*Proceedings of the 34th European Modeling & Simulation Symposium (EMSS), 009*
19th International Multidisciplinary Modeling & Simulation Multiconference

2724-0029 © 2022 The Authors.
doi: 10.46354/i3m.2022.emss.009

# Enhancing a Simulation-Based Allocator Software with Machine Learning and Genetic Algorithms for Improving the Gate Assignment Problem

Alejandro Murrieta-Mendoza[1*], Roberto Salvador Félix Patrón[1], and Miguel Mújica Mota[1]

[1]Aviation Academy, Amsterdam University of Applied Sciences, Rhijnspoorplein 2, Amsterdam, 1091GC, The Netherlands

[*]Corresponding author. Email address: a.murrieta.mendoza@hva.nl

## Abstract

Assigning gates to flights considering physical, operational, and temporal constraints is known as the Gate Assignment Problem. This article proposes the novelty of coupling a commercial stand and gate allocation software with an off-the-grid optimization algorithm. The software provides the assignment costs, verifies constraints and restrictions of an airport, and provides an initial allocation solution. The gate assignment problem was solved using a genetic algorithm. To improve the robustness of the allocation results, delays and early arrivals are predicted using a random forest regressor, a machine learning technique and in turn they are considered by the optimization algorithm. Weather data and schedules were obtained from Zurich International Airport. Results showed that the combination of the techniques result in more efficient and robust solutions with higher degree of applicability than the one possible with the sole use of them independently.

**Keywords:** Allocation; Optimization, Machine Learning, Airport, Genetic Algorithms, Simulation

## 1. Introduction

At an airport, landing and departing aircraft must be assigned to a given stand. At this place, passengers board or deboard the aircraft using gates (which are coupled to a stand) and allows ground handling to execute different operations such as refueling, catering, luggage handling, and other operations.

Not every turnaround(set of arrival and departure flights) can be assigned to any given stand. Aspects to consider when assigning a flight are the aircraft size, domestic or international aircraft (passport control and special terminals), airline preferences, interaction with nearby stands (might block some other stands due to the aircraft size), and other specific constraints given by the airport. Flight schedule, and turnaround time (time required to get the aircraft ready for the outbound flight) are crucial for the gate assignment process. As stands are coupled to gates, they will be used interchangeably in this paper..

Peak hours stress the assignment as many turnarounds request a stand to be assigned. Because stands are a limited resource, it is not always possible to assign stands with gates attached to the terminal. Consequently, aircraft either wait for a stand to become available or the aircraft is assigned to a remote stand. The first is problematic as loaded aircraft wait on the airside of the airport with systems consuming energy, polluting and disrupting the passengers. The latter would require transportation from the terminal to the remote stand, causing discomfort to passengers, crew, and ground handlers as the latter should travel long distances to service an aircraft. Resource availability shortages might worsen the problem as it is expected that aircraft movements will increase by 4.3% annually from 2015 until 2035 (ICAO, 2021).

Every day, airports assign stands to turnarounds considering constraints and aiming for maximizing the services, benefits, and capacity. This problem is known as the gate assignment problem and can be classified as a combinatory problem. The best combination is the one that provides the lowest cost while fulfilling all the constraints.

Even when airlines can solve this problem beforehand, daily operational issues at the airport, at the remote origin/destination airport, or weather conditions make flights arrive/depart late or sometimes earlier than expected. This disrupts the planned gate assignment. It is desirable to estimate these delays to make the gate assignment more robust. This also has an economical cost as in 2019, in the United States only, delays represented 33 billion USD (FAA, 2020).

The Airport Research Center (ARC) developed the commercial stand and gate allocation software CAST, which is used by the aviation industry worldwide. The software uses a heuristic approach to provide a near-optimal and in practice well proved solution for the gate assignment problem.

CAST is capable of allocating flight schedule demands to available aircraft gates and passenger gates at an airport, considering any operational constraints and preferences that are applied in the real world. The software was used to evaluate in how far flight time predictions can lead to a more robust allocation planning. As baseline for this investigation Machine Learning techniques, such as random forest regressor have been used to generate predictions based on historical flight scheduled and weather condition data.

The objectives of this paper are:

- To couple a commercial allocation simulator software to an external optimization algorithm for improving the allocation.
- To increase the gate assignment robustness by using a machine learning arrival/departure time predictor in combination with the optimization algorithm.
- Evaluate potential improvements of the allocation results by using a genetic algorithm.

The rest of the document is as follows. First, a literature review regarding the GAP and delay predictors is provided. Secondly, the algorithm overview is provided. Thirdly, the data and the delay predictor algorithm are explained. Fourthly, the optimization algorithm is presented. Then, results are shown and discussed. Finally, conclusions and future work are introduced.

## 2. State of the art

### 2.1. The Gate Assignment Problem

This problem has been tackled in many different ways for different objectives and using different optimization techniques as thoroughly detailed by (Gülesin Sena Daş, Gzara, & Stützle, 2020). These authors as well detailed the typical GAP model. Another model was presented by (Jiefeng & Bailey, 2001). These type of models normally only take some constraints into account as it is really complicated to mathematical model all constraints. In this sense, the introduction of a simulator overcomes these modeling issues. Another interesting model taking into account the retail activity was presented by (G. Sena Daş, 2017).

The use of metaheuristic algorithms to solve the gate assignment problem to minimize passengers' walking distance and the number of aircraft assigned to remote gates was implemented in (Aktel, Yagmahan, Özcan, Yenisey, & Sansarcı, 2017) using tabu search and simulated annealing. Marinelli, Dell'Orco, and Sassanelli (2015) implemented bee colony optimization and later they combined that algorithm with the biography-based optimization algorithm. Tabu search was implemented by Bi, Wu, Wang, Xie, and Zhao (2020) to maximize the number of passengers boarding and deboarding from the bridge instead of using busses due to remote gate allocation. Ant colony optimization to minimize delays, buffer time and matching aircraft with gates (Zhao & Cheng, 2014) . (Cheng, Ho, & Kwan, 2012) also implemented Genetic Algorithms, Tab search and simulated annealing to minimize passenger walking distance.

The use of genetic algorithms, a class of evolutionary algorithms, to solve the gate assignment problem is not new. Bagamanova and Mota (2020) applied genetic algorithms with Bayesian modeling to successfully create robust assignment schedules while minimizing aircraft waiting time and reducing taxi distances and they also used a similar approach to reduce the emission footprint in airports (Bagamanova & Mujica Mota, 2020). C. H. Yu and Lau (2014) used genetic algorithms and the large neighborhood search focusing on tow costs and passenger distance. Gu and Chung (1999) developed an algorithm based on genetic algorithms able to reassign delayed flights. Bolat (2001) also applied genetic algorithms to minimize the gates idle time. A variation of genetic algorithms called immune genetic algorithm was also proposed by (Wang, Zhu, & Xu, 2014). Kim, Feron, and Clarke (2013) applied a combination of genetic algorithm with tabu search to minimize the aircraft taxiing time and the passengers transit time.

Genetic Algorithms have also been proved to be effective in other types of aviation-related combinatory optimization problems such as aircraft trajectory optimization (Murrieta-Mendoza, Botez, & Félix Patrón, 2015) (Patrón & Botez, 2015), taxi scheduling (Liu & Guo, 2010), air traffic sector regrouping (Delahaye, Alliot, Schoenauer, & Farges, 1995), aircraft arrival sequencing and scheduling for multiple runaways (Hu & Di Paolo, 2009).

Genetic algorithms were selected for this problem due to two main reasons. Because it has been previously applied to solve the gate assignment problem as in (B. Yu, Guo, Asian, Wang, & Chen, 2019) and the as the previously mentioned references giving confidence that it can be applied for this same problem. The second one is Genetic algorithms have proven its efficiency for finding solutions to combinatory problems as well as for sub-optimal solutions.

### 2.2. Flight Delay predictors

Flight delay prediction is a topic that has been largely studied in the literature. Different approaches and applications can be obtained, such as classifying the delays to have a binary or a multi-class classification problem. In (Gui et al., 2020), researchers used neural networks applied to automatic dependent surveillance-broadcast (ADS-B) data to predict flight delays using multi-class classification (delays times coded into different classes). In (Truong, 2021), Bayesian networks augmented naïve Bayes (BNAN) were used to classify delays. Other studies focus on estimating an actual value in minutes. For our approach, it was required to provide delay prediction in terms of minutes to be incorporated to the gate allocation algorithm. In (Qu, Zhao, Ye, Li, & Liu, 2020), three different algorithms were implemented to classify delays: Denoising autoencoder with Levenberg-Marquart algorithm (SDA-LM), autoencoder with Levenberg-Marquart algorithm (SAE-LM), and denoising autoencoder (SDA), where SDA-LM performed better than the other two models.

Rebollo and Balakrishnan (2014) implemented random forest to estimate the delay in minutes at different time horizons. The error variated from 20 min (2 h forecast horizon) to 27.4 minutes for a 24 h forecast horizon.

B. Yu et al. (2019) applied a deep learning approach for flight prediction. In their study, they define the important factors that can be used to predict flight delays. They mention, for example, having data related to air traffic control and information about previous flights delay. If available, this data could increase the prediction capabilities of any algorithm. However, this data was not available for our project. They also compared results for the Mean Absolute Error (MAE) from their proposed algorithms, varying anywhere between 8.41 and 15.56 minutes. This MAE is an evaluation metric used with regression models. The MAE is defined as the mean error value of the individual prediction errors over all the data evaluated instances. The values obtained by Yu et al. are a good benchmark to compare our results. Guo et al. (2021) obtained results with a MAE between 8.73 minutes and 13.85 minutes.

Zoutendijk and Mitici (2021) describe an interesting approach to predicting flight to be implemented to the gate assignment problem as well. Their results show a MAE of less than 15 minutes per flight. In this study, they present a comprehensive literature review and show how other researchers obtain different MAE values, which are dependent on which data is available to the researchers.

Even if good results have already been obtained in the literature, this document presents an approach specific to our study case where a powerful airport allocation software, such as CAST, is coupled with real historical flight data from a large European airport, as well as weather data. Our results improve some of the results seen in the literature for which more data is available and provide a good indication of the actual flight delay in minutes for the purpose of our project.

### 2.3. Main Contribution

The main contributions of this paper are the synergies achieved by coupling of a commercial airport allocation software (in this study CAST ® by ARC) with an optimization algorithm in a general framework where both techniques benefit each other to generate a more efficient solution. An advantage of coupling the allocation software with an optimization algorithm is that the software can reflect the operational planning at an airport in a very high detail, considering actual and empirical restrictions, which are not always revealed in technical documents and sometimes hard to model and the optimization algorithm can focus on suggesting solutions validated in turn by the allocation software that has all the information. Furthermore, the introduction of machine learning predictors provides more realistic arrival/departing times. These times can be incorporated into the schedule to make it more robust.

## 3. Methodology

This section briefly explains the overview of the whole algorithm, available data, the development of the regression algorithm and the optimization algorithm.

### 3.1. The Algorithm Framework

The relationship between the different parts of this work is depicted in Figure 1 where first, the machine learning predictor algorithm estimates arrival and departure dates considering operational and weather parameters as will be discussed in Section 3.2. Predicted flight times are fed to the CAST allocation software and the optimization algorithm. Constant communication is established between the Optimization algorithm and the software which exchange constantly potential stand assignment solutions (the algorithm) and the cost of those solutions (the allocation software).

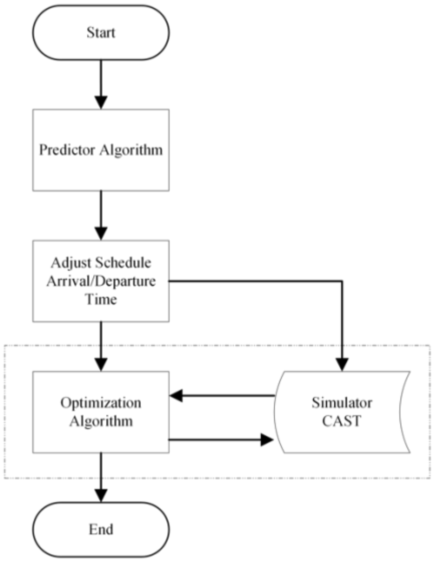


**Figure 1.** Relationship between the predictor algorithm, the optimization algorithm, and the allocation software.

Each one of these parts of the algorithm are briefly explained next.

## 3.2. Data and the Regression – Predictor algorithm

For the data regression algorithm, two data sets were used. One regarding the flights served by the Zurich airport, and the other containing the weather information.

The first dataset contains information for 323,461 flights including departures and arrivals, for the years 2019 and 2020. The relevant data used in this project can be seen in Table 1.

Information such as aircraft type and date and time have been transformed into simplified categories. The aircraft types have been converted into APC (*approach speed categorization*) aircraft codes (B-F), and the weekdays, months, and hours have been extracted from the date and times of flight. Furthermore, the flights have been categorized based on their flight time into Early (3 am – 11 am), Mid (11 am – 7 pm), and Late (7 pm – 3 am) based on peak hours of operation at Zurich airport.

An outlier detection and removal process has been applied when an airline has fewer than one flight per month to, or from the airport, a city pair has fewer than one flight per month or delays above 120 minutes, or flights earlier than 50 minutes based on the historical data distribution.

Weather data has been included to improve the prediction capabilities of our algorithm. The data is obtained every 30 minutes, and it has been linearly interpolated to each scheduled flight time. The data relevant to our algorithm can be seen in Table 2.

**Table 1.** Historical information data.

| Information | Format |
|---|---|
| Date and time (expected) | Datetime |
| Date and time (actual) | Datetime |
| Gate | Category |
| Aircraft type | Category |
| Runway | Category |
| Runway configuration | Category |
| Origin/Destination airport | Category |
| Date and time (expected) | Datetime |

**Table 2.** Weather data.

| Information | Unit |
|---|---|
| Wind direction | Degrees |
| Wind speed | Knots |
| Gust tip | Knots |
| Temperature | Degrees Celsius |
| Dew point | Degrees Celsius |
| Visibility | Meters |
| Precipitation | Code METAR |
| Wind direction | Degrees |

The prediction algorithm has been set up as a regression problem, i.e., the actual arrival or departure time in minutes will be predicted. For this, the time difference is calculated as expected arrival or departure minus actual arrival or departure. A direct solution provides prediction in minutes. As the delay in minutes, data distribution is non-uniform and presents a positive skew (the mode is lower than the median, which is also lower than the mean). A skew in the data could reduce the quality of a regression prediction algorithm. A Random Forest Regressor was implemented using scikit-learning (Pedregosa et al., 2011). Data was split into 80% for training and 20% for testing.

## 3.3. Optimization algorithm

Genetic Algorithms mimic the evolution process of species based on the evolution theory developed by Charles Darwin.. The most economical solution is reported as the candidate's optimal solution. Genetic Algorithms do not guarantee finding the optimal solution.

The general functioning of the optimization algorithm is shown in Figure 2. The blocks using CAST represent processes when the optimization algorithm communicates with CAST.. Each block of this diagrams is explained next.

### 3.3.1. *The Solution Cost*

There are different operational costs associated with assigning a turnaround to a stand. Some examples of these costs are tow operations' cost, aircraft size, airline stand preference, geometric stand dependencies, domestic or international flights, among others. These costs are defined by the user and vary between different airports. These costs are captured into CAST.

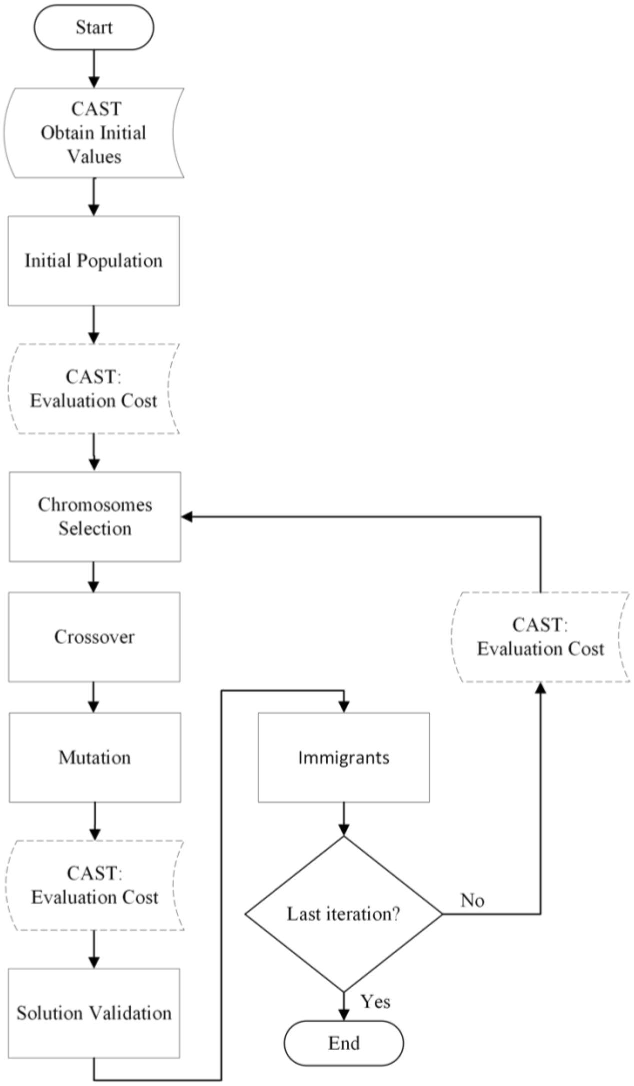


**Figure 2.** The Optimization algorithm schema

This way, the software provides all the elements composing the assignment cost. The total cost of assigning a flight *f* to a stand is given by Eq (1).

$$Assignation\ Cost\ CAST = \sum_{0}^{j} c_j \qquad (1)$$

Where $c_j$ stands for each different operational cost. Constraints such as assigning an over-dimensioned aircraft to a given stand that cannot hold it or assigning an international turnaround to a domestic stand result in a high-cost value. This value indicates that a constrain has been violated, thus the assignment is non-feasible.

An example of the assignment cost computation is given in Table 2 a hypothetical turnaround (FL123 :: FL 124) is assigned to four different stands. The cost of each assignment is computed with three simple costs as an illustration. Assignment costs are given in the row *Assigned Cost Cast* obtained in Table 2 by adding the three sub-costs. In this hypothetical example, stand 202 is not a feasible stand to be assigned to this turnaround as the international stand constraint is not respected resulting in a very high cost . For example, if the maximal cost value found in the assignments different than infinite was of 100, the high value was 100 times this value to be 10000 as in Table 3.

**Table 3.** Turnaround Assignment Costs Computation Example.

| Assign Turnaround FL123::FL124 to stands | | | | |
|---|---|---|---|---|
| Stand | 101 | 202 | 312 | 416 |
| Airline Preference Cost | 100 | 12 | 8 | 6 |
| Geometric Stand Cost | 0 | 0 | 0 | 1000 |
| International Stand Cost | 0 | 1000 | 0 | 0 |
| Assigned Cost CAST | 100 | 1012 | 8 | 1006 |

The total cost from an assignment can then be defined by the sum of all individual feasible cost assignments as in Eq. (2).

$$Assignation\ Cost = \sum_{0}^{i} Assignation\ Cost\ CAST_i \qquad (2)$$

As it is important to identify all the non-feasible flights for comparisons later in the results section, the Total Assignment Cost can be rewritten as in Eq (3) where the number of non-feasible flights is identified under the parameter hold multiplied by a large artificial value.

$$Total\ Assignation\ Cost = \sum_{0}^{i} Assignation\ Cost_i + (\#\ hold * hold\ cost) \qquad (3)$$

Where *hold cost* takes the high value described before. The objective function is then to minimize the Total Assignment Cost.

### 3.3.2. *The Chromosomes*

A solution consists of assigning all turnarounds to a unique stand or identifying a turnaround not assignable due to not having enough stand available. A turnaround is composed of inbound and outbound flights flown by the same aircraft, a tow-in and an outbound flight, or an inbound flight and a tow-off.

If the *turnaround* row order is kept constant A general chromosome can be encoded as in Eq (4).

$$Chromosme = [Stand_1, Stand_2, Stand_3, \dots, Stand_n] \qquad (4)$$

### 3.3.3. *Initial Population*

The initial population is the first part of the optimization algorithm. It consists of many different solutions (feasible or non-feasible). As mentioned before, the feasible solution provided by the allocation software is used as an input for the genetic algorithm. This solution is selected as the first chromosome of the population. This first chromosome is mutated many times to provide a different set of first solutions.

### 3.3.4. Evaluation

The population's chromosomes are evaluated on the cost of assigning turnarounds to stands. Each chromosome cost is computed as with Eq. 3. The most fitted individuals are those that provide the lowest cost. At this stage, it is verified if there are conflicts or violations. The evaluation cost is obtained from CAST along with the held flights. The chromosome is sent to CAST which in turn replies with the cost.

### 3.3.5. Selection

The selection process selects what chromosomes can reproduce to pass their genetic information. For this process, it is desired to select the most fitted chromosomes while giving chances to the least fitted chromosomes to be selected.

The tournament method was chosen as the selection method. Here, chromosomes compete against each other, and winners are determined using each chromosome's cost. The winners of the tournament are the chromosomes used as the base for reproduction.

This type of selection was selected due to the nature of the solution. Modifying a feasible solution quickly degenerates its quality as allocation rules are violated. It is desired to provide the best solution to reproduce. When the population is large, as in this problem, non-feasible solutions would as well emerge victorious bringing diversity to the population and avoiding getting stuck in a local optimal.

### 3.3.6. Reproduction: Crossover

Once the chromosomes are selected, they can reproduce and create a new generation. For this reproduction, the crossover method is selected. The crossover involves selecting two different chromosomes and interchanging their genes.

A random number of neighbor genes in the chromosomes is selected and exchanged within the chromosomes as in Figure 3 to create the new chromosome (offspring).

In an attempt to improve the quality of the offspring, the parents' genes are sorted by time. Gen G1,1 in Figure 3 begins for example at 9:00 AM, gen G1,2 begins at 11:00 AM, and so on. This helps to avoid time conflicts in the new chromosome.

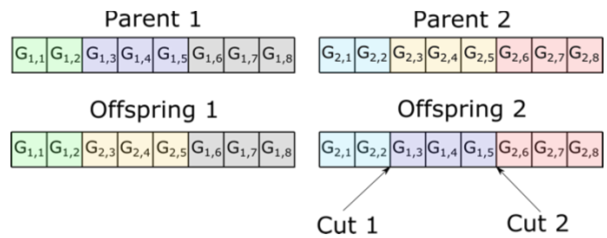


**Figure 3.** The Crossover Process.

### 3.3.7. Mutation

Each new chromosome has a small probability of mutating. In this process, three different strategies were developed. First, genes are selected based on the probability to mutate or not. The second strategy was to select all invalid genes and mutate them, the third and last strategy consisted in selecting a given proportion of the most expensive genes and mutating them. The mutation selection strategy is selected by the user.

The actual mutation can happen in two different strategies. The first is random replacement with statically valid genes where the genes to mutate are randomly replaced by other genes. The second strategy is a random exchange weighted according to statically valid gene costs. Here, the genes are statistically weighted to their cost. The genes with the statically lowest and valid costs have the highest probability of being selected and assigned to the gen to mutate. The mutation replacement strategy is selected by the user.

### 3.3.8. Immigrants

At the end of every iteration, there is a chance that some chromosomes get discarded from the population. This is called immigrants. To replace the discarded ones, new random chromosomes are created and added to the population. This new set of chromosomes are called immigrants and their purpose is to bring diversity to the population to try to avoid the algorithm getting stuck. If after the unique block in Figure 2, the population is below the population size, the lacking chromosomes are added via immigrants.

### 3.3.9. Solution Validation

The stand-allocation solution is sensible. A small change in a solution quickly degrades due to the complex rules in an airport and time constraints. The stochastic nature of the genetic algorithm tends to violate constraints and might create low-quality solutions. To improve these low-quality solutions, they are re-allocated using the heuristic available in CAST.

Finally, it is verified whether the population size is still the required one. If there is underpopulation, chromosomes are added until the specified number is reached. Whereas overpopulation does the opposite. The worst chromosomes are discarded.

## 4. Results

This section exposes results for this paper. First, the results for the prediction algorithm are shown and results are discussed. Then, preliminary results for the optimization algorithm are displayed and shown.

### 4.1. The Delay Prediction Algorithm

Table 4 shows the time of arrival results after the algorithm was ran taking the weather into account and excluding it from the training. This is the difference between the expected arrival/departure time and the real arrival/departure time.

**Table 4.** Prediction Results.

| | Arrivals | Departures |
|---|---|---|
| | Weather Data Excluded | |
| MAE | 10.19 | 11.81 |
| | Weather Data Included | |
| MAE | 9.68 | 11.8 |

The MAE provided by the prediction algorithm can be considered as acceptable when looking at the literature in similar topics. The reason for the high MAE value can be due to the skewness of the data, the low correlation between the delay in minutes and the weather variables, as well as possible low correlation between the other categorical values from the historical data. It can also be seen that it is more difficult to predict the arrival time, which can be because the data is spread out over a larger time span. Key data from previous flight delay for a specific aircraft is missing and would have been an important factor to improve our results (as per the literature review). This aligns with the results obtained by (Belcastro, Marozzo, Talia, & Trunfio, 2016) and (Qu et al., 2020) where the accuracy improved when weather was incorporated into the model.

To visualize the potential effect of the predicted times in scheduling, 5 random schedule days from the Zurich airport were selected from the year 2019. Allocation was carried out using CAST® with the original departure and arrival times. Then, the same days were re-allocated with CAST with the actual departure/arrival times. The number of non-allocated turnarounds due to time overlaps between the planned and the real times were counted. The same process is repeated, but the first allocation is carried out using the predicted schedule times due to delays and early arrivals. Finally, the number of non-allocated flights was counted. Results can be seen in Figure 4.

The number of turnarounds affected using the predicted arrival/departure times (black bars) diminishes compared to the original schedule. Day 3 is a good example of the potential of using predicted times when assigning stands to turnarounds. On this day, 17 turnarounds more than using the predicted were not assigned to any stands. This means that in arriving aircraft passengers must wait on the apron until a stand becomes available to unload the aircraft. This causes discomfort in passengers, and it also affects the planning of ground handling personnel.

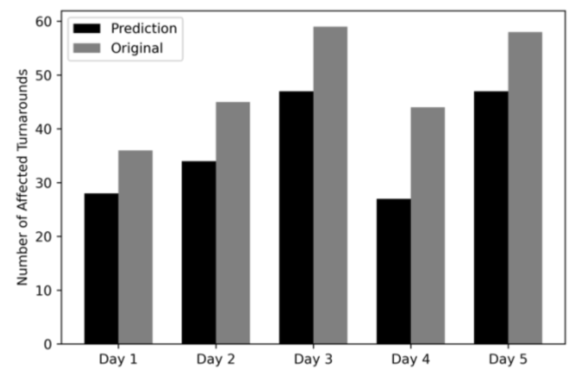


**Figure 4.** Number of affected flights after the re-allocation

### 4.2. The Optimization Algorithm

The optimization algorithm were evaluated using historical schedules from the Zurich International Airport. The scheduled stand assignment was allocated in CAST which in turn provided the cost. Then, the solution was ran with the developed optimization algorithm and cost comparison were carried out.

The selected date for the case study using a prototype of the algorithm described in this paper consisted of 362 turnarounds to be assigned to 177 stands. The historical allocation resulted in 23 turnarounds that were not assigned to stands (*hold*), the cost of the assigned flights was 125,048,355 units.

The allocation provided by the genetic algorithm can be seen in Figure 5. Note that the allocation is given in term of gates instead of stands. In this solution, 23 turnarounds were as well not assigned to stands and the cost of the assigned flights was reduced to 125,046,528 units. The solution provided by the prototype algorithm then corresponds to savings of 1827 units. Other solutions reported savings of 1246 or 918. This means that the optimization algorithm can improve the planned solutions.

Another study was carried out using a mid-size airport with 132 stands and 1000 flights scheduled using the final version of the algorithm detailed in this paper. The number of flights exceeds the number of available stands. A heuristic developed by ARC was used to find the solution to this allocation. The result provided a cost of 14,020,522 units where 14 turnarounds were not allocated. The cost of the allocated flights was 20522. The cost of each non-allocated flight is $10^6$.

The algorithm described in this paper was run 5 times for 10000 generations. The cost provided by the heuristic was the input for the genetic algorithm to improve this result as shown in Figure 6 where the

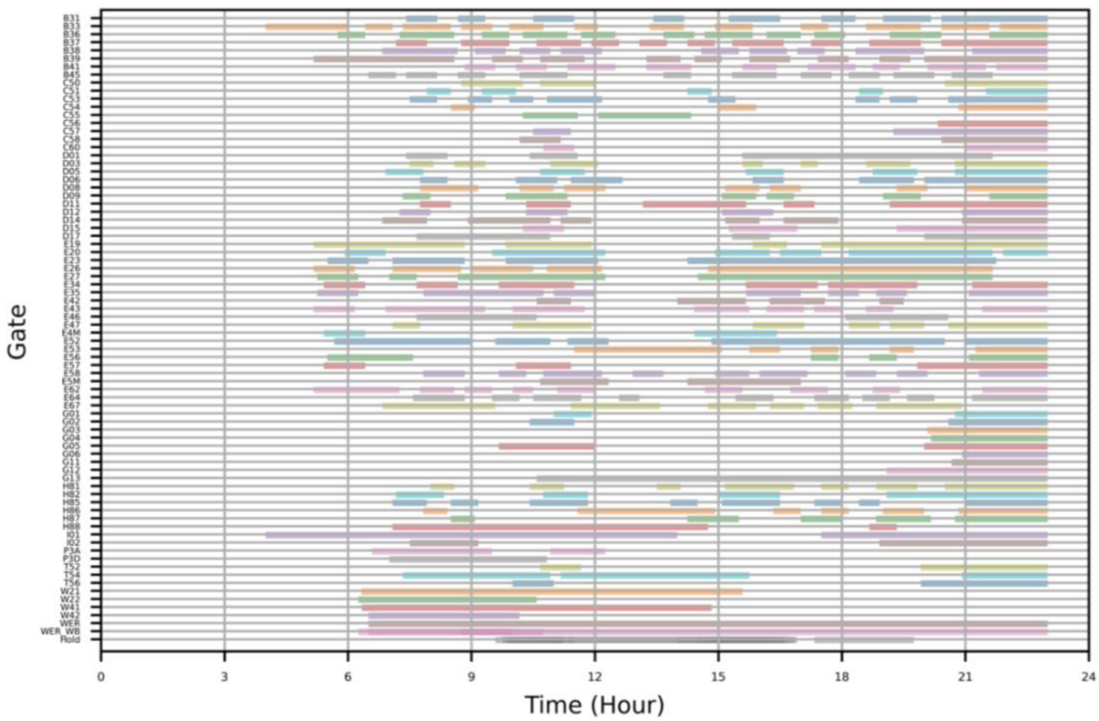


**Figure 5.** The Optimized Flight Schedule for a given day at Zurich International Airport

dotted line is the reference cost of 20,522 units.

This algorithm suggests that Test 2 and Test 3 are more expensive than the reference flight. Test 1 (20,318 units) and Test 4 (20,476 units) are more economical than the reference cost. However, this is because there are additional turnarounds allocated as shown in Figure 7.

The reason why the cost of allocation in Test 2 is higher than others (for example Test 5), could be that turnarounds were allocated in expensive stands. For example, in remote stands or stands where special equipment is required but not always available.

However, this is preferable to keeping an aircraft waiting on the apron. Figure 8 shows the final cost considering the non-allocated flight. Here it can be seen that compared to the reference (14,020,522 units), all flights are more economical than the reference. Even Test 2 reported similar costs to the other Tests. The large difference shown in Test 4 is because, for that test, only 1 additional test was allocated.

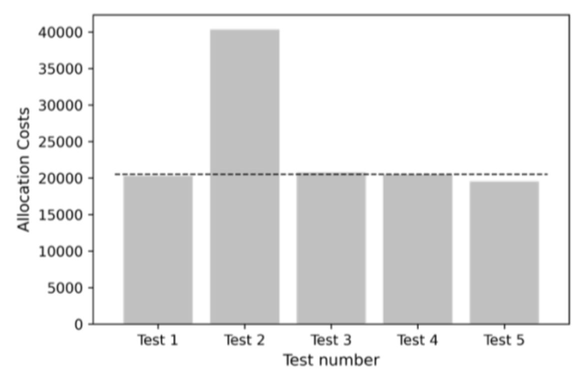


**Figure 6.**Optimization for Different Tests

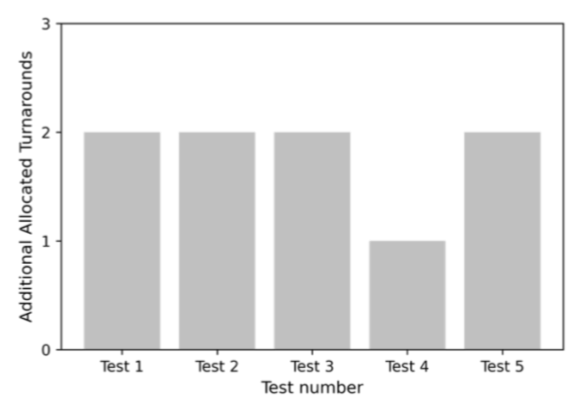


**Figure 7.**Additional turnarounds allocated by the algorithm

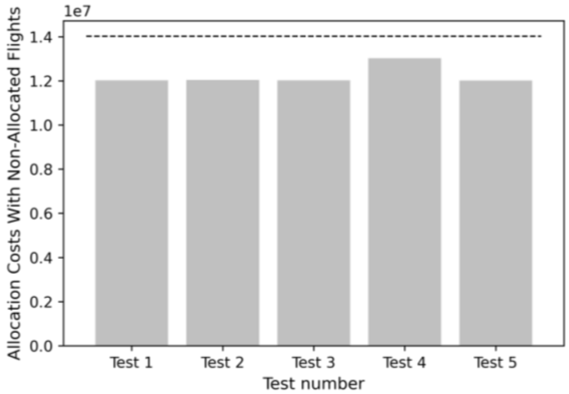


**Figure 8.**Additional turnarounds allocated by the algorithm

## 5. Conclusions

The work presented in this paper combined machine learning, optimization and a commercial allocation software as a cost model for a gate allocation optimization algorithm.

It can be concluded that the allocation software CAST can be used in combination with an optimization algorithm. It was possible to demonstrate that the outcome overcome the solutions provided by the software alone and with the combination of the ML technique for predicting the delay the allocation become more robust. The framework presented allows a more efficient approach that considers practical and relevant constraints opposed to the typical mathematical models available in literature.

The predictor delay algorithm provided good results which allowed to assign realistic predications of departure/arrival times. Results showed that when using the predicted arrival and departure times reduced the number of non-allocated flights when compared against the original schedule with the real historical arrival/departure times.

Future work aims to incorporate new parameters into the prediction model such as the leg number of each flight. For the optimization algorithm, other algorithms will be explored such as Tabu Search to see if better results can be found.

## Funding

This work received funding from the German Ministry for Transport and Digital Infrastructure via the M-Fund Project A-BOOST - NEXT GENERATION ALLOCATION (2021).

## Acknowledgements

The authors would like to thank the Zurich Airport for their help and the weather data provided. The authors would like to thank the Amsterdam University of Applied Sciences for the support in this research, as well as the Dutch Benelux Simulation Society (www.dutchBSS.org) and EUROSIM for disseminating the results of this work.

## References

Aktel, A., Yagmahan, B., Özcan, T., Yenisey, M. M., & Sansarcı, E. (2017). The comparison of the metaheuristic algorithms performances on airport gate assignment problem. *Transportation Research Procedia, 22*, 469-478. doi:https://doi.org/10.1016/j.trpro.2017.03.061

Bagamanova, M., & Mota, M. M. (2020). A multi-objective optimization with a delay-aware component for airport stand allocation. *Journal of Air Transport Management, 83*, 101757. doi:https://doi.org/10.1016/j.jairtraman.2019.101757

Bagamanova, M., & Mujica Mota, M. (2020). Reducing airport environmental footprint using a disruption-aware stand assignment approach. *Transportation Research Part D: Transport and Environment, 89*, 102634. doi:https://doi.org/10.1016/j.trd.2020.102634

Belcastro, L., Marozzo, F., Talia, D., & Trunfio, P. (2016). Using Scalable Data Mining for Predicting Flight Delays. *ACM Trans. Intell. Syst. Technol., 8*(1), Article 5. doi:10.1145/2888402

Bi, J., Wu, Z., Wang, L., Xie, D., & Zhao, X. (2020). A Tabu Search-Based Algorithm for Airport Gate Assignment: A Case Study in Kunming, China. *Journal of Advanced Transportation, 2020*, 8835201. doi:10.1155/2020/8835201

Bolat, A. (2001). Models and a genetic algorithm for static aircraft-gate assignment problem. *Journal of the Operational Research Society, 52*(10), 1107-1120. doi:10.1057/palgrave.jors.2601190

Cheng, C.-H., Ho, S. C., & Kwan, C.-L. (2012). The use of meta-heuristics for airport gate assignment. *Expert Systems with Applications, 39*(16), 12430-12437. doi:https://doi.org/10.1016/j.eswa.2012.04.071

Daş, G. S. (2017). New Multi objective models for the gate assignment problem. *Computers & Industrial Engineering, 109*, 347-356. doi:https://doi.org/10.1016/j.cie.2017.04.042

Daş, G. S., Gzara, F., & Stützle, T. (2020). A review on airport gate assignment problems: Single versus multi objective approaches. *Omega, 92*, 102146. doi:https://doi.org/10.1016/j.omega.2019.102146

Delahaye, D., Alliot, J., Schoenauer, M., & Farges, J. (1995). *Genetic Algorithms for Automatic Regrouping of Air Traffic Control Sectors.* Paper presented at the Evolutionary Programming.

FAA. (2020). *Cost of Delay Estimates 2019* (APO-100). Retrieved from https://www.faa.gov/data_research/aviation_data_statistics/media/cost_delay_estimates.pdf

Gu, Y., & Chung, C. A. (1999). Genetic Algorithm Approach to Aircraft Gate Reassignment Problem. *Journal of Transportation Engineering, 125*(5), 384-389. doi:doi:10.1061/(ASCE)0733-947X(1999)125:5(384)

Gui, G., Liu, F., Sun, J., Yang, J., Zhou, Z., & Zhao, D. (2020). Flight Delay Prediction Based on Aviation Big Data and Machine Learning. *IEEE Transactions on Vehicular Technology, 69*(1), 140-150. doi:10.1109/TVT.2019.2954094

Guo, Z., Yu, B., Hao, M., Wang, W., Jiang, Y., & Zong, F. (2021). A novel hybrid method for flight departure delay prediction using Random Forest Regression and Maximal Information Coefficient. *Aerospace Science and Technology, 116*, 106822. doi:https://doi.org/10.1016/j.ast.2021.106822

Hu, X.-B., & Di Paolo, E. (2009). An efficient genetic algorithm with uniform crossover for air traffic control. *Computers & Operations Research, 36*(1), 245-259. doi:https://doi.org/10.1016/j.cor.2007.09.005

Jiefeng, X., & Bailey, G. (2001, 6-6 Jan. 2001). *The airport gate assignment problem: mathematical model and a tabu search algorithm.* Paper presented at the Proceedings of the 34th Annual Hawaii International Conference on System Sciences.

Kim, S. H., Feron, E., & Clarke, J.-P. (2013). Gate Assignment to Minimize Passenger Transit Time and Aircraft Taxi Time. *Journal of Guidance, Control, and Dynamics, 36*(2), 467-475. doi:10.2514/1.57022

Marinelli, M., Dell'Orco, M., & Sassanelli, D. (2015). A Metaheuristic Approach to Solve the Flight Gate Assignment Problem. *Transportation Research Procedia, 5*, 211-220. doi:https://doi.org/10.1016/j.trpro.2015.01.013

Murrieta-Mendoza, A., Botez, R. M., & Félix Patrón, R. S. (2015). Flight Altitude Optimization Using Genetic Algorithms Considering Climb and Descent Costs in Cruise with Flight Plan Information. *SAE 2015 AeroTech Congress & Exhibition*, 9. doi:https://doi.org/10.4271/2015-01-2542.

Patrón, R. S. F., & Botez, R. M. (2015). Flight Trajectory Optimization Through Genetic Algorithms for Lateral and Vertical Integrated Navigation. *Journal of Aerospace Information Systems, 12*(8), 533-544. doi:10.2514/1.I010348

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res., 12*(null), 2825–2830.

Qu, J., Zhao, T., Ye, M., Li, J., & Liu, C. (2020). Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data. *Neural Processing Letters, 52*(2), 1461-1484. doi:10.1007/s11063-020-10318-4

Rebollo, J. J., & Balakrishnan, H. (2014). Characterization and prediction of air traffic delays. *Transportation Research Part C: Emerging Technologies, 44*, 231-241. doi:https://doi.org/10.1016/j.trc.2014.04.007

Truong, D. (2021). Using causal machine learning for predicting the risk of flight delays in air transportation. *Journal of Air Transport Management, 91*, 101993. doi:https://doi.org/10.1016/j.jairtraman.2020.101993

Wang, L. I., Zhu, Q.-L., & Xu, X.-F. (2014). Optimised assignment of airport gate configurations using an immune genetic algorithm. *Mathematical Structures in Computer Science, 24*(5), e240517. doi:10.1017/S0960129513000765

Yu, B., Guo, Z., Asian, S., Wang, H., & Chen, G. (2019). Flight delay prediction for commercial air transport: A deep learning approach. *Transportation Research Part E: Logistics and Transportation Review, 125*, 203-221. doi:https://doi.org/10.1016/j.tre.2019.03.013

Yu, C. H., & Lau, H. Y. K. (2014). *A Hybrid Algorithm for Solving Gate Assignment Problem with Robustness and Tow Considerations*, Cham.

Zhao, H., & Cheng, L. (2014). Ant Colony Algorithm and Simulation for Robust Airport Gate Assignment. *Mathematical Problems in Engineering, 2014*, 804310. doi:10.1155/2014/804310

Zoutendijk, M., & Mitici, M. (2021). Probabilistic Flight Delay Predictions Using Machine Learning and Applications to the Flight-to-Gate Assignment Problem. *Aerospace, 8*(6), 152. Retrieved from https://www.mdpi.com/2226-4310/8/6/152