Project Checkpoint 1 Report

(GITHUB)

Raymond Li
Georgia Institute of Technology
Atlanta, GA
cli420@gatech.edu

Junyang Tang
Georgia Institute of Technology
Atlanta, GA
jtang375@gatech.edu

Abstract

The optimization of the operations of the airport system relies heavily on passenger simulations. While various models and simulation techniques have been developed, improvements can be achieved by incorporating discrete events simulation (DES), agent-based modeling (ABM), and machine learning approaches. Our model aims to replicate the flow of individual passengers and aircraft through an airport facility while accounting for variables like traveler behavior, staffing levels, and resource limitations. The goal of the project is to give airport management a more useful tool for system optimization by merging various simulation approaches and machine learning algorithms.

1 System Description

We will only focus on two crucial aspects of our airport system simulation due to time constraints: passenger movement and staff movement. These two simulations will incorporate numerous sub-processes and variables to ensure a complete and accurate depiction of the system.

1.1 Passenger movement

There are four sub-processes we plan to implement for our simulation project:
i) **Passenger arrival** The Poisson distribution is well-known for modeling the arrival of passengers at an airport. Suppose a mean arrival rate λ, the probability of observing k arrivals during a time interval Δt is given by:

$$P(k \text{ arrivals in } \Delta t) = \frac{(\lambda \Delta t)^k e^{-\lambda \Delta t}}{k!}$$

We will incorporate the Poisson distribution in our model to capture the stochastic nature of passenger arrivals and help airport managers make informed decisions on resource allocation.
ii) **Check-in** We plan to model different check-in options (e.g. counters, kiosks, online check-ins), taking into account each passenger's preferences, each resource's availability, and queue lengths.
iii) **Security screening** The security screening process can be modeled as an M/M/c queuing system. Passengers arrive at the security checkpoint and wait in line in order to be screened. We assume that arrivals follow a Poisson process, service times follow an exponential distribution, and there are c servers (in this case, security lanes). We will discuss mathematical details for our models in the conceptual model section.
iv) **Boarding** We will model the boarding process as well. Factors such as boarding order and boarding gate allocation will be considered.

1.2 Staff simulation

i) **Staff planning** Simulation of the number of staff available is challenging since this is a result of interactions with passenger volume, flight schedules, and other operational requirements. We assume that the airport will act timely to ensure that adequate staffing levels will be ensured throughout the day.
ii) **Staff movement** We will also need to consider staff movement within the airport. Passengers may experience longer waiting times when airport staff need to switch workstations and has switch changeovers and breaks.

2 Conceptual model

2.1 Passenger movement

We will implement models for the following sub processes to simulate airport operations:

i) **Passenger Arrival** We have already discussed the Poisson distribution for modeling passenger arrivals. We will use this model to generate random passenger arrival patterns based on historical data.

ii) **Check-in** For the check-in process, we can use a multi-server queuing model for each check-in option (counters, kiosks, online check-ins). Suppose a mean arrival rate λ_i and mean service rate μ_i for each check-in option i. We can use the Erlang-C formula to calculate the probability $P(W_i > 0)$ that a passenger has to wait for service at check-in option i. Notice that online check-ins do not require any servers, and we may assume it has an extremely high service rate, and imagine this process as a single queue.

$$P(W_i > 0) = \frac{\frac{(c_i \rho_i)^{c_i}}{c_i! \cdot (1-\rho_i)}}{\sum_{k=0}^{c_i-1} \frac{(c_i \rho_i)^k}{k!} + \frac{(c_i \rho_i)^{c_i}}{c_i! \cdot (1-\rho_i)}}$$

In this formula: c_i is the number of servers (check-in counters or kiosks) for check-in option i.
ρ_i is the traffic intensity for check-in option i, which is the

ratio of the arrival rate.
λ_i to the product of the number of servers c_i and the service rate μ_i. It represents the proportion of time that the servers are busy. After obtaining the probability of waiting time greater than zero, we can obtain the wait time for each passenger:

$$W_{q_i} = \frac{P(W_i > 0)}{c_i \mu_i (1 - \rho_i)} \times \frac{1}{\mu_i}$$

This formula calculates the average waiting time in the queue by dividing the multiplication of the probability of waiting, $P(W_i > 0)$, and the inverse of the service rate, $\frac{1}{\mu_i}$, which represents the average service time, by the product of the number of servers, the service rate, and the idle time (1 - traffic intensity). The formula takes into account the impact of the number of servers, the service rate, and the traffic intensity on the waiting time in the queue.

iii) **Security screening** For the security screening process, we will still use the M/M/c queuing model. The model will incorporate the following factors:

- Arrival rate of passengers at the security checkpoint
- Service rate of each security lane (processing time per passenger)
- Number of security lanes (servers) available

We will also try to adopt Agent-Based Modeling (ABM) method where Each passenger and staff can be assigned unique characteristics.

The arrival of passenger agents at the checkpoint, their behavior in lines, and the assignment of security staff agents to various jobs may all be replicated using ABM. The model can depict how agents engage with one another, for as when a security guard directs a customer to a certain lane or when a passenger requests help with their things.

The ABM can also explore the effects of various policies or interventions, such as introducing a pre-screening program for frequent travelers or adjusting the number of security lanes during peak hours, by simulating these interactions in order to offer insights into the overall efficiency of the security screening process, identify potential bottlenecks, and provide insights into the efficiency of the process as a whole.

iv) **Boarding** As for our simulation of the boarding process, we plan to combine discrete event simulation (DES) to simulate the order of events, agent-based modeling (ABM) to simulate individual passengers/boarding staff as agents with unique traits and behaviors, and machine learning (ML) methods to optimize the boarding process based on past data. We utilize DES to describe events like boarding announcements, passenger arrival at the gate, boarding pass verification, and boarding the plane, and we apply the relevant probability distributions to simulate the times between events. Whereas ML methods like clustering algorithms organize passengers according to characteristics for boarding group assignments and regression approaches estimate boarding times, ABM records agent interactions such as queue formation and staff assistance. These methods help to efficiently allocate resources. These methods are used in our model to give a thorough and accurate depiction of the boarding procedure, allowing for the optimization of boarding tactics, resource allocation, and passenger experience.

2.2 Staff simulation

i) Staff planning We plan to simulate the staff scheduling process as primarily an optimization problem. Since, in real-world scenarios, optimal situations rarely occur, we will also consider adding constraints related to delayed response, required staff break times, or staff skill requirements. Due to time constraint, at the moment, we believe staff planning can be modeled as this optimization problem:

The objective function aims to minimize the total labor cost across all staff types and time periods, represented as $\min \sum_{s \in S} \sum_{t=1}^{T} C_s x_{st}$.

The parameters of the model include the number of time periods T, the demand for staff D_t during time period t, the cost per hour C_s of staff type s, and the set of staff types S. The decision variables are denoted as x_{st}, which represent the number of staff of type s assigned to time period t.

The model has two primary constraints: 1) Staff demand, ensuring that enough staff are assigned during each time period, represented as $\sum_{s \in S} x_{st} \geq D_t, \forall t$, and 2) Non-negativity, ensuring that staff assignments are non-negative, represented as $x_{st} \geq 0, \forall s, \forall t$. By considering these factors and adjusting the model accordingly, we aim to create a more realistic and comprehensive staff scheduling simulation.

2) Staff movement We plan to use ABM to simulate staff movement. Each staff member will be considered as an individual agent with specific attributes and behaviors. Key parameters include a set of all agents (staff members) A, a set of all locations within the airport L, distance d_{ij} between locations i and j, speed v_s of staff type s.

3 Development Platform

We will utilize Python as the primary programming language for implementing our project, given its extensive range of libraries that facilitate the integration of DES, ABM, Machine Learning, and optimization modules. To streamline our development process, we will write our code in a Jupyter Notebook environment. The following libraries will be employed for the various components of our project:

- Discrete Event Simulation: We will use SimPy, a powerful and flexible DES library designed for Python, to model and simulate the airport processes.
- Agent-Based Modeling: Mesa, an ABM framework in Python, will be employed to create complex agent

behaviors and interactions within the airport environment.
- Optimization: We will leverage PuLP and SciPy, two powerful Python libraries for linear programming and optimization, to optimize staff scheduling and other aspects of the project.
- Machine Learning: Scikit-learn will serve as our primary machine learning library, with TensorFlow being utilized for more advanced machine learning models or deep learning, if necessary.

Using this well-rounded set of tools and libraries, we can develop a comprehensive and efficient simulation platform for our airport system project.

4 Literature Survey

Airport operations simulations are crucial for analyzing and optimizing the performance of airport systems. Over the years, numerous researchers have developed models, simulators, and simulation techniques to simulate various aspects of airport operations. While these simulations have been helpful in understanding and improving airport operations, there is still room for improvement.

One widely adopted approach in airport operations simulations is the use of discrete event simulation (DES) models. These models simulate airport operations in real-time by breaking down events into discrete time intervals. For example, Kierzkowski and Kisiel [2] developed a simulation model of the security control system operation at Wrocław Airport, a regional airport situated in south-western Poland in Europe. The paper presents a discrete event simulation model implemented in the FlexSim environment. The model uses input data such as the structure of the security control system, the procedure of implementing the security control process, and the flight schedule to analyze the efficiency of the system.

Another popular approach is agent-based modeling (ABM), which simulates the behavior of individual agents within a system. Several studies have used ABM to simulate airport operations, including Janssen, Sharpanskykh, & Curran [1] who developed operational and security models using ABM to simulate security risks and risk management implementation. Wu and Chen [3], who examined the effects of passenger characteristics and terminal layout on airport retail revenue through an agent-based simulation approach. Their study highlights the potential for utilizing ABM to gain valuable insights into optimizing airport retail revenue by understanding the complex interplay of passenger behavior and terminal design.

While these models have provided valuable insights into airport operations, they have limitations. For example, DES models may oversimplify the behavior of passengers and aircraft, while ABM models may overlook the impact of resource constraints on airport operations. Therefore, there is a need to develop more comprehensive models that integrate multiple simulation techniques. To address these limitations, we plan to develop a new simulation model that integrates both DES and ABM techniques. Our model will simulate the movement of individual passengers and aircraft through an airport system while taking into account factors such as passenger behavior, staff scheduling, and resource constraints. By using a combined DES-ABM approach, we aim to create a more realistic and accurate simulation of airport operations that can help airport managers optimize their systems.

In addition to the integration of multiple simulation techniques, our model will incorporate machine learning algorithms to optimize airport operations. For example, our model can be used to predict passenger flow, identify potential bottlenecks, and optimize staff schedules based on historical data. These predictions can help airport managers make informed decisions and improve the overall efficiency of airport operations.

In summary, while there have been many existing models and techniques used in airport operations simulations, there is still a need for more comprehensive and accurate models. By integrating multiple simulation techniques and incorporating machine learning algorithms, we hope to improve upon existing airport operations simulations and provide airport managers with a more effective tool for optimizing their systems.

5 Dataset Description

5.1 Passenger movement

We acquired one of our datasets from Kaggle. This dataset is from the U. S. Department of Transportation's Bureau of Transportation Statistics which tracks the performance of domestic flights in 2015. This dataset tracks the departure and arrival destinations as well as the delays and cancellations of flights. We will be using this to model boarding for passengers in the airport. We also have data from LAX Airport. This is the data of passenger traffic through specific times in this airport.

6 Initial Results

The code simulates the journey of passengers through the check-in and security checkpoints at an airport, using the SimPy library to create the simulation environment for discrete event systems. The check-in desks and security checkpoints are modeled as SimPy resources with variable capacity, currently set at 3 and 2, respectively, for testing purposes.

The simulation starts with each passenger checking in and then proceeding to the security checkpoint, where they undergo a security check before proceeding to their gate. Random arrival times and service times are generated using the random module and an exponentially distributed arrival time method is used for each passenger.

To make the simulation more realistic, additional features will be included based on data found online. A discrete event simulation will be added to model flights arriving and departing from the airport, and the agent arrival process will be adjusted to better reflect real-world scenarios. A learning algorithm will also be developed to adjust agent arrivals based on queue times, potentially incorporating a value system for flights. Additionally, the simulation will include workers at the airport, who will impact the number of stations open at each checkpoint.

Currently, the simulation provides basic insights into the average time passengers spend at the check-in and security checkpoints. However, with the inclusion of more realistic scenarios and additional features, the simulation will offer more accurate insights and valuable information.

7 Division of labor

Raymond will focus on the airport simulation while Junyang will handle data cleaning and integration. Both of us will work on implementing a learning model with the passengers and collaborate on the project paper.

References

[1] Sybert Janssen, Alexei Sharpanskykh, and R Curran. 2019. Agent-based modelling and analysis of security and efficiency in airport terminals. *Transportation Research Part C: Emerging Technologies* 100 (2019), 142–160.

[2] Artur Kierzkowski and Tadeusz Kisiel. 2017. Simulation model of security control system functioning: A case study of the Wroclaw Airport terminal. *Journal of Air Transport Management* 64 (2017), 173–185. https://doi.org/10.1016/j.jairtraman.2016.09.008

[3] Cheng-Lung Wu and Yen Chen. 2019. Effects of passenger characteristics and terminal layout on airport retail revenue: An agent-based simulation approach. *Transportation Planning and Technology* 42, 2 (2019), 167–186. https://doi.org/10.1080/03081060.2019.1565163