x_{1n} x_{21} \ldots x_{2n} \ldots x_{m1} \ldots x_{mn})^T \quad (2)$$

B. Principles of the Quadratic Assignment Problem

Since we have to determine a stand for each aircraft, depending on the assignment to all other aircraft within a fixed period of time, this gives rise to a quadratic assignment problem (QAP). A QAP is a binary quadratic program and is, therefore, characterized by a quadratic, possibly non-convex objective function, constraints of linear equations and inequations and binary decision variables. Thus, the QAP that we use to model our gate assignment problem will have the following structure:

$$\frac{1}{2}\, x^T Q x + c^T x + d \quad \rightarrow t.b.minimized \quad (3)$$

$$s.t. \qquad \begin{aligned} Ex &\leq f \\ Gx &= h \end{aligned} \quad (4)$$

$$x_{ik} \in \{0,1\}\ \forall i \in \{1,\ldots,m\}, k \in \{1,\ldots,n\} \quad (5)$$

In the following sections, we will deduce the constraints and the objective function of our QAP in detail.

C. The Constraints

We must ensure that first, each aircraft is assigned to one stand (6) and second, a certain stand blocked by at most one aircraft (7). Since we aim to determine the assignment of stands over one day, the time domain has to be considered. Therefore, the period from the on-block time (ONB) to the off-block time (OFB) is used to define the occupation of a stand. Furthermore, the ONB is the sum of the actual time of arrival (ATA) and the time required to taxi to the position. In both cases, ONB and OFB are assumed to be known based on an estimation of the ATA and STD. For each aircraft $i \in \{1,\ldots,m\}$, we denote the ONB and OFB by a_i and d_i, respectively. We assume that there are no delays. However, a time buffer is included in a_i and d_i to avoid collisions during maneuvering. Finally, the following constraints are included in the optimization problem:

$$\sum_{k=1}^{n} x_{ik} = 1\ \forall i \in \{1,\ldots,m\} \quad (6)$$

$$\begin{aligned} &x_{ik} x_{jk} (d_j - a_i)(d_i - a_j) \leq 0 \\ &\forall i,j \in \{1,\ldots,m\}, i \neq j, \forall k \in \{1,\ldots,n\} \end{aligned} \quad (7)$$

In addition, we want to introduce the option of assigning certain stands to selected aircraft only. The reason for this is twofold: first, the prevailing terminal structure may require us to consider border controls; and second, selected aircraft types, airlines or general aviation traffic are only permitted to use certain areas or selected stands. In this case, we have to define at least two sets of stands. Since we are aiming to develop a model for Munich airport, we introduce a subset of stands $P_1 \subset P$ and a subset of aircraft types $A_2 \subset A$ that must only use a remote stand. All other aircraft types are permitted to use either a remote stand or a terminal stand directly linked to the terminal by a passenger bridge. The following constraint prevents an aircraft $i \in A_2$ from using a stand $k \in P_1$, where P_1 denotes the set of stands at the terminal:

$$\sum_{k \in P_1}^{n} x_{ik} = 0\ \ \forall i \in A_2 \quad (8)$$

For the sake of completeness, $P_2 := P \backslash P_1$ is the set of remote stands and $A_1 := A \backslash A_2$ the set of aircraft that is allowed to use any stand. However, we do not have to consider P_2 and A_1 explicitly at the moment, as we are not making any further restrictions.

D. Reformulation of Constraint (7)

The inequality constraint (7) is quadratic. We will use big-M linearization in order to linearize it. First, note that (7) is equivalent to the following constraint:

$$\begin{aligned} &x_{ik} = 1 \Rightarrow x_{jk}(d_j - a_i)(d_i - a_j) \leq 0 \\ &\forall i,j \in \{1,\ldots,m\}, i \neq j, \forall k \in \{1,\ldots,n\} \end{aligned} \quad (9)$$

Big-M linearization is a common method of transferring constraints like (9) into linear ones [12]. Let us define:

$$M := \max_{i,j \in \{1,\ldots,m\}} (d_j - a_i)(d_i - a_j) \quad (10)$$

Then, (9) can be equivalently written as:

$$\begin{aligned} &x_{jk}(d_j - a_i)(d_i - a_j) \leq (1 - x_{ik})M \Leftrightarrow \\ &x_{jk}(d_j - a_i)(d_i - a_j) + x_{ik}M \leq M \\ &\forall i,j \in \{1,\ldots,m\}, i \neq j, \forall k \in \{1,\ldots,n\} \end{aligned} \quad (11)$$

In fact, if $x_{ik} = 1$ for some $i \in \{1,\ldots,m\}, k \in \{1,\ldots,n\}$, the inequalities in (11) are reduced to the original constraints $x_{jk}(d_j - a_i)(d_i - a_j) \leq 0$ for all $j \in \{1,\ldots,m\}$. On the other hand, if $x_{ik} = 0$, then the inequalities in (11) reduce to $x_{jk}(d_j - a_i)(d_i - a_j) \leq M$, which are satisfied independently of whether $x_{jk} = 0$ or $x_{jk} = 1$, due to the definition of M. Consequently, (7) can be replaced by the linear constraints (11).

E. The Objective Function

The objective function to be minimized quantifies the total time-related costs as the product of the number of passengers and the time it takes each passenger to get to the new gate. For $i,j \in \{1,\dots,m\}$ and $k,l \in \{1,\dots,n\}$ we define: $m_{(ik),(jl)} := t_{kl}y_{ij}$. As a matter of principle, the total costs depend on the assigned stands, and we retrieve the following objective function:

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{n}\sum_{l=1}^{n} x_{ik}\,(t_{kl}y_{ij})\,x_{jl} = x^T M\, x \tag{12}$$

The matrix $M := \left(m_{(ik),(jl)}\right) \in \mathbb{R}^{mn\times mn}$ does not necessarily have the symmetrical structure that solvers of binary quadratic programs often require. One way to overcome this issue is to substitute the matrix M by $\widetilde{M} := M + M^T$. This would just augment the objective function by the factor 2. Therefore, the value should be reduced accordingly.

F. Summary – the Basic AGAP

The following provides a summary of the complete basic AGAP, which we will use as a base for further extensions:

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{n}\sum_{l=1}^{n} t_{kl}y_{ij}x_{ik}x_{jl} \quad \rightarrow t.b.minimized! \tag{12}$$

$s.t.$

$$\sum_{k=1}^{n} x_{ik} = 1 \;\; \forall i \in \{1,\dots,m\} \tag{6}$$

$$\sum_{k\in P_1} x_{ik} = 0 \;\; \forall i \in A_2 \tag{8}$$

$$x_{jk}\left(d_j - a_i\right)\left(d_i - a_j\right) + x_{ik}M \le M \quad \forall i,j \in \{1,\dots,m\}, i \ne j, \forall k \in \{1,\dots,n\} \tag{11}$$

$$x_{ik} \in \{0,1\} \;\forall i \in \{1,\dots,m\}, k \in \{1,\dots,n\} \tag{13}$$

The objective function (12) sums up the total walking time for all passengers. The constraints (6) and (8) ensure that each aircraft is assigned to exactly one stand and selected aircraft are banned from certain stands, respectively. As we consider a whole day of operation or a closed time interval, blocking and clearing stands over time are enabled by constraint (11), which arises from (7), using big-M linearization. The decision variables may be either zero or one, which is ensured by (13). The resulting optimization problem is a binary quadratic program.

IV. Beyond the Classic AGAP

A. Passenger Transportation to Remote Stands

Until now, we have addressed the classic AGAP without considering the available ground infrastructure in detail. When remote stands are used, a bus is usually deployed to transport passengers from the aircraft to the terminal building and vice versa. Aircraft parking at a terminal stand directly in front of the terminal may need only an aircraft tug for push-back, but no buses. The number of buses and their capacities are limited, and must be taken into consideration in order to avoid shortages resulting in delays and inconveniences. For this reason, we add some further restrictions to the classic AGAP. For any aircraft $i \in \{1,\dots,m\}$, we define the required number of buses $n_{bus,i}$ for the case that the aircraft is assigned to a remote stand by:

$$n_{bus,i} = \left\lceil \frac{n_{pax,i}}{cap_{bus}} \right\rceil \tag{14}$$

where $n_{pax,i}$ denotes the actual number of passengers of aircraft i and cap_{bus} is the bus capacity, i.e. the number of passengers that can be transported by one bus. The total number of available buses is denoted by n_{bus}. The buses have to be blocked for arriving and departing flights. Therefore, we have to consider the time interval A_i; a bus starts its trip from the terminal to the remote stand and back if aircraft i is assigned to a remote stand. The same logic is applied to departing flights. We define A_i, and the time interval D_i for the deployment of buses for departing flights as follows:

$$\begin{aligned} A_i &:= [a_i - \tau_{i1}, a_i + \tau_{i2}], \qquad i \in \{1,\dots,m\} \\ D_i &:= \left[t_{b,i} - \tau_{i3}, t_{b,i} + \tau_{i4}\right], \qquad i \in \{1,\dots,m\} \end{aligned} \tag{15}$$

The parameters τ_{i1} and τ_{i2} denote the time taken by a bus to get to the remote stand, plus wait time for passengers to disembark and drive time to the terminal, respectively. The parameter τ_{i3} denotes the time spent waiting for passengers at the gate and taking them to the remote stand for departure, τ_{i4} denotes the time needed to return to the terminal and $t_{b,i}$ is the estimated time of boarding. Note that the durations $\tau_{i1},\dots,\tau_{i4}$ are assumed to be fixed. Of course, in reality they might depend on the actual remote stand which is assigned to the aircraft i. We determine the set of all other aircraft j for each aircraft i that requires buses within the time intervals A_i and D_i, by:

$$\begin{aligned} G_i &:= \left\{ j \in \{1,\dots,n\} \middle| \, A_i \cap A_j \ne 0 \,\vee\, A_i \cap D_j \ne 0 \right\} \\ H_i &:= \left\{ j \in \{1,\dots,n\} \middle| \, D_i \cap A_j \ne 0 \,\vee\, D_i \cap D_j \ne 0 \right\} \end{aligned} \tag{16}$$

The set G_i contains all aircraft j that also demand buses within the arrival time interval of aircraft i. Similarly, the set H_i contains all aircraft j that also demand buses within the departure time interval of aircraft i.

Finally, we obtain the following additional constraints for our model taking into account the requirement of buses for aircraft that are assigned to remote positions:

$$\sum_{k\in P_2}\sum_{j\in G_i} n_{bus,i}x_{jk} \leq n_{bus} \quad \forall i\in\{1,\dots,m\}$$
$$\sum_{k\in P_2}\sum_{j\in H_i} n_{bus,i}x_{jk} \leq n_{bus} \quad \forall i\in\{1,\dots,m\} \tag{17}$$

As we have mentioned before, luxury cars are available to premium customers who travel in first class or have the highest achievable frequent traveler status. The deployment of these cars can be modeled in a similar way, but is not addressed in detail.

B. Aircraft Wingspan

Another key element might be the consideration of the aircraft wingspan. One objective in terminal planning is to place as many aircraft as possible in front of the terminal to enable (de)boarding by passenger bridges. In fact, space is limited and therefore remote stands are also used. Furthermore, the distance between two adjacent stands at some terminals is only sufficient to place single-aisle aircraft with a smaller wingspan next to each other. For this reason, all stakeholders may benefit from more flexible steering and assignment of stands when a mix of single-aisle and widebody aircraft is frequently handled.

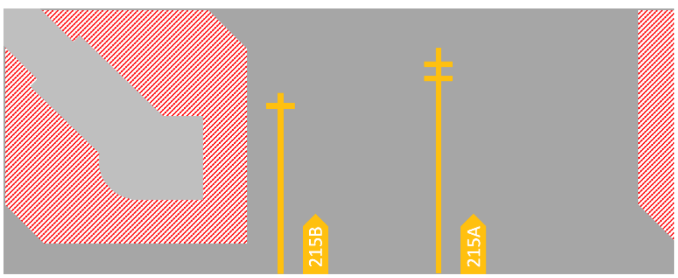

Figure 1: A gate position with two taxi guidance lines

How can we enable widebody aircraft to use stands at the terminal without violating the minimum clearance? Answering this question is straightforward: we omit one or two stands. When it comes to efficiency, however, the issue is not straightforward. In general, there is a tradeoff between providing all stands with a passenger bridge either only to single-aisle aircraft or to omit stands instead. To make it as efficient as possible, our aim is to use every space and ensure that the minimum clearance is met at the same time. Furthermore, terminal stands may have two or even more (parallel) taxi guidance lines to increase the utilization of available space to accommodate aircraft of different wingspans. Figure 1 shows terminal stand no. 215 at MUC, which has two parallel taxi guidance lines 215A and 215B. Furthermore, we assume that the passenger bridge can be used to handle all aircraft types at either stand. No positioning restrictions are implied by the passenger bridge.

Our next step is to develop restrictions, which we will add to our model. We indicate the wingspan of each aircraft $i \in \{1,\dots,m\}$ by $span_i$. Moreover, the distance between two adjacent stands k and $k+1$ is denoted by $dis(k)$. Finally, we indicate the set of all aircraft that are on the ground at the same time as aircraft i, by:

$$L_i := \{j \in \{1,\dots,m\}\backslash\{i\} \,|\, [a_i,d_i] \cap [a_j,d_j] \neq \emptyset\} \tag{18}$$

An initial formulation of constraints that ensure parking without collision is provided by:

$$\frac{span_i}{2}x_{ik} + \frac{span_j}{2}x_{j(k+1)} + s \leq dis(k)$$
$$\forall i\in\{1,\dots,m\}, \forall j \in L_i, \forall k \in \{1,\dots,n-1\} \tag{19}$$

The constraints require that for any i, j, k, the sum of the halves of the wingspans of both aircraft i and j and a minimum clearance s is less than the distance between the adjacent stands k and $k+1$, if aircraft i and j are assigned to these stands.

The constraints (19) only consider pairs of adjacent stands in ascending order. This is, however, not yet sufficient to ensure a minimum clearance between aircraft in all cases. One way to overcome this issue would be to introduce constraints that consider any combination. However, this increases the complexity enormously and should be avoided.

Therefore, we will take a different approach, which outlines the case of terminal stands with two guidance lines and three aircraft parking at adjacent stands: considering the wingspans only to the right was not sufficient to ensure a minimum clearance; we must also consider the distances to the left. For each $k \in \{2,\dots,n\}$, let $dis_l(k)$ denote the distance of stand k to the stand adjacent to the left. Accordingly, for each $k \in \{1,\dots,n-1\}$, let $dis_r(k)$ denote the distance to the stand adjacent to the right. Any distance is the greatest one between possibly two guidance lines of two adjacent terminal stands. We add further constraints to our model to ensure a minimum clearance in both directions:

$$\underbrace{\left(p_{i_1 i_2 k} - q_{i_2 i_3 k}\right)}_{f_1}\underbrace{\left\lfloor\frac{x_{i_1 k} + x_{i_2(k+1)} + x_{i_3(k+2)}}{3}\right\rfloor}_{f_2} \leq 0$$
$$\forall i_1 \in \{1,\dots,m\}, \forall i_2, i_3 \in L_{i_1}, \forall k \in \{1,\dots,n-2\} \tag{20}$$

The terms p_{ijk} and q_{ijk} reflect the relation between the available width and the required width as the sum of the wing spans of two aircraft i and j and the minimum clearance s.

$$p_{ijk} := \begin{cases} 1 \; if \left(\frac{span_i}{2} + \frac{span_j}{2} + s\right) - dis_l(k+1) \geq 0 \\ -1 \; else \end{cases} \tag{21}$$

$$q_{ijk} := \begin{cases} 1 \; if \left(\frac{span_i}{2} + \frac{span_j}{2} + s\right) - dis_r(k+1) \geq 0 \\ -1 \; else \end{cases} \tag{22}$$

Unfortunately, the constraints (20) are non-linear inequalities. However, they can be reformulated into linear ones. Note that the factor f_1 may only have the values 2, 0 or -2. An assignment $x_{i_1k} = x_{i_2(k+1)} = x_{i_3(k+2)} = 1$ is not valid if, and only if, $\left(p_{i_1i_2k} - q_{i_2i_3k}\right) = 2$ holds. These considerations allow us to conclude linear inequalities in the following way:

$$\begin{gathered} \left(p_{i_1i_2k} - q_{i_2i_3k}\right)\left(x_{i_1k} + x_{i_2(k+1)} + x_{i_3(k+2)}\right) \leq 5 \\ \forall i_1 \in \{1, \dots, m\}, \forall i_2, i_3 \in L_{i_1}, \forall k \in \{1, \dots, n-2\} \end{gathered} \tag{23}$$

In fact, the left-hand side of one of the inequalities is greater than 5 if, and only if, the available width is not sufficient.

C. The Objective Function

We already introduced an objective function for the classic AGAP. The objective function (12) was the sum of the walking time of all passengers. However, this has at least two disadvantages: first, we do not prioritize customers who have very short connecting times and may miss their onward flights; furthermore, our aim is to ensure a five-star service level, so premium customers should preferably connect in less time. Second, when remote stands are used, buses[5] have to be provided, at the very least, for passenger transportation, which also incurs monetary costs.

In the following section, we extend the classic objective function and first, split the passenger flow matrix Y into three matrices Y^E, Y^J and Y^F. The matrix Y^E contains the number of passengers that travel in economy class and do not have frequent traveler status, Y^J contains the number of passengers that travel in business class or have a frequent traveler status and Y^F contains all premium customers that fly in first class or have the highest achievable frequent traveler status. In addition, we prioritize passengers with short connecting times by adding a weighting factor, which is the reciprocal of the connecting time γ_{ij} between two aircraft $i, j \in \{1, \dots m\}$:

$$\gamma_{ij} := \begin{cases} \nu_{j1} - \nu_{i2} \; if \; \nu_{j1} > \nu_{i2} \\ 1 \; else \end{cases} \tag{24}$$

[5] Aircraft tugs could be deployed depending on the apron layout. The deployment of aircraft tugs causes additional costs.

The connecting time between two aircraft i and j is the remaining time between the time ν_{j1} that the gate of flight j will be closed and the time ν_{i2} the passengers are able to disembark aircraft i. Moreover, we prioritize business and first-class customers by introducing another two weighting factors $\mu^J > 0$ and $\mu^F > 0$, which are model parameters that should be selected adequately, but follow no particular rule. Finally, the latest modification of the objective function is given by:

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{n}\sum_{l=1}^{n} \frac{1}{\gamma_{ij}} x_{ik} x_{jl} t_{kl} \left(y_{ij}^E + \mu^J y_{ij}^J + \mu^F y_{ij}^F\right) \tag{25}$$

Furthermore, monetary costs are added, which may arise from the use of ground infrastructure. In the following, we only consider bus transportation as a prime example. This is done by taking the required number of buses $nr_{bus,i}$ for each aircraft i that is assigned to a remote stand $k \in P_2$. Depending on the airport, the costs p_B for the bus service are either fixed or subject to the period of use. In this paper, we only assume fixed costs. Subsequently, we can add the following expression:

$$\sum_{i=1}^{m}\sum_{k \in P_2} x_{ik} \, nr_{bus,i} \, p_B \tag{26}$$

Another important aspect relates to the skillful assignment of remote stands and terminal stands, which are equipped with one or even up to three passenger bridges to handle the Airbus A380. In Europe, using passenger bridges for smaller regional aircraft is uncommon. The level of the terminal building is usually too high, and the tilt of the bridge would be too steep for safe (de)boarding. Therefore, it makes more sense to assign remote stands to regional aircraft and assign terminal stands to larger aircraft. However, there is no requirement to fully prohibit the assignment. If all remote stands are occupied, it is still possible to assign terminal stands to regional aircraft, however without using passenger bridges. Furthermore, the opposite situation usually occurs for long-haul flights, as we do not want a large number of passengers to have to travel on a bus before spending hours on a flight. Moreover, some single-aisle and almost all widebody aircraft have a second door in front of the wing, which allows passengers to (de)board simultaneously by means of two passenger bridges. This not only expedites the (de)boarding process, but also ensures that passengers in economy remain separate from those in first/business class. A more comfortable boarding process can be ensured for the latter group. Therefore, we either penalize or favor the assignment of some stands to certain groups of aircraft by adding costs.

We split the set of terminal stands P_1 into two sets T_1 and T_2. The set T_1 contains all stands with only one passenger bridge, whereas T_2 contains all stands with two passenger bridges. The first term adds costs $\tilde{p}_R$ for a regional aircraft i that belongs to the set A_R of aircraft, which cannot use a passenger bridge at either T_1 or T_2, but is nevertheless assigned to a stand $k \in T_1 \cup T_2$. The second term adds costs $\tilde{p}_L$ to an aircraft i if it belongs to the set of aircraft A_L with two doors in front of the wing, but is nevertheless assigned to a stand $k \in T_1$. Finally, the third term adds costs $\tilde{p}_A$ if an aircraft is assigned to a remote stand $k \in P_2$. We add the following cost term to the objective function:

$$\sum_{i \in A_R} \sum_{k \in T_1 \cup T_2} x_{ik}\, \tilde{p}_R + \sum_{i \in A_L} \sum_{k \in T_1} x_{ik}\, \tilde{p}_L + \sum_{i \in A_L} \sum_{k \in P_2} x_{ik}\, \tilde{p}_A \qquad (27)$$

D. The Final Optimization Problem

For the sake of completeness, the final QAP follows the objective function, which is the sum of (25), (26) and (27) as well as the constraints (6), (8), (11), (13), (17) and (23).

V. Simulation and Results

A. Testbench and Data Input

To solve the optimization problem and enable performance benchmarking of the solutions gathered, we implemented a model in a Matlab environment. Based on Matlab 2019b, we use IBM CPLEX 12.9 to solve the underlying problem on a late 2014 model iMac with a 4 GHz Quad-Core Intel Core i7 processor and a memory of 16 GB 1600 MHz DDR3.

One key element of our case study is the integration of a graph with weighted and directed edges in our model, which enables us to determine the total walking time required by passengers to get from one gate or remote stand to another, including optional bus transportation at MUC. Adapting the weights of the edges allows us to simulate peak times, at which longer walking and waiting times are expected. Furthermore, we derive the distance between two stands from another graph, which comprises the ground taxi paths and the stands. The passenger flow matrix provides information about the number of passengers that change from one flight to another, start or terminate their journeys. The data on the scheduled flights is derived from opensource platforms, e.g. an airport's official homepage. The data contains the origin or destination, the aircraft type and usually the registration of a particular aircraft. Observing a whole day of operation allowed us to gain information about the aircraft cycles. We selected August 19, 2019 for our case study and counted a total of 682 scheduled flights departing or arriving, among 151 aircraft[6] using the infrastructure of terminal 2. The aircraft belong to either the Lufthansa Group, including regional partners, or Star Alliance members. Ultimately, a total of 379 assignments have to be determined. Furthermore, the number of passengers must be known. In this case, we do not use real data, which is usually provided by airlines. A database containing aircraft and airline-specific seating configuration properties is included in the testbench instead. We determine the origin, destination and the number of passengers by random in order to gather the passenger flow matrix. For ground handling, 35 buses with a capacity of 90 passengers, 10 aircraft tugs and 18 luxury cars exclusively used for premium customers are available during the whole day of operation.

B. Objective Function and Monetary Costs

The objective function sums up time costs [min], monetary costs [€] and operational penalty costs [-]. As we minimize a single objective function, we can either sum up all costs in relation to each other by introducing weighting factors or regardless of the units. In our work, we assume all weighting factors to be one, which means a time unit equals a monetary unit and a monetary unit equals a penalty unit. The costs based on [13] for using a bus p_B, a luxury car p_C and an aircraft tug p_T during a period of 30 minutes are assumed with $p_B =$ 99.50 € , $p_T = 196.00$ € and $p_C = 55.00$ €. Penalty costs are assumed with $\tilde{p}_A = \tilde{p}_L = \tilde{p}_R = 100$.

C. Heuristics

As stated before, the problem cannot be solved exactly at once for the whole day of operation due to the presence of nearly 1.76 billion binary decision variables to be processed, and for two operational reasons: first, the period from the ONB until the time the stand is safely cleared to accommodate the next aircraft is subject to delay. This makes it hard to pre-plan for an extensive period. Second, a solution has to be determined by a certain time; in this case the determination of a nearly optimal or *satisficing* solution is more desirable. However, for benchmarking purposes, we also executed two runs to find the exact solutions for $m_1 = 10$, $n_1 = 13$ and $m_2 = 14$, $n_2 = 26$. The execution times elapsed are 20.73s and 5,415.34s. In both cases, we already limited the number of available stands to obtain a solution in a reasonable timeframe. The immense increase in execution time demonstrates impressively the need for heuristics to solve the underlying problem. For these reasons, heuristics should be used. A greedy algorithm and heuristics that sort aircraft according to their arrival time are used in [14] to solve the classic AGAP. A stand is assigned to an arriving aircraft as soon as the stand is cleared by the

[6] Distinguished by the aircraft registration.

departing aircraft. On the other hand, heuristics based on simulated annealing and tabu search are used to solve the AGAP in [15]. At this stage, we choose a rather simple method and create subproblems. In every subproblem we consider α aircraft, $\alpha < m$. Then, we determine the optimal assignment for these α aircraft, whereby the aircraft are initially arranged by the ONB in an ascending order. However, we only assign stands to β aircraft, $\beta < \alpha$. The remaining aircraft will be considered again in the next subproblem. This process is repeated until a stand is assigned to every aircraft.

D. Numerical Results

Using the same input data in all simulations of a whole day of operation with 379 aircraft, we executed 4 runs with different values for α and β.

Table 1: Results for the determination of 379 assignments subjected to the parameters α and β

Run	α	β	Value of OF	Execution time [s]
1	6	2	503,716	2,113
2	6	4	503,112	1,204
3	8	4	503,624	4,490
4	10	4	500,694	16,247

The results show that the higher the number of aircraft considered in a subproblem, the longer the execution time will be for a fixed β (run 2 to 4). The reason is that the computing effort rises exponentially with α. However, the value of the objective function presented in the fourth column is only slightly reduced. Furthermore, the selection of a higher α and fixed β may not lead to a reduction in the value of the objective function, since this depends on the solution of the subproblems and the aircraft considered. Even the selection of a smaller β does not necessarily lead to a reduction in the value of the objective function (run 1 compared to run 2). For these reasons, selecting $\alpha = 6$ and $\beta = 4$ might be preferable.

VI. Conclusion and Outlook

In this paper, we introduced new constraints to the extensively discussed AGAP to enable a more efficient assignment in consideration of the aircraft wingspan, gate properties, such as the number of available passenger bridges, and parallel alternate taxi lines. A case study was carried out over a whole day of operation at the five-star Munich Airport. We divided the problem into subproblems using a heuristic, and solved the resulting subproblems in due time. It is fair to say that the validity of the solution depends on the number of aircraft considered and assigned in a subproblem. The number of aircraft should be small from an operational viewpoint, in order to retrieve solutions in due time; however, to ensure better results, as many aircraft as possible should be considered at once. Therefore, in the continuation of this work, we will first focus on determining appropriate weighting factors in the objective function; second we will focus on the heuristics to allow a longer planning horizon with more aircraft, and third, on methods to allow reassignments in the event of disruption. Moreover, we will consider the seat assignment in consideration of the available connecting time and the tradeoff between seat reservation fees offered and charged by airlines and the amount passengers are willing to pay to enhance the feeling of catching an onward flight, when making a reservation for a seat in the front rows near to the closest exit.

References

[1] Skytrax, URL: www.skytraxratings.com, derived: 01/31/2020

[2] Department of Aviation Hartsfield-Jackson Atlanta International Airport, "Monthly airport traffic report," November 2018 to October 2019, URL: www.atl.com/business-information/statistics, derived 01/14/2020.

[3] United States Department of Transportation, Bureau of Transportation Statistics, Data Base, "Atlanta, GA Hartsfield-Jackson Atlanta International Airport," URL: www.transtats.bts.gov/airports.asp, derived: 01/14/2020.

[4] Heathrow Airport Holdings Limited, "Annual report and financial statements for the year ended 31 December 2018," 2019.

[5] Munich Airport, "Annual traffic report 2018, air traffic statistics," 2019.

[6] A. Bouras, M.A. Ghaleb, U.S. Suryahazmaja, and A.M. Salem, "The airport gate assignment problem: A survey," The Scientific World Journal, 2014.

[7] Ruixin Wang, Cyril Allignol, Nicolas Barnier, Jean-Baptiste Gotteland, "Departure management with robust gate allocation," 13th USA/Europe Air Traffic Management Research and Development Seminar, 2019.

[8] J. Guépet, R. Acuna-Agost, O. Briant, and J. Gayon, "Exact and heuristic approaches to the airport stand allocation problem," European Journal of Operations Research, vol. 246, no. 2, pp. 597-608, 2015.

[9] U. Dorndorf, A. Drexl, Y. Nikulin, E. Pesch, "Flight gate scheduling, State-of-the-art and recent developments," Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 584, Kiel, 2005.

[10] A. Drexl, Y. Nikulin, "Mulitcriteria airport gate assignment and Pareto simulated annealing," IIE Transactions, vol. 40, no. 4, pp. 385-397, 2008.

[11] T. Obata, "Quadratic assignment problem: evaluation of exact and heuristic algorithms," Rensselaer Polytechnic Institute, Troy, NY, USA, 1979.

[12] F. Heinicke, A. Simroth, G. Scheithauer, A. Fischer, „A railway maintenance scheduling problem with customer costs," EURO Journal in Transportation and Logistics, Vol. 4(1), pp. 113-137, 2015.

[13] Fraport, "List of service charges, valid as of January 1, 2020," 2020

[14] Ahmed Thanyan AL-Sultan, "The airport gate assignment problem: scheduling algorithms and simulation approach," Doctoral thesis, Okayama University, Japan, 2012.

[15] H. Ding, A. Lim, B. Rodrigues, Y. Zhu, "The over-constrained airport gate assignment problem," Computers & Operations Research 32, pp. 1867-1880, 2015.