A Stochastic Discrete Event Simulation of Airline Network and Maintenance Operations

Sara Varenna

Delft University of Technology

TU Delft

A Stochastic Discrete Event Simulation of Airline Network and Maintenance Operations

by

Sara Varenna

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday April 13, 2023.

Student number:	5367441	
Project duration:	February 15, 2022 - April13, 2023	
Thesis committee:	Dr. ir. B.F. Santos,	TU Delft, supervisor & chair
	Dr. M. Baptista,	TU Delft, examiner
	Dr. ir. E. van Kampen,	TU Delft, examiner
	ir. F. van de Peppel,	KLM Operations Control

Cover Image: KLM Boeing 777-300 by Johan Scholten

Acknowledgements

I would like to start by thanking my supervisor Dr. Bruno Santos, for the critical and encouraging guidance, and for the multiple attempts to put deadlines on my otherwise never-ending thesis.

Secondly, I would like to thank my former colleagues from KLM. First and foremost, Frank van de Peppel, who has given me the huge opportunity of developing this thesis work at KLM Operations Control, and has supported me with valuable expertise. Then, Paul van Kessel, for always being available to answer my questions, discuss my doubts, and give me critical feedback. Finally, I would like to thank all my colleagues at KLM Operations Control, for sharing their knowledge with me.

Also, I would like to thank my family. My brother Marco for always being a role model for me, and my parents Mario and Flavia for always believing in me, especially when six years ago I decided that psychology wasn't my thing.

Speaking about family, the last two (and a half) years would have not been the same without my family away from home, Frankie, Dane, Yazzy, Fra, Lucas and Siti. Thank you for the joy you gave me throughout both the fun and stressful times.

Last but definitely not least, a huge thank you goes to my boyfriend Antonio. Whether with a hug or with your critical thinking, you always knew how to support me.

Sara Varenna
Delft, March 2023

Contents

List of Figures

List of Abbreviations

A-CDM	Airport Collaborative Decision Making
ABS	Agent Based Simulation
AMP	Aircraft maintenance program
AOG	Aircraft On Ground
ATM	Air Traffic Management
CBM	Condition Based Maintenance
CM	Condition Monitoring
DD	Deferred Defect
DES	Discrete Event Simulation
DOO	Day of Operations
EASA	European Aviation Safety Agency
ETOPS	Extended Range Operations
FAA	Federal Aviation Administration
HT	Hard Time
IATA	International Air Transport Association
MEL	Minimum Equipment List
MILP	Mixed Integer Linear Programming
MPD	maintenance planning document
MRO	Maintenance Repair and Overhaul
NR	Non-Routine
NSRE	Non Safety Related Equipment
OC	On Condition
OC	Operations Control
OTP$_n$	OTP with n minutes slack time
OTP	On Time Performance
SD	System Dynamics
TAT	Turn Around Time

Introduction

Airline operations planning is complex, as it requires managing the resources needed to execute hundreds and sometimes thousands of flights each day. The complexity involved requires dividing the planning of operations into steps executed sequentially, often by different airline departments. This includes, for example, defining a flight schedule, scheduling when the aircraft undergoes maintenance, or defining which aircraft should execute a flight. Although developed separately, the final plans and schedules interact during operations, due to the sharing of resources. These interactions, however, are hard to evaluate, especially due to the uncertainty connected with airline operations.

Some works in the literature have addressed this problem by developing simulation models of airline operations, but they have always focused either on network operations, i.e. all operations connected with executing flights, including crew management, or maintenance operations, i.e. all dynamics regarding executing maintenance on an aircraft, from scheduling maintenance interventions to spare parts management. This thesis work sets out to overcome this separation, by proposing a model capable of simulating and investigating both network and maintenance dynamics.

Given these premises, the research objective that this work tries to reach can be formulated as follows:

To develop a stochastic simulation model of airline network and maintenance operations to evaluate plans and policies from both domains, in an integrated environment.

The remainder of this thesis report is structured as follows: Part I presents the scientific paper that describes the simulation model structure and implementation. Part II includes the literature study developed in support of the research.

I

Scientific Paper

A Stochastic Discrete Event Simulation of Airline Network and Maintenance Operations

Sara Varenna,*

Delft University of Technology, Delft, The Netherlands

Abstract

The complexities associated with airline operations require operations planning to be divided into multiple problems solved sequentially by the respective departments: (1) network planning, and (2) maintenance planning. Furthermore, airline operations take place in an intrinsically uncertain environment, which requires the development of robust plans and the use of effective recovery policies. Despite the close interaction of network and maintenance plans in this dynamic environment, it is current airline practice to evaluate plans from the two domains separately, thus not representing airline operations from an integrated perspective. To this end, a modular, stochastic, discrete event simulation model of airline operations named ANEMOS (Airline Network and Maintenance Operations Simulation) is presented in this paper. The model integrates network and maintenance operations dynamics, allowing the evaluation of plans, policies, and scenarios from both domains. The model is validated using data provided by a major European airline, and it is shown that the simulated results closely resemble the airline's historical operational performance. Finally, the model's capabilities are demonstrated with a case study investigating the effects of adding a second reserve aircraft to a fleet of fifty wide-body aircraft. Results show that the second reserve is capable of reducing cancellations by 55%, but the lost revenue associated with keeping an aircraft non-operational make it a very costly solution, with the avoided costs of disruptions quantified at 6.2% of the lost profit.

1 Introduction

Airline operations often require the planning and management of thousands of flights a week, operated by tens of thousands of crews, while ensuring the airworthiness of hundreds of aircraft through the application of strict maintenance regulations. As if the management of all the required resources was not complex enough, the notoriously low profit margins that airlines generate, which were on average 3.1% in 2019 (IATA, 2019), require operations to be planned as close to optimality as possible, to maximise revenues and minimize costs.

Although the optimization of all operations in one step would theoretically lead to an optimal solution, the complexities involved require airline operations planning to be divided into multiple steps, executed by different airline departments. A first distinction which exists both in literature and in real-life dynamics is made between network and maintenance operations. While network operations involve the execution of flights, from the development of a schedule to the assignment of crews to each flight, maintenance operations deal with guaranteeing the airworthiness of the aircraft, from planning maintenance slots to scheduling which tasks should be executed in each of them. Within both the network and maintenance domains, operations are optimized in steps solved sequentially over time. This process stretches over months if not years, starting with the design of the flight schedule and the planning of heavy maintenance checks, up until the day of operations, with the assignment of each flight to a specific aircraft, and the scheduling of single maintenance tasks into a certain maintenance opportunity.

To add even more complexity, airlines operate in an intrinsically uncertain environment, where disruptions caused by several reasons including bad weather conditions, airspace and airport congestion, or technical problems, can easily spread through the network due to the interconnectivity of resources. On one hand, this requires that operations are planned robustly so that disruptions can be either avoided or mitigated. On the other hand, when disruptions occur, the airline must be capable of applying effective disruption recovery policies to efficiently restore the undisrupted plans.

As a result, the definition of plans and schedules, and the applications of scheduling, rescheduling, or resource management policies become subordinated to inter-departmental negotiations of time management, since the time an aircraft can spend flying, receiving maintenance, and staying on the ground as a form of operational buffer is a limited resource. These negotiations, however, are often based on experience as it is hard to evaluate how plans and policies used in the different domains will affect the other domains, especially in an uncertain environment. As an outcome, the application of plans and policies that are not optimal when framed in full airline operations dynamics can generate high costs for a carrier.

It becomes clear how airlines would benefit from a model capable of investigating how decisions made in a domain would affect operations as a whole. The literature on this subject, however, is scarce. The proposed works tend to focus on either network (Jacobs et al., 2005; Rosenberger et al., 2002) or maintenance (Duffuaa and Andijani, 1999; Öhman et al., 2020; Iwata and Mavris, 2013) operations, simulating the other domain in a simplified manner. This allows them to test some plans and policies, but it does not allow them to capture the full picture of airline dynamics.

To overcome this limitation, this paper presents a modular, stochastic discrete event simulation model of airline operations named ANEMOS (Airline Network and Maintenance Operations Simulation). ANEMOS is designed to simulate the network and maintenance operations of the inter-continental fleet of hub-and-spoke carriers. Its purpose is to provide a framework to test policies, plans, and scenarios involving both network and maintenance operations with the goal of understanding the performance of the system as a whole. Its dynamic structure allows the evaluation of plans and policies at the strategic, tactical, and operational levels. To give some examples, the model can be used to test the performance of a flight schedule in combination with a maintenance schedule, to investigate the effects of anticipating maintenance task execution, or to evaluate how different recovery policies influence the outcome of disruptions. In addition to this, ANEMOS also allows the evaluation of the effects of some external factors such as increased hub congestion on the airline's performance.

The remainder of the paper is structured as follows. Section 2 introduces the problem context, the various steps taken for network and maintenance operations planning, and presents an overview of comparable works found in the literature. Section 3 presents the structure of ANEMOS and explains each of its modules in detail. Section 4 describes how ANEMOS has been implemented into a simulator in collaboration with a major European airline, and introduces a case study used to demonstrate its capabilities. The obtained results are presented in Section 5, and Section 6 draws the conclusions of this work.

2 Problem Context

This section has the objective of providing the reader with the instruments for understanding the remainder of the paper. First, the airline planning process is introduced. As the planning of network and maintenance operations are treated separately both in the literature and in real-life operations, two separate sections are provided for addressing them. Then, an overview of comparable literature is presented, with the objective of identifying the existing gap that this paper tries to fill.

2.1 Network Planning

The network planning process is generally divided into four steps: (1) schedule design, which defines a flight schedule, (2) fleet assignment, which decides which aircraft subtype will cover each flight leg, (3) tail assignment, which assigns a certain sequence of flights and maintenance slots to each aircraft, and (4) crew scheduling, which defines rosters and pairings for cabin and cockpit crew. Given the complexity of each of these problems, a sequential approach is usually used in reality, where each step is considered independently from the others. While this section gives an overview of each of these problems, the interested reader is redirected to Barnhart and Cohn (2004) and Belobaba et al. (2000) for a more detailed review.

Schedule design is the first step in the network planning process. It is a strategic problem that takes place from one year before operations, but small changes to the schedule can be made up until the day of operations (Belobaba et al., 2000). Designing the schedule is a critical task, as it determines the products (the flights) that the airline will sell, and, as a consequence, the market share that the airline will be able to capture (Barnhart and Cohn, 2004). Given the complexities associated with decision-making at the network level, the uncertainty associated with competitor's decisions, airport slots limitations, and passengers' fidelity aspects, schedule planning is often executed by partially modifying the schedule of the previous years, rather than developing a new schedule from scratch (Barnhart and Cohn, 2004). This can be done by, for instance, retiming flights within a window around the originally scheduled time (Levin, 1971), or by pre-determining flights candidate to be cancelled or added, and choosing a certain set of them (Lohatepanont and Barnhart, 2004).

Fleet assignment is a tactical problem that consists of assigning a certain aircraft subtype to each flight leg in the network (Belobaba et al., 2000). The main objective of this planning step is to match supply to demand by minimizing the number of spilled passengers and spoiled seats (Sherali et al., 2006), which respectively represent the passengers that exceed the provided capacity, and the number of seats that remain unsold. Further complications to this analysis regard the possibility of recapturing some of the spilled capacity through similar flights or itineraries (Sherali et al., 2006). Another objective that is kept in mind during fleet assignment regards the minimization of operating costs (Barnhart and Cohn, 2004) where, for example, efficient aircraft should be assigned to longer routes. Constraints that must be considered in the fleet assignment problem include considerations on fleet-route compatibility and routing feasibility, meaning that it must be made possible to select feasible sequences of flights (routings) to be executed by each aircraft.

The **tail assignment** problem is carried out in a tactical-operational context, and it involves the assignment of routings, i.e. sequences of flights, to each aircraft in the fleet (Belobaba et al., 2000). For hub and spoke carriers, it is common practice to combine flights in short sequences of flights (often two flights) starting and ending at a hub named rotations. Rotations can then be assembled into longer routings guaranteeing that, in case of disruptions, operations can be easily restored by cancelling a rotation. Given that the availability of an aircraft to execute a flight is dependent on the scheduled maintenance the aircraft needs to undergo, routings and maintenance interventions are often planned together in the literature in what is known as the maintenance routing problem. In real-life operations, the tail assignment problem and the scheduling of maintenance for each aircraft originate from a negotiation between the maintenance and operations departments.

The last step in the planning framework is the **crew scheduling problem**. This problem (Belobaba et al., 2000) consists of assigning cabin and cockpit crew to flights while ensuring compliance with complex regulations. Given the complexity of the problem, standard practice is to solve it in two steps. The first step, called the crew pairing problem, consists of generating feasible pairings, i.e. sequences of flights with a duration of one to five days. Then, pairings are put together in longer sequences in the crew rostering problem, that generate rosters for each crew member. Given the already ample scope of this research, and that ANEMOS does not simulate crew dynamics, crew-related problems will not be further addressed.

2.2 Maintenance Planning

Maintenance planning consists of scheduling maintenance interventions for each aircraft, in which maintenance tasks can be executed. Given that maintenance is a strongly regulated field, maintenance scheduling must abide by strict regulations, under the surveillance of authorities such as the European Aviation Safety Agency (EASA) in Europe and the Federal Aviation Administration (FAA) in the United States (Regattieri et al., 2015).

In general, maintenance can be divided into scheduled and unscheduled maintenance, where the execution of the former is dictated by authority-approved plans, while the latter is initiated by deviations from nominal, regulated conditions (Ackert, 2010). **Scheduled maintenance** can be divided into three categories of tasks based on the initiation of their execution, hard-time, on-condition, and condition-monitoring (Kinnison and Siddiqui, 2013). Hard-time tasks, also known in the industry as **requirements**, must be executed at fixed intervals, which can be defined in terms of calendar days, flight hours, or flight cycles. When more than one interval type is defined, the interval which is reached first dominates. On-condition tasks are executed whenever a certain condition is reached; often, they derive from a hard-time inspection. Finally, condition-monitoring tasks are triggered by models and part monitoring systems that attempt to anticipate a failure event.

Despite an effort to prevent failures, **unscheduled maintenance** can occur. Whenever this happens, regulations might require that it is corrected immediately, in which case it is known as **Non-Routine (NR)** maintenance, or it can be deferred by a limited amount of time, leading to a **Deferred Defect (DD)**. DDs can be again divided into **Minimum Equipment List (MEL)** items, and **Non-Safety-Related Equipment (NSRE)**, where in the former case limits on task execution are dictated by regulations, while in the latter they are imposed by the operator. MEL items are defined in accordance between aircraft manufacturers and operators, and, being usually part of redundant systems, a failure in a MEL item does not cause a complete loss of airworthiness (Kinnison and Siddiqui, 2013), although an aircraft with open MELs can in some cases be limited in the routes that it can execute (Obadimu et al., 2020). NSREs, as the name suggests, are tasks that do not lead to safety-related issues, but an airline may decide to execute them within a certain time window, such as cabin-related work.

In order to execute tasks, maintenance interventions must be scheduled for each aircraft. The time slots in which a maintenance intervention can be scheduled are commonly called **maintenance slots** in the industry, and the bundle of tasks scheduled within a maintenance slot takes the name of **work package**.

Although each task could be theoretically scheduled individually, airlines often group tasks with a longer, but similar interval into blocks, which are regularly executed during slots known as **letter checks** (Ackert, 2010). Currently, many airlines make use of three types of letter checks, A, C, and D checks, which are usually executed at intervals of respectively 2-3 months, 18-24 months, and 6-10 years (Deng et al., 2020). Given the long interval and duration, which can go from one day for an A-check to weeks for a D-check, letter checks are scheduled months if not years in advance. Since during the execution of letter checks the number of available aircraft for operations is reduced, the schedule of letter checks is used as input during the design of the flight schedule.

Although some requirements are executed with months-long intervals, other requirements must be executed with short intervals, sometimes before each flight can depart. This type of tasks is often executed at an airport aircraft stand in between flights in what is known in the industry as **line maintenance**. Work packages that are executed in line maintenance are generally scheduled the day before operations.

Often, an aircraft will need to undergo maintenance that can neither be executed in line maintenance nor in letter checks. This can happen, for example, when a task requires to be executed in the hangar, but its due date falls before the next scheduled letter check. This can be the case if a DD is found, or for several requirements whose interval is too short to fit in A-checks. To accommodate these cases, airlines often make use of **additional maintenance slots**, which are assigned to the aircraft based on needs. The scheduling of these maintenance interventions is usually the result of a negotiation between the maintenance and operations departments, to guarantee both the availability of resources for the maintenance slot execution and the existence of feasible aircraft routings.

It is essential that tasks are executed timely on each aircraft, in order to retain its airworthiness. In fact, although in some cases the postponement of a task past its due date can be granted by regulators (Shaukat et al., 2020), the regular course of events is that whenever due dates of scheduled or unscheduled maintenance are not met, aircraft are grounded until airworthiness is achieved again. This case is known in the industry as **Aircraft On Ground (AOG)**. Given the operational costs associated with AOGs, which have been estimated to reach $150,000 for a two-hour grounding (Boeing, 2000), airlines always try to prevent tasks from going due. Although this can generally be achieved for requirements, MELs with short due dates or unforeseen events such as bird strikes can still cause such events.

2.3 Irregular Operations

Disruptions regularly occur during airline operations. Various reasons can be at the origin of disruptions, including weather, staff, airport and airspace congestion, and technical reasons, just to cite some. The International Air Transport Association (IATA) provides unique codes (EUROCONTROL, 2019) to identify the cause of delays and disruptions. When discussing disruptions in airline operations, it is important to notice that when delays and disruptions are generated, they propagate in the network as the availability of resources is constrained. Given this fact, a distinction can be made between primary and propagated delays, where **primary delays** have an intrinsic cause, while **propagated delays** are those generated by the delay of connecting resources.

For airlines, disruptions are expensive. Passengers whose itinerary is delayed or cancelled are entitled to compensations according to regulations (EC) No 261/2004 (Cook and Tanner, 2015), which causes the airline to incur in what are known as *hard costs*. A negative passenger experience also causes a loss of fidelity, which translates into passengers choosing to fly with different airlines in the future. This causes a loss of future revenue known as *soft costs*, which are highly non-linear and constitute a big part of the total disruption costs (Cook and Tanner, 2015).

Given the costs associated with disruptions, airlines put in place proactive strategies to limit the number of occurring disruptions and, whenever a disruption occurs, they put in place efficient recovery actions to limit the costs to a minimum. Acting proactively (Abdelghany and Abdelghany, 2018) consists of acting on the operations planning step to generate schedules and routings that are more flexible and robust, and inherently less prone to disruptions.

Increased flexibility can be achieved through two general strategies (Aloulou et al., 2010; Abdelghany and Abdelghany, 2018): the use of *time flexibility*, and the use of *resource flexibility*. **Time flexibility** (Aloulou et al., 2010) refers to the strategical use of buffer time in the schedule, so that delay can be (partially) absorbed, and its spreading mitigated. Buffer time in the schedule can be added in two forms: as flight time buffer, and as turn around buffer. **Resource flexibility**, on the other hand, consists in aligning resources in a way that, in case of disruptions, the schedule can be recovered easily. This includes, for example, using aircraft routings that give many opportunities for swapping aircraft (Ageeva, 2000), or routings that include many short cycles, as opposed to big loops, that can be cancelled in case of disruptions (Rosenberger et al., 2003). This also includes the use of reserve aircraft, which are aircraft that are kept at a base (generally at the hub for hub-and-spoke carriers) during the day of operations, ready to substitute other disrupted airplanes. The use of a reserve aircraft is a very easy (although expensive) way of increasing resource robustness in operations.

Despite every proactive action taken, disruptions can occur. Whenever this happens, a solution must be found to recover operations. Each airline has a dedicated team, usually referred to as Operations Control (OC) that makes decisions on what solution to adopt. Disregarding crew-related causes, typical disruptions that the OC needs to solve include flight delays, aircraft unavailability, and airport congestion Hassan et al. (2021). On the other hand, the tools and strategies that the OC can use include:

- **Delaying a flight**: this is probably the most simple strategy that can be applied, as it involves letting the delay propagate in the network.
- **Flight cancellation**: often used in the form of cancelling rotations (Rosenberger et al., 2004).
- **Swapping aircraft:** it concerns assigning flights or rotations to a different aircraft, when more buffer time can be achieved, or when it allows the cancellation of cheaper flights.
- **Use of a reserve aircraft:** involves an aircraft swap where one of the aircraft is a designated reserve aircraft.
- **Delaying the start time of a maintenance slot**: as it is done for flights, a maintenance slot can start after its planned start time.
- **Postponing maintenance to a different opportunity**: if the tasks that are scheduled in a maintenance slot will not go due, and if the airline's policies allow it, a work package can be fully or partially postponed to be executed in a later opportunity.
- **Speed control:** allows the reduction of block time for delay absorption. As delay absorption capabilities are a function of the time spent flying, it is more effective on longer flights (Marla et al., 2017). However, considerations on fuel consumption and environmental impact must be made.
- **Shortening of ground operations:** Turnaround operations can be executed partially or faster for absorbing delays (Evler et al., 2022).
- **Aircraft ferrying:** it consists of moving an aircraft between stations, without passengers on board. As it is a very expensive solution, it is rarely used in reality (Rosenberger et al., 2003).

2.4 Comparable Literature

Simulation has been used in literature in the field of airline maintenance and network operations planning and optimization. While some works use a simulation framework as an instrument for testing their models (Barnhart et al., 2002; Aloulou et al., 2010; Vos et al., 2015), a few papers focus on the simulation models themselves, with the objective of using them to assess scenarios and to support decision making (Mota et al., 2017; Duffuaa and Andijani, 1999; Jacobs et al., 2005; Iwata and Mavris, 2013). While some of these works are developed by airlines, who are interested in evaluating what-if scenarios in their operations (Jacobs et al., 2005; Duffuaa and Andijani, 1999; Öhman et al., 2020), other models are mainly developed for research purposes, to allow model testing and comparison (Rosenberger et al., 2002).

Jacobs et al. (2005) describe how the operations simulation model OPiuM (from Operational Plan Management) is used by KLM OC. According to the authors, when the Network Department proposes a schedule to OC, OC evaluates the schedule feasibility using OPiuM, before accepting it. The model assesses the schedule by simulating disruptions and recovery, allowing aircraft swaps, the use of a reserve aircraft, reducing maintenance time, and cancelling flights. Further details on the model are not discussed in the paper, which is more focused on model implementation rather than model architecture.

Duffuaa and Andijani (1999), Iwata and Mavris (2013), and Öhman et al. (2020) propose operations simulation models that focus on maintenance dynamics. The goal of Duffuaa and Andijani (1999) is to enable Saudi Arabian Airlines to evaluate the impact of different maintenance policies on airline operations. The presented framework is modular and includes interactive modules such as a planning and scheduling module for maintenance planning, a supply and inventory module for allowing different spare parts management policies, an organization module for stations availability and personnel rules, and an airline operations module to simulate the interaction between maintenance and network operations. The work of Duffuaa and Andijani, however, only presents a framework for airline simulation, while the implementation and interaction of the modules are not explained.

Öhman et al. (2020) also collaborate with a partner airline, but in this case their purpose is to evaluate a specific maintenance scheduling policy. This policy, which they call *frontlog buffer*, implies including in a work package a set of tasks whose due date allows the postponement to a future opportunity, so that if any DD or NR task is found, the frontlog can be postponed, allowing the work package not to exceed its scheduled end time. They develop a DES that includes heuristic algorithms for aircraft routing, and maintenance slots and tasks scheduling and rescheduling, with both requirements and DDs considered. Given the purpose of testing the frontlog buffer policy, the focus of the simulation is on maintenance operations, while network operations are simplified to a sequence of deterministic flight and turnaround times, with the only recovery option being flight delay and reassignment to a different aircraft.

Another approach that simulates mission and maintenance operations is proposed by Iwata and Mavris (2013). However, it is relevant to notice that the focus of this work is on military, rather than airline operations. The model is a DES with a modular structure that includes mission, maintenance, and parts logistics. The model can be used for assessing maintenance policies such as postponing tasks execution and parts logistics.

The work of Jacobs et al. (2005), Duffuaa and Andijani (1999) and Öhman et al. (2020) makes clear that airlines value the insights that operations simulation can provide. Rosenberger et al. (2000, 2001, 2002) and Lee et al. (2003), on the other hand, present SimAir, a model for simulating airline operations developed for use in academia. The objective of this model is to provide researchers with the possibility of testing their models and solutions in a common framework, allowing comparison. This framework has been indeed widely used in the literature for testing models, see for example Lan et al. (2006); Ben Ahmed et al. (2017); Rosenberger et al. (2004). SimAir is a discrete event simulator capable of simulating airline schedules and recovery strategies, including turnaround and block time, weather, influences from other airlines, and crew and passenger flow. However, maintenance is simulated in a simplified manner, by only considering regular maintenance stops and unscheduled maintenance in between flights with a certain probability. Three modules are at the basis of SimAir. The *controller module* keeps track of the simulation and, whenever it detects a disruption, it calls the *recovery module*, which finds a solution to recover operations. Finally, the *events generator module* is responsible for defining stochastic processes' occurrence and duration. The modular structure allows simple adaptation of the model to specific needs, such as the use of different recovery strategies in the recovery module.

Using a more strategic perspective, and orienting their work to both academia and industry, Pohya et al. (2021) present a DES framework capable of evaluating the effects of using specific products, technologies and policies in the long run, throughout the life cycle of an aircraft or fleet. They propose a modular DES model that simulates the complete lifetime of an aircraft, from its purchase to the flights and maintenance executed on it, up until its retirement. The long-term perspective used makes this a very useful model for evaluating the effects of high-level, strategic policies on the overall life cycle of an aircraft. However, due to this wide perspective, both network and maintenance operations are simulated in a simple manner: the aircraft fly whichever flight is departing first, and maintenance slots are not scheduled, but rather executed at fixed intervals or when pre-defined degradation levels of components are reached.

From the presented literature overview emerges that despite simulation models of airline operations have been proposed in the literature, they focus on either network or maintenance operations while modelling the other aspect in a simplified manner. As a consequence, these models are not capable of evaluating the integrated performance of airline operations, in which network and maintenance plans and policies closely interact with each other. In conclusion, it can be stated that a gap in the literature exists in the form of a simulation model of airline operations that includes the simulation of both network and maintenance operations. From the point of view of the industry, such a model could be used to facilitate the negotiations between departments when it comes to defining constraints and requirements for planning and scheduling, to test the obtained plans, and to evaluate the effects of disruption scenarios and of specific recovery policies. In academia, this model would allow testing the effectiveness of proposed optimization models in a stochastic and network-maintenance integrated environment. This would be especially valuable for testing works such as Lagos et al. (2020), who have recently integrated the aircraft routing and maintenance tasks scheduling problem within the same model.

3 Methodology

In this section, the general structure of ANEMOS is presented. As already introduced, ANEMOS is a modular, stochastic, discrete event simulation model of the network and maintenance operations of hub-and-spoke carriers. The model is aircraft-based, meaning that it simulates the dynamics each simulated aircraft goes through including flights and maintenance slots, but it disregards passenger connections and crew rosters. Although the model is developed with a special interest in intercontinental operations, it could be easily adapted to short-haul dynamics.

The input to the model comprises a list of aircraft with their subtypes, a flight schedule with fleet assignment, a list of the maintenance slots available for each aircraft subtype, and a list of requirements and DDs for each subtype. For the DES, the input must describe a deterministic or stochastic measure of each simulated activity. More details on each of the described input is given in the following sections.

Similarly to other models presented in the literature such Rosenberger et al. (2002); Duffuaa and Andijani (1999); Pohya et al. (2021), ANEMOS is developed with a modular structure, which allows freedom for changing and adapting the single modules to a simulation's needs. Figure 1 shows the four modules that make up the simulation, along with the interaction flows that connect them.

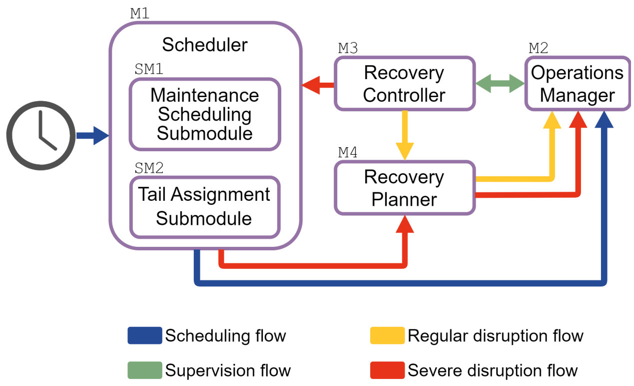


Figure 1: ANEMOS modules and interaction flows betweeen them

A simulation clock regularly calls the *Scheduler* (M1 in Figure 1) which assigns to each aircraft a feasible sequence of maintenance slots and flights to be flown. In practice, this module is made up of two submodules: the *Maintenance Scheduling Submodule* (SM1), and the *Tail Assignment Submodule* (SM2), which are called sequentially. The output of the Scheduler is the input to the second module, the *Operations Manager* (M2), which includes separate discrete event processes for each of the simulated aircraft and manages the dynamics of the simulation. The *Recovery Controller* (M3) monitors the aircraft processes and intervenes whenever a disruption in the original schedule is found. Generally, the recovery controller calls the last module, the *Recovery Planner* (M4) to find an optimal solution for the disruption at hand. However, if the occurring disruption impacts flights falling after the end of the recovery window considered by the Recovery Planner, then the Recovery Controller calls the Tail Assignment Submodule first to provide a long-term solution. This solution is then given as input to the Recovery Planner, that optimally solves the disruption within its considered recovery window.

The following sections describe each module and submodule in detail. Since the scheduler is made up of the sequential call of its submodules, its description is split up into two parts.

3.1 The Maintenance Scheduling Submodule

The Maintenance Scheduling Submodule (MSS) is the submodule responsible for maintenance slots and tasks scheduling. When given a list of available maintenance slots for each aircraft subtype and a list of tasks for each registration, it assigns slots to specific aircraft and schedules the execution of tasks within the assigned slots. The scheduling window of the MSS starts at the end of the recovery window of the Recovery Planner and covers a fixed number of weeks. This section gives an overview of the maintenance tasks and slot types that are simulated and presents an Integer Linear Programming (ILP) formulation of the MSS.

3.1.1 Maintenance Tasks

In the simulation, each aircraft has a set of tasks that need to be scheduled and executed. Each task is characterized by its arrival date, i.e. the date when a DD is found or when the previous instance of a requirement is completed, ready date, i.e. the day from which the task can be executed, and due date, i.e. the date before which the task must be performed. Also, each task has an estimated duration and labour hours for its execution. It is assumed that a set of tasks can be scheduled within a maintenance slot if the duration of each task is shorter than that of the maintenance slot, and if the total number of labour hours associated with the set of tasks does not exceed the slot's maximum allowed labour hours. Finally, the task is characterized by a location where it can be executed. hangar tasks can only be executed in the hangar, while it is assumed that platform tasks can either be executed within the hangar or on the platform.

ANEMOS simulates requirements and deferred defects. Simulated **requirements** are characterized by an interval defined in calendar days. When the input real-life requirements are characterized by an interval expressed in flight hours or flight cycles, this is translated into calendar days assuming a fixed number of flight hours and flight cycles flown per day. This simplification is made necessary by the fact that adjustments to the expected due date of a task based on the effectively flown flight hours and flight cycles would require multiple calls to the MSS, which would not be sustainable from a run-time perspective. Requirements are simulated so that a new instance is generated whenever the previous one is executed. The initialization of requirements is done so that the arrival date of the first instance of a requirement is chosen randomly in the time that goes from one interval before the simulation start date and the simulation start date. The arrival date is then shifted by a fixed number of days so that tasks are not due right at the beginning of the simulation. The scope of considered requirements should be limited based on their interval. Requirements with an interval shorter than the time intercurring between two MSS calls should be excluded since there would be no opportunity to schedule them before their due dates. Also, requirements that are included within letter-checks work packages should be excluded, since the scope of the simulation is limited to maintenance up to A-checks.

Differently from requirements, **deferred defects** are one-off tasks. Their arrival is simulated similarly to how it is proposed by van Kessel et al. (2022). In their work, van Kessel et al. (2022) notice that the arrival of corrective maintenance tasks is not independent of other tasks, but more than one task often arrives at the same time. Therefore, they make use of an exponential distribution fitted to historical data to determine the inter-arrival time in between tasks' arrival events and then use a weighted choice to determine how many tasks should arrive at the same time, allowing a maximum of four concurrent tasks. The tasks' duration and type are then determined, respectively from a fitted exponential distribution and weighted choice. Differently from the work of van Kessel et al., who defines the inter-arrival time at the fleet level and later determines which registration the task regards, ANEMOS considers a separate task inter-arrival process for each simulated aircraft. Also, ANEMOS assumes that tasks only arrive at the beginning of each day. Given the discrete nature of the problem due to this assumption, a weighted choice is used to determine the number of days between DDs' arrival, and again another weighted choice is used to determine how many tasks should arrive on an arrival day. The specific DDs arriving are then sampled from historically arrived DDs. At each call of the MSS, the DDs arriving before the next call are disclosed. This is necessary because disclosing the arrival on each day would require the MSS to be called too often, which would not be feasible for run-time reasons.

As stated before, each task is characterized by its **ready date**, i.e. the date from which it can be executed. In real-life operations, the ready date of a task is generally limited by the availability of parts for its execution. Since requirements are scheduled, repetitive tasks, it is assumed that the required parts for their execution will always be available, and their ready date coincides with their arrival date. For DDs, on the other hand, the ready date depends on parts availability. If a task does not require any part, then it can be executed from its arrival date. On the other hand, if any part is required, then the historical date of part availability is used to compute its ready date. If the part appears available after the execution date of the task, then the historical execution date of the task is used to compute the simulated ready date.

One last notice should be made on the **tasks going due**, i.e. those occasions where a task is not scheduled in time and exceeds its due date. As explained in Section 2.2, in real-life operations an aircraft with a task that is gone due is grounded until the task is executed. However, in practice, the event of a task going due is extremely rare and mainly linked to MEL items with a very short interval. On the other hand, most AOGs are caused by unpredictable circumstances such as bird strikes. Given the limited flexibility available to the model to find solutions for avoiding AOG situations, it is decided to keep the processes of task execution and aircraft grounding independent from each other, so that a task going due does not have direct effects on the simulation dynamics. In order to keep an existing line of requirements, an instance of a requirement that goes due is assumed to be executed at a fixed fraction of the interval of the requirement. The process that manages the grounding of the aircraft is described along with the Operations Manager module.

3.1.2 Scheduled Maintenance Slots

Three types of scheduled maintenance slots are included in the simulation: Line Maintenance (LM) slots, Flexible (Flex) slots, and Mandatory Hangar (MH) slots. The scope of slots included in the simulation is limited to slots with a duration comparable to A-checks, which for wide-body aircraft is around 24 hours. Furthermore, it is assumed that all maintenance is carried out at the hub. Each slot is characterized by its start and end date, its assigned aircraft type, the number of labour hours that can be scheduled within it, and the maximum duration and labour hours of each task that can be scheduled in it.

LM slots include all the maintenance that can be executed on the platform between flights. They are not simulated within the aircraft processes included in the Operations Manager, but they are modelled as weekly 'bins' that include all the tasks scheduled in line maintenance for the week. It is then assumed that a feasible assignment of tasks throughout the week can be made. An LM slot is defined for each aircraft each week, and tasks can be scheduled within it if their ready date and due date allow scheduling between its start and end date. When a task is executed within an LM slot, it is assumed to be executed at a fixed fraction of its interval where, for DDs, the interval is defined as the time between the task arrival and due date. When the computed execution date falls out of the start and end date of the LM slot, the date is moved to the closest time boundary.

The available **Flex slots** are part of the input of the model. They can be defined for one or more weeks, and they are repeated over the simulated time window. The input must also specify the location in which they are executed, i.e. on the platform or in a hangar. These maintenance slots are not necessarily used, but they are only simulated when the Scheduler assigns them to an aircraft, meaning that at least one task is scheduled within them. Differently from LM slots, these slots are simulated within the aircraft processes of the Operations Manager.

MH slots are also defined recursively but, unlike the Flex slots, they are mandatory, i.e. they should always be assigned to an aircraft, independently from the fact that any additional task is scheduled in their work package. This type of slot represents all maintenance slots that are scheduled to execute a set of routine tasks that are not directly considered within the simulation. This includes, for example, A-checks or slots scheduled to execute cabin modifications. For this reason, unlike Flex slots, this kind of slot is characterized by a nominal duration independent of the tasks scheduled within them, and it is assumed that a certain number of labour hours, defined for each slot, can be executed during this time. Differently from airlines' common practice, these types of slots are simulated so that they can be assigned to any aircraft, rather than being assigned to a specific registration in advance. This is the case because when providing a weekly slots list as input to the model, it would be hard to define a feasible sequence of aircraft-slot assignment, given that in practice these slots do not repeat at a fixed pace. Finally, it must be specified that in order to avoid infeasibility issues, the activation of MH slots is not actually forced by the MSS and Recovery Planner, but their cancellation is strongly disincentivized.

3.1.3 MSS: Mathematical Formulation

Sets and subsets		
A		Aircraft
T		Tasks
S		Flex slots and MH Slots
L		Line Maintenance slots
C		Weeks included within the scheduling window
A^s	$\subseteq$ A	Aircraft that can be assigned to slot s
S_{F}	$\subseteq$ S	Flex slots
S^t	$\subseteq$ S	Slots in which task t can be executed
S^c	$\subseteq$ S	Slots in week c
L^t	$\subseteq$ L	Line Maintenance slots in which task t can be executed
T^a	$\subseteq$ T	Tasks of aircraft a
T^s	$\subseteq$ T	Tasks that can be executed in slot s
Decision variables		
$\delta_{A_{as}}$	$\in \{0,1\}$	1 if slot s is assigned to aircraft a, 0 otherwise
$\delta_{T_{ts}}$	$\in \{0,1\}$	1 if task t is scheduled in slot s, 0 otherwise
δ_{U_t}	$\in \{0,1\}$	1 if task t is not scheduled in any slot, 0 otherwise
Parameters		
$W_{S_{sa}}$		Cost of assigning slot s to aircraft a
$W_{T_{ts}}$		Cost of scheduling task t in slot s
W_{U_t}		Cost of leaving task t unscheduled
P_{TL_t}		Labor hours required to execute task t
P_{SL_s}		Maximum labor hours that can be assigned to slot s
M		Large constant

$$\text{Minimize:} \sum_{t \in T} \left(\sum_{s \in S^t \cup L^t} W_{T_{ts}} \delta_{T_{ts}} + W_{U_t} \delta_{U_t} \right) + \sum_{s \in S} \sum_{a \in A^S} W_{S_{sa}} \delta_{A_{as}} \tag{1a}$$

Subject to:

$$\sum_{s \in S^t \cup L^t} \delta_{T_{ts}} + \delta_{U_t} = 1 \qquad \forall t \in T \tag{1b}$$

$$\sum_{a \in A^s} \delta_{A_{as}} \leq 1 \qquad \forall s \in S \tag{1c}$$

$$\sum_{t \in T^a \cap T^s} \delta_{T_{ts}} \leq M \delta_{A_{as}} \qquad \forall s \in S, \forall a \in A^S \tag{1d}$$

$$\sum_{a \in A^s} \delta_{A_{as}} \leq \sum_{t \in T^s} \delta_{T_{ts}} \qquad \forall s \in S_{\mathrm{F}} \tag{1e}$$

$$\sum_{s \in S^c} \delta_{A_{as}} \leq 1 \qquad \forall c \in C \forall a \in A^s \tag{1f}$$

$$\sum_{t \in t^s} P_{\mathrm{TL}_t} \delta_{T_{ts}} \leq P_{\mathrm{SL}_s} \qquad \forall s \in S \cup L \tag{1g}$$

$$\delta_{A_{as}} \in \{0,1\} \qquad \forall s \in S, \forall a \in A^s \tag{1h}$$

$$\delta_{T_{ts}} \in \{0,1\} \qquad \forall t \in T, \forall s \in S^t \tag{1i}$$

$$\delta_{U_t} \in \{0,1\} \qquad \forall t \in T \tag{1j}$$

The mathematical formulation of the MSS is derived from the work of van Kessel et al. (2022). The objective function described in Equation (1a) minimizes the costs associated with assigning a task to a specific slot or unassigning it. The fact that a task is not assigned to any slot is only penalized if the task goes due within the end of the scheduling window. In fact, tasks with a due date that is far in the future will have future opportunities to be scheduled and executed. The cost of assigning a task to a specific maintenance slot varies per task type and slot. Since requirements are recurring tasks, they should be scheduled as close as possible to their due date, in order to minimize the *wasted interval*, i.e. the fraction of requirement interval that is lost due to the anticipation of its execution. On the other hand, DDs should be executed as soon as possible. However, all tasks should be executed with some anticipation with respect to their due date, in order to add a buffer for maintenance postponement in case of disruptions. Following this reasoning, the weight of assigning a task to a maintenance slot follows the function displayed in Figure 2, where the cost of assigning a requirement to a slot generally decreases for later slots, while the cost of assigning a DD increases for slots starting later in time. After the preferred anticipation is reached, the cost of the assignment for all tasks then increases at a higher rate.

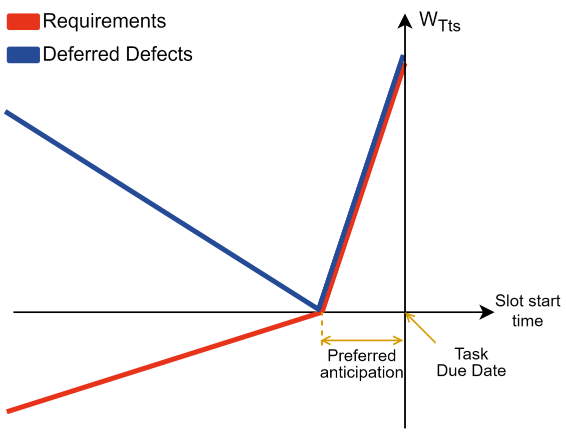


Figure 2: Cost of assigning a task to a slot, based on task type and slot date

The second term of the objective represents the cost of activating a maintenance slot with the exception of LM slots, which are not penalized because the tasks that fit in line maintenance can always be executed in this context. This cost assumes different values for different types of slots. MH slots should, in theory, be mandatory. However, in order to avoid infeasible situations, the use of MH slots is strongly incentivized by giving a high negative value to their activation weight. Flex slots should only be used when necessary, and the use of shorter slots should be preferred. Therefore, the cost of assigning a Flex slot to an aircraft is made up of two components: a fixed value connected to the activation of a slot dependent on its location, and a value proportional to the duration of the slot.

The *fixed scheduling window* is defined as the time that goes from the beginning of the scheduling window to the next scheduled call of the MSS. When MH slots and Flex slots fall within this time window and they are assigned to a different aircraft than the one they were assigned to in the previous call of the MSS, they should receive an additional penalty. The purpose of this is twofold: first, in real-life operations rescheduling maintenance slots close to their start date can cause a waste of resources such as planned man-hours or hangar space, which is why rescheduling a slot close to its start date should be avoided in real life, and, therefore, in the simulation. Second, the MSS works in a close relationship with the Tail Assignment Submodule (TAS) which in previous calls of the Scheduler had found a tail assignment solution based on the previous input of the MSS. Changing the assignment of maintenance slots within the fixed scheduling window could cause incompatibility with the previous plans of the TAS, and, as a consequence, it could cause unnecessary flight cancellations.

Equations 1b - 1j describe the constraints of the model. Constraints (1b) impose that all tasks are either scheduled within a slot or unassigned. Constraints (1c) ensure that no more than one aircraft is assigned to a slot. Constraints (1d) allow a task to be assigned to a slot only if the slot is assigned to the task's aircraft. Constraints (1e) impose that a slot can be activated only if at least one task is assigned to it. Note that this set of constraints does not apply to MH slots since they should always be assigned. Constraints (1f) restrict the assignment of a slot to an aircraft to one per week. Constraints (1g) limit the total labour scheduled within a slot's work package to the slot's maximum allowed labour. This set of constraints, along with Constraints (1b) are the only two sets of constraints that interest LM slots in addition to Flex and MH slots. This is because each aircraft has a pre-assigned weekly LM slot which does not require activation. Finally, Equations 1h - 1j describe the decision variables' domain.

The use of subsets ensures that feasibility constraints of tasks assignment to a maintenance slot and slot assignment to an aircraft are guaranteed. In particular, a maintenance slot can be assigned to an aircraft if their subtype match, while a task can be assigned to a maintenance slot when its duration is shorter than the maximum allowed duration of a slot, when its associated labour hours are within the maximum labour hours that a slot allows per task, when their location match, when the slot falls between the task's ready date and due date, and when the task's aircraft is compatible with the slot in terms of subtype.

3.2 The Tail Assignment Submodule

The Tail Assignment Submodule (TAS) takes the output of the MSS as input and assigns a feasible sequence of flights to each aircraft, considering the preassigned maintenance slots. In addition to flights, the TAS is also capable of assigning a certain number of reserve slots, i.e. time slots during which an aircraft is scheduled to act as a reserve aircraft. Similarly to the MSS, the TAS considers a scheduling window that goes from the end of the recovery window for the Recovery Planner to a fixed number of weeks after the call day.

3.2.1 Rotations and Reserve Slots

Since ANEMOS simulates the operations of hub-and-spoke carries with one hub, the assignment of flights to aircraft is not done at the flight level, but rather at the *rotation* level. As a reminder, a **rotation** is a sequence of flights that departs from the hub and arrives back at the hub. For this reason, a rotation comprises at least two flights, but it can also include more than two, with multiple intermediate stops at outstations. The schedule that is given as input to ANEMOS is a weekly collection of rotations, with their included flights' scheduled departure and arrival times, and assigned fleet type.

As already anticipated, the TAS does not only assign rotations to aircraft but also *reserve slots*, which identify when a registration should act as reserve aircraft. **Reserve slots** do not require any action from the aircraft to which they are assigned and are therefore not simulated within the aircraft processes of the Operations Manager. On the contrary, they act as placeholders to guarantee that the number of desired reserves is present at the hub every day. In addition to this, they identify the aircraft that are acting as reserves, so that specific recovery policies involving the reserve aircraft can be implemented by the Recovery Planner. The number of daily reserve slots and their start and end times are inputs of the model.

In conclusion, the objective of the TAS is to assign to each aircraft a sequence of rotations and reserve slots which is feasible in itself and with respect to the maintenance slots which are pre-assigned to each aircraft. Since the proposed TAS models rotations and reserve slots in the same way, the term *segment* will be used to refer to either one of these entities.

3.2.2 TAS: Mathematical Formulation

Sets and subsets		
R		Rotations and reserve slots (segments)
A		Aircraft
A^r	$\subseteq A$	Aircraft that can be assigned rotation or reserve slot r
OV		set of unordered sets $(r,t), r \in R, t \in R$ where r and t overlap in time
Decision variables		
$\delta_{R_{ra}}$	$\in \{0,1\}$	1 if rotation or reserve slot r is assigned to aircraft a, 0 otherwise
δ_{U_r}	$\in \{0,1\}$	1 if rotation or reserve slot r remains unassigned, 0 otherwise
Parameters		
$W_{R_{ra}}$		Cost of assigning rotation or reserve slot r to aircraft a
W_{U_r}		Cost of leaving rotation or reserve slot r unassigned

$$\text{Minimize: } \sum_{\substack{r\in R\\ a\in A^r}} W_{R_{ra}}\delta_{R_{ra}} + \sum_{r\in R} W_{U_r}\delta_{U_r} \tag{2a}$$

Subject to:

$$\sum_{a\in A^r} \delta_{R_{ra}} + \delta_{U_r} = 1 \qquad \forall r \in R \tag{2b}$$

$$\delta_{R_{ra}} + \delta_{R_{ta}} \leq 1 \qquad \forall (r,t) \in \text{OV}, \forall a \in A^r \cap A^t \tag{2c}$$

$$\delta_{R_{ra}} \in \{0,1\} \qquad \forall r \in R, \forall a \in A^r \tag{2d}$$

$$\delta_{U_r} \in \{0,1\} \qquad \forall r \in R \tag{2e}$$

The objective of the model, presented in Equation (2a) minimizes the costs of the assignment of a segment to a specific aircraft, and of leaving segments unassigned. In particular, the cost of unassigning segments is the highest, since cancelling rotations and reserve slots should always be avoided.

The cost of assigning a segment to an aircraft depends on the nature of the segment. While the cost of assigning a reserve slot is constant, the cost of assigning a rotation to an aircraft is dependent on the aircraft type. In fact, although the fleet assignment is an input to the model, it can happen that a reassignment is necessary to avoid cancellations. Categories of preferred subtypes are defined, so that if a feasible assignment cannot be done within the originally assigned subtype, then a rotation can be assigned to other subtypes according to the preference.

The constraints to the model are included in Equations 2b - 2e. Constraints (2b) are the cover constraints, that impose that each segment is either assigned to one aircraft, or unassigned. Note that an unassigned rotation from the TAS solution is deemed cancelled in the simulation only if it falls before the next scheduled call of the maintenance scheduler. This is done because changes in the slots assignment can lead to changes in the rotations assignments and to cancellations. Constraints (2c) prevent two overlapping segments from being assigned to the same aircraft. The overlapping segments are pre-computed, also considering a buffer before and after each rotation. The feasible assignment of a segment to an aircraft while considering the aircraft's pre-assigned maintenance slots is achieved by reducing the feasible subsets A^r. Reductions of these subsets can also be used to reduce the feasibility of aircraft-route assignments. Constraints (2d) and Constraints (2e) define the domain of the decision variables.

3.3 The Operations Manager

The Operations Manager is the module responsible for the discrete event simulation dynamics. Figure 3 shows a graphical overview of the module and of the processes included in it. The Operations Manager includes three types of discrete event processes: the aircraft process, the AOG process, and the hub disruption process. One aircraft process is built for each of the simulated aircraft, and each process includes the sequence of flights and maintenance the aircraft goes through. In order to determine which rotation

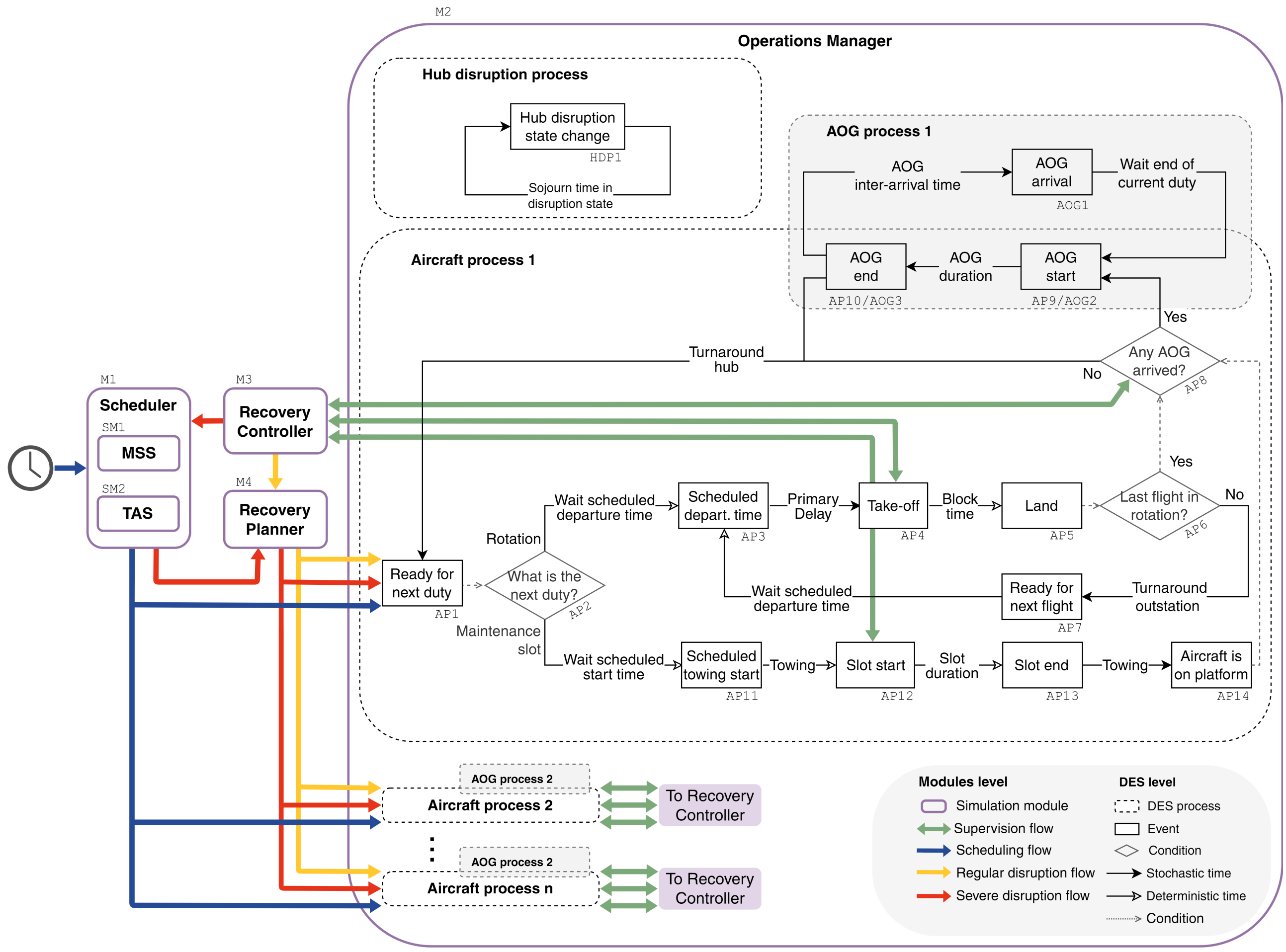


Figure 3: Expansion of Figure 1 that details the Operations Manager and its discrete event processes: the aircraft processes, i.e. the sequences of activities and events each aircraft goes through within the discrete event simulation, the AOG processes, which manage the grounding of the aircraft, and the hub disruption process, which keeps track of the disruption state at the hub

or maintenance slot each aircraft should execute next, each process takes as input the solution of the Scheduler or, in case of disruption, of the Recovery Planner. Note that, in terms of the DES, each aircraft follows simulation dynamics that are independent of the other aircraft. At the same time, an interaction between the processes is indirectly achieved through the Scheduler and the Recovery Planner, which assign and re-assign rotations and maintenance slots at the fleet level.

The second type of process included in the Operations Manager is the AOG process. One AOG process is built for each simulated aircraft, with the objective of determining when an aircraft should be subject to an AOG.

Finally, the last process included in this module is the hub disruption process, of which there is one only copy. This process keeps track of the disruption state at the hub, which influences the primary delays experienced by departing flights. In the following section, each of these processes is described in detail.

3.3.1 The Aircraft Process

This process describes the sequence of activities an aircraft goes through. At the beginning of the simulation, all aircraft are located at the hub, ready to execute the next assigned rotation or maintenance slot, which will be generally defined as *duty* (Block AP1 in Figure 3). If the next duty scheduled for the aircraft (AP2) is a rotation, then the aircraft waits for its scheduled departure time. Once this is reached (AP3), the rotation becomes the aircraft's current duty and its assignment cannot be changed anymore. At this point, the aircraft can experience a primary departure delay, which is summed to the propagated delay the aircraft is experiencing from previously executed duties. This delay represents a combination of all delays that are not technical or propagated delays, including delays related to crew, weather, and congestion, to cite some. When the delay time has passed, the aircraft takes off (AP4), and it reaches its destination after flying for a certain amount of time (AP5). The aircraft undergoes the turnaround activities at an outer station (AP7) and then waits for the scheduled departure time of the next flight in the rotation (AP3). The flight activities are then repeated until the aircraft lands back in the hub (AP5) after executing the last flight in the rotation (AP6). At this point, or during the duration of its ground time at the hub, the aircraft can experience a grounding (AP8), as determined by the AOG process of the corresponding aircraft (AP9, AP10). The aircraft then undergoes turnaround activities at the hub, and it's again ready for the next duty (AP1).

If the next duty of the aircraft (AP2) is a maintenance slot, the aircraft must wait for the scheduled start time. In particular, if it is a hangar maintenance slot the aircraft must wait for the scheduled start of towing (AP11), after which it is towed to the hangar, while if it is a platform slot, then the aircraft simply waits for the scheduled start time of the slot (AP12). In the slot, both scheduled tasks and non-routines are executed, and when the slot ends (AP13), the aircraft is towed back to the platform, if not already there. Once on the platform (AP14), the aircraft must wait for the turnaround time to elapse before it can start flying again (AP1). Before the next duty starts, the aircraft can be grounded, in accordance with the AOG process (AP9, AP10).

The duration of all cited activities, i.e. the time elapsed between two subsequent events, can assume a stochastic or deterministic value based on the simulation's needs. The arrows in Figure 3 define an activity as deterministic or stochastic as implemented in the case study proposed in Section 4.

3.3.2 The AOG Process

As already anticipated when discussing the effects of a task going due, AOG situations are modelled as independent from tasks dynamics. An AOG process is defined for each aircraft, and it interacts with the corresponding aircraft process as shown in Figure 3.

When an AOG arrives for an aircraft(AOG1), i.e. when there is a finding that requires the grounding of the aircraft, the aircraft should stop executing duties. If the aircraft is on the ground at the hub at the time of the arrival, then the AOG starts right away. However, if the aircraft is currently executing a rotation or a maintenance slot, then the start of the AOG is postponed until the end of the execution of the aircraft's current duty. This is necessary to avoid having big disruptions within a rotation while both the Scheduler and the Recovery Planner are rotation-based rather than flight-based. Once the aircraft reaches the hub or it is back on the platform after undergoing scheduled maintenance, the AOG can start (AOG2), and end (AOG3) after a stochastically determined duration. During the duration of the AOG, the aircraft cannot execute any duty. Once an AOG ends, a new one arrives after a certain time named the *AOG inter-arrival time* has elapsed.

In real-life operations, and so in ANEMOS, AOGs can require days to be solved. In the case of long groundings, an AOG slot may overlap with a scheduled maintenance slot, in which case a call to the recovery module would generally lead to the cancellation of the maintenance slot. However, AOGs are opportunities in which the aircraft is on the ground available for receiving maintenance, and there is therefore no reason why a work package should not be executed as scheduled during a grounding. Therefore, it can be assumed that maintenance slots whose duration is shorter than that of the AOG by a certain multiplicative factor can be executed within the AOG slot. If this condition does not apply, then the maintenance slot is postponed to after the AOG time has elapsed.

3.3.3 The Hub Disruption State Process

As previously explained, a departing flight can experience a primary delay that aggregates delays different from technical and propagated delays, which can emerge from the simulation dynamics. This includes delays caused by crew, airport, and weather disruptions, to cite some.

While at outstations flights often depart hours or days apart, at the hub many flights depart within a short period of time. Given that primary departure delays often appear to be linked to airport congestion, the departure delay of flights departing in a short time window cannot be assumed to be independent of one another. In order to account for this, the disruption state of the hub airport is modelled as a discrete-state, discrete-time process, and the primary delay of the departing flights is expressed as a function of the current disruption state.

To describe this process, two parameters must be set: the number of categorical disruption states considered, and the time steps, or time brackets that should be used to discretize time. The disruption state of the hub is initialized at the lowest level at the beginning of the day of operations. Then, the sojourn time in this state, i.e. the number of brackets during which the disruption state at the hub remains unvaried, is determined by sampling from an exponential distribution and by rounding the obtained value to the nearest integer. After the sojourn time has elapsed, the new state is determined by means of a transition probability matrix, which describes the probability of transitioning from each state to each of the other considered states. The new sojourn time can now be computed, and so on. The process continues until it is initialized again at the beginning of the next morning, where a daily initialization is necessary because it is uncommon for an airport to be congested at night or at the early hours of the morning.

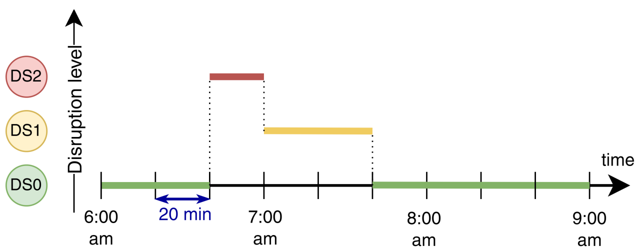


Figure 4: Example of the evolution of the hub disruption state over three hours, for three disruption states, and 20 minutes long brackets.

Figure 4 shows an example of the evolution of the hub disruption state over three hours, when three disruption states are considered, and time is discretized in 20 minutes brackets. The disruption state is initialised at the minimum disruption level (DS0) at 6:00. The state changes after two time brackets and transitions to disruption level DS2, where it remains until 7:00. After that, there is a transition to disruption level DS1, followed by a sojourn time of two time brackets, and so on.

This stochastic process can be easily translated into a discrete event process with one recurring event of state change (HDP1), separated by an activity of duration corresponding to the sojourn time. This makes the process easily integrable within the Operations Manager in the form of the hub disruption process.

3.4 The Recovery Controller

The Recovery Controller supervises the aircraft processes of the Operations Manager to detect when disruptions occur and, when this happens, it requests a recovery action. The Recovery Controller interacts with each aircraft process in correspondence with three events: when a flight takes off (AP4 in Figure 3), when a maintenance slot starts (AP12), and when an AOG arrives (AP8). At these points in time, it estimates the time at which the aircraft will be ready to start its next scheduled duty, given the current state. For a flight, this estimate is done by summing average turnaround times and flight duration, while the duration of maintenance slots and AOGs is assumed to be known from the start.

This estimate of the next ready time of the aircraft is then compared to the currently expected departure time of the next duty: if the estimated ready time falls after the expected departure time by a minimum defined value, then a recovery action is deemed necessary. Notice that the *expected*, instead of the *scheduled* departure time of a duty is considered, due to the fact that both maintenance slots and rotations can be delayed by the Recovery Planner. If previous recovery calls have already delayed a duty, and it is expected that that delay will not be increased, then there is no need to call the Recovery Planner again.

This concept can be better clarified with an example. An aircraft is flying a rotation (Rotation 1) that includes two flights, connecting the hub (Airport A) to an outstation (Airport B), with a flight that lasts on average 9 hours. The following duty the aircraft is scheduled to fly is a rotation (Rotation 2) scheduled and expected to depart at 21:00 UTC. The average turnaround time at the hub is two hours. The aircraft is now departing from Airport B at 11:00 UTC, with a delay of two hours. The recovery controller computes that the aircraft will be ready to fly the next rotation 11 hours from now (9hr of flight plus 2hr of turnaround), i.e. at 22:00 UTC. Since Rotation 2 was expected to depart before this time, the Recovery Controller detects that a recovery action is necessary. Let us now assume that Rotation 1 had already departed from the hub with a two hours delay and that the Recovery Planner had already decided that Rotation 2 was supposed to be delayed by one hour to fit the schedule. In this case, the expected departure time of Rotation 2 would be 22:00 UTC and no recovery action would be requested by the Recovery Controller.

When the Recovery Controller detects that a recovery action is needed, the general procedure involves calling the Recovery Planner to find a solution within a relatively short recovery window, lasting for a time that is in the order of days. However, in some cases, a disruption can be so severe that it affects duties not included within the recovery window of the Recovery Planner. These disruptions generally occur due to the arrival of AOGs that last for days. In these cases, it is necessary to call the TAS to define a long-term solution over the TAS' scheduling window, before the Recovery Planner can be called. Note that a call to the MSS is not necessary, because if any maintenance slot overlaps with such long AOGs, it is automatically included within them.

3.5 The Recovery Planner

The Recovery Planner is called whenever a disruption occurs, with the objective of finding a short-term feasible solution. The model acts on a recovery window with a duration in the order of days, and it must produce a solution that is compatible with the assignment of rotations and slots that do not fall within the recovery window. The implemented Recovery Planner is based on the ILP presented by Vink et al. (2020), which models one parallel time-space network for each aircraft in the considered fleet. Since the Recovery Planner proposed for ANEMOS is rotation-based rather than flight-based, the time-space network is collapsed into a timeline, where the only airport from which arcs generate and terminate is the hub. The allowed recovery options include delaying or cancelling a rotation or a maintenance slot, changing the appointed aircraft to fly a rotation, using a reserve aircraft, swapping maintenance slots, or postponing maintenance slots to a future opportunity.

Nomenclature

Sets and subsets	
A	Aircraft
R	Rotations
DR	Delayed rotations. Pair (r,d) denotes rotation r being delayed by d time units
S	Maintenance slots
DS	Delayed slots. Pair (s,d) denotes maintenance slot s being delayed by d time units
SW	Aircraft swap. Ordered pair (s,t) denotes the feasible swap of maintenance slots s and t
M	Free maintenance arcs. Pair (s,m) denotes the postponement of slot s to free maintenance arc m
G	Ground arcs
N	Nodes
O^{na}	Arcs originating in node n that interest aircraft a
T^{na}	Arcs terminating in node n that interest aircraft a

Sets and subsets (continued)		
A^r	$\subseteq A$	Aircraft a that can be assigned rotation r
A^{rd}	$\subseteq A$	Aircraft a that can be assigned delayed rotation (r, d)
DR^r	$\subseteq DR$	Delayed rotation derived from rotation r
R^a_{orig}	$\subseteq R$	Rotations currently assigned to aircraft a
S^a	$\subseteq S$	Maintenance slots of aircraft a
DS^s	$\subseteq DS$	Delayed maintenance slots derived from slot s
SW^s	$\subseteq SW$	Ordered pairs of maintenance slots that can be swapped where the first slot is s
M^s	$\subseteq M$	Free maintenance arcs that can be used for postponing slot s
G^a	$\subseteq G$	Ground arcs that can be used by aircraft a
N^a	$\subseteq N$	Nodes that interest aircraft a

Decision variables		
$\delta_{R_{ra}}$	$\in \{0, 1\}$	1 if rotation r is assigned to aircraft a, 0 otherwise
$\delta_{DR_{rda}}$	$\in \{0, 1\}$	1 if delayed rotation (r, d) is assigned to aircraft a, 0 otherwise
δ_{CR_r}	$\in \{0, 1\}$	1 if rotation r is cancelled, 0 otherwise
δ_{S_s}	$\in \{0, 1\}$	1 if slot s is kept active, 0 otherwise
$\delta_{DS_{sd}}$	$\in \{0, 1\}$	1 if delayed slot (s, d), 0 otherwise
δ_{CS_s}	$\in \{0, 1\}$	1 if slot s is cancelled, 0 otherwise
$\delta_{SW_{st}}$	$\in \{0, 1\}$	1 if slot s and t are swapped, 0 otherwise
$\delta_{G_{ga}}$	$\in \{0, 1\}$	1 if aircraft a uses ground arc g, 0 otherwise
$\delta_{M_{sm}}$	$\in \{0, 1\}$	1 if slot s is moved to flexible maintenance arc m, 0 otherwise
z_{Γ_a}	$\in \{0, 1\}$	1 if any rotation initially assigned to aircraft a is reassigned, 0 otherwise

Parameters	
$W_{R_{ra}}$	Cost of assigning rotation r to aircraft a
$W_{DR_{rda}}$	Cost of delaying rotation r by d time units and assigning it to aircraft a
W_{CR_r}	Cost of cancelling rotation r
W_{S_s}	Cost of keeping slot s active
$W_{DS_{sd}}$	Cost of delaying slot s by d time units
W_{CS_s}	Cost of cancelling slot s
$W_{SW_{st}}$	Cost of swapping slot s and t
$W_{M_{sm}}$	Cost of postponing slot s to flex maintenance arc m
$W_{G_{ga}}$	Cost of aircraft a using ground arc g
W_{Γ_a}	Cost of changing the rotation assignment of aircraft a
$P_{B_{na}}$	Node balance at node n of aircraft a. Equal to 1 if n is an origin node, to -1 if n is a termination node, and to 0 if n is a central node.

Mathematical Formulation

$$
\begin{aligned}
\text{Minimize: } & \sum_{r\in R}\sum_{a\in A^r} W_{R_{ra}}\delta_{R_{ra}} + \sum_{r\in R} W_{CR_r}\delta_{CR_r} + \\
& + \sum_{(r,a)\in DR}\sum_{a\in A^{rd}} + W_{DR_{rda}}\delta_{DR_{rda}} + \\
& + \sum_{s\in S} W_{S_s}\delta_{S_s} + \sum_{s\in S} W_{CS_s}\delta_{CS_s} + \\
& + \sum_{(s,d)\in SD} W_{DS_{sd}}\delta_{DS_{sd}} + \\
& + \sum_{(s,t)\in SW} \frac{1}{2} W_{SW_{st}}\delta_{SW_{st}} + \\
& + \sum_{(s,m)\in M} W_{M_{sm}}\delta_{M_{sm}} + \\
& + \sum_{g\in G^a}\sum_{a\in A} W_{G_{ga}}\delta_{G_{ga}} + \\
& + \sum_{a\in A} W_{\Gamma_a} z_{\Gamma_a}
\end{aligned} \tag{3a}
$$

Subject to:

$$
\sum_{a\in A^r} \delta_{R_{ra}} + \sum_{a\in A^{rd}} \sum_{\substack{(r,d)\\ \in DR^r}} \delta_{DR_{rda}} + \delta_{CR_r} = 1 \quad \forall r \in R \tag{3b}
$$

$$
\begin{aligned}
& \delta_{S_s} + \sum_{\substack{(s,d)\\ \in DS^s}} \delta_{DS_{sd}} + \delta_{CS_s} + \\
& \quad + \sum_{\substack{(s,m)\\ \in M^s}} \delta_{M_{sm}} + \sum_{\substack{(s,t)\\ \in SW^s}} \delta_{SW_{st}} = 1 \qquad \forall s \in S
\end{aligned} \tag{3c}
$$

$$
\begin{aligned}
& \sum_{\substack{r\in\\ R\cap O^{na}}} \delta_{R_{ra}} - \sum_{\substack{r\in\\ R\cap T^{na}}} \delta_{R_{ra}} + \sum_{\substack{(r,d)\in\\ DR\cap O^{na}}} \delta_{DR_{rda}} + \\
& \quad - \sum_{\substack{(r,d)\in\\ DR\cap T^{na}}} \delta_{DR_{rda}} + \sum_{\substack{s\in\\ S\cap O^{na}}} \delta_{S_s} - \sum_{\substack{s\in\\ S\cap T^{na}}} \delta_{S_s} + \\
& \quad + \sum_{\substack{(s,d)\in\\ DS\cap O^{na}}} \delta_{DS_{sd}} - \sum_{\substack{(s,d)\in\\ DS\cap T^{na}}} \delta_{DS_{sd}} + \\
& \quad + \sum_{\substack{(s,m)\in\\ M\cap O^{na}}} \delta_{M_{sm}} - \sum_{\substack{(s,m)\in\\ M\cap T^{na}}} \delta_{M_{sm}} + \\
& \quad + \sum_{s\in S^a} \sum_{\substack{(s,t)\in SW^s\\ t\in O^{na}}} \delta_{SW_{st}} - \sum_{s\in S^a} \sum_{\substack{(s,t)\in SW^s\\ t\in T^{na}}} \delta_{SW_{st}} + \\
& \quad + \sum_{\substack{g\in\\ G\cap O^{na}}} \delta_{G_{ga}} - \sum_{\substack{g\in\\ G\cap T^{na}}} \delta_{G_{ga}} = P_{B_n a} \\
& \qquad \forall a \in A, \forall n \in N^a
\end{aligned} \tag{3d}
$$

$$
\sum_{r\in R^a_{\text{orig}}} (\delta_{R_{ra}} + \delta_{CR_r}) + \sum_{\substack{(r,d)\\ \in DR^r}} \delta_{DR_{rd}} \ge \left|R^a_{\text{orig}}\right| (1 - z_{\Gamma_a}) \qquad \forall a \in A \tag{3e}
$$

$$
\delta_{SW_{st}} = \delta_{SW_{ts}} \qquad \forall (s,t) \in SW \tag{3f}
$$

$$
\delta_{R_{ra}} \in \{0, 1\} \qquad \forall r \in R, \forall A \in A^r \tag{3g}
$$

$$\delta_{DR_{rda}} \in \{0,1\} \qquad \forall (r,d) \in DR, \forall A \in A^d r \tag{3h}$$

$$\delta_{CR_r} \in \{0,1\} \qquad \forall r \in R \tag{3i}$$

$$\delta_{S_s} \in \{0,1\} \qquad \forall s \in S \tag{3j}$$

$$\delta_{DS_{sd}} \in \{0,1\} \qquad \forall (s,d) \in DS \tag{3k}$$

$$\delta_{CS_s} \in \{0,1\} \qquad \forall s \in S \tag{3l}$$

$$\delta_{SW_{st}} \in \{0,1\} \qquad \forall (s,t) \in SW \tag{3m}$$

$$\delta_{G_{ga}} \in \{0,1\} \qquad \forall a \in A, \forall g \in G^a \tag{3n}$$

$$\delta_{M_{sm}} \in \{0,1\} \qquad \forall (s,m) \in M \tag{3o}$$

$$z_{\Gamma_a} \in \{0,1\} \qquad \forall a \in A \tag{3p}$$

Equation (3a) describes the objective of the ILP, which is a cost-minimization of all considered recovery options. The cost of assigning a rotation to an aircraft is dependent on the aircraft type, and, in particular, on aircraft subtype preference groups, similarly to what is done in the TAS. Delayed rotations are modelled as copies of the original rotations, departing and arriving at later nodes, in an approach that was initially proposed by Levin (1971). For this reason, a rotation can only be delayed by pre-defined discrete amounts of time. The cost of delaying a rotation and assigning it to an aircraft depends both on the aircraft subtype and on the duration of the delay. The latter component is assumed to vary linearly with the delay duration. In addition to the base cost of the assignment, the weight should be increased whenever the assignment does not correspond to the original assignment of the rotation, in order to favour keeping the plan as it is.

Similarly to rotations, slots can be executed as originally planned, delayed or cancelled. However, differently from rotations, slots cannot be freely reassigned, but they can only follow simple swap patterns. This means that if three aircraft A, B, and C are respectively assigned maintenance slots a, b, and c, it is possible to do a swap such as $A \to b, B \to a$, but not a swap such as $A \to b, B \to c, C \to a$. Furthermore, cancelling a slot should, in principle, not be allowed. This is because leaving some tasks un-executed would lead to the grounding of the aircraft. However, in order to avoid infeasible situations, cancelling a slot is allowed, at a very high cost. Another recovery option included in the model is the possibility of postponing maintenance slots to *flexible maintenance arcs*, which are arcs generated any time an aircraft is on the ground at the hub for a time that allows fitting a maintenance slot. Note that, especially for what concerns hangar slots, the assumption of always having the available resources to provide maintenance to an aircraft is a strong assumption, that does not necessarily represent actual operations. However, this option can be used to investigate scenarios of maximum maintenance flexibility.

Ground arcs connect other activity arcs, and represent the time during which an aircraft is on the ground in between duties. In general, ground arcs can be assigned a cost of zero, unless particular conditions require otherwise.

Finally, a binary slack variable is used to avoid unnecessary involvement of an aircraft in the recovery strategy. If any of the rotations previously assigned to an aircraft is reassigned to a different one, then the slack variable is activated and a penalty applies. This term acts in parallel with respect to the additional weight incurred by each re-assigned rotation, and it allows for a reduction of the number of aircraft involved in the recovery solution.

The model is subject to constraints defined in Equations 3b -3p. Constraints (3b) ensure that a rotation is either executed as scheduled, delayed or cancelled. Constraints (3c) impose that a slot is either executed as planned, delayed, cancelled, swapped or postponed to a flexible maintenance arc. Constraints (3d) ensure the balance of the network. For each aircraft, the origin and termination nodes are defined based on their assigned rotations and slots before any recovery action is taken. In particular, the origin node corresponds to the time at which their current (or last) duty is expected to end, or, if an AOG has arrived, it corresponds to the expected end time of the AOG. The termination node is imposed on the time when the first duty not included within the recovery window is scheduled to start. All central nodes are put in correspondence with duties arriving or departing, always considering a possible buffer before and after the duty for either turnaround or towing operations. At each node, the sum of entering and exiting arcs must be equal to the balance of the node, i.e. 1 for origin nodes, -1 for termination nodes, and 0 for central nodes. Constraints (3e) activate the slack variables that avoid the re-assignment of rotations for each aircraft. Constraints (3f) is required given the formulation of the slot swaps. In fact, the decision variables associated with slot swaps refer to ordered pairs of slots (s,t) which can be swapped. This constraint imposes that if slot s is swapped with slot t, then slot t is also swapped with slot s. Finally, the Constraints (3g) - 3p define the domain for the included decision variables.

As it is done for the MSS and the TAS, also in this case the limitations on assignments are imposed by means of subsets rather than through explicit constraints. The assignment of a rotation or of a delayed rotation to an aircraft is not permitted when the rotation falls out of the aircraft's origin and termination nodes. Also, the re-assignment of a rotation or of a delayed rotation to a different aircraft is not permitted when a minimum anticipation, i.e. the time intercurring between the current time and the time when the rotation is expected to depart, is not guaranteed. The reduction of subsets can also be used to limit the types of aircraft that can fly specific routes.

Two Flex slots can be swapped when their scheduled work package would fit in the destination slot in terms of duration and scheduled labour, and when no task in either work package would go due before the new assigned slot's scheduled start. Also, there should be a match in aircraft subtype, slot location, and slot type for the swap to be possible. Flexible maintenance arcs are specifically generated for each maintenance slot assigned to each aircraft so that the expected duration of the slot fits within the maintenance arc.

Finally, it can be observed that reserve slots are not explicitly considered by the model. However, the objective function weights and the reduced subsets can easily be used to impose airline-specific policies. For example, the assignment of a rotation to an aircraft when the latter has a reserve slot scheduled can be disincentivized by assigning a higher cost of the specific rotation-aircraft assignment weight. As another example, if an airline policy requires a reserve aircraft to be available at the beginning of each day of operations, even at the cost of cancelling rotations scheduled in the coming days, the subsets A^r can be reduced to prevent the assignment of rotations to aircraft when this would cause the overlap with reserve slots scheduled on the coming days.

4 Case Study

This section presents a general implementation of ANEMOS that was developed in collaboration with a major European airline and introduces an example case study used to validate the model and test its capabilities. The proposed case study investigates the effects of adding a reserve aircraft to our partner airline's fleet in different operational disruption scenarios.

4.1 An Implementation of ANEMOS

ANEMOS has been implemented into a simulation tool. The simulation has been developed in collaboration with a major European hub-and-spoke carrier, which provided the historical data necessary to define all input parameters and distributions that are described in this section. The fleets considered in the implementation include the following aircraft subtypes: Boeing 787-9, 787-10, 777-200, and 777-300.

4.1.1 MSS

Before identifying specific parameters for the MSS, some parameters regarding the scope and arrival of the simulated task must be defined. Requirements are limited to those having an interval between 15 days and three months, as this is the interval of A-checks for the considered fleets. The simulation of the arrival of DDs requires the definition of the limits of two weighted choices. A weighted choice between one to seven days is used to determine the number of days between DDs' arrival, and again a weighted choice between one to five is used to determine how many tasks should arrive on an arrival day. Different probabilities for the weighted choices and different pools of historical DDs are used for each simulated aircraft fleet.

The MSS is called once a week, and its scheduling window goes from 3 days to three weeks after its call day. The weights of the parameters of the MSS are determined following the logic presented in Section 3.1. The values to be given to each parameter of the objective function are determined using a hierarchical logic (van Kessel et al., 2022), which involves defining a hierarchy of importance associated with each scheduling decision. The hierarchy is defined as follows, from highest to lowest importance: (1) assignment of MH slots, (2) assignment of maintenance tasks, (3) maintaining the assignment of slots in the fixed scheduling window unchanged, (4) reducing the use of ground time, i.e. avoiding the unnecessary activation of maintenance slots, (5) scheduling tasks with the preferred anticipation.

In the case at hand, the value of some objective function weights can be defined by one component, such as the cost of leaving a task unassigned (W_{U_t}). In other cases, the value of these weights derives from the combination of multiple cost components, such as in the case of the weight of assigning a slot to an aircraft, which depends both on slot duration and location. For the latter cases, some *bridging costs* (C) are defined in the hierarchy and later assembled into the final objective function weights (W). The hierarchy is translated into the following weights:

C_{MH}	-5×10^{12}	Activating a MH slot.
W_{U_t}	5×10^{7} 10^{7}	Unassigning task t, if t is a dd Unassigning task t, if t is a requirement
C_{fix_s}	10^{6}	Changing the assignment of a slot s in the fixed scheduling window. 0 if s is not in the fixed window.
C_{S_s}	10^{5} 10^{4}	Activating flex slot s, if s is a hangar slot Activating flex slot s, if s is a platform slot
C_{sd}	10^{3}	Additional cost per hours of slot duration, when a slot is activated
C_{ant_t}	10^{2} -4×10^{1}	Anticipating the execution of requirement t with respect to its due date by one day Anticipating the execution of dd t by one day

The cost W_{U_t} is higher for deferred defects, since in real-life operations the scheduling of requirements is less critical, in the sense that an opportunity can always be made available for their execution. The remainder of the weights, which are compound weights, can now be defined. Calling d_s the duration of slot s, $W_{S_{sa}}$ can be defined as:

$$W_{S_{sa}} = \begin{cases} C_{MH} + C_{fix_s}, & \text{if } s \text{ is a MH slot} \\ C_{S_s} + C_{sd} \cdot d_s + C_{fix_s}, & \text{if } s \text{ is a flex slot} \end{cases}$$

The cost of scheduling a task t in maintenance slot s depends on the number of anticipation days with which the task would be executed. Calling this anticipation a_{ts}, the parameter can be determined as:

$$W_{T_{ts}} = C_{ant_t} \cdot a_{ts}$$

4.1.2 TAS

The recovery window of the TAS goes from three days to two weeks after the call of the TAS. Its parameters are determined using the same hierarchical logic used for the MSS. The hierarchy is defined as follows: (1) assignment of segments (2) assignment of rotations to the preferred aircraft type. Subtype preference groups are introduced to determine which aircraft subtype should be used in the case of the unavailability of the originally assigned subtype. The following groups are defined:

- Group 1: B787-9, B777-200
- Group 2: B787-10, B777-300

Due to run time reasons, rotations assignment is not allowed outside of the preference group, meaning that for example a rotation originally scheduled for a B787-9 cannot be assigned to a B777-300. This is done by reducing the model subsets.

The hierarchy is translated into the following weights:

W_{U_r}	10^5	Unassigning segment r, if r is a reserve slot
	10^4	Unassigning segment r, if r is a rotation
$W_{R_{ra}}$	10^2	Assigning rotation r to aircraft a, if r is a rotation and a is not of its originally assigned subtype. 0 if r s a reserve slot or a is of the preferred subtype.

The fact that the cost of unassigning a reserve slot is higher than that of unassigning a rotation is justified by the policies of our partner airline, which require to have reserves available at the beginning of each operational day, even at the cost of cancelling rotations during the planning phases.

4.1.3 Operations Manager

The operations manager is implemented using SimPy (Team SimPy, 2020), which is an open-source, process-oriented library for DES written and accessible in Python language. Being Python-based, it uses object-oriented logic, and it is platform-independent. Furthermore, being process-oriented, it allows for a simple definition of aircraft processes as a sequence of activities, without having to manage complex events queues. The package has been used in the literature in works such as Iwata and Mavris (2013).

All processes included in the Operations Manager, i.e. the aircraft processes, the AOG processes and the hub disruption state process are implemented in the form of a DES, so for each of the activities included in them a duration or a characterizing stochastic distribution must be defined. All the parameters used are derived from the analysis of historical data.

The Aircraft Process

Primary delays are assumed to only be positive, meaning that a flight cannot depart before its scheduled departure time. Being non-negative, they can be described by a simple probability of experiencing a delay, and by an analytical distribution limited at values greater than zero that describes the duration of the delay. Primary delays are assumed to be independent of the fleet type, but dependent on the station at which they occur. The same simple probability and analytical distribution are used for all outstations due to the limited availability of data for some outstations. Four other sets of parameters are used for the hub, one for each of the considered hub disruption states.

All the simple probabilities and the analytical distributions are obtained from historical data. In particular, analytical distributions are fitted on historical data by minimizing the Residual Sum of Squares (RSS). Python package Distfit (Erdogan Taskesen, 2020) is used for this purpose. The procedure to categorize historical data in disruption states at the hub is discussed later, with the hub disruption state process.

In this simulation, the **block time** corresponds to the flight time. In reality, block time is a random variable depending on many factors, including weather, ATM, and the airlines themselves, since increasing the flight speed is a used method to reduce flight delays, which is especially effective for intercontinental flights (Marla et al., 2017). Due to the many factors that influence block time duration, some of which make block time duration directly dependent on the delay currently being experienced by a flight, this model assumes flight time to be fixed for each route.

The duration of **turnaround** activities depends on many factors, such as the airport where the turnaround happens, the congestion level at the time of arrival of a flight, direct decisions of the carrier, and the available time before the following flight. This makes it a challenge to isolate the duration of turnaround activities, as opposed to the total time the aircraft is on the ground in between flights, in historical data. In order to make a distinction, only flights that departed with a delay due to IATA delay code 93, i.e. propagated delay, are considered in the analysis. For these flights, it is assumed that the time needed for turnaround operations corresponds to the ground time between arrival and departure of the limiting flights.

Due to the limited availability of data caused by the filtering of flights with delay code 93, turnaround activities are assumed to be independent of the aircraft type. Three empirical distributions are obtained: one for turnaround time at the hub, and two for turnaround time at outstations. Two separate turnaround time distributions are used for outstations because at certain airports only shorter technical turnaround activities are executed, as suggested by the bimodal empirical probability density function (PDF) of the full dataset. Outstations are categorized into short or regular turnaround stations based on the mean registered turnaround time. Once again, analytical distributions are fitted on historical data by minimizing the RSS.

The **towing time** is only considered when an aircraft needs to undergo maintenance in the hangar. The towing time to and from the hangar is assumed to be fixed to one hour.

The **duration of a maintenance slot** depends on its work package, and on the non-routine labour originating from findings happening during the slot. Since tasks that can be executed on the platform generally do not lead to any findings, it is assumed that NRs can only happen within slots executed in the hangar. The probability of NRs coming up in a hangar work package is determined from historical data. The total NR labour hours executed within the historical maintenance slots are also computed and used to fit an analytical curve for each aircraft subtype. At the beginning of each hangar slot, the historical probability of experiencing non-routines is used to determine whether there will be some findings in the work package or not. If there should be some findings, the total required NR labour is sampled from the reference distribution and added to the work package.

Note that this model of non-routines assumes that the findings are independent of the number of tasks scheduled within the slot's work package, the specific tasks, the slot's duration, and the frequency of maintenance. Although it has been observed that there is indeed a dependency between slot scheduled duration and non-routine labour executed within it, this dependency is complex, and it is deemed out of the scope of this work.

Once the NR labour hours are sampled, The duration of the maintenance slot can be computed as the maximum of the following two values:

- The maximum duration among the tasks included in the slot's work package
- The sum of the labour hours associated with the tasks included in the work package (including non-routine labour), divided by the available workforce in the maintenance slot.

This way of measuring the duration of a maintenance slot is a simplification since it implies that all tasks can be executed in parallel, possibly by more than one person at a time. However, this way of measuring slots duration was validated in previous research work done by our partner airline.

The AOG Process

The AOG inter-arrival time and duration are defined by stochastic distributions fitted on historical data by minimization of the RSS. Different distributions are obtained for each aircraft type.

The Hub Disruption State Process

The hub disruption process is characterized by four disruption states, and time is discretized in 20-minutes brackets. The process is initialized every day at 6:00 UTC. Both the exponential distribution describing the sojourn time in a state and the transition probability matrix are derived from historical data. To determine the duration of the disruption state, historical flights are grouped in twenty-minute brackets based on their actual departure time. Each bracket is then characterized by a disruption state based on the mean departure delay observed within that bracket. To do so, arbitrarily defined minimum and maximum mean delays are defined for each state. Adjacent brackets characterized by the same state are then counted, and this measure of sojourn time is used to determine the best-fitting exponential distribution by RSS minimization. The transition probability matrix can easily be determined by computing the empirical probability of transitioning from one state to another.

Historical data on primary delays is classified in the considered disruption states based on the state of the bracket in which it falls. The categorized data is then used to determine one stochastic distribution of primary delay for each disruption state, as previously described.

4.1.4 Recovery Controller

The recovery controller requires a recovery action when it computes that the next duty of an aircraft will experience an increase in expected delay of at least ten minutes.

4.1.5 Recovery Planner

The Recovery Planner works on a recovery window that covers three and a half days. At the end of the recovery window operations must be resumed as originally scheduled, and for this reason, duties are included in the recovery space if their *arrival* time falls within the recovery window. In order to better resemble our partner airline's operations, the possibility of postponing maintenance to free maintenance arcs is excluded from the solution of the Recovery Planner. Also, with the objective of reducing passenger disruption on the day of operation, it is imposed that the designated reserve aircraft should always be available at the start of each day. As a consequence, the Recovery Planner cannot assign a rotation to an aircraft if this overlaps with a reserve slot assigned to the aircraft over the coming days. Finally, rotations are allowed to be reassigned to aircraft of any subtype, and the preference group logic described for the TAS applies.

The parameters of the Recovery Planner are once again defined based on a hierarchy, this time formulated in terms of avoidance preference, from the recovery action that should be avoided the most, to the recovery action that is considered most acceptable. The following order is defined: (1) cancelling maintenance slots, (2) cancelling rotations (3) changing the aircraft subtype assignment of a rotation (4) swapping maintenance slots, (5) changing the aircraft assignment of a rotation, (6) delaying the start time of a maintenance slot, and (7) delaying the start time of a rotation. The following costs are derived from the hierarchy:

W_{CS_s}	3×10^6	Cancelling slot s, if s if an MH slot
	10^6	Cancelling slot s, if s if a flex slot
W_{CR_r}	10^5	Cancelling a rotation
$\mathrm{C}_{\mathrm{type}_{ra}}$	2×10^4	Assigning rotation r to aircraft a, if aircraft a's subtype is not included in r's preference group
	10^4	Assigning rotation r to aircraft a, if aircraft a's subtype is within to rotation r's preference group. 0 if a is of the originally assigned subtype.
$\mathrm{W}_{SW_{st}}$	10^3	Swapping two maintenance slots
W_{Γ_a}	2×10^2	Involving aircraft a in the recovery solution
$\mathrm{C}_{\Delta_{ra}}$	10^2	Assigning rotation r to aircraft a, if the assignment is different from what previously planned
C_{DS}	2.5×10^1	Delaying a slot by one minute
C_{DR}	2×10^1	Delaying a rotation by one minute

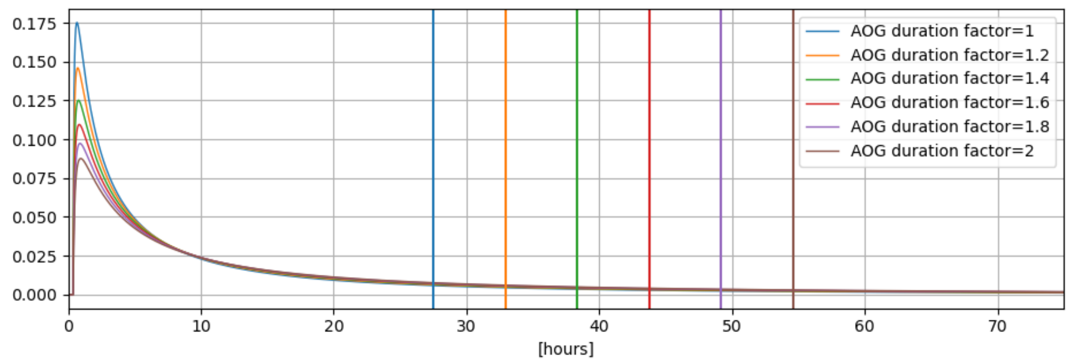


Figure 5: PDF of the AOG duration for different values of the AOG duration factor. The vertical lines represent the expected AOG duration in the corresponding disruption scenario.

The defined costs are used once again to define the compound weights. The cost of assigning a rotation to an aircraft can be defined as follows:

$$W_{R_{ra}} = C_{\text{type}_{ra}} + C_{\Delta ra}$$

For what concerns delaying duties, copies of the original rotation arcs are generated with a delay of 5, 10, 20, 40, 60, 120, 180, and 240 minutes. For maintenance slots, copies are created with a delay of 5, 10, 20, 40, 60, 120, and 180 minutes. The chosen values are denser for shorter delays because these values are more commonly observed, and because they allow the avoidance of more drastic recovery interventions that are often required for significant delays that are in the range of the hours. Also, the maximum delay allowed for a maintenance slot is shorter than that allowed for a rotation given the lower resource flexibility associated with a maintenance slot in terms of, for instance, manpower and hangar space. For the same reason, delaying a maintenance slot is more expensive than delaying a rotation. Calling d_d the delay imposed on a slot or rotation, the weights of assigning delayed duty arcs are defined as:

$$W_{DS_{sd}} = C_{DS} \cdot d_d$$

$$W_{DR_{rda}} = W_{R_{ra}} + C_{DR} \cdot d_d$$

The costs of using a maintenance slot as planned (W_{S_s}) and of using a ground arc ($W_{G_{ga}}$) are set to zero.

4.2 Case study: Additional Reserve Aircraft

In the operations of our partner airline, one widebody reserve aircraft is kept available every day to solve disruptions regarding the intercontinental fleet. In this case study, the value of adding an additional reserve aircraft for intercontinental operations is investigated. Furthermore, given the post-COVID shortage of parts and available technical labour, it is investigated how an increased average duration of AOGs would affect the general performance of operations and the effects of an additional reserve aircraft in these situations. All scenarios consider a fleet of fifty aircraft (or 51 when a reserve is added) including the already cited four aircraft subtypes. The simulated schedule is a weekly schedule that includes 263 rotations, and the simulated maintenance slots are the slots that were historically available during that week. Each scenario is simulated a hundred times over 180 days, in containers with 16 allocated cores and 60GB of available memory. The ILP models are solved with Gurobi 10.0.1. The required run time is 15 hours for each simulated scenario.

4.2.1 Scenarios Generation

Two variables model this case study: the number of reserve aircraft available, and the distribution describing the duration of AOGs. In order to generate different scenarios to evaluate, different distributions describing AOG duration must be chosen. In the baseline scenario, AOGs for all aircraft types are modelled as having a duration described by a lognormal distribution. Using the generally accepted parametrization of the lognormal distribution in μ and σ, the expected value can be expressed as:

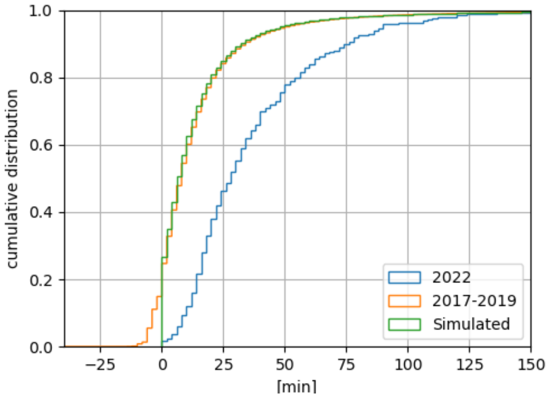
$$E[X] = e^{\mu + \frac{1}{2}\sigma^2} \tag{4}$$

Given this property, new scenarios are generated by multiplying the scale parameter e^{μ} by a constant, so that the new distributions are characterized by an expected value that is equal to the original expected value multiplied by the same constant. The multiplicative constant, which will be called *AOG duration factor*, is set to 1, 1.2, 1.4, 1.6, 1.8 and 2. The originated distributions and their respective expected values for an example aircraft type are shown in Figure 5.

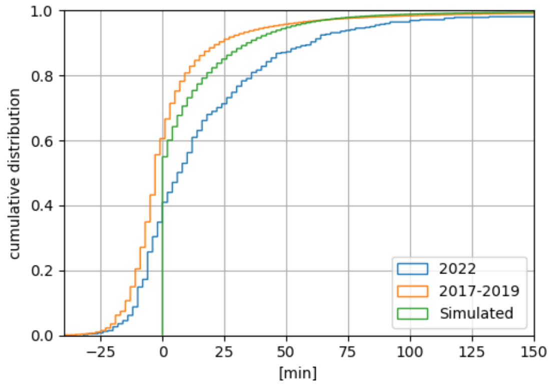
4.3 Model Validation

The model is validated by comparing the results obtained for the baseline case of the case study (AOG duration factor of 1, 1 reserve aircraft), with historically observed operational performance. For what concerns network operations, departure delays and flight cancellations are used as comparative performance indicators.

Due to the extensive dataset needed to implement the model of delays dependent on the hub disruption state, the historical dataset used to implement the de-



(a)



(b)

Figure 6: Empirical CDFs of the departure delays of simulated and historical flights departing from the hub (6a) and from outstations (6b). Two historical curves are shown. The 2017-2019 curve corresponds to data used for building the delay model, while the 2022 curve refers to flights included in the simulated schedule, corresponding to a specific week of operations.

lay model comprises data on flights flown between 2017 and 2019. However, the proposed case study refers to a schedule and maintenance plan implemented in 2022. For this reason, the simulated delays are compared to historical data of two sets of flights: the flights flown between 2017 and 2019, and the flights flown during the week of the simulated schedule in 2022.

Figure 6a and Figure 6b show the empirical cumulative distribution functions (CDF) of the departure delay of flights departing from the hub and from outstations, respectively. The curve of the simulated data closely follows the 2017-2019 curve for departure delays from the hub, with the sole difference that there are no departures before the scheduled departure time since they are not allowed by the model. When considering departure delays from outstation, a similar trend is observed between simulated and 2017-2019 flights, although simulated departure delays tend to be higher than 2017-2019 ones. This can be explained by the assumption of fixed flight leg duration, which in some routes leads to a systematic accumulation of delay that propagates to the following flight legs.

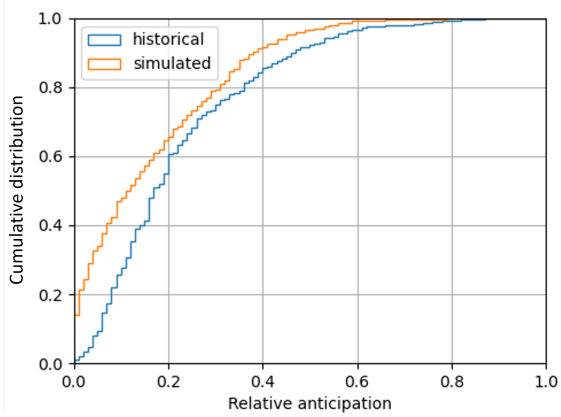
A comparison of the simulated delays and delays from 2022 shows a significant underestimation of delays in the simulated results. This can be explained by the historical period from which 2022 data is retrieved, characterized by widespread ground personnel shortages leading to strong disruptions in numerous airports worldwide. This result, however, does not invalidate the proposed case study, since the case study focuses more on flight cancellations rather than delays, and historical data of our partner airline show that no cancellation was caused by delay propagation in 2022.

The second validated performance measure regards cancellations, for which the historical value to be used as a validating comparison should be discussed. Many reasons can lead to the cancellation of a flight, of which the most common are bad weather, crew disruptions, commercial decisions, technical problems, and delay propagation. Since the model disregards crew, and it models weather problems only in the form of primary delay, the only reasons that can lead to a cancellation of a flight in the simulation are technical problems and delay propagation. However, as already mentioned, delay propagation does not cause any cancellations thanks to the buffer time scheduled at the hub for intercontinental operations. Therefore, technical cancellations are the only ones that are considered. Furthermore, since maintenance and AOGs only happen at the hub within the simulation, and since cancellations can only be done at the rotation level rather than the flight level, only historical cancellations of full rotations are considered.

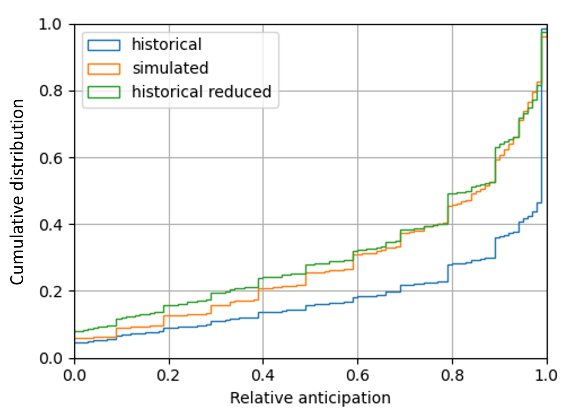
Given these premises, the values of the simulated cancellations, 0.11% can be compared to the historical value found for 2022, 0.19%. The simulated value is 42% lower than the historically experienced, but it must be observed that cancellations of intercontinental flights are a rare event, which in reality emerges from complex interactions between departments, and is therefore intrinsically hard to simulate.

Furthermore, several reasons for this underestimation of cancellations can be given. First, the model generally has more flexibility in recovery than it is available in reality. In real-life operations, some aircraft cannot fly specific routes due to regulations, and the opening of some MEL items on a specific aircraft can also limit the routes the aircraft can fly. Second, A-checks are simulated in a simplified manner, so that they have more operational flexibility than in reality. Third, the scope of maintenance limited to A-checks excludes heavy maintenance which is more likely to exceed the scheduled time, causing disruptions.

When focusing on maintenance validation, it is hard to do a direct comparison of ground time or total workload because of the limited scope of maintenance tasks and slots considered. For this reason, a focus on task execution is used. The model is capable of executing between 99.3% and 99.5% of tasks for all aircraft subtypes apart for the B777-200, for which an execution rate of 96.1% is simulated. The tasks that the model is incapable of scheduling are generally tasks that are longer than the available maintenance slots, or tasks for which a maintenance opportunity cannot be made available due to the short time between the task's ready date and due date. In the case of the B777-200, the re-



(a)



(b)

Figure 7: Empirical CDFs of the relative task anticipation of simulated and historical requirements (7a) and deferred defects (7b). For DDs, a third curve showing the relative anticipation of historical tasks not found and executed on the same day is shown.

duced execution rate is due to the inclusion of recurring tasks that in real-life operations are executed in specifically designated maintenance slots which are longer than the ones included in the scope of the simulation.

The scheduling logic used by the simulator is validated by considering the *tasks' relative anticipation*, defined as the ratio between the number of days intercurring between a task execution date and due date and the number of days intercurring between the task arrival date and due date. This concept can be seen as a generalization of the *lost interval* as it was defined for requirements. By validating the lost interval, especially for what concerns requirements, the simulated workload can be indirectly validated since executing tasks at the same interval leads to the same number of needed repetitions of the task over a period of time.

Figure 7 shows the empirical CDF of the relative task anticipation of the simulated and historical tasks. Simulated requirements (Figure 7a) follow the trend of historical requirements, as they tend to be executed close to their due date, to minimize the requirements' lost interval. However, they are generally executed closer to their due date than in reality, which can be explained by the setting of the preferred anticipation for requirements execution of the MSS to zero.

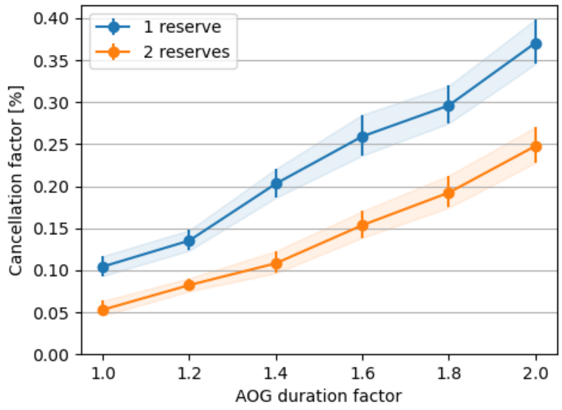


Figure 8: Cancellation factor for different AOG duration factor and reserve aircraft scenarios

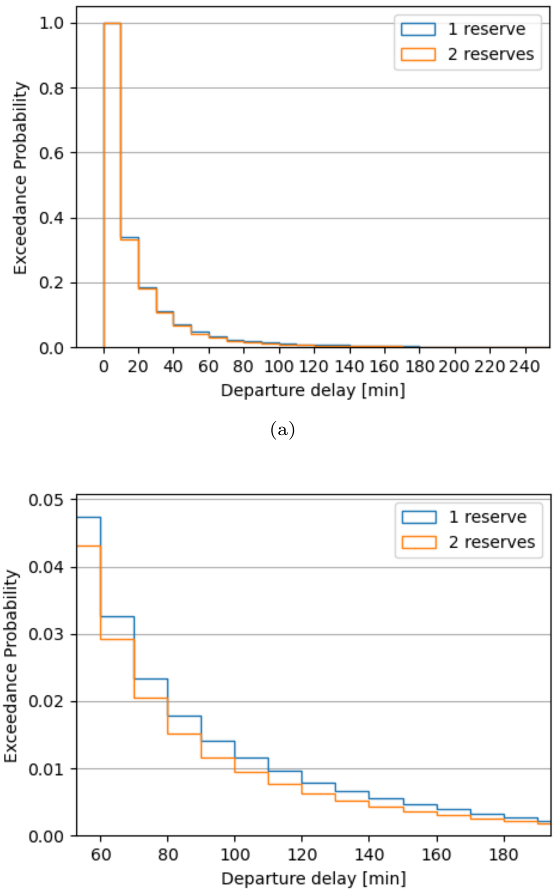
For what concerns DDs, Figure 7b shows that they follow the correct trend of being executed early after their finding. In this case, the simulated results are compared with two historical curves. When compared to the full dataset of historical DDs, a general postponement of simulated task execution is observed. This is because historically, many DDs are generated from crew complaints registered within the Aircraft Maintenance Log (AML), i.e. a book located on board of each aircraft that can be used to report any MEL problem detected on the aircraft during operations. When an aircraft undergoes maintenance, the AML is checked and the included tasks are often executed on the same day on which they are found, without needing to be scheduled. If these tasks are excluded from the historical dataset, the second historical curve shown in Figure 7b is obtained (historical reduced), which closely resembles the curve of simulated tasks.

5 Results

In this section, the results of the case study on the reserve aircraft are discussed. Four main indicators are used for the evaluation of the results of this case: (1) the cancellation factor (CF), i.e. the percentage of rotations that are cancelled, (2) the arrival delays, (3) the costs of disruptions, and (4) the avoided disruption costs when a second reserve is used.

5.1 Operational Performance

The results for the **cancellation factor** for all scenarios are shown in Figure 8 along with the 95% confidence interval computed using the bootstrap technique. The increase in AOG average duration causes a significant increase in the CF, with the number of cancellations almost doubling for an AOG duration factor of 1.4, and becoming more than triple for an AOG duration factor of 2. Adding a second reserve aircraft, on the other hand, allows for a reduction of the expected cancellations. At the baseline, the second reserve halves the number of cancellations, bringing the CF from 0.11% to 0.05%. As the value of the AOG duration factor increases, the impact in terms of the number of cancellations that the reserve can avoid in-



(a)

(b)

Figure 9: Exceedance probability curve of departure delays, for an AOG duration factor of 1: full plot (9a) and detail (9b)

creases, and then it stabilizes at around 0.10%.

When considering delays, Figure 9a shows the empirical exceedance probability curve of **departure delays**, which described the probability of observing a delay greater than a specified value. The results shown are those obtained with an AOG duration factor of 1, but other scenarios show a similar impact of the second reserve aircraft. A detail of delays between one and four hours is shown in Figure 9b. The use of a second reserve aircraft reduces the probability of observing a delay longer than one hour by 0.4% (from 3.3% to 2.9%) and the probability of observing a delay longer than two hours by 0.1% (from 0.8% to 0.7%). Further research is needed to fully understand the impact of the second reserve in this case. In fact, Although this improvement in observed delay might look negligible, it is of comparable order of magnitude with the reduction in CF, and it might have a significant impact on costs derived from passengers' misconnections.

5.2 Economic Impact

The economic impact of the simulated scenarios is computed considering the costs of cancellations and delays, including costs associated with European regulations on passenger compensation and soft costs related to passenger satisfaction. In the model used, the computation of delay costs, unlike cancellations, disregards the cost of passengers' lost connections. This means that if a flight experiences a one-hour delay, then the passengers are assumed to arrive at their final destination with a one-hour delay, disregarding any possible misconnections. This leads, in general, to an underestimation of delay costs. The resulting costs are given per unit of time, in the form of monetary units.

The obtained results on **disruption costs** are shown in Figure 10, where the two columns on the right show the cost components and the column on the left show the total costs, obtained from their summation. The effects of an increased AOG duration factor are significant, with a cost increase of +30% for an AOG duration factor of 2. The primary contributors to this cost increase are cancellation costs. This can be explained by the high costs associated with the cancellation of a flight, rather than with its delay, especially considering that cancellation costs, differently from delay costs, account for passenger misconnections.

In order to evaluate the impact of adding a second reserve aircraft on operational costs, the **avoided disruption costs**, i.e. the difference between the costs incurred with one and two reserves, are computed. Results are presented in Figure 11. For low values of AOG duration factors, the impact of the second reserve aircraft is comparable for delay and cancellation costs. For higher values of AOG duration factor, the avoided costs of delay remain stable, while the avoided cancellation costs increase significantly. This can be explained by the fact that as AOGs become longer, the probability of experiencing disruptions that could lead to both delays and cancellations increases, but it is more cost-efficient to use the reserve aircraft to avoid cancellations, rather than delays. Furthermore, the much higher costs associated with cancellations with respect to delays, lead to a higher impact on cancellation costs, rather than delay costs, when a comparable number of disruptions of the two types are avoided.

These results can be used to determine whether the use of a second reserve aircraft would be economically convenient. Per time unit, the profit generated by an operating aircraft is estimated to be 1,000 MU, while the use of a second reserve leads to savings between 62 MU and 102 MU for different AOG disruption scenarios. This result shows how using a reserve aircraft is an expensive measure, which turns out to be not economically advantageous, despite the benefits obtained in terms of operational performance.

6 Conclusions

This paper presented a stochastic discrete event simulation model of airline operations named ANEMOS (Airline Network and Maintenance Operations Simulation). The main purpose of the model is to provide a framework for testing how plans and policies made in the network and maintenance domain of an airline interact and perform from an integrated perspective. This work is novel in its approach since both in the literature and in airlines practices network and maintenance operations are optimized and

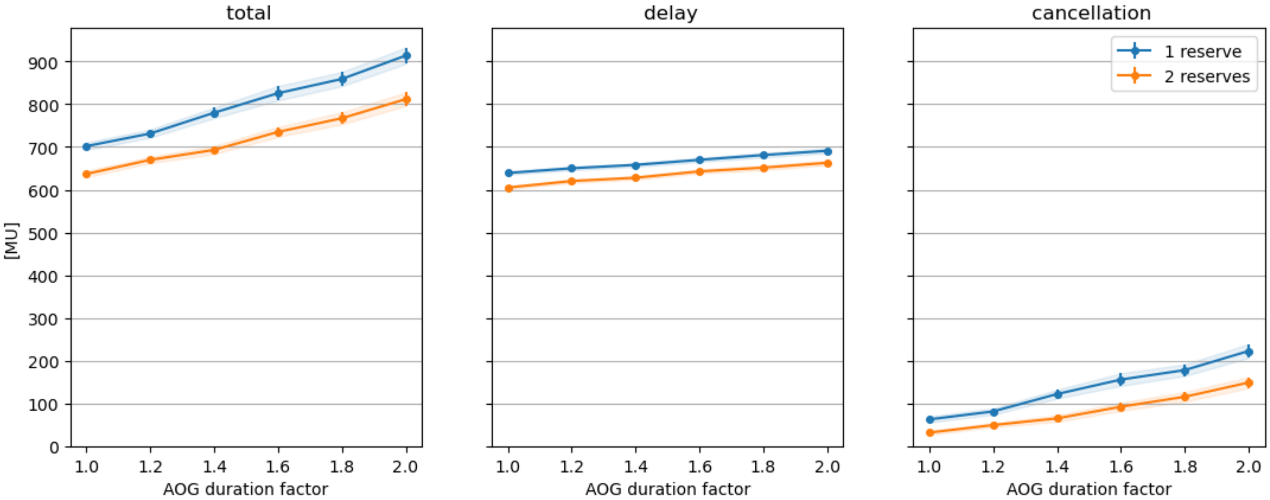


Figure 10: Delay, cancellation, and total costs of disruptions for different AOG duration factors and reserve aircraft scenarios.

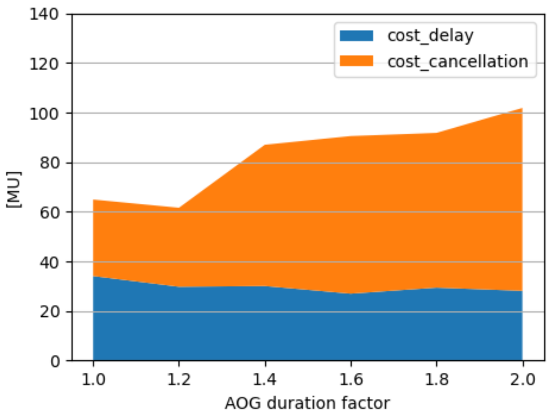


Figure 11: Disruption costs avoided through the use of a second reserve aircraft for different AOG duration factors.

evaluated separately, and their interaction is generally disregarded.

ANEMOS is modular, dynamic, and stochastic. The modular structure allows for an easy adaptation to a simulation's needs, while its dynamicity allows the evaluation of plans and policies made with both a strategic-tactical, and operational perspective. This includes, for example, the evaluation of flight schedules and maintenance scheduling policies, as well as recovery strategies. Finally, the use of a stochastic DES allows for the evaluation of the robustness of plans and policies in a stochastic environment which well represents the uncertainty that airlines face in their operations.

A case study developed in collaboration with a major European carrier is presented to validate the model and show its capabilities. The case study considers a fleet of fifty aircraft of four different subtypes, and it investigates the effects of the use of a second reserve aircraft for the simulated fleet. The impact of the second reserve is quantified in terms of cancellation factor, delay reduction, and economic value. Simulated results of the baseline scenario are compared with historical data, and it is shown that the model closely resembles historically observed operational performance.

Despite ANEMOS is capable of evaluating several scenarios and plans in its current form, a number of directions for future improvements can be named. First, the proposed model simulates the operations of a hub-and-spoke airline with one hub. The model could be modified in the future to allow for the simulation of multiple hubs and point-to-point carriers, which would mean transforming all the modules into being flight-based rather than rotation-based. Secondly, the expansion of the model so that passenger and crew flows are considered would allow for testing a wider range of plans such as crew rosters, and it would lead to a better quantification of airline performance in terms of passenger misconnections. Third, better dependencies between simulated events could be achieved. This includes, for example, coupling the event of a task going due with the arrival of an AOG, or introducing a dependency between the quantity of work that can be executed in line maintenance and the scheduled ground time. Finally, widening the scope of the simulated maintenance slots to include maintenance heavier than A-checks would allow a better evaluation of maintenance-flights operations interaction since heavy maintenance is generally more likely to exceed the scheduled duration.

References

A. Abdelghany and K. Abdelghany. *Airline network planning and scheduling*. John Wiley & Sons, 2018.

S. Ackert. BasicsofAircraftMaintenance Programsfor-Financiers Evaluation & Insights of Commercial Aircraft Maintenance Programs. Technical report, 2010.

Y. Ageeva. *Approaches to Incorporating Robustness into Airline Scheduling L 1 1 200*. PhD thesis, 2000.

M. A. Aloulou, M. Haouari, and F. Zeghal Mansour. Robust aircraft routing and flight retiming. *Electronic Notes in Discrete Mathematics*, 36(C):367–

374, 2010. ISSN 15710653. doi: 10.1016/j.endm.2010.05.047.

C. Barnhart and A. Cohn. Airline schedule planning: Accomplishments and opportunities. *Manufacturing and Service Operations Management*, 6(1):3–22, 2004. ISSN 15234614. doi: 10.1287/msom.1030.0018.

C. Barnhart, T. S. Kniker, and M. Lohatepanont. Itinerary-based airline fleet assignment. *Transportation Science*, 36(2):199–217, 2002. ISSN 00411655. doi: 10.1287/trsc.36.2.199.566.

P. Belobaba, A. Odoni, and C. Banhart. The global airline industry. *Climatic Change*, 44(1/2):27–58, 2000. doi: 10.1023/A:1005579306109.

M. Ben Ahmed, W. Ghroubi, M. Haouari, and H. D. Sherali. A hybrid optimization-simulation approach for robust weekly aircraft routing and retiming. *Transportation Research Part C: Emerging Technologies*, 84:1–20, 2017. ISSN 0968090X. doi: 10.1016/j.trc.2017.07.010.

Boeing. Boeing Rapid Response Center Proves Its Worth In First Year, 2000.

A. Cook and G. Tanner. The cost of passenger delay to airlines in Europe. Technical report, 2015.

Q. Deng, B. F. Santos, and R. Curran. A practical dynamic programming based methodology for aircraft maintenance check scheduling optimization. *European Journal of Operational Research*, 281(2):256–273, 2020. ISSN 03772217. doi: 10.1016/j.ejor.2019.08.025.

S. O. Duffuaa and A. A. Andijani. An integrated simulation model for effective planning of maintenance operations for Saudi Arabian Airlines (SAUDIA). *Production Planning & Control*, 10(6):579–584, 1 1999. ISSN 0953-7287. doi: 10.1080/095372899232876.

Erdogan Taskesen. Distfit: a Python Library for Probability Density Fitting, 2020.

EUROCONTROL. CODA Digest All-causes delay and cancellations to air transport in Europe. Technical report, 2019.

J. Evler, M. Lindner, H. Fricke, and M. Schultz. Integration of turnaround and aircraft recovery to mitigate delay propagation in airline networks. *Computers and Operations Research*, 138, 2022. ISSN 03050548. doi: 10.1016/j.cor.2021.105602.

L. Hassan, B. Santos, and J. Vink. Airline disruption management: A literature review and practical challenges. *Computers & Operations Research*, 127:105137, 2021. doi: 10.1016/j.cor.2020.105137.

IATA. Industry Statistics Fact Sheet. Technical report, 2019.

C. Iwata and D. Mavris. Object-oriented discrete event simulation modeling environment for aerospace vehicle maintenance and logistics process. In *Procedia Computer Science*, volume 16, pages 187–196. Elsevier B.V., 2013. doi: 10.1016/j.procs.2013.01.020.

P. H. Jacobs, A. Verbraeck, and J. B. Mulder. Flight scheduling at KLM. In *Proceedings - Winter Simulation Conference*, volume 2005, pages 299–306, 2005. doi: 10.1109/WSC.2005.1574264.

H. A. Kinnison and T. Siddiqui. *Aviation maintenance management*, volume McGraw Hill. 2013.

C. Lagos, F. Delgado, and M. A. Klapp. Dynamic optimization for airline maintenance operations. *Transportation Science*, 54(4):998–1015, 2020. ISSN 15265447. doi: 10.1287/TRSC.2020.0984.

S. Lan, J. P. Clarke, and C. Barnhart. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science*, 40(1):15–28, 2006. ISSN 15265447. doi: 10.1287/trsc.1050.0134.

L. H. Lee, Huei Chuen Huang, Chulung Lee, Ek Peng Chew, W. Jaruphongsa, Yean Yik Yong, Zhe Liang, Chun How Leong, Yen Ping Tan, K. Namburi, E. Johnson, and J. Banks. Discrete event simulation model for airline operations: SIMAIR. In *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693)*, pages 1656–1662. IEEE, 2003. doi: 10.1109/WSC.2003.1261616.

A. Levin. Scheduling and Fleet Routing Models for Transportation Systems. *Transportation Science*, 5 (3):232–255, 1971. doi: 10.1287/trsc.5.3.232.

M. Lohatepanont and C. Barnhart. Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment. *Transportation Science*, 38(1):19–32, 2004. ISSN 00411655. doi: 10.1287/trsc.1030.0026.

L. Marla, B. Vaaben, and C. Barnhart. Integrated Disruption Management and Flight Planning to Trade Off Delays and Fuel Burn. *Transportation Science*, 51(1):88–111, 2017. doi: 10.1287/trsc.2015.0609.

M. M. Mota, G. Boosten, N. De Bock, E. Jimenez, and J. P. de Sousa. Simulation-based turnaround evaluation for Lelystad Airport. *Journal of Air Transport Management*, 64:21–32, 2017. ISSN 09696997. doi: 10.1016/j.jairtraman.2017.06.021.

S. O. Obadimu, N. Karanikas, and K. I. Kourousis. Development of the minimum equipment list: Current practice and the need for standardisation, 1 2020. ISSN 22264310.

M. Öhman, M. Hiltunen, . K. Virtanen, and J. Holmström. Frontlog scheduling in aircraft line maintenance: From explorative solution design to theoretical insight into buffer management. 2020. doi: 10.1002/joom.1108.

A. A. Pohya, J. Wehrspohn, R. Meissner, and K. Wicke. A modular framework for the life cycle based evaluation of aircraft technologies, maintenance strategies, and operational decision making using discrete event simulation. *Aerospace*, 8(7), 2021. ISSN 22264310. doi: 10.3390/AEROSPACE8070187.

A. Regattieri, A. Giazzi, M. Gamberi, and R. Gamberini. An innovative method to optimize the maintenance policies in an aircraft: General framework and case study. *Journal of Air Transport Management*, 44-45:8–20, 2015. ISSN 09696997. doi: 10.1016/j.jairtraman.2015.02.001.

J. Rosenberger, A. Schaefer, D. Goldsman, E. Johnson, A. Kleywegt, and G. Nemhauser. SimAir: a stochastic model of airline operations. In *2000 Winter Simulation Conference Proceedings (Cat. No.00CH37165)*, volume 2, pages 1118–1122. IEEE, 2000. doi: 10.1109/WSC.2000.899074.

J. M. Rosenberger, A. J. Schaefer, D. Goldsman, E. L. Johnson, A. J. Kleywegt, and G. L. Nemhauser. A Stochastic Model of Airline Operations. Technical report, Georgia Institute of Technology, Atlanta, 10 2001.

J. M. Rosenberger, A. J. Schaefer, D. Goldsman, E. L. Johnson, A. J. Kleywegt, and G. L. Nemhauser. A stochastic model of airline operations. *Transportation Science*, 36(4):357–377, 2002. ISSN 00411655. doi: 10.1287/trsc.36.4.357.551.

J. M. Rosenberger, E. L. Johnson, and G. L. Nemhauser. Rerouting aircraft for airline recovery. *Transportation Science*, 37(4):408–421, 2003. ISSN 00411655. doi: 10.1287/trsc.37.4.408.23271.

J. M. Rosenberger, E. L. Johnson, and G. L. Nemhauser. A robust fleet-assignment model with hub isolation and short cycles. *Transportation Science*, 38(3):357–368, 2004. ISSN 00411655. doi: 10.1287/trsc.1030.0038.

S. Shaukat, M. Katscher, C. L. Wu, F. Delgado, and H. Larrain. Aircraft line maintenance scheduling and optimisation. *Journal of Air Transport Management*, 89, 2020. ISSN 09696997. doi: 10.1016/j.jairtraman.2020.101914.

H. D. Sherali, E. K. Bish, and X. Zhu. Airline fleet assignment concepts, models, and algorithms. *European Journal of Operational Research*, 172(1):1–30, 2006. ISSN 03772217. doi: 10.1016/j.ejor.2005.01.056.

Team SimPy. SimPy: Discrete event simulation for Python, 2020. URL `https://simpy.readthedocs.io/en/latest/index.html`.

P. J. van Kessel, F. C. Freeman, and B. F. Santos. Airline maintenance task rescheduling in a disruptive environment. *European Journal of Operational Research*, 2022. ISSN 03772217. doi: 10.1016/j.ejor.2022.11.017.

J. Vink, B. F. Santos, W. J. Verhagen, I. Medeiros, and R. Filho. Dynamic aircraft recovery problem - An operational decision support framework. *Computers and Operations Research*, 117, 2020. ISSN 03050548. doi: 10.1016/j.cor.2020.104892.

H. W. M. Vos, B. F. Santos, and T. Omondi. Aircraft schedule recovery problem - A dynamic modeling framework for daily operations. In *Transportation Research Procedia*, volume 10, pages 931–940. Elsevier, 2015. doi: 10.1016/j.trpro.2015.09.047.

II

Literature Study
previously graded under AE4020

1
Introduction

Until the coronavirus pandemic hit the world in 2020, commercial aviation has been growing continuously, with revenue passenger kilometres (RPKs) increasing on average by 5.5% every year (IATA, 2019a). At the same time, the average airline has a net profit margin of only 3.1% (IATA, 2019b). It is therefore evident that airlines must, from one side, meet market growth with operations that increase in dimension and complexity, and, on the other side, reduce costs to a minimum, to actually generate profit. For meeting both these objectives, it becomes fundamental to plan and manage operations in order to closely match demand and supply while reducing operating and maintenance costs, as well as costs connected to disruptions.

But how can airlines plan and manage their operations optimally? This can be done before the day of operations by developing effective flight schedules and aircraft rotations that are not susceptible to disruptions, and by producing maintenance plans that make aircraft as available as possible for flying. Optimal operations can also be achieved by developing models and policies to be used on the day of operations so that when disruptions occur, they can be recovered at a minimum cost. Giving an overview of the models proposed in the literature on the topics of airline network and maintenance operations planning, scheduling, and disruption management is the first objective of this work.

Furthermore, given the complex dynamics of this field, and the stochasticity that characterizes it, it is clear that any work that addresses operations effectiveness and flexibility has to deal with simulation at a certain level, either as the main instrument for solving the problem at hand or as test framework for a developed model. Therefore, the second objective of this work is to introduce simulation techniques, identify a suitable one to be used in the field of airline operations, and investigate how simulation has been used in the literature in this field.

Given the broad scope used, and in order to meet the imposed time constraints, a decision is made to focus on aircraft-centred operations, and to disregard crew-related planning and operations, although the author recognizes the importance of this field in airline dynamics.

The remainder of this work is organised as follows. chapter 2 gives an overview of airline network and maintenance operations planning. chapter 3 addresses irregular operations: how disruptions originate and spread within the network, how they are solved when they occur, and how they can be prevented and mitigated by including robustness in operations. chapter 4 presents some simulation techniques and explains how simulation has been used in the literature in the context of airline operations. Based on the works presented in the previous chapters, chapter 5 draws the conclusions of this work, by identifying a gap in the literature and introducing a research objective that will be addressed in the thesis project this literature study supports.

2

Operations Planning

During the last decades, up until the coronavirus pandemic hit, the airline industry has been growing continuously, with an expansion in RPK of 4.2% in 2019 IATA (2019a). At the same time, the average airline has a net profit margin of only 3.1% IATA (2019b). For these reasons, it is evident how planning becomes fundamental to meet both market growth opportunities and the necessity of keeping costs to a minimum. In this chapter, an introduction to the airline planning process is given. In particular, section 2.1 deals with network-related planning, section 2.2 focuses on maintenance planning, and section 2.3 explores the interaction between the network and maintenance planning processes.

2.1. Network Planning

The airline planning process comprises (Barnhart and Cohn, 2004) (1) schedule design, which defines a flight schedule, (2) fleet assignment, which decides which aircraft subtype will cover each flight leg, (3) maintenance routing, which assigns a certain sequence of flights to each aircraft, and (4) crew scheduling, which defines rosters and pairings for cabin and cockpit crew. Given the complexity of each of these problems, a sequential approach (Figure 2.1) is usually used in reality, where each step is considered independently from the others (Barnhart and Cohn, 2004).

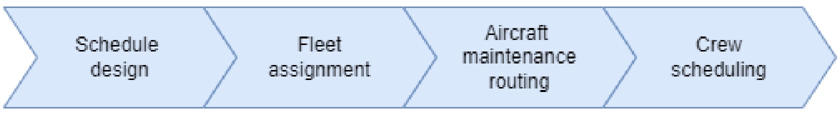


Figure 2.1: Airline planning process (Based of Barnhart and Cohn (2004))

Schedule design is the first step in the airline planning process. It is a strategic problem, as it takes place from one year before operations, but small changes to the schedule can be made up until the day of operations (DOO) (Belobaba et al., 2000). Designing the schedule is a critical task, as it determines the products (the flights) that the airline will sell, and, as a consequence, the market share that the airline will be able to capture (Barnhart and Cohn, 2004).

In general, schedule design can be subdivided into two subproblems, (1) frequency planning, i.e. deciding how many flights should be operated on a route, and (2) timetable development, which refers to deciding at what time the flights should be flown. Both these processes are quite complex (Belobaba et al., 2000), as considerations at the Origin-Destination (OD) market level and network level must be kept into account, in addition to consideration of available fleet and competition. In addition to this, (semi-)automating the process of schedule design through operations research models is even more complex, given that data such as competitors' schedules (Barnhart and Cohn, 2004) and the expected market share (Belobaba et al., 2000) is dependent on the schedule that an airline operates.

Given these limitations, and given passengers' fidelity aspects, schedule planning is often executed by partially modifying the schedule of the previous years, rather than developing a new schedule from scratch

(Barnhart and Cohn, 2004). This can be done by, for instance, retiming flights within a window around the originally scheduled time (Levin, 1971), or by pre-determining flights candidate to be cancelled or added, and choosing a certain set of them (Lohatepanont and Barnhart, 2004).

Fleet assignment is a tactical problem that consists of assigning a certain aircraft subtype to each flight leg in the network (Belobaba et al., 2000). The main objective of this planning step is to match supply to demand by minimizing the number of spilled passengers and spoiled seats (Sherali et al., 2006), which respectively represent the passengers that exceed the provided capacity, and the number of seats that remain unsold. Further complications to this analysis regard the possibility of recapturing some of the spilled capacity through similar flights or itineraries (Sherali et al., 2006). Another objective that is kept in mind during fleet assignment regards the minimization of operating costs (Barnhart and Cohn, 2004) where, for example, efficient aircraft should be assigned to longer routes. Constraints that must be considered in the fleet assignment problem include considerations on fleet-route compatibility and routing feasibility since the aircraft in a fleet are limited and feasible routings must be built for each registration.

The **maintenance routing** problem (Belobaba et al., 2000) is carried out in a tactical-operational context, and it involves the assignment of routings, i.e. sequences of flights, to each aircraft in the fleet. Given the strong interdependence of routings (also called rotations and lines of flight in the literature) and maintenance slot planning, the problems of assigning maintenance slots and flights to aircraft are usually solved together.

The last step in the planning framework is the **crew scheduling problem**. This problem (Belobaba et al., 2000) consists of assigning cabin and cockpit crew to flights while ensuring compliance with complex regulations. Given the complexity of the problem, standard practice is to solve it in two steps. The first step, called the crew pairing problem, consists of generating feasible pairings, i.e. sequences of flights with a duration of one to five days. Then, pairings are put together in longer sequences in the crew rostering problem, that generate rosters for each crew member.

By now, it should be clear that these planning steps are strongly interrelated to each other so that each step must produce a valid input for the following steps. Schedule design, for instance, is limited by the number and types of fleets available for operations. Also, fleet assignment solutions must guarantee the existence of feasible aircraft routings, which is often a challenge, especially for point-to-point carriers (Barnhart and Cohn, 2004). This is why many authors, also from early approaches (Levin, 1971), integrate at least two planning problems in their optimization models. Recent literature (Papadakos, 2009, Shao et al., 2017, Sherali et al., 2013) propose methods for integrating more than two steps of the planning process, guaranteeing higher solution quality.

In this section, an introduction to some flight scheduling and fleet assignment models (subsection 2.1.1), as well as to aircraft maintenance routing models (subsection 2.1.2) is given. Since airlines still use a sequential approach in their planning, and since the purpose of this work is to understand how operations are carried out in real life, as well as to find opportunities for adding flexibility to operations, a focus is put on more traditional approaches, while fully integrated approaches are disregarded. It is worth mentioning that a trend in network planning has regarded the inclusion of robustness in the planning process. This topic is further discussed in section 3.3. Finally, for the reasons introduced in chapter 1, crew planning models are not further considered.

2.1.1. Schedule Design and Fleet Assignment

Schedule design and fleet assignment are strongly interrelated with each other, especially in the fact that schedule design is constrained by the available fleet. This is why these two planning steps are sometimes treated together in literature (Lohatepanont and Barnhart, 2004) and in reality, and are presented together in this section. Both fleet assignment (Abara, 1989) and schedule design (Levin, 1971) have been discussed for many decades, usually adopting a connection network (Abara, 1989) or a time-space network (Clarke et al., 1996, Hane et al., 1995) model representation (Sherali et al., 2006).

Abara (1989) and Hane et al. (1995) are two of the first works that address the fleet assignment problem, both assuming a fixed daily schedule. Abara (1989) use a connection network as a base for their model while maximizing aircraft profit, computed as a function of operating cost and revenue for each flight-fleet assignment. Opting for a different approach, Hane et al. (1995) use a time-space representation for their model, which maximizes profits by also taking into account a pre-determined recapture rate for each assignment. This formulation, in addition to multiple reduction strategies proposed by the authors, allow them to solve large-scale problems including more than 2500 daily flights and 11 fleets. On the other hand, as Hane et al. recognize themselves, their approach is not able to determine aircraft rotations, while other works such as Abara (1989) are.

The models proposed by Abara (1989) and Hane et al. (1995) solve the fleet assignment problem by ensuring the feasibility of the basic leg coverage, flow balance, and fleet count constraints, but they disregard any other relevant limitation. Clarke et al. (1996) improve the work of Hane et al. (1995) by adding maintenance and crew related constraints. For what concerns maintenance, they include maintenance opportunity constraints for overnight maintenance and maximize line maintenance opportunities in between flights. Furthermore, they reduce crew costs by penalizing solutions that leave the crew inoperative for long periods of time.

The approaches presented so far consider static demand for flight legs, disregarding or modelling simplistically network effects such as spilling and recapturing dynamics. A model improved in this sense is presented by Barnhart et al. (2002), who propose a daily itinerary-based fleet assignment model that integrates the passenger mixed flow model with fleet assignment. The model is expressed as a mixed integer linear programming (MILP) model that minimizes fleet assignment costs while considering spilling and recapturing effects. The authors present a heuristic solution that allows solving instances including more than 2000 flights and demonstrate through simulation that their approach performs better than traditional fleet assignment models when the solution is tested in an environment with stochastic demand.

The works introduced so far mainly focus on fleet assignment, in some cases (Abara, 1989) combining it with the aircraft routing problem. Other models, on the other hand, focus on schedule optimization. Early work in this sense is Levin (1971). They propose a model to optimize the schedule by retiming flights within small time windows. In the model, each flight is represented by multiple arcs with different departure times, among which only one is chosen. As objective, the authors minimize the number of aircraft needed to cover the schedule. A solution based on a branch and bound algorithm is presented but not implemented. Although the model is quite simple and does not take into account considerations regarding for instance different fleet types or maintenance, the concept of retiming flights within a certain time window is used by various authors in later works (Aloulou et al., 2010, Burke et al., 2010, Lan et al., 2006). These works are often expressions of a recent trend to investigate how robustness can be embedded in the airline schedule, as it is better explained in section 3.3.

Years later, Lohatepanont and Barnhart (2004) develop a much more complete model that optimizes both the flight schedule and fleet assignment. Given unconstrained demand and a list of mandatory and optional flights as input, the model is capable of choosing which of the optional flights to operate and which fleet to use to operate them. Passenger spilling and recapturing are taken into account by including similar objectives and constraints as in Barnhart et al. (2002), and demand-supply effects are considered by means of *demand correction terms* that estimate the modification of unconstrained demand when possible flight legs are cancelled. The model is expressed as a MILP that minimizes costs including operating costs, spill, recapture, and changes in demand. Since the model is intractable despite the use of a heuristic solution, an approximated version of the problem that substitutes demand correction terms with modified recapture parameters is introduced. The authors show that the modified model is capable of solving complex instances including more than 800 mandatory and optional flights and of delivering significantly improved solutions with respect to those proposed by human planners, in terms of revenue and number of flights included.

2.1.2. Aircraft Maintenance Routing

The aircraft maintenance routing problem has received great attention in the literature. Despite this problem is usually solved with a focus on network operations rather than maintenance, maintenance has progressively received more attention, and, while early approaches considered generic maintenance constraints such as imposing a maintenance opportunity at regular intervals (Abara, 1989, Barnhart et al., 1998, Sriram and Haghani, 2003), later works include aircraft-specific considerations on remaining flight hours and tasks going due (Lagos et al., 2020, Sanchez et al., 2020, Sarac et al., 2006). Recent approaches to the problem focus on defining rotations to embed robustness in operations, as explained in section 3.3.

An early approach to the aircraft routing problem is that of Abara (1989) presented in the previous section. This work, however, disregards maintenance in the definition of maintenance routings. Barnhart et al. (1998), on the other hand, develop an integrated model for fleet assignment and aircraft maintenance routing problems. They propose a string-based approach, where strings are sequences of flight legs starting and finishing at a maintenance station while satisfying maintenance feasibility constraints of maximum flight hours and flight time. The model is proposed as a MILP model that minimizes the total cost of assigning fleets a certain set of strings that include all flights in the schedule while meeting continuity constraints. The challenge met in solving this problem is given by the number of existing feasible strings, that must be somehow pre-computed for the model to select them. The authors opt for a branch and price algorithm as a solution method, so that only a limited number of strings must be considered. The authors show that the model can be used for real-life sized long and short-haul problems by finding a solution to a network including more than 1000 flights, 89 aircraft, and 9 fleet types. Given the freedom of constraints that can be imposed on the strings, this string-based formulation has been used by many authors in the literature (Ageeva, 2000, Rosenberger et al., 2004, Sarac et al., 2006).

Differently from Barnhart et al. (1998), the MILP model developed by Sriram and Haghani (2003) considers only short and medium-haul flights pre-assembled in daily routes, that must be assigned to aircraft to meet maintenance constraints in a one-week horizon. The aircraft are assumed to undergo maintenance during the night at selected maintenance stations and are requested to undergo short maintenance every four days and to have a long maintenance opportunity every week. The model is solved heuristically and the quality of the solution is tested on small instances, showing a gap of 5% to the optimal solution. Although this model is capable of solving instances including up to 75 cities and 58 aircraft, it is hard to compare it to the model of Barnhart et al. (1998), since Sriram and Haghani consider pre-defined sequences of daily flights, which do not allow as much flexibility as it is guaranteed by the work of Barnhart et al.. Furthermore, the model proposed by Barnhart et al. is capable of including flight hours constraints in the formulation, while Sriram and Haghani define but are unable to solve a version of their model including flight hours considerations. On the other hand, this model shifts the interest towards maintenance constraints, as it is capable of considering maintenance availability constraints at each maintenance station, and of including different types of maintenance stops at different time intervals.

The models introduced so far can be framed in a tactical context, where aircraft are required to undergo maintenance at predefined intervals specified in flight hours or calendar days. However, such models do not respond to the necessity of modelling more concrete and aircraft-specific maintenance requirements, which can however be imposed in an operational context. Sarac et al. (2006) are the first authors to propose an operational model that addresses the problem of developing daily aircraft routings that satisfy aircraft-level maintenance constraints. The authors develop a MILP model that assigns strings to aircraft so that the remaining flight hours of an aircraft are not exceeded and that a night stop at a suitable maintenance station is guaranteed to all aircraft with tasks going due the following day. The MILP is developed using a connection network structure, and it is solved with a branch and price algorithm, given the impossibility of pre-computing all feasible strings. The model is tested on a fleet of 32 aircraft covering a network of 19 stations with 175 daily flights, where it is capable of providing optimal solutions.

A further step forward in integrating aircraft routing and maintenance scheduling is done by Lagos et al. (2020). In their work, they not only define daily aircraft routes, but also schedule tasks that arrive dynamically to night maintenance slots. The authors model the problem as a Markov decision process, where a sequence of task-maintenance opportunity allocation and rotations assignment over the operating horizon must be chosen for minimal total costs. Costs are modelled as AOG costs and outsourcing costs, i.e. the costs of having a critical and non-critical task going due, respectively. Various approximated dynamic programming strategies are proposed by the authors to solve the problem. The model is tested on a fleet of thirty domestic aircraft, with an operating horizon of thirty days of operations. The case study shows the ability of the model to schedule both one-off and repetitive tasks in various resource availability settings. To the best of the author's knowledge, this work is the most complete when considering the integration of aircraft routing and maintenance task scheduling. At the same time, some assumptions such as the execution of maintenance only during the night make it usable only in the short-haul context.

The previously presented models use the perspective of an airline in defining aircraft routing and maintenance slot assignment. However, maintenance is often outsourced to Maintenance, Repair, and Overhaul (MRO) companies (Al-kaabi et al., 2007), and constraints from their side might apply. Sanchez et al. (2020) change perspective with respect to previous authors by using the perspective of MROs. The authors develop a model that takes as input the preferred rotations of multiple airlines and flight hours intervals allowed for each aircraft, and assigns medium and long-term maintenance slots to aircraft, possibly modifying the preferred aircraft rotations when infeasibility occurs. The authors propose an event-based formulation that they express in two sequential multi-objective MILP models: after the first model defines an initial maintenance slot assignment that minimized infeasibility while considering flight hours constraints, the second model allows tail reassignment to obtain a feasible solution that optimizes resource usage. The event-based definition and the proposed iterative solution allow fast computational time, which makes the algorithm suitable for solving big instances. This is proven by testing the model on a data set including 8 maintenance stations, 529 aircraft, and 16,000 flights over 30 days with a solution time of around 1.5 hours.

It is worth noticing that Sarac et al. (2006), Lagos et al. (2020), and Sanchez et al. (2020) try to schedule tasks as close to their due date as possible. As it will be explained in section 2.2, this can result in operations more prone to disruptions and AOG situations.

2.2. Maintenance planning

Aircraft maintenance is a strongly regulated field, where aircraft airworthiness must continuously be ensured by operators, under the surveillance of authorities, such as the European Aviation Safety Agency (EASA) in Europe and the Federal Aviation Administration (FAA) in the United States (Regattieri et al., 2015).

In general, maintenance can be divided into scheduled and unscheduled maintenance, where the execution of the former is dictated by authority-approved plans, while the latter is initiated by deviations from nominal, regulated conditions (Ackert, 2010). The foundation of every scheduled maintenance program is given by the Maintenance Steering Group-3 (MSG-3), a document that describes the methodology that must be followed when developing a maintenance program (Ahmadi et al., 2007). In particular, the MSG-3 is at the base of the Maintenance Planning Document (MPD), a document produced by the aircraft manufacturer that describes all the scheduled maintenance tasks that must be executed on an aircraft (Ackert, 2010). This authority-approved document is then used to develop each airline's Aircraft Maintenance Program (AMP), which must include the tasks required by the MPD, integrated with additional tasks provided by authorities and other parties such as engine manufacturers (Ackert, 2010). Although the MPD is the foundation on which the AMP is built, it must be stressed that it is the responsibility of the operator to define a maintenance program that guarantees operations safety and reliability, which means that tasks intervals might have to be adapted to particular operating conditions of a certain airline or specific aircraft (Kinnison and Siddiqui, 2013).

In general, **scheduled maintenance** can be divided into three categories based on the initiation of their execution, Hard Time (HT), On Condition (OC), and Condition Monitoring (CM) (Kinnison and Siddiqui, 2013). HT tasks, also known in the industry as **requirements**, must be executed at fixed intervals, which can be defined in terms of Calendar Days (CD), Flight Hours (FH), or Flight Cycles (FC). When more than one interval type is defined, the interval which is reached first dominates. OC tasks are executed whenever a certain condition is reached; often, they derive from an HT inspection. Finally, CM tasks are triggered by models and part monitoring systems that attempt to anticipate a failure event.

Many requirements, especially those with short intervals, are executed on a daily basis on the aircraft. However, airlines often group tasks with a longer, but similar interval into blocks, which are regularly executed during what are known in the industry as **letter checks** (Ackert, 2010). Currently, many airlines make use of three types of letter checks, A, C, and D checks, which are usually executed at intervals of respectively 2-3 months, 18-24 months, and 6-10 years (Deng et al., 2020).

Despite an effort to prevent failures, **unscheduled maintenance** can occur. Whenever this happens, regulations might require that it is corrected immediately, in which case we talk about **non-routine (NR)** maintenance, or it can be deferred by a limited amount of time, leading to a **Deferred Defect (DD)**. DDs can be again divided into **Minimum Equipment List (MEL)** items, and **Non-Safety-Related Equipment (NSRE)**, where in the former case limits on task execution are dictated by regulations, while in the latter they are imposed by the operator. MEL items (Kinnison and Siddiqui, 2013) are defined in accordance between aircraft manufacturers and operators, and, being usually part of redundant systems, a failure in a MEL item does not cause a complete loss of airworthiness, although some limitations such as Extended Range Operations (ETOPS) limitations may apply. NSREs, as the name suggests, are tasks that do not lead to safety-related issues, but an airline may decide to execute them within a certain period of time, such as cabin-related work. A schematic of the maintenance tasks framework just introduced is presented in Figure 2.2.

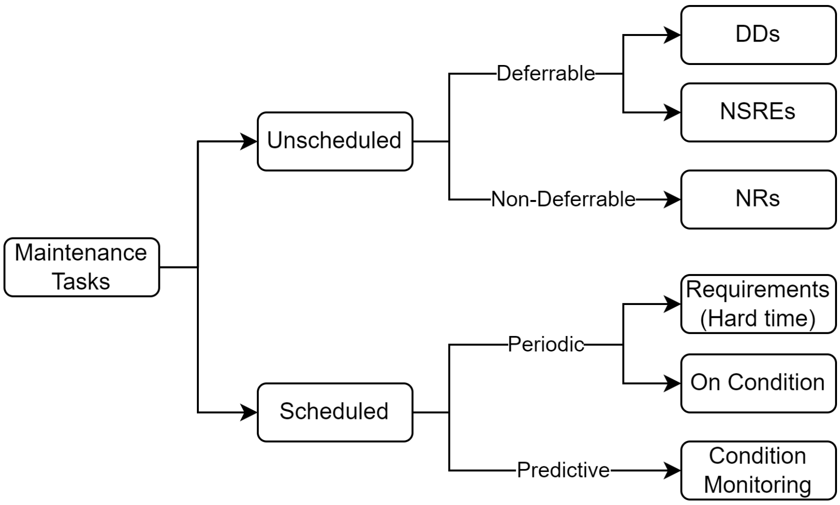


Figure 2.2: Maintenance tasks framework

Given that each aircraft needs to undergo hundreds of periodic tasks (Öhman et al., 2020) in addition to all unscheduled maintenance tasks found over time, one can see how the management of maintenance programs is a complex matter, where all tasks must be executed in time, in order to retain the airworthiness of the aircraft. In fact, although in some cases the postponement of a task past its due date can be granted by regulators (Shaukat et al., 2020), the regular course of events is that whenever due dates of scheduled or unscheduled maintenance are not met, aircraft are grounded until airworthiness is achieved again. This case is known in the industry as **Aircraft On Ground (AOG)**, which has the potential of causing operational costs of more than $70,000 for a Boeing 777 aircraft (Badkook and Basem Badkook, 2016).

For all these reasons, it is vital that efficient and effective planning of maintenance tasks is carried out in order to meet regulations, but at the same time to reduce costs to a minimum. Costs must be taken into account as maintenance expenses represent, on average, 10.5% of an airline's operating costs (IATA, 2020). This requires, for instance, to schedule requirements as close as possible to their due date in order to reduce the *wasted interval* given by the anticipation of a task. At the same time, scheduling tasks too close to their due dates can lead to situations prone to disruptions, and including a buffer in the form of task anticipation can allow an overall reduction of costs (Shaukat et al., 2020). This is why many maintenance planning optimization models have been developed by researchers, dealing with strategic (Boere, 1977, Deng et al., 2020), tactical (Lagos et al., 2020, Shaukat et al., 2020), and operational (Callewaert et al., 2018, Papakostas et al., 2010) maintenance planning and scheduling.

Maintenance planning is a topic that has been investigated in the literature from many different perspectives, and the reader interested in the topic is referred to Van Den Bergh et al. (2013) for a review. In this work, the topic of maintenance planning covers three maintenance planning aspects. subsection 2.1.2 already presented the topic of the maintenance routing problem, where maintenance slots and aircraft rotations are optimized at the same time. This topic has received wide attention in the literature, but at the same time, it is usually more network-operations centred, and only accounts for some maintenance-related constraints. In this chapter the focus is shifted towards maintenance-centred topics: subsection 2.2.1 deals with long-term, strategic maintenance planning regarding letter checks, while subsection 2.2.2 discusses task-level maintenance scheduling.

2.2.1. Letter Checks Scheduling

The task of scheduling letter checks in usually done manually by schedulers, who rely on their experience to develop a feasible maintenance schedule (Deng et al., 2020). In fact, given the complexity of this planning, feasibility, rather than optimality, is the goal the planners want to achieve (Boere, 1977). However, given that a C-check for a Boeing 777 can cost anywhere between \$375K and \$2.8M (Ackert, 2010), it is evident how even a small improvement in checks planning could lead to significant savings. Despite this fact, literature regarding long-term maintenance scheduling is scarce (Deng et al., 2020).

Boere (1977) is among the first to address this problem, by developing a decision support tool based on a non-stochastic simulation in addition to a scheduling heuristic. The scheduling heuristic generates a feasible schedule for A, C, and D-checks over a five years window, with the objective of minimizing the lost interval of the included checks. The optimization is then executed manually by the human scheduler, who generates multiple schedules using different input parameters, and chooses the best-performing schedule on the basis of output indicators. This approach allows the inclusion of constraints such as periods where heavy checks cannot be scheduled and available resources. After being used for two years at Air Canada, the tool allowed the improvement of average interval utilization from 86% to 95%. Despite the relevant results achieved, this is just an elementary tool that supports decision-making through manual optimization.

Many years later, the problem of long-term maintenance scheduling regained the interest of academia, with Deng et al. (2020) proposing a model for scheduling A and C-checks in a four-year planning horizon. The model is formulated in a dynamic programming framework, where a sequence of daily check scheduling actions must be chosen with the objective of minimizing the lost interval. In order to make the problem tractable, the authors propose a forward induction approach based on reduction strategies. First, the daily action space is reduced by only considering actions that meet pre-determined aircraft maintenance priorities and that lead to states for which a solution that does not include grounded aircraft exists. Then, the state space is reduced by grouping states according to interval utilization at every step, and further considering only one out of these multiple states. The algorithm is tested over a four-year planning horizon on a fleet of 45 aircraft of a European airline by comparing the obtained planning with the planning manually generated by the airline planners. The results show a reduction in the number of A and C-checks of more than 7%, which the authors estimate in savings between \$1.1 and \$3.4 million in just maintenance-related costs (disregarding increased aircraft availability). Furthermore, the results show that human planners tend to schedule more checks closer to their due date with respect to the plan provided by the model, leading to a less robust maintenance schedule.

In a later phase, Deng and Santos (2022) extended the work presented by Deng et al. (2020) by considering the uncertainty connected to maintenance slot duration and aircraft flight hours while planning maintenance checks. They formulate the problem as a Markov decision process where the state of the system evolves in two phases, the first depending on the previous step and the decisions made, and the second one depending on information disclosed after the decisions have been made. They reduce the decision space on the basis of a check priority, defined as the expected remaining utilization for each aircraft, and they propose a lookahead approximate dynamic programming methodology to solve the problem. At each forward step of the decision process, C and D-checks are scheduled first on the basis of a deterministic forecast of the necessary additional and unused maintenance C and D-check slots. Then, Monte Carlo simulations are used to produce a stochastic forecast of the additional and unused A and B-check maintenance slots, which is used to determine the optimal A and B-check assignments. The model is tested on cases including up to 50 aircraft and a three-year planning horizon, and the results are compared to those obtained through the deterministic model presented by Deng et al. (2020). The model leads to a worsening in flight hours utilization of less than 1% with respect to Deng et al. (2020), while it allows a substantial reduction of extra slots needed for both A-checks and C-checks.

2.2.2. Maintenance Tasks Scheduling

Maintenance task scheduling is, in general, the problem of assigning tasks to a specific maintenance slot. This problem can be solved using different horizon perspectives, ranging from authors that consider a four-year planning horizon (Witteman et al., 2021), to authors that develop a decision support model for operational use (Callewaert et al., 2018). The topic of task scheduling has received attention in literature only in recent years, but with growing interest. In this section, an overview of the approaches used is given, following a strategic-to-operational logic.

Witteman et al. (2021) consider the problem of optimally assigning both recurring and deferred defects that must be executed in the hangar to pre-defined A and C-check maintenance slots. They formulate the problem as a bin-packing problem, where the available maintenance opportunities are the bins and tasks are the items to fill the bins with. In order to obtain results in a reasonable time, they propose a constructive heuristic as a solution method. In the heuristic, tasks with the same interval as a type of check are assigned to the checks first. The remaining tasks are organized in decreasing order by cost and considered one by one. For each task, a list of the bins that would allow a feasible allocation in terms of the task due date and required labor is produced, and the task is then assigned to the bin with the greatest availability of labor hours. If the task is a recurring task, then the next instance of the task is created and added to the list of tasks, and the algorithm is repeated for the next task in the sorted list. They test their algorithm using data from a fleet of 45 aircraft, and a four-year maintenance slot plan developed using the model proposed by Deng et al. (2020). The constructive heuristic is capable of solving all the considered instances in less than fifteen minutes, and when results in terms of maintenance costs are compared to those obtained through an exact optimization method, the observed gap is never higher than 5%.

Using a more tactical perspective, Shaukat et al. (2020) propose a task scheduling model for line maintenance. The scheduling horizon used by the authors is in the order of weeks, with 3-4 weeks being considered while testing. The model takes fixed aircraft rotations as input, and schedules tasks to a specific time within a line maintenance slot, keeping into account workforce resource constraints. The model focuses on recurring tasks, and it is formulated as a MILP model that minimizes costs associated with executing tasks too early or too close to their due date. Also, some tasks are considered not mandatory, and they can remain unscheduled, at a cost. Two solution methods are proposed. The first one solves the problem to optimality by means of a branch-and-bound algorithm where resource constraints are added progressively to maintain the tractability of the problem. The second approach replicates the approach used by a partner airline, where the assignment of a task to a slot and the definition of their specific start time are done sequentially. The model is tested on small instances considering up to 13 aircraft and 3 maintenance stations over a four-week planning horizon. Although the optimal algorithm provides solutions with objective function on average 3.5% better than the heuristics, it requires run times of up to more than one hour despite the reduced size of the considered instances. However, although the run time makes the optimal solution approach not suitable for usage in real life, one could argue that the approximated approach leads to good enough results for use in real life. Other limitations of the model can be identified in the fact that tasks are scheduled without trying to bundle together tasks of a specific aircraft, and that only required labour hours are considered while scheduling, while task duration is disregarded.

Another work that uses a tactical perspective for task scheduling is Lagos et al. (2020), which has already been presented in section 2.1. To the best of our knowledge, this work is the only one that combines task scheduling with the aircraft routing problem.

Other authors consider an operational time frame in their models. Papakostas et al. (2010) focus on the assignment of tasks to maintenance slots during the day of operations. The model is a simulation-optimization model that optimally assigns line maintenance tasks to maintenance stations throughout the day. First, it determines the possible alternatives of tasks-maintenance station allocation, then, it simulates them through Monte Carlo simulation, and finally chooses the alternative associated with the highest utility or aircraft operability, defined by the authors as "the aircraft's ability to meet the operational requirements in term of operational reliability, operational risk and costs". The utility is computed as a sum of the costs associated with the maintenance assignment, the wasted useful life, the operational risk associated with an assignment, and the simulated delay associated with an alternative. Due to the necessity of simulating all possible alternatives, and to the explosion of the number of such alternatives as the number of tasks and maintenance opportunities increase, this framework is applicable in the short term to a single aircraft at a time and for a limited number of tasks. These limitations, however, are overcome by the fact that, as the authors suggest, the model is developed to be applied in a Condition Based Maintenance (CBM) framework, for a limited number of continuously monitored components.

Finally, Callewaert et al. (2018), develop an operational decision support framework that, during the execution of a maintenance check, suggests the postponement or addition of tasks when maintenance operations are experiencing delays or are ahead of time. Whenever called, the system determines the status of the operations as regular, delayed, or ahead of time. When the latter two cases are found, an action from the system is required, and a solution approach similar to that of Papakostas et al. (2010) is used. In the case where a delay is found, the tasks that still need to be executed are divided into critical and non-critical, based on their due date or prognostic indicators. Then, a list of all critical tasks and a combination of none to all non-critical tasks are ranked based on their cost, operational risk, and associated delay, by means of a Monte Carlo simulation. Differently from Papakostas et al. (2010), the cost instead of utility is used to evaluate the alternative solutions. When an ahead-of-time situation is found, a very similar procedure takes place, where, however, additional tasks are added to the initial work package. The framework is tested through simulation on a data set including 47 tasks, and savings of more than 2,000 are predicted. The small dimension of the data set represents well an operational setting, but no information about run time is provided, for assessing the usability in operations.

A limitation of both Papakostas et al. (2010) and Callewaert et al. (2018) is that when considering the possibility of delaying maintenance to future opportunities, only the currently considered tasks are taken into account. This could cause over-scheduling maintenance for future opportunities, leading to later delays. Another limitation is that they both require task-specific data on the probability of failure in the near future. For Papadakos (2009) this is not necessarily a problem, given the CBM perspective used. Callewaert et al. (2018) suggest the use of probabilistic distributions for this purpose, which could be hard to obtain for all maintenance tasks. Finally, due to the number of generated alternatives, the scalability of these solutions is limited. However, given the operational setting, scalability might not be a key characteristic.

One last observation must be made on all the considered task scheduling models: most of them try to systematically schedule all tasks as close as possible to their due date. Although this approach reduces the wasted interval, it can also lead to operations that are susceptible to disruptions. The only exception is Shaukat et al. (2020), who recognise the problem, and define a preferred buffered due date by which maintenance must be executed.

2.3. Network and Maintenance Operations Interaction

Up until now, network planning and maintenance planning have been considered as two separate topics. The only point of contact is the maintenance routing problem, which, however, tends to be network-centred, with the exception of the work of Lagos et al. (2020) that integrates the maintenance routing and tasks scheduling problems. This separation is partially explicable by the complexity that would derive from the combined optimization of network and maintenance problems, and, at the same time, by the fact that airlines often outsource maintenance to external MRO companies (Al-kaabi et al., 2007), leading to only partial control over maintenance processes and scheduling. At the same time, maintenance and network operations are in close relation to one another, and some works have taken into account this relationship at various levels.

Some authors consider the relationship between maintenance and network simply by considering the effects of using a maintenance policy on total operational costs. Callewaert et al. (2018) consider the costs of network operations delays due to delays in maintenance operations in their task scheduler. Similarly, Regattieri et al. (2015) develop a framework to evaluate corrective and preventive maintenance policies in conjunction with inventory policies, and they consider downtime costs in their operational costs computation. However, in these works, the maintenance-network relationship is still very limited.

Other authors focus on maintenance operations, but acknowledge the effect that a maintenance policy has on maintenance slot reliability and, as a consequence, on network operations. Öhman et al. (2020) evaluate the effects of including a frontlog in maintenance work packages. In other words, they evaluate the effects of anticipating certain tasks so that in each work package there would be tasks that can be postponed without going due, creating flexibility in maintenance operations. When unscheduled maintenance needs to be executed in the check, these tasks can then be rescheduled allowing a buffer in operations. This policy is in contrast to the traditional approach of scheduling tasks as close as possible to their due date in order to optimize interval utilization (see for example Lagos et al. (2020), Sarac et al. (2006)), and, as the authors recognize, this approach deals with a trade-off between the increased workload due to executing some tasks more frequently, and the stability in required workforce and flexibility in operations. Although the focus of their work remains on maintenance-related costs, they also acknowledge an improvement in slots reliability that would lead to reduced delays in network operations that, however, they do not quantify.

Sachon and Paté-Cornell (2000) investigate the direct effect of some maintenance policies on operations. They use probability risk analysis to evaluate the impact of maintenance policies regarding personnel qualification, maintenance timing, and the number of allowed pending MELs on delays, cancellations, and safety. They do so by developing three interacting models representing management decisions, and ground and flight operations, in combination with a Monte Carlo simulation. The downside of this approach is that it allows the evaluation of the impact of one component at a time. At the same time, it allows the evaluation of rare scenarios for which only a limited amount of historical data points is available.

When studying influences between maintenance and network operations many authors make use of simulation, due to the complex interactions that exist between these two systems. Among the authors that make use of simulation, Iwata and Mavris (2013) develop a discrete event simulation (DES) model (see section 4.1 for more details) for simulating the influence of inventory policies and maintenance policies on military operations. The policies that are considered include scheduling policies such as postponing maintenance at the cost of allowing limited mission profile, inventory policies such as the number of spare parts available for substitution, and geographical policies such as the availability and location of maintenance stations. A limitation of this approach is that the authors only simulate a small number of tasks and related components.

Duffuaa and Andijani (1999) introduce a framework for airline operations simulation with a focus on maintenance operations. Their goal is to enable Saudi Arabian Airlines to evaluate the impact of different maintenance policies on airline operations. The framework includes modules for inventory and spare parts, tasks scheduling manpower and equipment, as well as quality control, all interacting with a network operations module. The advantage of this work with respect to previously cited ones such as Öhman et al. (2020) is that in this case the impact of maintenance policies on network operations is evaluated. However, it is not clear from the paper to which level of detail network operations would be modelled.

More recently, Pohya et al. (2021) present a discrete event simulation framework capable of evaluating the effects of using specific products, technologies and policies in the long run, throughout the life cycle of an aircraft or fleet. They propose a modular discrete event simulation model that simulates the complete lifetime of an aircraft, from its purchase to the flights and maintenance executed on it, up until its retirement. The capabilities of the model are demonstrated through a case study of the effects of performing engine wash on an aircraft's engines. The long-term perspective used makes this a very useful model for evaluating the effects of high-level, strategic policies on the overall life cycle of an aircraft. However, due to this wide perspective, both network and maintenance operations are simulated in a simple manner: the aircraft fly whichever flight is departing first, and maintenance slots are not scheduled, but rather executed at fixed intervals or when pre-defined degradation levels of components are reached.

3

Irregular Operations

We refer to not following the schedule as planned as irregular operations. Irregular operations are not an uncommon event, as only 77.6% flights arrived on time in 2019 (within a 15 minutes window around the scheduled arrival time, OTP_{15}), and 1.7% of flights were cancelled in the same year (EUROCONTROL, 2019). This chapter deals with disruptions in operations and with how they can be prevented and solved. section 3.1 introduces disruptions in terms of how they are generated and propagate, section 3.2 focuses on how disruptions are solved when they occur, while section 3.3 explains how disruptions can be avoided or mitigated by increasing operations robustness.

3.1. Disruptions

Disruptions regularly occur during airline operations. In 2019, for instance, only 59.4% of flights arrived within the scheduled arrival time (EUROCONTROL, 2019). Various reasons can be at the origin of disruptions, including weather, staff, airport and airspace congestion, and technical reasons, just to cite some. The International Air Transport Association (IATA) provides unique codes (EUROCONTROL, 2019) to identify the cause of delays and disruptions.

A good reason for understanding the mechanisms of disruptions is that they are expensive. Passengers whose itinerary is cancelled are entitled to compensations according to regulations (EC) No 264/2004 (Cook and Tanner, 2015). The industry refers to these costs are hard costs. In addition to hard costs, however, another category of less measurable costs must be considered, originated by the loss of fidelity of passengers, which causes passengers to choose different airlines in the future (Vos et al., 2015). These costs, known as soft costs, are highly non-linear (Cook and Tanner, 2015) and constitute a big part of the total disruption costs (Cook and Tanner, 2015). Given the complexity of computing hard and soft costs, disruption costs are usually estimated in the literature. Reference values for these estimations are provided by Cook (2015).

When discussing disruptions in airline operations, it is important to notice that when delays and disruptions are generated, they propagate in the network as the availability of resources is constrained. Given this fact, a distinction can be made between root and propagated delays. **Propagated delays** are those generated by the delay of previously, connecting resources. **Root delays**, on the other end, are primary delays with an intrinsic cause. Since EUROCONTROL (2019) finds that 39% of delays registered in 2019 were attributable to reactionary delays, it is evident that understanding the dynamics that regulate delay propagation in the network is important. The literature that deals with delay propagation is presented in subsection 3.1.2.

Finally, **turnaround** and **block time** operations are critical operations with delay generating and delay absorbing capabilities (Arikan et al., 2017, Fricke and Schultz, 2009), and this is why considerations on them are included in this section. Given the abundance of literature on turnaround operations (Schmidt, 2017), subsection 3.1.1 is dedicated to this topic. For what concerns models on block time, they are briefly included in subsection 3.1.2 when discussing delay propagation.

3.1.1. Turnaround Operations

During turnaround operations, many different and sometimes interdependent activities take place, including passenger deboarding and boarding, cabin cleaning and catering, fueling, baggage unloading and loading, water, and waste removal, and aircraft turnaround maintenance check. Many authors (Fricke and Schultz, 2009, Schmidt, 2017) find that passengers handling and cabin activities, as well as fueling, are critical to guarantee a non-delayed turnaround. This is due to the sequential nature of these activities, where cabin cleaning and catering, as well as fueling, normally take place while passengers are not on board the aircraft. Fueling can in some cases take place while passengers are on board, but regulations (Fricke and Schultz, 2009) require that additional safety measures are in place, including the presence of the fire brigade at the stand (Evler et al., 2022).

Turnaround operations have been investigated using different approaches, including critical path analysis (Adeleye and Chung, 2006), analytical models (Wu, 2005, Wu and Caves, 2002a, 2004b), and simulation (Mota et al., 2017). Early work regarding turnaround uses a critical path framework for investigating turnaround operations (Schmidt, 2017). In this framework, the duration of the turnaround can be estimated as the duration of the critical, i.e. the longest among the processes that take place sequentially during turnaround operations. Adeleye and Chung (2006), for instance, use discrete event simulation to model turnaround operations in a critical path framework. Their approach is quite simplistic, as they individuate five operations paths in which they assume fueling and catering operations to be a path on their own, independent from other activities. These are strong assumptions, especially considering that they do not represent the dynamics that happen in reality and that they are found, by other authors (Fricke and Schultz, 2009, Schmidt, 2017), to be part of the critical path. Possibly due to these assumptions, the authors find the baggage handling path to be critical. Some other limitations of this approach are that the simulation does not take into account disruptions due to weather, air traffic management (ATM), and crew or passengers delays.

A different approach is used by Wu and Caves (2002a), Wu and Caves (2004b) and Wu (2005), who progressively develop an analytical model for turnaround time (TAT) duration estimation. The passenger handling process and cargo and baggage handling process are modelled as semi-Markov chains where the process moves from the aircraft's arrival to its departure, through regular and disrupted states. While the baggage handling process is simplified to only unloading and loading activities, the passenger handling process is defined more in detail by including passenger disembarking and boarding, cabin cleaning, passengers and crew lateness, ATC flow limitations, departure operations, and weather disruptions. Other activities, such as fuelling and maintenance checks are assumed independent from the processes described above and are simulated through DES. The final turnaround duration estimated by the model is the longest process among the three described above. Despite this approach being more advanced than that proposed by Adeleye and Chung (2006), it must be noted how the assumption of the independence of fueling is made here as well.

Yet another approach used to assess turnaround processes is simulation. Mota et al. (2017) develop a discrete event simulation that includes turnaround operations and taxi times in order to investigate how different airport layouts influence ground operations. In their stand module they simulate stairs, passengers and baggage handling, fueling, water service, and cleaning, and for each operation, they take into account stochastic vehicle positioning times. A taxi module is also developed to simulate aircraft ground traffic dynamics. The model is developed to evaluate different possible layouts for Lelystad Airport in the Netherlands. In particular, the authors consider three possible airside configurations, for which they analyze turnaround performance and reliability.

A limitation of the approaches considered so far is that they all consider turnaround operations to be independent of arrival delay. Fricke and Schultz (2009) partially relax this assumption by investigating how buffer time between turnaround activities changes in case of delays. For this purpose, they assume a given critical path made of deboarding followed by parallel catering, cleaning, and fueling, and finally passengers boarding, as found in previous research. Turnaround operations are modelled stochastically through a Monte Carlo simulation, as well as the buffer time between activities, which is modelled as dependent on arrival delay. The results show that the buffer time is reduced when an arrival delay happens, giving delay absorption capabilities to turnaround operations. When arrival delays of up to thirty minutes are simulated, turnaround operations are capable of absorbing an average of one-third of accumulated delay.

The possibility of absorbing delay during turnaround operations has received more attention from a ground management perspective than from an airline operations perspective. The only exception is Evler et al. (2022) who recently presented a model that integrates turnaround operations into aircraft recovery (see subsection 3.2.1 for more details). However, from a ground management perspective, the concept of exploiting turnaround delay absorption capabilities has increasingly received attention in recent years, with the development of the concept of a Ground Manager (GMAN) (Evler et al., 2018, Oreschko et al., 2012) in the context of Airport Collaborative Decision Making (A-CDM) (EUROCONTROL).

A GMAN is a tool that should predict target off-block time for the aircraft at an airport, and optimally allocate resources based on those predictions. A proof of concept for a GMAN is presented by Oreschko et al. (2012).In their research, the authors use an improved model based on Fricke and Schultz (2009). The objective of the paper is to show the potential of a GMAN in predicting target off-block time when increasingly certain information is known about aircraft arrival delays.

Evler et al. (2018) do a further step in the development of a GMAN, by presenting a MILP that optimizes aircraft turnaround operations for delay recovery. The authors consider various strategies including the parallelization of activities, allowing additional resources, and reducing the execution of tasks such as cleaning and catering, with the objective of minimizing the total cost of operations including delay costs and recovery actions costs. The model is tested using Monte Carlo simulation on a case limited to four feeders and one intercontinental flight. The results show that the model is capable of reducing mean Turnaround Time (TAT) by up to seven minutes and that the standard deviation of operations cost is also reduced, meaning more predictable operations.

3.1.2. Delay Propagation

Many different approaches have been used in literature to investigate the propagation of delays within an aircraft rotation (Wong and Tsai, 2012, Wu, 2005), and within a network (Arikan et al., 2013, Wu and Law, 2019). Techniques used include semi-Markov chain and discrete event simulation (Wu, 2005), survival analysis (Wong and Tsai, 2012), analytical models (Arikan et al., 2013), and bayesian networks (Wu and Law, 2019).

Wu (2005) investigate the inherent delay of aircraft rotations, i.e. the delay that the authors assume to be embedded in the structure of the rotation itself when flights and buffer times are fixed. A two-module model is used to simulate operations, in combination with a Monte Carlo simulation. The turnaround module is the discrete event semi-Markov chain developed by Wu and Caves (2002a, 2004b) and Wu (2005) as introduced in subsection 3.1.1. The en-route module, on the other hand, models the block time as a random variable to be extracted from a probability distribution. The authors validate the model with data obtained from a point-to-point carrier, where they find that real-world delays are significantly higher than simulated ones. However, the results also show that the model is capable of following trends in delay propagation within a rotation. The model is then used to evaluate different scenarios where root delays are created in operations, and the performance of the schedule is measured in terms of the ability to absorb them.

A different perspective is used by Wong and Tsai (2012), who focus on the factors that influence arrival and departure delays, rather than on how delay propagates in a rotation or in a network. The authors use survival analysis in their work, by considering as survival time the time a delay will last, given that it has started or it has lasted for a certain amount of time. The survival function of arrival and departure delays are estimated from data using the Kaplan-Meier estimator, and the effects of various factors on delays are evaluated using the Cox proportional hazard model. The authors find that aircraft type, flight operations, and cargo mail handling have the biggest influence on departure delays, while arrival delays are mostly affected by block buffer time and weather. The authors suggest that the departure and arrival delay models can be expanded to evaluate the propagation of delay in the network by recursively applying the two models. However, this model expansion is not implemented.

The models of Wu (2005) and Wong and Tsai (2012) respectively investigate and give the basis for investigating the propagation of delay in a rotation, rather than at a network level. This limitation is overcome by Arikan et al. (2013), who develop an analytical model for investigating delay propagation at the whole US network level. In their model, they consider stochastic block time, which leads to the generation of root and propagated delay. They show that a log-Laplace distribution is capable of modelling a random variable that considers both block time and propagated delay, and they evaluate the network through analytical analysis. In contrast to Wu (2005) their analysis shows that previous approaches of deterministic delay propagation overestimate the propagation of delays, and through their models, they identify bottlenecks in airport capacity and critical flights in an airline's schedule.

All the models named so far make assumptions on statistical distributions and independence of subsequent delays of flights. The latter assumption, however, is especially limiting because it disregards the delay absorbing and amplifying capacity of the network that was proved by Fricke and Schultz (2009). A different approach is used by Wu and Law (2019), who develop a Bayesian network in a delay propagation tree framework to investigate delay propagation and delay causes. In the Bayesian network, flights are connected through resources such as aircraft, crew, and passengers. Historical data is used to train the model, but the model is capable of testing hypothetical scenarios for which historical data is not available, predicting delay propagation in the network, and diagnosing the primary cause of a delay.

3.2. Reactive Approaches to Disruptions

Whenever disruptions occur, a solution must be found to recover operations. Each airline has a dedicated team, usually referred to as operations control (OC) that makes decisions on what solution to adopt. Disregarding crew-related causes, typical disruptions that the OC needs to solve include flight delays, aircraft unavailability, and airport disruptions Hassan et al. (2021). On the other hand, the tools and strategies that the OC can use include:

- **Delaying a flight**: this is probably the most simple strategy that can be applied, as it involves letting the delay propagate in the network.
- **Flight cancellation**: often used in the form of cancelling short cycles, i.e. short sequences of flights that start and end at an airport (Rosenberger et al., 2004).
- **Swapping aircraft:** it concerns swapping two rotations, when more buffer time can be achieved, or when it allows the cancellation of cheaper flights.
- **Use of a reserve aircraft:** reserve aircraft are usually kept on standby at the hub, available for taking over other aircraft's rotations
- **Speed control:** allows the reduction of block time for delay absorption. As delay absorption capabilities are a function of the time spent flying, it is more effective on longer flights (Marla et al., 2017). Furthermore, considerations on fuel consumption must be made.
- **Use of ground operations:** Turnaround operations can be executed partially or faster for absorbing delays (Evler et al., 2022).
- **Aircraft ferrying:** it consists of moving an aircraft between stations, without passengers on board. As it is a very expensive solution, it is rarely used in reality (Rosenberger et al., 2003).

In the literature, the problem of operations recovery has been widely addressed (Clausen et al., 2010, Hassan et al., 2021). A big challenge in solving this problem, as opposed to, for example, network planning problems, is that in the operations recovery framework, run time is a fundamental factor to consider. Some authors (Vink et al., 2020) have suggested that in an operational context a solution should be provided within one minute. Early works on this topic have focused on solving big problems, while often allowing run times in the order of 10 minutes (Arikan et al., 2017, Bisaillon et al., 2010, Evler et al., 2022), which would not be usable in an operational context. Furthermore, these approaches tend to work in a static manner, in which a solution is provided on the basis of some input, but the solution cannot be improved from previously found solutions. These approaches are presented in subsection 3.2.1. More recent works (Vink et al., 2020, Vos et al., 2015) overcome these limitations by developing dynamic approaches, as described in subsection 3.2.2.

3.2.1. Static Approaches

Among the first authors to address the aircraft recovery problem, we find Teodorovi and Guberini (1984). They develop a model that defines new aircraft rotations when one or more aircraft are unavailable while minimizing total passenger delay. The authors assume one fleet and only allow delaying flights as a recovery policy. The MILP model exploits a connection network structure, and it is solved by means of a branch and bound algorithm. The authors present a simple example of the model application, which is quite far from a real-life operational environment.

Later works expand the considered disruptions types and recovery options. Rosenberger et al. (2003) consider flight delays and aircraft unavailability, as well as airport limited capacity and closure, and find a recovery solution that allows delaying and cancelling flights and swapping aircraft. The authors do not consider aircraft ferrying and inter-fleet swaps, as they claim that these options are rarely used by airlines, respectively due to cost and the necessity of crew considerations. The proposed solution is a string-based routing model that pre-computes maintenance feasible routes for each aircraft and then chooses one of them. Differently from already presented string-based models such as Barnhart et al. (1998), here flights are not necessarily included in the selected routes, in order to allow for flight cancellation as a recovery option. For what concerns the objective, the model minimizes the cost of cancellations, delays, and aircraft swaps. In an attempt to improve operational costs, the authors also suggest an alternative model that minimizes passenger and crew misconnections. Both models can be solved by means of a commercial solver after a reduction of the aircraft to be considered in the solution is done heuristically. When both models are tested using the airline simulator SimAir (Rosenberger et al. (2001), see section 4.1 for more details) on fleets of up to 96 aircraft, significantly improved flight cancellations and passengers misconnections results are obtained with respect to a simple short cycles cancellation recovery policy. Furthermore, the model is fast, as it delivers a solution within 16 seconds even for the biggest instances.

As Hassan et al. (2021) observe, in recent years authors started focusing on recovery models that would consider not just aircraft recovery, but also crew and passengers recovery in integrated recovery models. An example that integrates aircraft and passengers recovery is given by Bisaillon et al. (2010), who won a challenge (Palpant et al., 2009) proposed by the French Operational Research and Decision Support Society (ROADEF) to address the integrated recovery of aircraft and passengers. The challenge consists in developing a cost-minimizing model that allows recovery after flight, aircraft, and airport disruptions while considering maintenance constraints, airport capacity constraints, and preference for the final positioning of the aircraft. Bisaillon et al. (2010) present a solution based on a three-phases large neighbourhood search heuristic. The *construction* and *repair* phases are repeated iteratively and they find an initial feasible solution by delaying, cancelling, and reinserting flights in new rotations, and passengers are rerouted on the new flights. Then, a better solution for passengers' itineraries is found by iteratively delaying flights and re-accommodating passengers. The algorithm is tested on the real-life-sized instances provided for the challenge, where it is always capable of providing a feasible solution within the 10 minutes time window allowed by the challenge.

Another integrated approach is proposed by Arikan et al. (2017), who also include crew recovery in their model. This model allows the use of a wide range of recovery options, including aircraft ferrying, cruise speed control, the use of spare aircraft and reserve crew, and crew rerouting and deadheading. The model is based on a modified connection network where single aircraft, passengers, and crew flow as entities. This formulation allows the modelling of each entity separately, including for instance aircraft fleet and maintenance constraints or crew mandatory rest periods. The model is formulated mathematically as a conic quadratic mixed integer programming model, after the reformulation of a non-linear model, where non-linearity is due to fuel consumption constraints. Using preprocessing and problem reduction strategies, the model is able to find optimal solutions for a network including up to almost 500 flights, and approximated solutions for a complete network including more than 1,200 daily flights. Furthermore, the model produces an approximated solution in less than two minutes for smaller instances, and in less than 10 minutes for the complete network.

Until recent years, not much interest was placed on using turnaround operations as a recovery option. This might be due to the fact that ground operations are not necessarily under the direct control of an airline, since other external operators such as ground handlers are often involved. Evler et al. (2022) have recently presented an integrated aircraft, crew, and passenger recovery model that includes options for turnaround quick procedures, as well as stand assignment for shorter transfer time and quick passenger transfer. In addition to this, resource-specific constraints are used to represent the limited availability of airport resources. The model is a cost-minimizing MILP based on a connection network structure. A solution to the model is found for each hub bank within a rolling horizon framework, in order to make the model tractable. The authors test their model on data of a small hub and spoke airline operating 17 aircraft and 85 daily flights, only considering delays of up to 90 minutes as disruptions. The results show that integrating turnaround recovery options into aircraft recovery leads to a reduction of costs of more than 20% for some instances. Furthermore, for the less disrupted instances, action on turnaround operations is capable of avoiding any spread of delay to later hub banks. The greatest shortcoming of the model is its long run time, which reaches up to 30 minutes for the solution of one hub bank, in the small case considered.

3.2.2. Dynamic Approaches

All the authors mentioned in subsection 3.2.1 consider a static approach to disruption management, where all disruptions are known at the time of finding a solution. This approach does not depict the dynamic environment of operations, where a new solution must be found whenever a new disruption comes up, and new solutions must be based on the previously chosen one. These limitations were found to lead to a misestimation of solution costs of up to 80% (Vos et al., 2015).

A different modelling framework is proposed by Vos et al. (2015). The proposed model iterates between an aircraft selection algorithm that reduces the dimension of the solution space by limiting the aircraft considered for recovery, and a linear programming model that minimizes the cost of the solution. The model is capable of addressing flight delays and cancellations, aircraft unavailability, and airport closures by delaying and cancelling flights and swapping aircraft. Furthermore, the use of a time-space network representation for each aircraft allows the inclusion of aircraft-specific constraints such as maintenance and passenger capacity. The cost function takes into account both hard and soft costs, but it is unable to consider costs derived from passenger misconnections. The model is tested in a simulation environment using data provided by Kenya Airways, and the results show lower solution costs than those obtained using a static approach. However, run times are quite long, as the model requires up to 15 minutes for providing a solution.

Vink et al. (2020) also present a dynamic recovery model. Their work is similar to Vos et al. (2015) in terms of the iterative use of a heuristic aircraft selection algorithm and linear programming model, as well as for the use of the parallel time-space network formulation. Furthermore, the same disruption types and recovery options are considered. However, the model proposed by Vink et al. (2020) also includes considerations regarding the station where aircraft are located at the end of the disrupted period, maintenance constraints, and costs incurred due to disruptions of connecting passengers, and it also makes an attempt to keep the initial schedule as unchanged as possible. Furthermore, the model provides a sequence of solutions as new, better solutions are found while the optimization is running. When tested on historical data provided by a domestic American airline, the model is always capable of providing a first feasible solution within a few seconds, and the solution cost stabilizes within 50 seconds, by providing a solution that is on average 6% worse than the solution where no aircraft selection algorithm is used.

3.3. Proactive Approaches to Disruptions

Although operations can be solved as they occur, often the generated solutions can be expensive. Another approach to manage irregular operations concerns acting proactively (Abdelghany and Abdelghany, 2018), i.e. acting on the operations planning step to generate schedules and rotations that are more robust and inherently less prone to disruptions.

Before diving into how operations robustness can be achieved, it can be useful to try and define the concept of robustness. In general, robustness is made up of the combination of two components, *stability* and *flexibility* Burke et al. (2010). Whereas **stability** refers to the ability of the schedule to avoid the generation and spread of disruptions within the network, **flexibility** focuses on the recoverability of a schedule, i.e. on how easy and expensive it is to recover the schedule after disruptions occur.

As for how robustness can be achieved, two general strategies exist (Abdelghany and Abdelghany, 2018, Aloulou et al., 2010): the use of *time flexibility*, and the use of *resource flexibility*. **Resource flexibility** consists in aligning resources in a way that, in case of disruptions, the schedule can be recovered easily. This includes, for example, using aircraft routings that give many opportunities for swapping aircraft (Ageeva, 2000), or routings that include many short cycles, as opposed to big loops, that can be cancelled in case of disruptions (Rosenberger et al., 2003). Another generic strategy for adding robustness to operations is increasing time flexibility. **Time flexibility** (Aloulou et al., 2010) refers to the strategical use of buffer times in the schedule, so that delay can be (partially) absorbed, and its spreading mitigated. Buffer time in the schedule can be added in two forms: as block time buffer, and as turn around buffer. In reality, airlines use buffer time in their schedule in both block time(EUROCONTROL, 2019) and turn around (Wu, 2005).

However, buffer time comes at a cost. Ball et al. (2010) estimate that the cost of including buffer time in airline schedules was $3.7 billion in 2007. This cost, also known as **opportunity cost** (Wu and Caves, 2002b) is driven by the fact that aircraft do not generate revenues when they are not flying. Therefore, it is evident how a tradeoff exists between the use of buffers for guaranteeing robust schedules and operating more. This is probably why numerous authors in the literature (Aloulou et al., 2010, Ben Ahmed et al., 2017, Lan et al., 2006) only consider reallocating buffer time in the schedule rather than increasing it by reducing the number of scheduled flights. It is only in more recent times that authors have started investigating the possibility of adding buffers in the schedule from the schedule design stage of planning (Cadarso and Marín, 2011, 2013, Jamili, 2017, Kenan et al., 2018).

At this point, an observation must be made on the use of reserve aircraft for increasing operations robustness. The use of **reserve aircraft** is a very easy (although expensive) way of increasing resource robustness in operations. Using another perspective, they could also be seen as a particular use of buffer time, where a buffer is assigned in one big block to one aircraft, rather than distributed in the network. Despite the consideration of spare aircraft in operations recovery literature (Arikan et al., 2017), and their use in reality, to the best of the author's knowledge, no paper that considers aircraft reserves in the context of generating robustness was ever published. However, although the question of how many reserve aircraft should an airline use can sound trivial, the question becomes more challenging when a buffer-oriented point of view is used, and the possibility of strategically including 'concentrated' buffers in the schedule is considered.

This section gives an overview of the approaches that have been used in literature to increase operations robustness. subsection 3.3.1 presents works that make use of time flexibility by allocating the available buffer in the schedule, subsection 3.3.2 includes works that add time flexibility by adding a buffer in the schedule during the schedule design phase, while subsection 3.3.3 focuses on papers that propose strategies that increase resource flexibility.

3.3.1. Time Flexibility by Allocating Buffer

In the literature, many authors address the problem of reallocating slack time in the schedule, with the objective of reducing the propagated delay or reducing passenger misconnections. The most common approach, used for instance by Lan et al. (2006), Aloulou et al. (2010), and Ben Ahmed et al. (2017) deals with determining optimal aircraft rotations and flights departure time within a small time window. For what concerns aircraft routings, the optimal slack time allocation concerns placing the available buffer time strategically in between flights, in order to limit the spread of propagated delay (Aloulou et al., 2010). An additional effort for reallocating slack time can be done by allowing flight retiming within a small time window as proposed by Levin (1971). Limiting the adjustment of the schedule to a short time window around the originally planned departure time allows assuming an unchanged demand for the adjusted flight (Burke et al., 2010). Furthermore, schedule adjustments are common practice in airline operations, where scheduled departure times are revised starting from several weeks before operations up until the DOO (Lan et al., 2006).

Lan et al. (2006) propose to apply an aircraft maintenance and routing model and an aircraft retiming model in sequence in order to optimally distribute slack within the schedule. The first model is a string-based stochastic routing model, where the schedule is supposed to repeat daily. The objective function minimizes the expected propagated delay, and it is made deterministic by pre-computing the expected delay associated with each considered string on the basis of historical data of an American carrier. When aircraft rotations are determined, a second stochastic MILP problem is used to retime flights so that expected passenger misconnections are minimized. The model does so by choosing one out of several copies of each flight, all included within a small time window around the original departure time. Once again, the model is made deterministic by precomputing the expected passenger misconnections given a certain combination of copies of subsequent flights. Given the number of possible strings and flight copies, a branch and price algorithm is used to solve both models. The two models are tested using historical data from a major carrier. The results show a noticeable improvement in operations, with a 1.6% improvement in OTP_{15}, as well as 40% passenger misconnections reduction when allowing a ±15 min rescheduling time window.

Similarly to Lan et al. (2006), Ben Ahmed et al. (2017) develop two models to be solved sequentially. The first model is a MILP that finds optimal weekly aircraft routings that include regular maintenance stops by minimizing a surrogate measure of robustness expressed as a function of turnaround time between subsequent legs. The underlying assumption here is that more slack time between subsequent resources will lead to more robust operations. After aircraft rotations are found, a heuristic-simulation model is used to retime flights. This approach iteratively shifts flights to the right or left of a certain time window, based on the results of a Monte Carlo simulation that produces flights reactionary delays and network OTP performance estimates. The models are tested on real weekly schedules of Qatar Airways using SimAir, and the results show relevant improvement of operations, with a 9.8-16% improvement of simulated OTP_{15}, for different instances.

The aircraft routing and flight retiming problems are unified in one model by Aloulou et al. (2010), who propose a MILP model that minimizes a surrogate measure of robustness. This surrogate robustness is an improved function of that proposed by Ben Ahmed et al. (2017), where this time a component dependent on passengers' connection time is also considered. The model is tested on some of the instances of the ROADEF Challenge (Palpant et al., 2009) by simulating disruptions and using a push-back recovery strategy. Results are not extensively discussed, but an improvement in flights delay and passenger misconnection is observed.

3.3.2. Time Flexibility by Adding Buffer

Another way of increasing time flexibility concerns strategically adding buffer time in the schedule, rather than just allocating it. Early works that apply this concept use analytical models to determine the optimal buffer time to be added to flights within a rotation (Wu and Caves, 2002b) or isolated flights (Wu and Caves, 2004a). Only recently, authors have started developing optimization models capable of developing schedules on the basis of market demand, while considering robustness constraints or objectives. We can distinguish these works into two categories, respectively having a focus on reducing passenger misconnections (Cadarso and Marín, 2011, 2013), and on reducing the effects of propagated delay (Jamili, 2017, Kenan et al., 2018).

An early model that tries to optimally add buffer time to the schedule is Wu and Caves (2002b). The authors develop an analytical model that determines the optimal block and ground buffers to be assigned to each flight in a daily rotation with the objective of minimizing disruption and opportunity costs. The model is tested on one test rotation by comparing the outcome of the optimal schedule with the original one, and the results show an improvement in expected punctuality up to 40% for all legs. However, it must be mentioned that this work adds buffers between the flights of an aircraft route without considering the final departing time of such flights. This might lead to solutions that are not valuable from a market offer perspective or to solutions that are not even feasible due to the excessive shifting of the last flights in the rotation, which can, for instance, end up being scheduled during the night.

Another work that analytically addresses the allocation of buffer time is Wu and Caves (2004a), whose objective is to determine the optimal allocation of buffer time for individual flight legs. They develop a piecewise linear function that links arrival and departure delays and re-elaborate the cost function already developed by Wu and Caves (2002b) to be a function of departure delay. They conclude that the buffer time for each flight depends on the punctuality pattern of inbound aircraft, which can be determined for each operated route, as well as on the relative weight of delay costs and opportunity costs that an airline experiences. although this approach is useful in determining how much buffer time should be included for each flight in the schedule, is it limited by the fact that it considers isolated flights, rather than flights in a full schedule.

Cadarso and Marín (2011) are among the first authors to consider robustness within network planning, and they do so by using a passenger-oriented perspective, where considerations are made on passengers' connecting time. They develop a MILP model capable of selecting the frequency and timing of flights, as well as assigning the selected flights to a specific fleet. The model is a time-space network that, given a time-dependent unconstrained demand between airports, accommodates passengers on itineraries comprising up to two legs. The model minimized a cost function, where one of the terms accounts for the costs of expected misconnected passengers in two-legs itineraries. These costs are computed assuming that the probability of observing a misconnection decreases exponentially as the connection time increases. In order to account for the fact that longer connection times are undesirable for passengers, the objective function also includes a cost proportional to the connection time.

The work of Cadarso and Marín (2011) is expanded in Cadarso and Marín (2013), where the objective is transformed into a profit-maximizing function that, with respect to the previous work, includes costs associated with recaptured passengers. A further improvement is that some flights are considered mandatory in the schedule; the computation of misconnected passengers is not improved with respect to the previous model. Both Cadarso and Marín (2011) and Cadarso and Marín (2013) are tested on a test instance including 23 airports and 77 aircraft, and they compare the results to those obtained using a version of the model that do not consider robustness costs. Although the non-robust solutions show smaller costs associated with passenger dissatisfaction, they are also associated with higher expected misconnections. It is, however, worth noticing that neither one of the models is tested in a simulation environment.

In both Cadarso and Marín (2011) and Cadarso and Marín (2013) robustness is integrated into the model by promoting buffer time during passengers' connections. However, these approaches disregard the effect of accumulated delay within an aircraft' route. A different approach is used by Jamili (2017). They develop a MILP model capable of defining flight frequency, fleet assignment, and aircraft routes, given the demand of origin-destination markets. The model is capable of accommodating passengers in itineraries comprising up to two legs, and it maximizes profit, computed as the difference between revenues and operating costs. Robustness is included in the model by imposing some buffer time between the flight legs. The buffer before a flight leg is computed as a function of the number of take-offs and landings that precede the leg in the operating aircraft's route, and of the expected block time disruption accumulated over the preceding flights. The model is tested on small instances including six airports and six aircraft using two different solution heuristics, and results are briefly presented. The solutions are not evaluated in any way to test their effective added robustness. The model shows two main limitations: it considers a fixed demand for a market, independent of the time of the day, and it imposes a rule-based buffer time in the schedule without optimizing its allocation.

The limitations encountered in Jamili (2017) are partially overcome by Kenan et al. (2018), who develop a model in which the objective function maximizes the expected profit while considering the expected costs due to the propagated delay. They develop a two-stage stochastic programming model that determines which optional flight legs should be added to a mandatory schedule while considering the demand associated with each optional flight leg as a stochastic variable assumed to follow a truncated normal distribution. Robustness is included by adding a term in the objective function, which computes the costs associated with the expected propagated delay of a flight leg given the previous flight in the operating aircraft's route. The model is solved using the Sample Average Approximation algorithm, which allows the solution of small test instances. In order to be able to solve bigger problems, the authors reformulate the model in a string-based format and present three variants of the column generation algorithm to solve it. The authors test the model in its second formulation on instances comprising up to 59 aircraft, 45 airports, and 228 legs, where they are able to solve the problem to optimality within less than four hours. However, in this case, as for the previously cited authors, the model is not tested in a simulation environment. Some shortcomings can be identified in this model. First, demand is considered per leg instead of per origin-destination market, making the model suitable only for point-to-point airlines. Also, the timing of flight legs is fixed, which considerably limits the freedom in schedule development.

3.3.3. Resource Flexibility

Increased resource flexibility at the aircraft level (i.e. disregarding crew) has been achieved in two ways in literature: by increasing the number of short cycles in the schedule, so that they can be cancelled during operations recovery Rosenberger et al. (2004), and by increasing the number of opportunities for swapping aircraft (Ageeva, 2000, Burke et al., 2010).

Rosenberger et al. (2004) propose a fleet assignment model that maximizes the number of short cycles included in the rotations and reduces hub connectivity, defined as the number of flight legs in a route that starts and ends in different hubs only stopping at spokes in between. According to the authors, such an approach would increase operations robustness by avoiding disruption spreading between hubs, and by allowing faster schedule recovery thanks to the short cycle cancellation strategy. After demonstrating analytically that reduced hub connectivity leads to a higher lower bound to the number of cycles included in a rotation, the authors focus on minimizing hub connectivity by developing four models that consider a combination of operational costs and hub connectivity as objective or constraint. The models are based on the work of Barnhart et al. (1998), already introduced in subsection 2.1.2 However, Rosenberger et al. (2004) only consider strings that start and end at a hub. SimAir is used to run a 500 days simulation on three different schedules and fleet scenarios to test and compare the models. Results show a marginal improvement in on-time performance, cancellations and necessary swaps required when using the proposed models with respect to the original model by Barnhart et al. (1998).

A different approach to increase robustness is used by Ageeva (2000), who develops an aircraft routing model that maximizes the aircraft swap opportunities in the routings. The authors find a solution in a two-step algorithm. In the first step, an optimal solution that disregards robustness is found using the string-based aircraft routing model by Barnhart et al. (1998). Then, alternative optimal solutions are found, and a measure of their robustness is produced as a function of the time multiple aircraft have a certain overlapping time at an airport. The best-performing solution is then chosen as the result of the algorithm. The model is tested on small instances including up to 37 flights, and the results show that the second step of the algorithm is capable of significantly increasing time overlaps at the airports. However, the solution is not tested by means of simulation, to actually check if more robust operations are achieved.

Burke et al. (2010) also consider swap opportunities, along with other strategies, to improve operations robustness. In their work, they develop a weekly aircraft and maintenance routing and flight retiming model that optimizes slack allocation, swap opportunities, and reserve aircraft appointments. The authors develop a weekly time-space multi-commodity flow model to represent the schedule with multiple flight copies for flight retiming purposes. Two objectives are considered in the model, schedule reliability and flexibility. The reliability of a schedule is expressed as the sum of the probability of each flight in the schedule being delayed and must be minimized. Flexibility, on the other hand, is related to swap opportunities and is defined as the sum of the probability of both flights involved in a swap leaving on time, for all swap opportunities existing in the schedule. The necessary probabilities are computed on the basis of historical distributions for turnaround and block time. The model is solved using a multi-meme memetic algorithm that combines a genetic algorithm with three local search operators. Each chromosome represents a final schedule, and whenever a new generation is produced, a local search for a gene of a certain chromosome is executed with a certain probability. All non-dominated schedules are evaluated using the KLM operations simulation tool (Jacobs et al., 2005), and the results show an average 2.1% improvement of OTP_{15} with respect to the original schedule.

4

Operations Simulation

Simulation, and especially stochastic simulation, allows evaluating scenarios and models in an environment that tries to replicate real-life (Iwata and Mavris, 2013). This is why simulation is very useful in the airline operations field, where resources, schedules, and disruptions interact with each other in a complex manner. This chapter gives an introduction to existing simulation techniques and software for implementation (section 4.2) and to how simulation has been used in literature for simulating airline operations (section 4.1).

4.1. Airline Operations Simulation

Simulation is widely used in literature in the field of airline maintenance and network operations planning and optimization. While some works use a simulation framework as an instrument for testing their models (Aloulou et al., 2010, Barnhart et al., 2002, Vos et al., 2015), other papers focus on simulation models themselves, with the objective of using simulation to assess scenarios and to support decision making (Duffuaa and Andijani, 1999, Iwata and Mavris, 2013, Jacobs et al., 2005, Mota et al., 2017). While some of these works are developed by airlines, who are interested in evaluating what-if scenarios in their operations (Duffuaa and Andijani, 1999, Jacobs et al., 2005), other models are mainly developed for research purposes, to allow model testing and comparison (Rosenberger et al., 2002).

Jacobs et al. (2005) describe how the operations simulation model OPiuM (from Operational Plan Management) is used by KLM OC. According to the authors, when the Network Department proposes a schedule to OC, OC evaluates the schedule feasibility using OPiuM, before accepting it. The model assesses the schedule by simulating disruptions and recovery, allowing aircraft swaps, the use of a reserve aircraft, reducing maintenance time, and cancelling flights. Further details on the model are not discussed in the paper, which is more focused on model implementation rather than model architecture.

As already introduced in section 2.3, Duffuaa and Andijani (1999) and Iwata and Mavris (2013) present two simulation-based works that involve both network and maintenance operations. Duffuaa and Andijani (1999) present a framework for airline operations simulation with a focus on maintenance operations. Their goal is to enable Saudi Arabian Airlines to evaluate the impact of different maintenance policies on airline operations. The presented framework is modular and includes interactive modules such as a planning and scheduling module for maintenance planning, a supply and inventory module for allowing different spare parts management policies, an organization module for stations availability and personnel rules, and an airline operations module to simulate the interaction between maintenance and network operations. The work of Duffuaa and Andijani, however, only presents a framework for airline simulation, while the implementation and interaction of the modules are not explained.

Another approach that simulates mission and maintenance operations is proposed by Iwata and Mavris (2013). However, it is relevant to notice that the focus of this work is on military, rather than airline operations. The model is a DES with a modular structure that includes mission, maintenance, and parts logistics. Contrary to the approach of Duffuaa and Andijani (1999), the authors describe the simulation dynamics, which include task-wise failure and maintenance operations simulation. The model can be used for assessing maintenance policies such as postponing tasks execution and parts logistics.

The work of Jacobs et al. (2005) and Duffuaa and Andijani (1999) makes clear that airlines value the insights that operations simulation can provide. Rosenberger et al. (2000, 2001, 2002) and Lee et al. (2003), on the other hand, present SimAir, a model for simulating airline operations developed for use in academia. The objective of this model is to provide researchers with the possibility of testing their models and solutions in a common framework, allowing comparison. This framework has been indeed widely used in literature for testing models, see for example Ben Ahmed et al. (2017), Lan et al. (2006), Rosenberger et al. (2004). SimAir is a discrete event simulator capable of simulating airline schedules and recovery strategies, including turnaround and block time, weather and influences from other airlines, and crew and passenger flow. However, maintenance is simulated in a simplified manner, by only considering regular maintenance stops and unscheduled maintenance in between flights with a certain probability. Three modules are at the basis of SimAir. The *controller module* keeps track of the simulation and, whenever it detects a disruption, it calls the *recovery module*, which finds a solution to recover operations. Finally, the *events generator module* is responsible for defining stochastic processes' occurrence and duration. The modular structure allows simple adaptation of the model to specific needs, such as the use of different recovery strategies in the recovery module.

Using a more strategic perspective, and orienting their work to both academia and industry, Pohya et al. (2021) develop a discrete event simulation of the whole life cycle of an aircraft, as already introduced in section 2.3. Their model includes modules of network operations and scheduled and unscheduled maintenance. A flight schedule is performed so that each aircraft executes the first departing flight as long as the departure time does not violate an airport curfew. Scheduled maintenance is executed either at fixed intervals or when a certain level of degradation of an aircraft component is reached, while unscheduled maintenance is simulated through stochastic failures of components. However, the biggest added value of this model with respect to previously cited ones is its capability of estimating costs and revenue throughout the life of an aircraft. This includes revenue obtained from executing flights and costs of purchase and ownership of the aircraft, scheduled and unscheduled maintenance, fuel, and crew. This allows, despite the quite simple operations simulation model, to effectively evaluate the effects of the use of strategic policies, technologies and products in the long run.

4.2. Implementing a Simulation

Simulation is an approach that allows the evaluation of different scenarios, that would be impractical to test in reality (Iwata and Mavris, 2013). In particular, simulation allows the assessment of scenarios in which complex and in some cases stochastic interactions occur, as it can lead to the identification of emergent properties of the system, that static methods are not capable of evaluating. The one downside of simulation (Iwata and Mavris, 2013) is that the run times are often long, especially when a stochastic simulation is used, and Monte Carlo techniques need to be applied to reach statistically significant results. Furthermore, (Iwata and Mavris, 2013), both model validation and statistical analysis of the results are critical steps for generating reliable conclusions.

Many different simulation techniques exist, with system dynamics (SD), discrete event simulation (DES), and agent-based simulation (ABS) being the most used in operations research (Jahangirian et al., 2010). **SD** (Maidstone, 2012) is a usually deterministic and continuous technique that models a system as composed of stocks or containers, flows between these stocks, and delays by which actions on the system take place. **DES** (Ross, 2013) differs from SD as it is a discrete and usually stochastic simulation technique that models a system as a network of queues between which single entities move. It is worth noticing that **discrete time simulation**, i.e. a simulation where time advances in fixed time steps, can be seen as a special case of a discrete event simulation. In the literature, this approach is referred to as *activity-oriented* paradigm Matloff (2008), which is better explained in subsection 4.2.2. Finally, **ABS** (Maidstone, 2012) is the most recent technique among the three. It does not model the system as a whole, but it models autonomous agents that interact with each other and with the environment, originating emergent properties that characterize the system. Recent works have increasingly used these three techniques in combination with each other, in what is known as hybrid simulation (Brailsford et al., 2019).

Examples of the use of these simulation techniques in the airline industry exist, although each technique is usually used for simulating different aspects of the industry. Liehr et al. (2001) use SD to investigate airline business cycles, by modelling the interaction between capacity demand and supply. Bouarfa et al. (2016) use ABS to model the dynamics of decision-making within OC while solving disruptions. DES is the most widely used simulation technique when it comes to simulating airline operations dynamics. This can be easily seen by the fact that all the simulation models introduced in the previous section (Duffuaa and Andijani, 1999, Iwata and Mavris, 2013, Jacobs et al., 2005, Rosenberger et al., 2001) choose DES as simulation approach.

Given the straightforwardness of modelling airline operations as aircraft moving in between flights, turnaround and maintenance operations, DES is the chosen simulation technique for the project that this work supports. For this reason, subsection 4.2.1 deepens the topic of DES as a simulation approach and subsection 4.2.2 gives an introduction to software and libraries for DES implementation.

4.2.1. Discrete Event Simulation

In a general framework, DES can be seen as the simulation of how three types of variables evolve (Ross, 2013). The variables include (Ross, 2013):

- **Time variable:** keeps track of the the simulated time
- **Counter variables:** counter of how many times events have happened
- **System state variables:** describe the system at a certain point in time

The idea in discrete event simulation, as the name suggests, is that time is not discretized in fixed increments, but rather in uneven steps determined by the happening of *events*, which are the only moments in time when the system state changes. Therefore, whenever an event happens, the time, counter, and system state variables are updated, and their evolution in time is the simulation output. This technique can be adapted to any problem in which the changes in the state are not continuous, but rather attributable to identifiable events. The advantage of this technique with respect to continuous time simulation is that it usually runs faster since only a reduced number of time steps need to be considered.

This general framework is quite abstract, and it can be hard to develop a simulation using this perspective. A more straightforward way of understanding DES is through a process-based framework. In this new framework, DES can be seen as a simulation technique that models reality as a network of *queues* and *activities* in which *entities* flow while consuming (possibly renewable) available resources (Brailsford et al., 2014). A representation of such a network is presented in Figure 4.1.

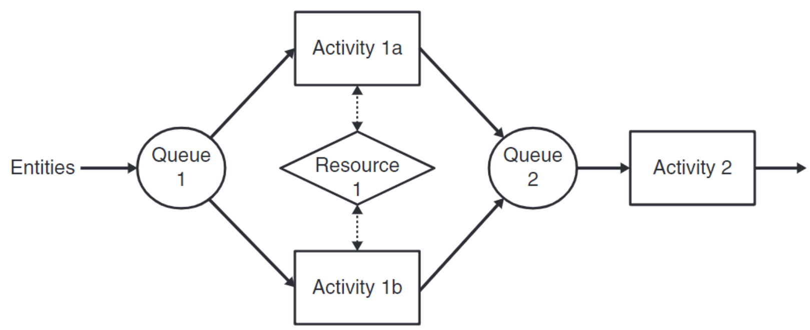


Figure 4.1: DES network model (Brailsford et al., 2014)

According to this view, a DES model is made up of four key blocks (Brailsford et al., 2014):

- **Entities**: single items represented in the system. In the airline operations framework, an entity could be for instance a single aircraft, a crew, or a passenger.

- **Queues**: queues where entities line up to receive an action. An example could be a queue at the runway or a queue for receiving handling services.

- **Activities**: work done on or for entities, that can require some time, possibly stochastic in nature. For example, the time to taxi from a stand to the runway.

- **Resources**: resources necessary to perform an activity on an entity, such as available push backs, or hangar space to execute maintenance.

From this perspective, events can be seen as the arrival of an entity in the system, and its transition between queues and activities (Brailsford et al., 2014). Using this process-based perspective, it is immediate to see that DES is a good approach whenever processes are defined clearly, but interactions between entities and resources are complex. Since airline operations match this description, DES is a good approach for modelling them.

Finally, it must be stressed that any simulation developed using a process-oriented perspective can be translated into an event-oriented perspective, by finding the relationship between activities and queues and the system state. There is no unique way of modelling the system state, but the state should give a unique description of a system at a certain point in time. State variables must therefore be adapted to the problem at hand, but they can for instance include the number of entities in a queue, the number of entities undergoing an activity, or the total number of entities in the system.

4.2.2. Software for Simulation

When implementing a DES, modellers can use three different supports (Brailsford et al., 2014): spreadsheets, simulation software, and programming languages. While the use of spreadsheets is quite limiting in terms of complexity that can be implemented, both the other methods allow the development of complex simulations. Examples of commercial software used in airline operations literature include for example AnyLogic, Arena, Extend, Simio, and WITNESS (Brailsford et al., 2014, Iwata and Mavris, 2013, Schmidt, 2017). While the advantage of this software is that it is user-friendly and allows the visualization of the simulation through a graphical user interface, it allows limited modelling flexibility (Jacobs et al., 2005), and it usually requires accessing a license, which can limit how the model is shared and run on different machines (Iwata and Mavris, 2013). These limitations are overcome by developing a simulation directly using a programming language, at the cost of less straightforward implementation (Brailsford et al., 2014, Iwata and Mavris, 2013, Jacobs et al., 2005). However, given the flexibility provided by the programming language-based approach, many researchers (Duffuaa and Andijani, 1999, Iwata and Mavris, 2013, Jacobs et al., 2005, Lee et al., 2003, Rosenberger et al., 2001) have chosen it for their work.

When using a programming language for developing a simulation, a distinction can be made between three approaches (Matloff, 2008):

- **Direct use of a programming language.** This approach can be hard due to the necessity of managing multiple parallel activities (Matloff, 2008). It is used in the literature for instance by Rosenberger et al. (2001) and Lee et al. (2003), who develop SimAir in C++.

- Use of a **programming language specifically developed for simulation.** Duffuaa and Andijani (1999) use SLAM language in their maintenance and operations simulation.

- Use of a **library developed for simulation**, which can be accessed using a specific programming language. Jacobs et al. (2005) for instance use DSOL, which is Java-based, while Iwata and Mavris (2013) use SimPy, which is Python-based.

Furthermore, when discussing approaches to simulation implementation, a distinction in the used *World View* can be made. This refers to the paradigm behind the simulation implementation (Matloff, 2008). Three main paradigms are used (Matloff, 2008):

- **Activity-oriented**: simulated time evolves in fixed intervals, and at each interval, a check is made if any event has happened. This approach can be quite slow, as it considers many simulation time steps where nothing actually happens.

- **Event-oriented**: the focus is on events, which are stored in event lists, and time steps are determined by the first pending event. The advantages of this approach are that run times can be short, and it is very flexible (Matloff, 2008). However, the disadvantage is that as the events list and their interaction increase, the implementation can become fragmented, and hard to understand (Derrick et al., 1989).
- **Process-oriented**: The simulation is modelled in the form of the process an entity goes through, instead of every single event of, for instance, the start and end of an activity (Matloff, 2008). This paradigm allows to easily understand the system structure (Derrick et al., 1989), but it can require a longer run time, due to the use of threads (Matloff, 2008).

The last two of these paradigms correspond to the two perspectives introduced in subsection 4.2.1, where the focus is either put on the system state and on the events that change the state (event-oriented) or on the process made of queues and activities the single entity goes through (process-oriented).

Among the named implementation approaches, some focus can be put on SimPy (Team SimPy). SimPy is an open-source, process-oriented library for DES written and accessible in Python language. Being Python-based, it uses an object-oriented logic, and it is platform-independent. Furthermore, being open-source allows for great flexibility in sharing simulation models and adapting the code to one's needs. Finally, it allows for fast Monte Carlo simulations, in addition to being able to handle simulation with a great number of entities (Iwata and Mavris, 2013).

5
Research Gap

This short chapter states the conclusions of this work. section 5.1 identifies a gap in the reviewed literature, and section 5.2 defines a research objective for the thesis project that this literature study supports.

5.1. Research Gap

In this work, an overview of network and maintenance operations planning and disruption management was given. As it can be understood from the previous chapters, a great number of works in the literature are oriented towards developing optimization models for delivering better operations and/or lower costs for airlines, being this in the planning, scheduling, or recovery phase of network and maintenance operations. However, the results provided by these models are often not tested in a stochastic simulation environment, which best represents the dynamic nature of operations. This is true even when the authors (Cadarso and Marín, 2013, Jamili, 2017, Kenan et al., 2018) claim that their model should provide results less prone to disruptions.

At the same time, it must be recognized that the literature on stochastic simulation of airline operations is quite limited. In particular, the available works focus on either network (Rosenberger et al., 2001) or maintenance (Öhman et al., 2020) operations, despite a close link between these two aspects of operations exists. Also, when the integration between these two worlds is considered, the authors focus on military operations (Iwata and Mavris, 2013), or they model in a simple manner one or both aspects of operations (Duffuaa and Andijani, 1999, Pohya et al., 2021).

Therefore, it can be seen how a gap in the literature exists in the form of a stochastic simulation model that is capable of evaluating the effects of airline plans and policies on both the network and maintenance sides of operations, which have so far been considered separately. Such a model would have a positive impact both in academia, where it would allow the evaluation of results provided by optimization models, and in the airline industry, where it could be used for evaluating flight schedules, levels of resource availability, and policies used for both planning and recovery.

5.2. Research Opportunity

Given the gap just discussed, the thesis project that this work supports will focus on the following research objective:

To develop a stochastic simulation model of airline network and maintenance operations in order to allow the evaluation of flight schedules, tail assignment policies, maintenance slots and tasks scheduling policies, operations recovery policies, and available resources.

Bibliography

J. Abara. Applying Integer Linear Programming to the Fleet Assignment Problem. *Interfaces*, 19(4):20–28, 1989. ISSN 0092-2102. doi: 10.1287/inte.19.4.20.

A. Abdelghany and K. Abdelghany. *Airline network planning and scheduling*. John Wiley & Sons, 2018.

S. Ackert. BasicsofAircraftMaintenance ProgramsforFinanciers Evaluation & Insights of Commercial Aircraft Maintenance Programs. Technical report, 2010.

S. Adeleye and C. Chung. A simulation based approach for contingency planning for aircraft turnaround operation system activities in airline hubs. *Journal of Air Transportation*, 11(1), 2006.

Y. Ageeva. *Approaches to Incorporating Robustness into Airline Scheduling L 1 1 200*. PhD thesis, 2000.

A. Ahmadi, P. Söderholm, and U. Kumar. An overview of trends in aircraft maintenance program development: Past, present, and future. Technical report, 2007.

H. Al-kaabi, A. Potter, and M. Naim. An outsourcing decision model for airlines' MRO activities. *Journal of Quality in Maintenance Engineering*, 13(3):217–227, 8 2007. ISSN 13552511. doi: 10.1108/13552510710780258.

M. A. Aloulou, M. Haouari, and F. Zeghal Mansour. Robust aircraft routing and flight retiming. *Electronic Notes in Discrete Mathematics*, 36(C):367–374, 2010. ISSN 15710653. doi: 10.1016/j.endm.2010.05.047.

M. Arikan, V. Deshpande, and M. Sohoni. Building reliable air-travel infrastructure using empirical data and stochastic models of airline networks. *Operations Research*, 61(1):45–64, 2013. ISSN 0030364X. doi: 10.1287/opre.1120.1146.

U. Arikan, S. Gürel, and M. S. Aktürk. Flight network-based approach for integrated airline recovery with cruise speed control. *Transportation Science*, 51(4):1259–1287, 2017. ISSN 15265447. doi: 10.1287/trsc.2016.0716.

B. Badkook and E. Basem Badkook. Study to Determine the Aircraft on Ground (AOG) Cost of the Boeing 777 Fleet at X Airline. *Technology, and Sciences (ASRJETS) American Scientific Research Journal for Engineering*, 25(1):51–71, 2016. ISSN 2313-4402.

M. Ball, C. Barnhart, M. Dresner, M. Hansen, K. Neels, A. Odoni, E. Peterson, L. Sherry, A. Trani, B. Zou, R. Britto, D. Fearing, P. Swaroop, N. Uman, V. Vaze, and A. Voltes. Total Delay Impact Study A Comprehensive Assessment of the Costs and Impacts of Flight Delay in the United States. Technical report, 2010.

C. Barnhart and A. Cohn. Airline schedule planning: Accomplishments and opportunities. *Manufacturing and Service Operations Management*, 6(1):3–22, 2004. ISSN 15234614. doi: 10.1287/msom.1030.0018.

C. Barnhart, N. L. Boland, L. W. Clarke, E. L. Johnson, G. L. Nemhauser, and R. G. Shenoi. Flight string models for aircraft fleeting and routing. *Transportation Science*, 32(3):208–220, 1998. ISSN 00411655. doi: 10.1287/trsc.32.3.208.

C. Barnhart, T. S. Kniker, and M. Lohatepanont. Itinerary-based airline fleet assignment. *Transportation Science*, 36(2):199–217, 2002. ISSN 00411655. doi: 10.1287/trsc.36.2.199.566.

P. Belobaba, A. Odoni, and C. Banhart. The global airline industry. *Climatic Change*, 44(1/2):27–58, 2000. doi: 10.1023/A:1005579306109.

M. Ben Ahmed, W. Ghroubi, M. Haouari, and H. D. Sherali. A hybrid optimization-simulation approach for robust weekly aircraft routing and retiming. *Transportation Research Part C: Emerging Technologies*, 84: 1–20, 2017. ISSN 0968090X. doi: 10.1016/j.trc.2017.07.010.

S. Bisaillon, J.-F. Cordeau, G. Laporte, and F. Pasin. A large neighbourhood search heuristic for the aircraft and passenger recovery problem. *4OR*, 9(2):139–157, 2010. doi: 10.1007/s10288-010-0145-5.

N. J. Boere. Air Canada Saves with Aircraft Maintenance Scheduling. *Interfaces*, 7(3):1–13, 1977. doi: 10.1287/inte.7.3.1.

S. Bouarfa, H. A. Blom, and R. Curran. Agent-Based Modeling and Simulation of Coordination by Airline Operations Control. *IEEE Transactions on Emerging Topics in Computing*, 4(1):9–20, 1 2016. ISSN 21686750. doi: 10.1109/TETC.2015.2439633.

S. Brailsford, L. Churilov, and B. Dangerfield. *Discrete-Event Simulation and System Dynamics for Management Decision Making*. Wiley, 2014.

S. C. Brailsford, T. Eldabi, M. Kunc, N. Mustafee, and A. F. Osorio. Hybrid simulation modelling in operational research: A state-of-the-art review, 2019. ISSN 03772217.

E. K. Burke, P. De Causmaecker, G. De Maere, J. Mulder, M. Paelinck, and G. Vanden Berghe. A multi-objective approach for robust airline scheduling. *Computers and Operations Research*, 37(5):822–832, 2010. ISSN 03050548. doi: 10.1016/j.cor.2009.03.026.

L. Cadarso and . Marín. Integrated robust airline schedule development. In *Procedia - Social and Behavioral Sciences*, volume 20, pages 1041–1050, 2011. doi: 10.1016/j.sbspro.2011.08.113.

L. Cadarso and . Marín. Robust passenger oriented timetable and fleet assignment integration in airline planning. *Journal of Air Transport Management*, 26:44–49, 2013. ISSN 09696997. doi: 10.1016/j.jairtraman.2012.10.004.

P. Callewaert, W. J. Verhagen, and R. Curran. Integrating maintenance work progress monitoring into aircraft maintenance planning decision support. In *Transportation Research Procedia*, volume 29, pages 58–69. Elsevier B.V., 2018. doi: 10.1016/j.trpro.2018.02.006.

L. W. Clarke, C. A. Hane, E. L. Johnson, and G. L. Nemhauser. Maintenance and crew considerations in fleet assignment. *Transportation Science*, 30(3):249–260, 1996. ISSN 00411655. doi: 10.1287/trsc.30.3.249.

J. Clausen, A. Larsen, J. Larsen, and N. J. Rezanova. Disruption management in the airline industry-Concepts, models and methods. *Computers and Operations Research*, 37(5):809–821, 2010. ISSN 03050548. doi: 10.1016/j.cor.2009.03.027.

A. Cook. European airline delay cost reference values - updated and extended values (Version 4.1). Technical report, 2015.

A. Cook and G. Tanner. The cost of passenger delay to airlines in Europe. Technical report, 2015.

Q. Deng and B. F. Santos. Lookahead approximate dynamic programming for stochastic aircraft maintenance check scheduling optimization. *European Journal of Operational Research*, 299(3):814–833, 6 2022. ISSN 03772217. doi: 10.1016/j.ejor.2021.09.019.

Q. Deng, B. F. Santos, and R. Curran. A practical dynamic programming based methodology for aircraft maintenance check scheduling optimization. *European Journal of Operational Research*, 281(2):256–273, 2020. ISSN 03772217. doi: 10.1016/j.ejor.2019.08.025.

E. J. Derrick, O. Balci, and R. E. Nance. A comparison of selected conceptual frameworks for simulation modeling. In *Proceedings of the 21st conference on Winter simulation - WSC '89*, pages 711–718, New York, New York, USA, 1989. ACM Press. ISBN 0911801588. doi: 10.1145/76738.76829.

S. O. Duffuaa and A. A. Andijani. An integrated simulation model for effective planning of maintenance operations for Saudi Arabian Airlines (SAUDIA). *Production Planning & Control*, 10(6):579–584, 1 1999. ISSN 0953-7287. doi: 10.1080/095372899232876.

EUROCONTROL. Airport collaborative decision-making. URL https://www.eurocontrol.int/concept/airport-collaborative-decision-making.

EUROCONTROL. CODA Digest All-causes delay and cancellations to air transport in Europe. Technical report, 2019.

J. Evler, E. Asadi, H. Preis, and H. Fricke. Stochastic Control of Turnarounds at HUB-Airports. In *Proceedings of the Eighth SESAR Innovation Days, Salzburg, Austria*, pages 3–7, 2018.

J. Evler, M. Lindner, H. Fricke, and M. Schultz. Integration of turnaround and aircraft recovery to mitigate delay propagation in airline networks. *Computers and Operations Research*, 138, 2022. ISSN 03050548. doi: 10.1016/j.cor.2021.105602.

H. Fricke and M. Schultz. Delay Impacts onto Turnaround Performance-Optimal Time Buffering for Minimizing Delay Propagation Data Analytics and Machine Learning View project PROFIFUEL View project. In *USA/Europe Air Traffic Management Research and Development Seminar (ATM2009)*, 2009.

C. A. Hane, C. Barnhart, E. L. Johnson, R. E. Marsten, G. L. Nemhauser, and G. Sigismondi. The fleet assignment problem: 1 solving a large-scale integer program. Technical report, 1995.

L. Hassan, B. Santos, and J. Vink. Airline disruption management: A literature review and practical challenges. *Computers & Operations Research*, 127:105137, 2021. doi: 10.1016/j.cor.2020.105137.

IATA. Air Passenger Market Analysis. Technical report, 2019a.

IATA. Industry Statistics Fact Sheet. Technical report, 2019b.

IATA. Airline Maintenance Cost Executive Commentary. Technical report, 2020.

C. Iwata and D. Mavris. Object-oriented discrete event simulation modeling environment for aerospace vehicle maintenance and logistics process. In *Procedia Computer Science*, volume 16, pages 187–196. Elsevier B.V., 2013. doi: 10.1016/j.procs.2013.01.020.

P. H. Jacobs, A. Verbraeck, and J. B. Mulder. Flight scheduling at KLM. In *Proceedings - Winter Simulation Conference*, volume 2005, pages 299–306, 2005. doi: 10.1109/WSC.2005.1574264.

M. Jahangirian, T. Eldabi, A. Naseer, L. K. Stergioulas, and T. Young. Simulation in manufacturing and business: A review. *European Journal of Operational Research*, 203(1):1–13, 5 2010. ISSN 03772217. doi: 10.1016/j.ejor.2009.06.004.

A. Jamili. A robust mathematical model and heuristic algorithms for integrated aircraft routing and scheduling, with consideration of fleet assignment problem. *Journal of Air Transport Management*, 58:21–30, 2017. ISSN 09696997. doi: 10.1016/j.jairtraman.2016.08.008.

N. Kenan, A. Jebali, and A. Diabat. The integrated aircraft routing problem with optional flights and delay considerations. *Transportation Research Part E: Logistics and Transportation Review*, 118:355–375, 2018. ISSN 13665545. doi: 10.1016/j.tre.2018.08.002.

H. A. Kinnison and T. Siddiqui. *Aviation maintenance management*, volume McGraw Hill. 2013.

C. Lagos, F. Delgado, and M. A. Klapp. Dynamic optimization for airline maintenance operations. *Transportation Science*, 54(4):998–1015, 2020. ISSN 15265447. doi: 10.1287/TRSC.2020.0984.

S. Lan, J. P. Clarke, and C. Barnhart. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science*, 40(1):15–28, 2006. ISSN 15265447. doi: 10.1287/trsc.1050.0134.

L. H. Lee, Huei Chuen Huang, Chulung Lee, Ek Peng Chew, W. Jaruphongsa, Yean Yik Yong, Zhe Liang, Chun How Leong, Yen Ping Tan, K. Namburi, E. Johnson, and J. Banks. Discrete event simulation model for airline operations: SIMAIR. In *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693)*, pages 1656–1662. IEEE, 2003. doi: 10.1109/WSC.2003.1261616.

A. Levin. Scheduling and Fleet Routing Models for Transportation Systems. *Transportation Science*, 5(3): 232–255, 1971. doi: 10.1287/trsc.5.3.232.

M. Liehr, A. Größler, M. Klein, and P. M. Milling. Cycles in the sky: Understanding and managing business cycles in the airline market. *System Dynamics Review*, 17(4):311–332, 12 2001. ISSN 08837066. doi: 10.1002/SDR.226.

M. Lohatepanont and C. Barnhart. Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment. *Transportation Science*, 38(1):19–32, 2004. ISSN 00411655. doi: 10.1287/trsc.1030.0026.

R. Maidstone. Discrete Event Simulation, System Dynamics and Agent Based Simulation: Discussion and Comparison. Technical report, 2012.

L. Marla, B. Vaaben, and C. Barnhart. Integrated Disruption Management and Flight Planning to Trade Off Delays and Fuel Burn. *Transportation Science*, 51(1):88–111, 2017. doi: 10.1287/trsc.2015.0609.

N. Matloff. Introduction to Discrete-Event Simulation and the SimPy Language. Technical report, 2008.

M. M. Mota, G. Boosten, N. De Bock, E. Jimenez, and J. P. de Sousa. Simulation-based turnaround evaluation for Lelystad Airport. *Journal of Air Transport Management*, 64:21–32, 2017. ISSN 09696997. doi: 10.1016/j.jairtraman.2017.06.021.

M. Öhman, M. Hiltunen, . K. Virtanen, and J. Holmström. Frontlog scheduling in aircraft line maintenance: From explorative solution design to theoretical insight into buffer management. 2020. doi: 10.1002/joom.1108.

B. Oreschko, M. Schultz, and H. Fricke. Turnaround Prediction with Stochastic Process Times and Airport Specific Delay Pattern Airspace Optimization and Management View project Knowledge extraction from large-scale air traffic data View project. In *International Conference on Research in Airport Transportation (ICRAT)*, 2012.

M. Palpant, M. Boudia, C.-a. Robelin, S. Gabteni, and F. Laburthe Amadeus SAS. ROADEF 2009 Challenge: Disruption Management for Commercial Aviation. Technical report, 2009.

N. Papadakos. Integrated airline scheduling. *Computers and Operations Research*, 36(1):176–195, 1 2009. ISSN 03050548. doi: 10.1016/j.cor.2007.08.002.

N. Papakostas, P. Papachatzakis, V. Xanthakis, D. Mourtzis, and G. Chryssolouris. An approach to operational aircraft maintenance planning. *Decision Support Systems*, 48(4):604–612, 2010. ISSN 01679236. doi: 10.1016/j.dss.2009.11.010.

A. A. Pohya, J. Wehrspohn, R. Meissner, and K. Wicke. A modular framework for the life cycle based evaluation of aircraft technologies, maintenance strategies, and operational decision making using discrete event simulation. *Aerospace*, 8(7), 7 2021. ISSN 22264310. doi: 10.3390/AEROSPACE8070187.

A. Regattieri, A. Giazzi, M. Gamberi, and R. Gamberini. An innovative method to optimize the maintenance policies in an aircraft: General framework and case study. *Journal of Air Transport Management*, 44-45:8–20, 2015. ISSN 09696997. doi: 10.1016/j.jairtraman.2015.02.001.

J. Rosenberger, A. Schaefer, D. Goldsman, E. Johnson, A. Kleywegt, and G. Nemhauser. SimAir: a stochastic model of airline operations. In *2000 Winter Simulation Conference Proceedings (Cat. No.00CH37165)*, volume 2, pages 1118–1122. IEEE, 2000. doi: 10.1109/WSC.2000.899074.

J. M. Rosenberger, A. J. Schaefer, D. Goldsman, E. L. Johnson, A. J. Kleywegt, and G. L. Nemhauser. A Stochastic Model of Airline Operations. Technical report, Georgia Institute of Technology, Atlanta, 10 2001.

J. M. Rosenberger, A. J. Schaefer, D. Goldsman, E. L. Johnson, A. J. Kleywegt, and G. L. Nemhauser. A stochastic model of airline operations. *Transportation Science*, 36(4):357–377, 2002. ISSN 00411655. doi: 10.1287/trsc.36.4.357.551.

J. M. Rosenberger, E. L. Johnson, and G. L. Nemhauser. Rerouting aircraft for airline recovery. *Transportation Science*, 37(4):408–421, 2003. ISSN 00411655. doi: 10.1287/trsc.37.4.408.23271.

J. M. Rosenberger, E. L. Johnson, and G. L. Nemhauser. A robust fleet-assignment model with hub isolation and short cycles. *Transportation Science*, 38(3):357–368, 2004. ISSN 00411655. doi: 10.1287/trsc.1030.0038.

S. Ross. The Discrete Event Simulation Approach. In *Simulation*, pages 111–134. Elsevier, 2013. doi: 10.1016/b978-0-12-415825-2.00007-3.

M. Sachon and E. Paté-Cornell. Delays and safety in airline maintenance. *Reliability Engineering & System Safety*, 67(3):301–309, 3 2000. ISSN 09518320. doi: 10.1016/S0951-8320(99)00062-9.

D. T. Sanchez, B. Boyac, and K. G. Zografos. An optimisation framework for airline fleet maintenance scheduling with tail assignment considerations. *Transportation Research Part B: Methodological*, 133:142–164, 3 2020. ISSN 01912615. doi: 10.1016/j.trb.2019.12.008.

A. Sarac, R. Batta, and C. M. Rump. A branch-and-price approach for operational aircraft maintenance routing. *European Journal of Operational Research*, 175(3):1850–1869, 2006. ISSN 03772217. doi: 10.1016/j.ejor.2004.10.033.

M. Schmidt. A review of aircraft turnaround operations and simulations. *Progress in Aerospace Sciences*, 92: 25–38, 2017. ISSN 03760421. doi: 10.1016/j.paerosci.2017.05.002.

S. Shao, H. D. Sherali, and M. Haouari. A novel model and decomposition approach for the integrated airline fleet assignment, aircraft routing, and crew pairing problem. *Transportation Science*, 51(1):233–249, 2017. ISSN 15265447. doi: 10.1287/trsc.2015.0623.

S. Shaukat, M. Katscher, C. L. Wu, F. Delgado, and H. Larrain. Aircraft line maintenance scheduling and optimisation. *Journal of Air Transport Management*, 89, 2020. ISSN 09696997. doi: 10.1016/j.jairtraman.2020.101914.

H. D. Sherali, E. K. Bish, and X. Zhu. Airline fleet assignment concepts, models, and algorithms. *European Journal of Operational Research*, 172(1):1–30, 2006. ISSN 03772217. doi: 10.1016/j.ejor.2005.01.056.

H. D. Sherali, K. H. Bae, and M. Haouari. An integrated approach for airline flight selection and timing, fleet assignment, and aircraft routing. *Transportation Science*, 47(4):455–476, 2013. ISSN 15265447. doi: 10.1287/trsc.2013.0460.

C. Sriram and A. Haghani. An optimization model for aircraft maintenance scheduling and re-assignment. *Transportation Research Part A: Policy and Practice*, 37(1):29–48, 2003. ISSN 09658564. doi: 10.1016/S0965-8564(02)00004-6.

Team SimPy. SimPy: Discrete event simulation for Python.

D. a. Teodorovi and S. Guberini. Optimal dispatching strategy on an airline network after a schedule perturbation. Technical report, 1984.

J. Van Den Bergh, P. De Bruecker, J. Beliën, and J. Peeters. Aircraft maintenance operations: state of the art. Technical report, 2013.

J. Vink, B. F. Santos, W. J. Verhagen, I. Medeiros, and R. Filho. Dynamic aircraft recovery problem - An operational decision support framework. *Computers and Operations Research*, 117, 2020. ISSN 03050548. doi: 10.1016/j.cor.2020.104892.

H. W. M. Vos, B. F. Santos, and T. Omondi. Aircraft schedule recovery problem - A dynamic modeling framework for daily operations. In *Transportation Research Procedia*, volume 10, pages 931–940. Elsevier, 2015. doi: 10.1016/j.trpro.2015.09.047.

M. Witteman, Q. Deng, and B. F. Santos. A bin packing approach to solve the aircraft maintenance task allocation problem. *European Journal of Operational Research*, 294(1):365–376, 10 2021. ISSN 03772217. doi: 10.1016/j.ejor.2021.01.027.

J. T. Wong and S. C. Tsai. A survival model for flight delay propagation. *Journal of Air Transport Management*, 23:5–11, 2012. ISSN 09696997. doi: 10.1016/j.jairtraman.2012.01.016.

C.-L. Wu. Inherent delays and operational reliability of airline schedules. *Journal of Air Transport Management*, 11(4):273–282, 7 2005. ISSN 09696997. doi: 10.1016/j.jairtraman.2005.01.005.

C.-L. Wu and R. E. Caves. Modelling of aircraft rotation in a multiple airport environment. *Transportation Research Part E: Logistics and Transportation Review*, 38(3-4):265–277, 2002a. ISSN 13665545. doi: 10.1016/S1366-5545(02)00010-8.

C.-L. Wu and R. E. Caves. Towards the optimisation of the schedule reliability of aircraft rotations. *Journal of Air Transport Management*, 8(6):419–426, 11 2002b. ISSN 09696997. doi: 10.1016/S0969-6997(02)00042-X.

C. L. Wu and R. E. Caves. Modelling and optimization of aircraft turnaround time at an airport. *Transportation Planning and Technology*, 27(1):47–66, 2004a. ISSN 03081060. doi: 10.1080/0308106042000184454.

C. L. Wu and R. E. Caves. Modelling and simulation of aircraft turnaround operations at airports. *Transportation Planning and Technology*, 27(1):25–46, 2004b. ISSN 03081060. doi: 10.1080/0308106042000184445.

C. L. Wu and K. Law. Modelling the delay propagation effects of multiple resource connections in an airline network using a Bayesian network model. *Transportation Research Part E: Logistics and Transportation Review*, 122:62–77, 2019. ISSN 13665545. doi: 10.1016/j.tre.2018.11.004.