A greedy approach to the minimisation of deviations of the dynamic vehicle routing problem with electric taxiing systems

Master of Science Thesis

B.J.L. Tindemans

TU Delft

A greedy approach to the minimisation of deviations of the dynamic vehicle routing problem with electric taxiing systems

Master of Science Thesis

by

B.J.L. Tindemans

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday October 7^{th}, 2021.

Student number:	4466225	
Project duration:	December 2020 – October 2021	
Thesis committee:	Prof. Dr. M. Snellen	TU Delft, Chair
	Dr. M.A. Mitici	TU Delft, Supervisor
	PhD candidate. M. Zoutendijk	TU Delft, Co-supervisor
	Dr. A. Bombellini	TU Delft, External committee member

An electronic version of this thesis is available at http://repository.tudelft.nl/.

The image on the front page is taken from Beelen [4].

Acknowledgements

With this thesis report I conclude my master degree in Aerospace Engineering at Delft University of Technology. For the past six years, I have studied here with great enthusiasm and I am delighted to end this period with a project involving two of my most interesting subjects: optimization problems and sustainability. As TaxiBot is quite a novel concept, lots of technical and operational challenges arise for all parties involved. I hope, that this thesis contributes to the upcoming research in this field in order to eventually solve the sustainability matter within aviation.

I had the pleasure to work on my project with the help of SAS, developers of the TaxiBot ®. Therefore, I would like to express my gratitude for their help and support to steer my project into a usefull direction. Their experience and knowledge on the TaxiBot system itself but also the operations taking place at Schiphol Airport validated my work, resulting in a more realistic and all-inclusive model.

I would like to sincerely thank both of my supervisors Mihaela Mitici and Mike Zoutendijk, with whom I have had fruitful discussions. Mihaela in particular for letting me critically evaluate my own work by giving constructive feedback, and Mike specifically for the practical discussions and brainstorm sessions, as well as frequently taking the time to review my written work.

Furthermore, I would like to thank my friends for their help in the reviewing process and support throughout the project. Their comments and critical questions lead to a more mature and cohesive report.

Bas J.L. Tindemans
Delft, September 2021

Contents

List of Figures

List of Tables

Nomenclature

List of Abbreviations

A	Arrival
AAS	Amsterdam Airport Schiphol
AC	Aircraft
API	Application Programming Interface
ATA	Actual Time of Arrival
ATC	Air Traffic Control
ATD	Actual Time of Departure
C	Certified
CO	Carbon Oxide
CO_2	Carbon Dioxide
D	Departure
ECDT	Engine Cool-down Time
ESUT	Engine Start-up Time
ET	Electric Taxiing
ETS	Electric Taxiing Systems
ETV	Electric Taxiing Vehicle
FIR	Flight Information Region
FSA	Fleet Scheduling Assignment
GA	Genetic Algoritms
GAP	Gate Assignment Problem
GVRSP	Greedy Vehicle Routing and Scheduling Problem
IAI	Israel Aerospace Industries
KPI	Key Performance Indicator
MILP	Mixed-Integer Linear Programming

MLG	Main Landing Gear
NB	Narrow Body
NLG	Nose Landing Gear
NN	Neural Networks
PDF	Probability Density Function
SAS	Smart Airport Systems
STA	Scheduled Time of Arrival
STD	Scheduled Time of Departure
TB	TaxiBot
TOT	Turn Around Time
VRP	Vehicle Routing Problem
WB	Wide Body

List of Symbols

AC	Set of aircraft
ac_k	Aircraft k in the set
$AC_{current}$	Set of aircraft currently present in the nodal network
AC_{future}	Set of aircraft which will be present in the nodal network in the near future, i.e. next five minutes
ac_{total}	Total number of aircraft [-]
$AC_{watchlist}$	Set of aircraft which will be present in the nodal network in the far future, i.e. next half an hour
c_v	Coefficient of variation [-]
E	Set of edges
e_{total}	Total number of edges [-]
F	Flight schedule
F'	Updated flight schedule
G	Nodal network
N	Set of nodes
n_{total}	Total number of nodes [-]
$P_{no\ TB}$	Penalty for certified aircraft when taxiing without a TaxiBot [-]
P_{peak}	Penalty for certified aircraft during peak hours [-]
$P_{switching}$	Penalty for certified aircraft when switching to a different TaxiBot than scheduled [-]
P_{wait}	Penalty for certified aircraft per minute of waiting $[min^{-1}]$
S	Strategic schedule
S'	Tactical schedule
t	Time [s]
$T_{max\ wait}$	Maximum waiting time threshold for certified aircraft before they will taxi without a TaxiBot [min]
$T_{response}$	Time before the actual emerge time of aircraft to start up TaxiBots to move towards the starting node, set to five minutes [min]
TB	Set of TaxiBots
tb	TaxiBot
TB_{pool}	Pool of TaxiBots [-]

tb_{total}	Total number of TaxiBots [-]
v_e	Speed on edge e [m/s]
Δt	Timestep between two time stamps [s]
$\mu(penalty_{sum})$	Mean value of the sum of penalties [-]
$\sigma(penalty_{sum})$	Standard deviation of the sum of penalties [-]

Introduction

Aviation plays a big role in the current society. Even though it brings a lot of good to the world, unfortunately, it brings quite some negative consequences with it as well. 2.5% of the total global CO2 emissions are due to aviation [19]. As emissions from aviation continue to increase, solutions need to be found to counteract this. Historically, solutions were mostly sought in the airborne phase of a flight, however among onground operations are areas with huge potential as well. Multiple solutions have been found to electrify the taxiing of aircraft from and to the runway, one of which being the TaxiBot solution by Smart Airport Systems (SAS). However, as this solution is quite a novel concept, lots of technical and operational challenges arise. The routing and scheduling of all aircraft on the runways and taxiways is quite a complex puzzle to solve, especially with the addition of TaxiBots, increasing the total number of vehicles driving around. This research specifically focuses on the external electric taxiing solution, TaxiBot, and the implementation of it at an airport. Developments of such ET solutions are underway, however as they are a novel research direction, practical implications on the implementation of it arise. This research tries to fill that gap by strengthening the research in the field of TaxiBot routing and scheduling.

Next to that, as these ground phases take place at an airport, the surrounding environment directly benefits from this. Airports which seek to improve their sustainability benefit from such ETS. Royal Schiphol Group, the owner of Amsterdam Airport Schiphol (AAS) has set its mission to become emission free on the airport by 2030[1]. All ground bound vehicles ought to be driving on electricity or hydrogen, which includes the towing vehicles used. One of the ways Schiphol tries to reach this goal is to team up with SAS, one of the developers of the TaxiBot. Tests conducted in the first half of 2020 on the feasibility of TaxiBots on Schiphol resulted in the Proof of Concept of TaxiBot at Schiphol, however further research is necessary.

Important to note is that TaxiBot is used in many settings in this report. TaxiBot is the brand of the electric taxiing solution by SAS, however TaxiBot is also used as generic term for such a type of solution.

This thesis report is organized as follows. In Part I, the scientific paper is presented. Part II contains the relevant Literature Study that supports the research. Finally, in Part III, some additional results are presented.

[1] https://www.schiphol.nl/nl/schiphol-als-buur/pagina/emissievrij-in-2030/, accessed on 24-12-2020

I

Scientific Paper

A greedy approach to the minimisation of deviations of the dynamic vehicle routing problem with electric taxiing systems

Author: B.J.L. Tindemans*,
Supervisors: M. Mitici†, M. Zoutendijk‡

Faculty of Aerospace Engineering, Delft University of Technology, HS 2926 Delft, The Netherlands

Abstract

In order to reduce aircraft emissions during on-ground operations, electric taxiing systems (ETS) have been intensively researched to take over or assist in part of the taxiing phase of a flight. One of these ETS is the TaxiBot, deployed by Smart Airport Systems (SAS). While a number of papers have been researching this vehicle routing problem (VRP) in order to minimise costs, fuel consumption or another metric, most research uses deterministic input data. However, sudden changes are inevitable, disrupting the resulting schedule. In this paper, we propose both a strategic model and a tactical model for airport surface movement, taking into account stochastic delays to the flight schedule. The subsequent tactical schedule can be produced with only little deviations from the strategic schedule. The tactical model can be generated in the order of seconds, making it useful for real-life traffic management decision support systems. We found that the tactical schedule does not worsen remarkably. Still 47.9% of the flights can be towed by a TaxiBot, which was 48.5% in the strategic schedule. Different case studies have been performed in order to determine the effect of e.g. the size of the flight schedule and the number of TaxiBots used, on the aircraft taxiing coverage and TaxiBot efficiency. With this, both busy and calm days, now and in the near future are assessed and the optimum number of TaxiBots necessary can be determined. An upper limit is reached at an asymptote starting from 34 TaxiBots, while a lower limit is dependent on a trade-off between flight coverage and spare TaxiBot capacity.

Keywords: Electric Taxiing Systems, TaxiBot, VRP, FSA, Greedy Approach, Dynamic & Stochastic, Disruption Management

1 Introduction

Most research on the reduction of pollutants emitted by aircraft tends to focus on the airborne phase of the flight, however, sustainable improvements in aircraft efficiency often are found in the range of only small percentages. Moreover, aircraft are not continuously in the air, but rather also use their engines on the ground for a part of the mission. During these phases of the flight, the on-ground taxiing phase more specifically, larger improvements can be gained. During this phase, the engines of the aircraft are used, which is not optimal. *"During idle mode, an engines performance is less efficient due to the low combustor temperature. This induces higher fuel consumption, and emissions of hydrocarbon and CO,"* according to [Ithnan et al., 2013, p. 2]. As engines are not designed for this stage, relatively high fuel consumption and emissions occur which should be reduced as much as possible. Next to that, on the ground, many more options are present to make sure an aircraft is moved to its desired position. One such viable option is via electric taxiing systems (ETS). By using electrical energy for the taxiing phase instead of jet fuel, a big step can be taken towards a more sustainable solution. One of these ETS is the TaxiBot, designed by Smart Airport Systems (SAS), which is the only operational option at the moment of writing [Hospodka, 2014b].

As TaxiBots are a novel concept in airport operations, lots of technical and operational challenges arise when such systems are implemented at an airport. TaxiBots are yet another type of vehicles introduced at the apron, which should all be managed in a certain way. As aircraft and TaxiBots will make use of the runways and taxiways of an airport, ATC has control over these vehicles to make sure safety is maintained. Careful routing will need to be considered and strategic scheduling is needed for that. Therefore, a routing and scheduling solution needs to be found to make sure conflicts and collisions are avoided while making sure taxi time ([Roling

*MSc Student, Air Transport and Operations, Faculty of Aerospace Engineering, Delft University of Technology
†Supervisor, Assistant Professor
‡Supervisor, Doctoral Researcher

et al., 2015]), and all its other linked parameters, such as fuel consumption, are minimised ([Roling, 2009]).

The objective of the strategic schedule hence is the minimisation of total taxi time. However, sudden changes will always be present. A planning is ought to be made robust, however limitations are always present. Therefore, when for example an arriving aircraft is delayed, the old schedule might not be feasible anymore and a new, tactical (i.e. continuously monitoring) schedule needs to be developed. As an airport usually has to make instantaneous decisions, such a new schedule has to be generated within a reasonable time. Only then, airport operations can continue with no or limited disruptions. The objective of this research is to include disruptions into this deterministic problem, making it more stochastic and thus realistic.

The objective of the tactical schedule is focused on *Disruption Management*, reacting on deviations from the original flight schedule [Dorndorf et al., 2007]. Aircraft can be earlier or later than planned, resulting in an infeasible vehicle routing and towing assignment schedule. Dorndorf et al. present two objectives for this context. One is the maximisation of profits, which is equivalent to the minimisation of costs, and the other one is to return to the original plan as soon as possible. The first one will be used as general objective in both the strategic and tactical model. The second one can be motivated by the implicit costs emerging when changing the schedule and the customer dissatisfaction resulting from the possibly longer waiting times. [Dorndorf et al., 2007] This second objective will be implemented by use of penalties in the tactical model.

This paper is structured as follows. An overview of the relevant literature is given in section 2. Then, we give a description of the problem in section 3, followed by a description of the case studies in section 4. In section 5 we present the results of the different case studies. Lastly, section 6 presents our conclusions and recommendations for future work.

2 Literature Review

Different types of problems can be distinguished in the field of optimisation. Two relevant problem types used in airport operations research are the Vehicle Routing Problem (VRP) and the Fleet Scheduling Assignment (FSA). Extensive research has been done already in both fields, which has been summarised by [Atkin et al., 2010] for VRP and by [Dorndorf, 2005] for FSA. These problems have been extended over time by research in different subfields. [Schiffer and Walther, 2017] tried to expand to the different objective functions that can be used for the VRP, [Yan-Du et al., 2014] and [Smeltink et al., 2004] tried to minimise the computation time and [Sirigu, 2017], [Sirigu et al., 2018]. FSA problems mainly focus on assigning the aircraft to gates, however if the fields of VRP and FSA are combined and the set of vehicles is not limited to aircraft but also other ground vehicles, FSA can be reframed as the scheduling of aircraft to ground vehicles.

A number of different sustainable taxiing solutions have been researched, of which we consider the electrical solutions. In general, a split can be made between on-board and on-ground electric taxiing systems (ETS), next to single-engine taxiing. Each of these systems have been thoroughly compared (e.g. [Fordham et al., 2016], [Guo et al., 2014], [Hospodka, 2014a], [Ithnan et al., 2013], [Lukic et al., 2018], [Lukic et al., 2019] and [Pan et al., 2017]). As concluded by [Lukic et al., 2018], TaxiBot was at the moment of writing the only certified system and already used commercially. TaxiBots are electrified towing tractors able to taxi aircraft over a long distance from gate to runway or vice versa. A TaxiBot operates differently than conventional towing trucks and thus makes it possible to tow aircraft over longer distances [Hospodka, 2014b]. The aircraft fleet needs to be assigned to TaxiBots and routed over the airport surface. [Guillaume, 2018] developed a routing and scheduling model in order to optimise the number of automated guided towing vehicles for aircraft taxiing with respect to taxiing costs. Later on, [van Baaren, 2019] developed a similar VRP model which focused on optimising the number of ETS with respect to fuel consumption, emissions and energy usage. Van Baaren defined three towing vehicle designs, closely resembling the TaxiBot, which in turn have been used by [Kroese, 2021]. Kroese combined the VRP with a fleet scheduling assignment (FSA) resulting in a schedule of a fleet of TaxiBots, taking into account charging of the batteries.

The models used in these four studies are mixed-integer linear programming (MILP) models, which search for a global optimum. Other types of models used in literature are Genetic Algorithms (GA) [Jiang et al., 2013] or via Neural Networks (NN) [Sirigu et al., 2018], [Gotteland et al., 2001]. GA do not guarantee to give the optimal solution, moreover an approximation of the solution is not always guaranteed. However, computation times tend to be distinctly shorter than MILP or NN models. Long computation times for MILP and NN bring problems to airports as they are usually seeking to make decisions within a few minutes.

Previous models mainly made use of an MILP strategy in search of a global optimum, however as we developed a dynamic scheduling model we decided to develop a local optimisation model. In real life airport operations, a global optimum is never reached due to the lack of information about the future and moreover the fact that devations are expected. Therefore, a greedy approach is used which tries to find a local optimum. Furthermore, using an MILP model forces you to set all parameters, such as aircraft delays beforehand. This can be partially solved by using time windows, providing you with multiple moments to add new information. Examples are a runway configuration change or changes in the crew availability. However, as new flight data can arrive at any moment in time and not only in time windows, a local optimisation model is better suited.

Aforementioned VRP models try to minimise either costs, total taxiing time or emissions. The research focuses on deterministic airport operations. However, in real life operations sudden changes happen continuously. Aircraft arrive earlier or later than expected which will result in the defined schedule being disrupted. [Evertse and Visser, 2017] developed an MILP model in which the departure times are not deterministic, but rather allowed to deviate slightly from the scheduled time of departure. Slightly, in the sense that departure order and gate allocation is not changed. A number of papers have been written about dynamic VRPs ([Pillac et al., 2013], [Psaraftis et al., 2016], [Tas et al., 2013]). [Pillac et al., 2013] defined a taxonomy for VRP based on information evolution and quality as can be seen in Figure 1.

		Information quality	
		Deterministic input	Stochastic input
Information evolution	Input known beforehand	Static and deterministic	Static and stochastic
	Input changes over time	Dynamic and deterministic	Dynamic and stochastic

Figure 1: Taxonomy of VRP by information evolution and information quality [Pillac et al., 2013].

Aforementioned MILP models can be classified as static and deterministic. The work from [Evertse and Visser, 2017] can be seen as static and stochastic. In [Dorndorf et al., 2007], the authors focused on disruption management in flight gate scheduling. Adding disruptions to the schedule is recommended by [Yan-Du et al., 2014] as well, along with [Zaninotto et al., 2019] who recommended implementing a tactical planning module, adjusting input parameters, such as taxi routes and schedules, dynamically.
Our research will focus on dynamic and stochastic models. The dynamic aspect refers to the information evolution in which flight arrival/departure times will only be known at a certain moment in time and thus not all input data will be known beforehand. The stochastic aspect refers to the generation of the flight arrival/departure times. The input flight schedule will be generated from probability density functions.

Disruption management in the airline industry has been thoroughly investigated. Many sources for disruption can be thought of, such as severe weather conditions, corrective maintenance or gate or aircraft breaking down [Dorndorf et al., 2007]. Solutions found by [Su et al., 2021] which are used for flight scheduling, but can be used for ETS scheduling as well, are delaying, cancelling, swapping, or reallocating flights or using reserve equipment/crew. [Pei et al., 2021] constructed a quantitative scoring system for recovery suggestions. Six domains, part of a decision tree, were scored based on interviews, questionnaires, and operational data. These six domains are the following: flight density, aircraft properties, reasons for delay, route unavailability reasons, passenger properties, and delay severity. The decision tree then performed appropriate recovery actions, such as changing gates or crew, when the threshold values were exceeded, while complying with the order of priority of the six domains.
In this paper we combine aforementioned three fields of research: VRP, ETS and disruption management.

3 Methodology

The model consists of two separate parts, one to generate the strategic schedule and a second to generate a tactical schedule. The strategic schedule can be made some time before the day of operations, e.g. a couple of months beforehand, while the tactical schedule is continuously run on the day of operations itself. A functional flow diagram of the model set is shown in Figure 2. The block structure of three on the left represents the generation of the strategic schedule used as reference on the day of operation. The newly estimated times of arrival and departure, and the strategic schedule, are inputs to the tactical schedule, which is a continous monitoring process. A decision tree in the tactical model is used whenever an aircraft is delayed or early to determine the optimal solution, resulting in an updated planning. Finally, this will result in a tactical routing schedule, which can be used throughout the rest of the day of operation. A following aircraft will probably have a different time or arrival/departure as well, resulting in the need of a new tactical schedule for the remaining time of the day. Both models will be explained later on.

We consider the directed nodal network G with the set of nodes N and the set of edges E, so that $G = (N; E)$ and we consider two sets of vehicles; a set of aircraft AC and a set of TaxiBots TB. To this, time is added with the set of timestamps t. The aforementioned vehicles in the sets AC and TB can be present in the model in three different combinations. Table 1 gives an overview of these vehicle combinations. A description of all model sets and variables is the following:

Model sets

- $G = (N; E)$ Nodal network with nodes $n \in N$ and edges $e \in E$.
- $n \in N = \{1, ..., n_{total}\}$ Set of nodes in the nodal network, represented by either n or m.
- $e \in E = \{...n \cdot m...\}$ Set of edges in the nodal network, in which node n $\neq$ m and with speed v_e.
- $ac \in AC = \{1, ..., ac_{total}\}$ Set of aircraft arriving and departing. $ac = 0$ means no aircraft is attached in the aircraft-vehicle combination.
- $tb \in TB = \{1, ..., tb_{total}\}$ Set of TaxiBots. $tb = 0$ means no TaxiBot is attached in the aircraft-vehicle combination.
- $AC_{current} \subset AC$ Set of aircraft currently present in the nodal network
- $AC_{future} \subset AC$ Set of aircraft which will be present in the nodal network in the near future, i.e. next five minutes
- $AC_{watchlist} \subset AC$ Set of aircraft which will be present in the nodal network in the further future, i.e. next half an hour
- $TB_{pool} \subset TB$ Set of TaxiBots available to be called for a towing task
- $t \in \{0, ..., t_{total}\}$ Set of time stamps, with a Δt as a time step of 10 seconds, for one day of operations.

Model variables

- F Flight schedule input , as defined in section 3.3.
- F' Updated flight schedule with the actual times of arrival/departure, as defined in section 3.3.
- S Strategic schedule, output from the strategic model.
- S' Tactical schedule, output from the tactical model.
- P_{wait} Penalty for certified aircraft per minute of waiting $[min^{-1}]$.
- $P_{switching}$ Penalty for certified aircraft when switching to a different TaxiBot than scheduled [-].
- $P_{no\ TB}$ Penalty for certified aircraft when taxiing without a TaxiBot [-].
- P_{peak} Penalty for certified aircraft during peak hours [-].
- Δt Time step between two time stamps t_i and t_{i+1}.
- $T_{response}$ Time between the actual emerge time of aircraft and starting up TaxiBots to move towards the starting node, set to five minutes.
- $T_{max\ wait}$ Maximum waiting time threshold for certified aircraft before they will taxi without a TaxiBot.

Table 1: Vehicle combinations.

AC	**TB**	**State**
$ac = \{1...ac_{total}\}$	$tb = \{1...tb_{total}\}$	TaxiBoting
$ac = \{1...ac_{total}\}$	$tb = \{0\}$	Taxiing
$ac = \{0\}$	$tb = \{1...tb_{total}\}$	Traversing
$ac = \{0\}$	$tb = \{0\}$	Not Applicable

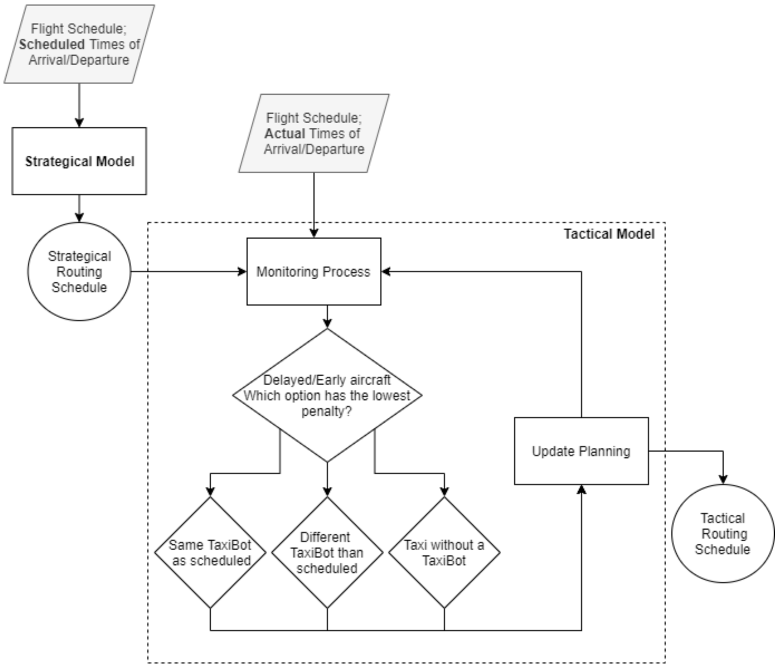

Figure 2: Functional flow diagram of the integration of strategic and tactical models. Figure adapted from [Dorndorf et al., 2007].

The Greedy Vehicle Routing and Scheduling Problem (GVRSP) is defined on a nodal network, where nodes can be a gate, runway or taxiway node and edges can be taxiways or service roads. Runway and gate nodes can connect to both taxiways and service roads, connecting the two different node structures. Each edge $(n, m) \in E$ has a travel time dependent on the distance and maximum speed. The number of vehicles AC and TB are known beforehand. The number of AC is based on the flight schedule, the number of TB is fixed within case studies, as described in section 4.

The goal of the strategic GVRSP is to locally minimise the total travel times, for all vehicle combinations. Next to that, it is aimed to let the aircraft that are certified to be towed by TaxiBot, make use of the TaxiBots as much as possible. The first goal is achieved by letting the vehicles move from begin to end node via the shortest path. The two shortest path algorithms that work for this type of weighed nodal network, having different distances and speeds, searching for the shortest path between one starting point and one ending point are Dijkstra's algorithm and the Bellman-Ford algorithm. According to [Abusalim et al., 2020] Dijkstra's is best suited for nodal networks such as the one considered here due to lower computation time and higher efficiency when solving the shortest path problem.

The goal of the tactical GVRSP is to locally minimise the deviations of the tactical model with respect to the strategic model, while also adhering to the goal of the strategic model. A greedy approach is used as a local heuristic. In real-life operations, the vehicle routing and scheduling problem cannot be globally optimised as changes in the flight schedule happen constantly and thus actual daily flight schedules cannot be fully known beforehand, due to the dynamic and stochastic aspect of the tactical model. Hence, the tactical model can only optimise locally by using the information that is known at that moment in time. As mentioned before, time is split into timestamps in order to go over the course of time and updates of the model are made at every time stamp. The model can be run for a full period of 24 hours, but can also be split in time windows if only specific hours are to be investigated. These can be set manually in the model. Both strategic and tactical model will be explained in the following subsections. Each of the parameters mentioned are set and do not differ between the models or any of the case studies explained later on.

3.1 Strategic Model

First the algorithm of the strategic model will be explained by means of a pseudocode, which can be found in Algorithm 1. The time starts at $t = 0$ (algorithm code line 2), which corresponds with 02:00. This time is chosen, rather than starting at 00:00 as at this hour the activities at the airport are at the lowest. Following this fact, it is assumed that this day of operations is independent from the previous day. Two sets of aircraft are continuously updated, a future set of aircraft AC_{future} and a current set of aircraft $AC_{current}$, which are initally set empty (line 3).

AC_{future} contains all aircraft that will emerge in the coming five minutes, named response time $T_{response}$. Emerging refers both to arrival at the runway exit and departure from the gate, after which the taxiing procedure takes place. Following this, the respective end nodes of both procedures are at the gate and ending at the runway entrance respectively. This default value of five minutes for the response time is taken as most of the TaxiBots called upon will be present within these five minutes before arrival or departure. According to 200 simulation runs, the mean traversing time lies around five minutes, and the median lies around 3 minutes. This default value is thus balanced between being on time and not waiting too long at a gate/runway.

$AC_{current}$ contains all aircraft that are currently moving around at the airport. At every time stamp, the flight schedule F is checked (line 9) and if an aircraft is scheduled to emerge in the near future, it is added to AC_{future} and the nearest TaxiBot will be requested (lines 10 to 13). Aircraft can either be certified to be towed by TaxiBots or not. If an aircraft is not certified, it will taxi from its starting node to its end node, i.e. from gate to runway or vice versa following the normal sequence of operations. If the aircraft is certified, the free TaxiBot closest by will be called upon and will drive towards the starting node five minutes before the actual emerging time. A maximum is set in order to make sure certified aircraft will not wait too long. If after the maximum time $T_{max\ wait}$ of five minutes, still no TaxiBot has been assigned, the certified aircraft will travel without one.

Once the aircraft emerges, it will be added to $AC_{current}$ and deleted from AC_{future} and again corresponding actions will follow, such as determining the shortest path in the Nodal Network G (lines 14 to 18). Here, it is assumed that runways only have one entrance/exit, the runway configuration is known upfront and runway sequencing is not taken into account. As soon as the TaxiBot is present, coupling will take place and the aircraft-TaxiBot combination will taxi to its end node (lines 19 to 22). The assumption is made here that (un)coupling can take place at the gate or runway which have sufficient space for these operations. As soon as the aircraft reaches its end node, the aircraft is deleted from $AC_{current}$ and the TaxiBot is free to be used again, if applicable (lines 23 to 29). While free, the TaxiBot will drive to one of the three waiting nodes, numbered 5, 108 & 110, dependent on which one is closest by. These three nodes are based on the strategic parking locations as defined by [Kroese, 2021] and are assumed to have sufficient space to fit all TaxiBots waiting.

Each time stamp, all ac in $AC_{current}$ and all tb in TB are moved forward with a distance equal to the edge speed multiplied with Δt (lines 6 & 8). The speeds of the vehicles ar assumed to be equal to the maximum velocity and acceleration or deceleration is not taken into account. If the end of the edge is reached, the vehicle moves towards the next edge, until the end of the last edge sequence is reached. Collision and conflict avoidance takes places at each time stamp (line 7). Only aircraft are subject to this as it is assumed that TaxiBots can maintain safe distance between each other. If all actions have taken place, the time is set forward with Δt, specified to be 10 seconds, based on [Roling et al., 2015] (line 32). This value limits computation time while still obtaining sufficient accuracy in the model. In the end, all aircraft and TaxiBot properties throughout the day will be returned, as well as the strategic routing schedule S.

3.2 Tactical Model

Secondly, the algorithm of the tactical model will be explained by means of a pseudocode, which can be found in Algorithm 2. The input for this model is the same input as for the strategic model, however now the new flight schedule F' is updated at every time stamp. Next to that, the strategic schedule S is used as input for the model (line 1).

The model structure is the same as before, however now there are three sets of aircraft. Next to the $AC_{current}$ and AC_{future}, use is made of a watch list $AC_{watchlist}$, all set empty at first (line 3). The very first time an aircraft comes into sight can be at two moments (line 9). The first one is at a set amount of time before the scheduled time of arrival/departure, which happens if the aircraft is either on time or delayed. The second one is when ATC finds out that the aircraft is earlier than scheduled. 30 minutes before arrival and 5 minutes before departure, the exact time is known and fixed. 30 minutes is taken for arrival, roughly based on the time it takes to land after entering Amsterdam Flight Information Region (FIR). As all the different operations happening just before aircraft departure, i.e. baggage loading, passengers taking place etc, are difficult to predict, only a 5-minute future outlook is used, which is the lower limit used by [Evertse and Visser, 2017]. In other words, only 5 minutes before the actual departure, this exact time is known.

If the aircraft is earlier or just in time, it is directly added to AC_{future}, whereafter the aforementioned corresponding actions follow (lines 13 to 20). One of these is the penalty decision tree chosing which TaxiBot is best to use (line 17), which will explained in section 3.2.1. If the aircraft is delayed, it is not known yet at what time exactly it will emerge and thus it is placed in $AC_{watchlist}$ (lines 10 to 12). At every following time stamp, $AC_{watchlist}$ is checked and as soon as the actual time of emergence is known, it will be added to AC_{future} and removed from $AC_{watchlist}$ again. Four examples of this process can be seen in Table 2. Again, if the aircraft actually emerges, it will be added to $AC_{current}$ and deleted from AC_{future} and the corresponding actions follow again (lines 21 to 29).

Table 2: Four examples of the process using aircraft sets.

Scenario	**Earliest update time**	**Information known at earliest update time**	**Starting set**	**Second Set**	**Third set**
Arriving aircraft (early) Scheduled time: 14:00 Actual time: 13:57	13:27	Aircraft is earlier Arrival time: 13:57	Future set 13:27	Current set 13:57	-
Arriving aircraft (delayed) Scheduled time: 14:00 Actual time 14:01	13:30	Aircraft is delayed Departing time: unknown	Watch list 13:30	Future set 13:31	Current set 14:01
Departing aircraft (on-time) Scheduled time: 14:00 Actual time: 14:00	13:55	Aircraft is exactly on time Departing time: 14:00	Future set 13:55	Current set 14:00	-
Departing aircraft (delayed) Scheduled time: 14:00 Actual time: 14:12	13:55	Aircraft is delayed Departing time: unknown	Watch list 13:55	Future set 14:07	Current set 14:12

3.2.1 Penalties

Penalties are given in the tactical model only. As soon as the aircraft is added to AC_{future}, the situation will be compared with the strategic model. The decision flow works as follows. The TaxiBot data used in the strategic model is retrieved. Then, three possible options will follow. The aircraft either travels with the same TaxiBot as was scheduled (Option 1), the aircraft will travel with a different TaxiBot, one that is closest by, (Option 2) or the aircraft will taxi without a TaxiBot (Option 3). The option that introduces the smallest penalty will be chosen by the model. If these options have the same penalty, option 1 is preferred over option 2 and option 2 is preferred over option 3. For each of these three options, a number of suboptions arises, of which only one is possible. The options are summed up below, followed by a visualisation of the options and their associated penalties in Figure 3. Different penalties apply for each of the three options, which are both fixed penalties and variable penalties based on the time such an action takes. Next to that, in order to make the decision tree a bit more complex, the flight density has been added to the scoring system. Whenever the airport experiences a busy period, currently set to a minimum of ten aircraft on the grid, penalties are scored double (P_{peak}). With this, busy and calm days can be modelled in a more realistic way, as penalties weigh heavier for busier days.

Choose the same TaxiBot as was scheduled (Option 1)

- The same TaxiBot as was scheduled is directly available (penalty is zero)
- The same TaxiBot first finishes its task (either towing or traversing and towing) and then travels to pick up the waiting aircraft (penalty for every minute the certified aircraft has to wait)

Choose another TaxiBot (Option 2)

- Another TaxiBot is directly available (penalty for switching and penalty for every minute the certified aircraft has to wait)
- The same TaxiBot as was scheduled is currently on his way for a taxiing task, but this task is cancelled and this TaxiBot goes to the waiting aircraft that needs a new TaxiBot instead. Another free TaxiBot is redirected to the other aircraft that had been waiting already for the initially assigned TB (penalty for waiting and penalty for every minute one of the certified aircraft has to wait)
- All TaxiBots are currently in use, so wait until one becomes available and couple with that one (penalty for switching and penalty for every minute the certified aircraft has to wait)

Taxi without a TaxiBot (Option 3)

- If all TaxiBots are currently in use and none will become available in the coming time within the threshold ($T_{max\ wait}$), the aircraft will taxi without a TaxiBot (penalty for taxiing without a TB)

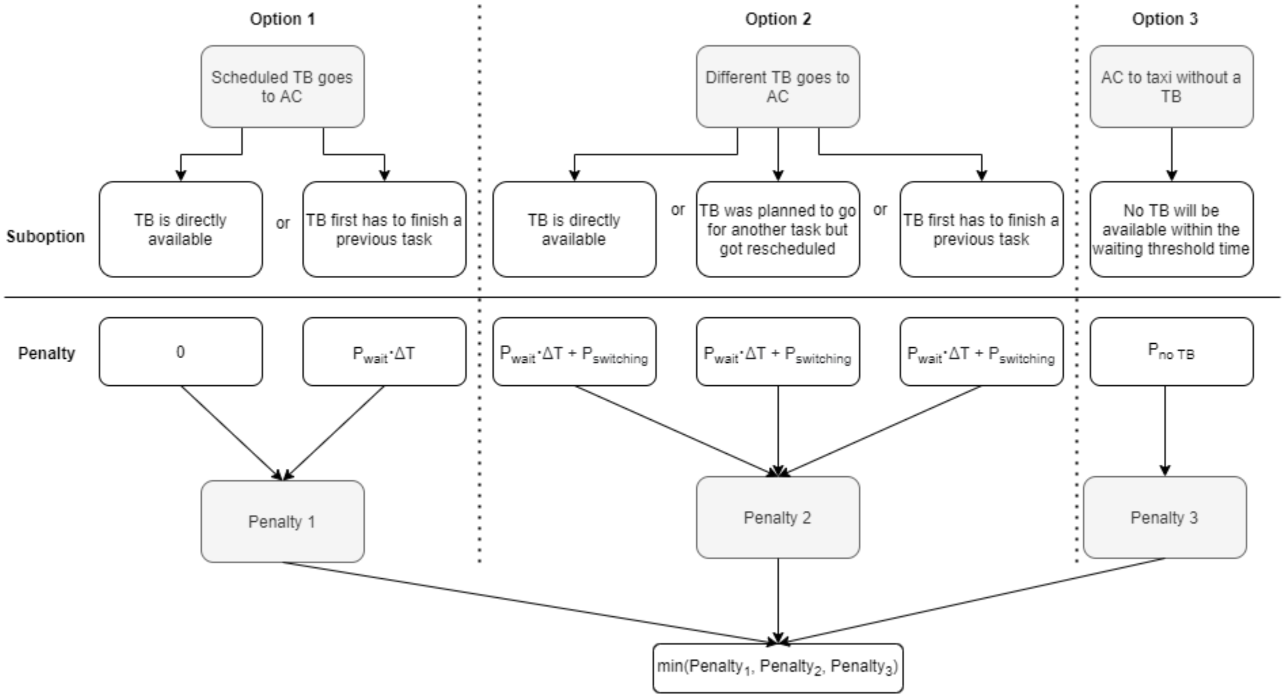

Figure 3: Visualisation of the options with its possible scenarios and the corresponding penalties.

The calculation of the penalties goes as follows. Suppose ac_k has a scheduled time of arrival of 12:00, hence as explained before, at 11:30 ac_k comes into play for the first time. Suppose ac_k has an actual time of arrival of 12:15, meaning that it will first be placed on $AC_{watchlist}$, until 11:45, when it will be placed on AC_{future}. 5 minutes before arrival, the tactical model looks at the strategic schedule S and finds that the scheduled TaxiBot was tb_u. First option 1 is assessed. Suppose tb_u is currently busy with another ac_j in $AC_{current}$ and this operation will approximately take 6 minutes. As this calculation takes place 5 minutes before arrival, these 5 minutes spare are subtracted from these 6 minutes, meaning that the aircraft probably has to wait for 1 minute before tb_u will be present. The penalty for option 1 then becomes $1min \cdot 10min^{-1}(P_{wait}) = 10$. Suppose plenty of other TaxiBots are available, and the one closest by, tb_v only needs 2 minutes to be at the correct pick up node. As this is less than 5 minutes, no penalty will be allocated for waiting, however the penalty for switching does apply, resulting in $0min \cdot 10min^{-1}(P_{wait}) + 50(P_{switching}) = 50$. As the aircraft does not have to wait 5 minutes before any TaxiBot will be present, option 3 is not applicable. As option 1 is lower than option 2 ($10 < 50$), option 1 will be chosen, even though ac_k has to wait for one minute after arrival and tb_u will be chosen. A visualisation of this decision can be seen in Figure 4

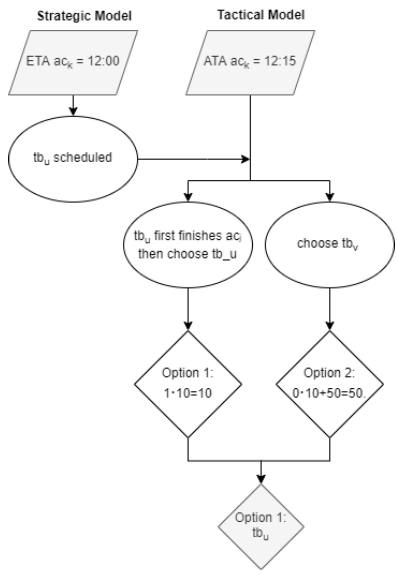

Figure 4: Visualisation of the decision proces for one certified aircraft to be taxied by a TaxiBot.

Algorithm 1 Simplified pseudocode of greedy vehicle routing problem: strategic model

```
Input Nodal network G with nodes N and edges E, Flight schedule F with aircraft AC
Set t=0
Set AC_current = ∅, AC_future = ∅, current set of TB = 20
while  t is in time window  do
   Close crossing nodes in case of runway closure
   Move each ac in AC_current on edge n, m in shortest path with speed v_e
   For each ac in AC_current, ensure conflict and collision avoidance
   Move each tb in the current set of tb on edge (n, m) with speed v_e
   Check for new emerging ac in the near future in F
   if ac will emerge (arrival/departure) within T_response  then
      Add ac to AC_future
      Let tb already go to the starting node (runway/gate) if possible
   end if
   if ac arrives/departs then
      Add ac to AC_current
      Delete ac from AC_future
      Determine Dijkstra's shortest path from starting node to end node
   end if
   if newly emerged ac is certified and TB_pool>0 then
      Couple with tb
      TB_pool -= 1
   end if
   if ac in AC_current reaches the end node (gate/runway) then
      Delete ac from AC_current
      if ac is certified then
         Uncouple ac and tb
         TB_pool += 1
      end if
   end if
   Move to next time window if end of current time window is reached
   print  new ac and tb positions
   t += Δt
end while
Return Aircraft and TaxiBot properties
Output Strategic routing schedule S
```

Algorithm 2 Simplified pseudo code of greedy vehicle routing problem: tactical model

Input Nodal network G with nodes N and edges E, Updated flight schedule F' with aircraft AC, Strategic schedule S
Set t=0
Set $AC_{current} = \emptyset$, $AC_{future} = \emptyset$, $AC_{watchlist} = \emptyset$, current set of TB = 20
while t is in time window **do**
  Close crossing nodes in case of runway closure
  Move each ac in $AC_{current}$ on edge (n, m) with speed v_e
  For each ac in $AC_{current}$, ensure conflict and collision avoidance
  Move each tb in the current set of tb on edge n, m with speed v_e
  Check for the first time if ac will emerge on time or will be delayed in F'
  if ac is delayed **then**
    Add ac to $AC_{watchlist}$
  end if
  if ac will emerge (arrival/departure) within $T_{response}$ **then**
    Add ac to AC_{future}
    Delete ac from $AC_{watchlist}$
    For each ac in AC_{future}, determine the different scenarios based on the strategic schedule S
    Determine the penalties per scenario and choose the one with the lowest penalty
    Let tb already traverse to the starting node (runway/gate) if possible
    $penalty_{sum}$ += penalty
  end if
  if ac emerges (arrival/departure) **then**
    Add ac to the $AC_{current}$
    Delete ac from AC_{future}
    Determine Dijkstra's shortest path from starting node to end node
  end if
  if newly emerged ac is certified and TB_{pool}>0 **then**
    Couple with tb
    TB_{pool} -= 1
  end if
  if ac in $AC_{current}$ reaches the end node (gate/runway) **then**
    Delete ac from $AC_{current}$
    if ac is certified **then**
      Uncouple ac and tb
      TB_{pool} += 1
    end if
  end if
  Move to next time window if end of current time window is reached
  print new ac and tb positions
  t += Δt
end while
Return Aircraft and TaxiBot properties
Output Strategic routing schedule S', $penalty_{sum}$, $penalty_{max}$, Penalty information P

3.3 Model Input

We consider a number of sets of input data. The nodal network, flight schedule, runway configuration, set of certified aircraft, penalties and operational (un)coupling times.

Nodal Network

The nodal network is adapted from [Guillaume, 2018] and represents Amsterdam Airport Schiphol (AAS). The fact that feasibility tests with the TaxiBot have been performed here during Q1 to Q3 2020 make this a reasonable choice [Schiphol, 2020]. Figure 5 shows the nodal network G with its nodes N and edges E and Figure 6 visualises the nodal network by overlaying it over AAS.

Each node contains the following attributes: its relative position, the nodes it is connected to and its node function. The latter refers to either service roads or taxiway roads and the specific function of end nodes, such as gates and runway entrances/exits. Light blue nodes 0 to 11 represent gates, nodes 94 to 99 represent the runway entrances/exits. Figure 7 visualises the gates associated with each of the gate nodes. Edges have the following attributes: the maximum driving speeds and the lengths. Blue edges represent service roads, only accessible for traversing TaxiBots, grey edges represent taxiway roads, accessible for aircraft and TaxiBots towing aircraft. The thick grey lines represent the six runways. Note that runway 18R-36L is not fully shown due to formatting purposes. Edge velocities, as adapted from [Guillaume, 2018], are the average speeds driven by aircraft, as analysed by [Roling et al., 2015]. These velocities were derived from real track data provided by AAS and ATC the Netherlands. Speeds around the apron are the maximum allowed speeds. As can be seen, no edges connect to runway 04-22 as this runway is not used in our model because only a minimal number of commercial flights (1.2% of total yearly flights) make use of this runway. A total of 110 nodes are used to construct the nodal network. Nodes are strategically placed on turns and to limit the edge length. This is as to maximise certain edge lengths of taxiway roads as conflict and collision avoidance is based on this edge length. Lastly, a restriction is set on runway crossing whenever this runway is in use.

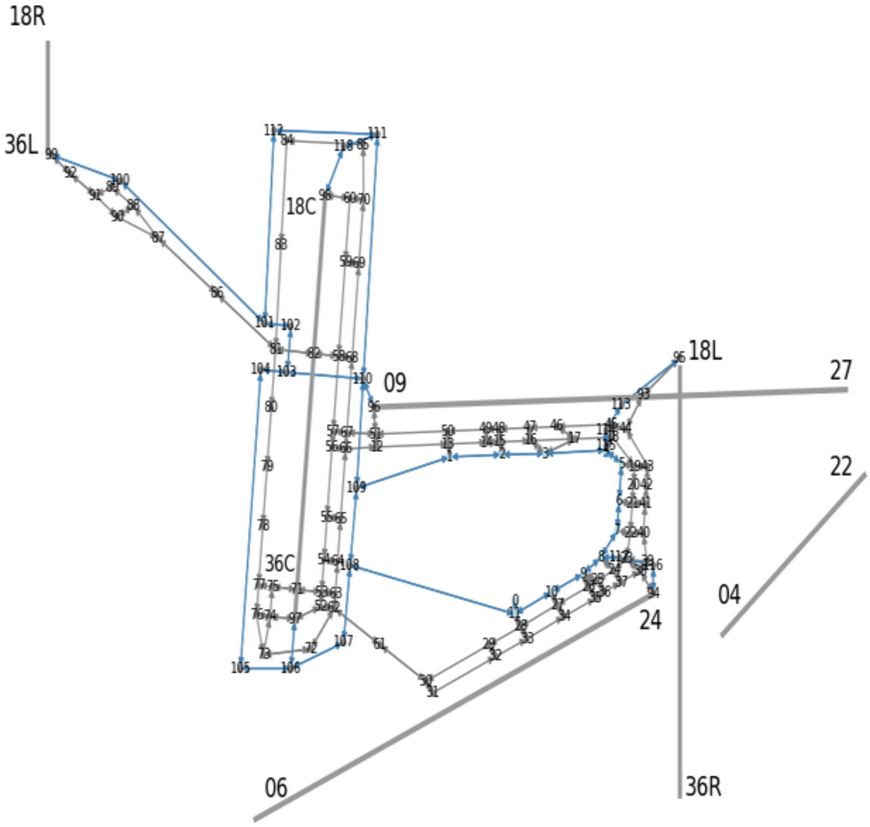

Figure 5: Nodal Network representing AAS, adapted from [Guillaume, 2018].

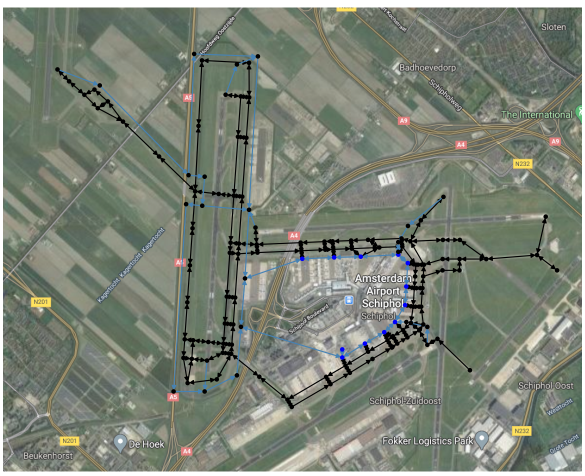

Figure 6: Overlay of the nodal network on a satellite image of Amsterdam Airport Schiphol.

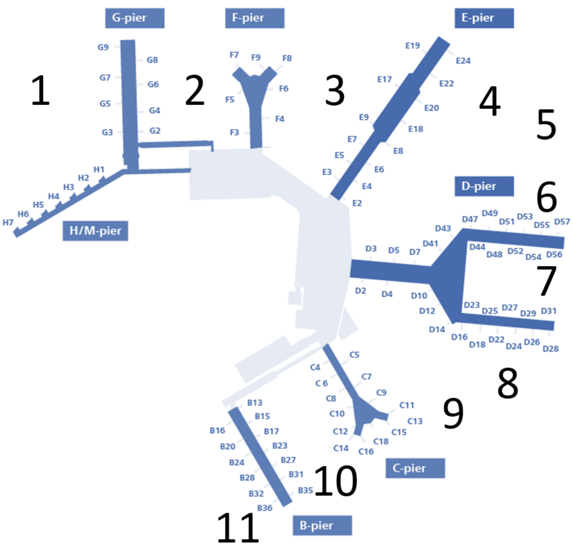

Figure 7: Gates associated to each of the gate nodes, adapted from [Kroese, 2021].

Conflict and collision avoidance is only applicable to taxiways and is based on a minimum separation of 60 meters [Kroese, 2021], [Gotteland et al., 2001]. Some edges in the nodal network have a length smaller than 60 meters, however as these locations represent a complex intersection, aircraft will already arrive with a slower speed than usual, justifying this 60 meters. At the airport apron, even lower taxiing speeds are common, hence [Salihu et al., 2021] suggests separation distances of 15m to 30m. As this separation will be ensured during normal operations by careful driving operations, conflict and collision avoidance is not required in this area.

Flight Schedule
The second input set is the flight schedule, which is taken from an online data collecting website[1]. A sample set is given in Table 3. The aircraft type is used to determine if the flight is certified to be towed by a TaxiBot. Aircraft that are certified are all types of Airbus A320 and Boeing B737 [Hospodka, 2014b], [TLD, 2019]. The TaxiBot deploying company SAS has the ambition to have all types of Airbus A220, Boeing B757, the current Embraer E-jet family and the COMAC family certified within the foreseeable future. Other types of data used are the gate the flight is scheduled to and the flight status, being arrival (A) or departure (D). Lastly, two sets of times are given. These arrival/departure times are the Scheduled Time of Arrival (STA) and Actual Time of Arrival (ATA) or Scheduled Time of Departure (STD) and Actual Time of Departure (ATD). For future reference, only STA and ATA will be used, which refers to both. The STA is the time as it is scheduled in advance, while the ATA is the actual time of arrival/departure determined after the action has taken place. The STA is used for the strategic schedule, while the ATA is used for the tactical schedule. If multiple aircraft are scheduled to use the same runway at one moment in time, it is assumed that appropiate runway sequencing takes place.

Table 3: Sample flight schedule input data.

ID	Aircraft type	Gate	Flight nr.	STA	ATA	Flight Status	Certified
0	BOEING 737-800 WINGLETS	C7	KL590	06:50	06:49	A	C
1	AIRBUS A350-900	D42	HV 6871	06:50	07:32	D	-

Runway Configuration
The third set of input is the runway configuration, taken from the website *DutchPlaneSpotters*[1] as well. This configuration is used for both the strategic and tactical schedule. When the strategic schedule is planned, this runway configuration is not known yet, as this is defined on the day itself, however it is assumed to be known. The reason for this is to compare the strategic with the tactical schedule with only one variable subject to change. When keeping the runway configuration the same while changing the arrival and departure times, a qualitative comparison can be made. The preference list as defined by the *MER commission* sets the combination 18R (and 18C in case two runways are used at the same time) for landings and 24 (and 18L in case two runways are used at the same time) for departures as the second favourable option, justifying this runway configuration used [Gordijn, 2016].

Penalty Parameters
Next to that, the penalty factors used can be found in Table 4. The base value for penalties P_{wait} is set arbitrarily to 10. The penalty for waiting is defined as the penalty base value times the number of minutes the aircraft has to wait. If a TaxiBot other than previously scheduled will taxi the aircraft, the penalty base value is multiplied by 5, resulting in a penalty of 50, in case of an aircraft being earlier than scheduled. If the aircraft decides to taxi without a TaxiBot, the penalty will be 100. As delayed aircraft have a higher priority to finish their route as soon as possible, the penalties for switching and taxiing for delayed aircraft are considered to be less important. For delayed aircraft, we prioritise arriving at their respective end node as fast as possible, resulting in lower penalty multiplication factors, respectively 4x and 8x. In practice, for early aircraft this means that the threshold value for switching of TaxiBots lies at 50/10 = 5 minutes, which means that after 5 minutes, the option of switching is most favourable. Following that same logic, the threshold for taxiing without a TaxiBot lies at 10 minutes. The penalty multiplication factors for delayed aircraft are relatively lower, meaning that the threshold values are lowered as well. As a default value, 5x is chosen as this is matching with the 5 minutes future outlook that is used for departing aircraft as the amount of time before the actual departure time is fixed. The default value of 10x for taxiing without a TaxiBot is set to be twice the previous value. For delayed aircraft these values are reduced with one minute (4x), and then doubled respectively (8x). As these penalty parameters are set rather arbitrarily, a sensitivity analysis on these parameters will follow in section 5.3.2.

[1] `https://schiphol.dutchplanespotters.nl`

Table 4: Overview of penalty factors used in the model.

Penalty - Early Aircraft	**Value**	**Penalty - Delayed Aircraft**	**Value**
Waiting on a TaxiBot - P_{wait} (min^{-1})	10	Waiting on a TaxiBot P_{wait} (min^{-1})	10
Switching of TaxiBot - $P_{switching}$ (5x)	50	Switching of TaxiBot - $P_{switching}$ (4x)	40
Taxiing without a TaxiBot - $P_{no\ TB}$ (10x)	100	Taxiing without a TaxiBot - $P_{no\ TB}$ (8x)	80

Operational (Un)coupling Times

As TaxiBots have a different method of operations than the usual towing trucks, the operation times significantly differ. For example, as TaxiBots will drive on the taxiway roads when coupled to the aircraft, ATC clearance is necessary beforehand. The different tasks for (un)coupling of towing trucks at airport operations, are summed to get to a total operation time, as can be seen in Table 5. Some of these task lengths are taken from [Kroese, 2021], however the TaxiBot deploying company SAS has the ambitions to minimise the task lengths as specified in the table.

A distinction can be made between certified and non-certified aircraft and arriving and departing aircraft. For each of these four situations, operational time is needed before and after the taxiing has taken place. No tasks take place for arriving non-certified aircraft after they have taxied to their respective end node, hence 0 seconds. The upper part of Table 5 represents the current way of operation when using towing trucks, the lower part represents the way of operations when using TaxiBots. Engine cool down time (ECDT) and engine start-up time (ESUT) are both set to be 5 minutes (300s) [Kumar et al., 2008] [Deonandan and Balakrishnan, 2010]. As engines must be sufficiently warmed up prior to departure and cooled down after arrival, this must be taken into account when calculating the time needed for the taxi operations. Therefore, for aircraft taxiing without a TaxiBot, take-off is postponed until the engines have warmed up sufficiently. However, in all other cases, this ECDT and ESUT is masked by other operations. Note however, that taxi-in must take at least five minutes in order to make sure the engines have cooled down sufficiently. For arriving aircraft, the ECDT can take place either during taxiing or when coupled to a TaxiBot. For departing aircraft, ESUT can take place while the TaxiBot taxies to the runway. The same assumption regarding a minimum taxi-out time applies here as well. From the simulation runs, it can be seen that the number of times the taxi-in and out times were shorter than five minutes is in the range of 1%, verifying this assumption. Lastly, pre-flight time, used to go over the pre-flight checklist, is assumed to be 45 seconds, however it is masked as well by general aircraft operating activities.

As can be seen from Table 5, taxi-in operations are four times longer for the aircraft-TaxiBot combination (240s with respect to 60s). On the other hand, taxi-out takes half as long for the aircraft-TaxiBot combination (405s with respect to 720s).

Table 5: Four different scenarios for coupling and uncoupling times taking place before and after taxiing, as defined for certified and non-certified aircraft.

	Non-Certified Aircraft			
	Arrival		**Departure**	
	Before	After	Before	After
	ATC clearance before taxi in (60s)	ECDT (masked)	Pre-flight time (masked) Pushback loading (45s) ATC clearance before pushback (60s) Pushback time (90s) Pushback unloading (105s) ESUT (300s) ATC clearance before taxi out (60s)	ATC clearance for line up (60s)
Total [s]:	60	0	660	60
	60		720	
	Certified Aircraft			
	Arrival		**Departure**	
	Before	After	Before	After
	Coupling time (45s) ATC clearance before taxi in (60s) ECDT (masked)	Decoupling time (135s)	Pre-flight time (masked) Coupling time (45s) ATC clearance before pushback (60s) Pushback time (90s) Command transfer (15s) ATC clearance before taxi out (0s)	ESUT (masked) Decoupling time (135s) ATC clearance for line up (60s)
Total [s]:	105	135	210	195
	240		405	

Assumptions

- As one of the six runways, Oostbaan 04/22, is not used often throughout the year, this runway is omitted from the nodal network and any applicable arrivals or departures here are deleted from the flight schedule. This assumption is justified as only 6058 of the total 497303 flights (1.2%) used this runway in 2019

[Bewoners-Aanspreekpunt-Schiphol, 2020].

- It is assumed that runways only have 1 runway entrance/exit at the beginning/end of the track.
- If more than 1 runway is used for arrival or departure, flights are alternately scheduled, following that multiple flights can arrive/depart at one moment in time.
- Runway sequencing and the subsequent minimum time separation between two arriving/departing aircraft is not taken into account.
- The runway configuration is assumed to be known for the strategic schedule.
- Coupling and uncoupling at the runway takes place on the runway node. It is assumed that such a holding node has sufficient space for aircraft and TaxiBot to (un)couple, while other aircraft and/or TaxiBots might pass meanwhile.
- The vehicles are assumed to be driving either at the maximum speed at that specific edge or are at standstill. Acceleration or deceleration hence is not taken into account.
- The number of TaxiBots is variable for different scenarios.
- We assume that TaxiBots have sufficient power to operate a full day and do not need charging throughout the day. It is assumed that charging takes place during lean hours and is not modelled.
- There will be sufficient operating personnel in order to use all TaxiBots throughout a day of operations.
- All cockpit crew are assumed to be familiar with the TaxiBot operations and are allowed to operate them.
- All other crew of operating parties are trained to operate all TaxiBot procedures, and no delays or inconsistencies will occur due to incompetence.
- No conflict and collision avoidance is necessary for TaxiBots that are driving on the service roads. Next to that, these TaxiBots can pass each other on these roads as well.
- If a runway is in use, runway crossing is not allowed.
- Conflict and collision avoidance for aircraft only takes places at taxiways and not at gates and runways. A minimum separation of 60m is guaranteed, which is not based on the size of both aircraft.
- At edges where service roads cross taxiway roads, it is assumed that appropriate right of way order is kept between aircraft and TaxiBots.
- The time span of one simulation takes exactly one day. These simulations are assumed to be independent of previous or future days.
- Once the actual time of arrival/departure is known a set time before, this time does not change anymore.
- As most recent flight schedules are affected by the Covid-19 crisis, input data from 2019 will be used, as this data is deemed more relevant than data from 2020.
- It is assumed that no cancellations or any other abnormal activities such as go-arounds take place.
- Delays can be maximum negative or positive 3 hours with respect to the scheduled time of arrival/departure. Only 56 of the 10335 flights assessed in the two weeks around the busy day go over this limit due to exceptional reasons. In order to limit the problem to realistic scenarios, this maximum is set.

4 Description of the Case Studies

First a case study regarding different scenarios follows in section 4.1, after which a case study regarding a sensitivity analysis follows in section 4.2.

4.1 Scenarios

We have solved the GVRSP for different scenarios, as can be seen in Table 6. The number of TaxiBots necessary is dependent on the number of aircraft certified to be taxied, which in turn is dependent on the number of aircraft arriving and departing and the list of aircraft types certified to be towed. The number of aircraft arriving and departing is not constant throughout a year but rather fluctuates. Summer periods tend to be busier and winter periods are calmer.

Table 6: Different scenario case studies performed.

ID	Scenarios	Nr of Aircraft	Of which certified	Nr of TaxiBot	Nr of Runs
#1	Busy day + Current set of certified aircraft + random A/D times (base scenario)	1427	757 (53.0%)	30	100
#2	Calm day + Current set of certified aircraft + random A/D times	954	499 (52.3%)	30	100
#3	Busy day + Future set of certified aircraft + random A/D times	1427	1138 (79.7%)	45	100
#4	Calm day + Future set of certified aircraft + random A/D times	954	721 (75.6%)	45	100

Busy vs. Calm Scenario

Four scenarios have been studied, two with a busy schedule (schedules 1 & 3) and two with a calm schedule (scenarios 2 & 4). Busy days are interesting as they showcase the maximum capacity and thus the maximum number of TaxiBots needed. A calm day will show the minimum number of TaxiBots needed at the airport. Hence, the number of TaxiBots needed fluctuates as well, resulting in a set of spare TaxiBots in the winter, while having a scarcity in the summer. The number of TaxiBots used in the schedule is set to 30 for the base scenario. The selection of this parameter is based on experimental results, as will be explained in section 4.2. The choice represents a scenario in which 90% of the runs the number of TaxiBots was sufficient to perform all tasks without having aircraft to wait for an available TaxiBot. The effect of changing this variable will be explained in the parametric study in section 5.3. This number is fixed for both busy and calm days in order to limit the number of variables. In real-life, this number of TaxiBots available is specified beforehand as not only variable but also fixed costs, such as personnel costs, are associated with the operability of a TaxiBot. Hence, one might expect to have only 2/3 of the TaxiBot fleet available on calm days.

The flight schedule as explained before in section 3.3 for the busy day used in the case studies is taken from August 8th 2019, which was one of the busiest days in that year, accompanied by other busy days in the weeks before and after. The latter is important as the probability density function (PDF) used in the model to set the ATA and ATD is derived from the actual delays of all flights in these two weeks around the assessed day. The reason why a PDF is used to randomly determine delays, instead of using multiple days as input, is to change only one variable at the time while keeping the rest fixed, i.e. only delays can vary, while the runway configuration, the number of aircraft etc stay the same. The flight schedule from the calm day used in the case studies is taken from January 26th 2019, which was the calmest day of the year. The percentage of certified aircraft is approximately the same for busy and calm days.

Current vs. Future Set of Certified Aircraft Types

As discussed before, the set of aircraft types that are certified to be towed will be extended in the near future. Hence, the case studies have been split up in a time period representing the current scenario (scenarios 1 & 2) and one representing the near future (scenarios 3 & 4). The percentage of certified aircraft is higher for the near future scenarios, however note that the current flight schedule is used still. It might be the case that in the near future, a higher number of certified aircraft will be used by airlines as the list of certified types in the near future contains newly designed aircraft as well. Therefore, this percentage might be higher and thus even more TaxiBots might be necessary as well. The number of TaxiBots used in these two scenarios are set to $30 \cdot 1.5 = 45$, as one and a half times more certified aircraft are scheduled to arrive and depart.

The ATA and ATD taken from the flight schedules represent the actual flight times on the day of operations, however this represents only one run for the tactical model. As arrival and departure delays are probabilistic, a set of multiple tactical model runs need to be performed in order to draw informed conclusions. This is based on a proper estimate of the distribution of the total sum of penalties, taken as the main output variable to base on. Based on the stabilisation of the coefficient of variation, as suggested by [Lorscheid et al., 2012], the number of runs necessary is 100.

In order to generate a different set of ATA and ATD for each run, we drew random delays for each flight from a lognormal probability density function. This distribution fits the data set best, based on literature. This distribution is selected as well by [Novianingsih and Hadianti, 2014] and [Lan et al., 2006], who both statistically tested gamma, lognormal, Rayleigh and Weibull distributions. Delay data is taken from all arrival and departure delays from the 14 days around the chosen busy and calm days (sample size 10000). These two weeks, one before and one after the actual chosen day, are similar in terms of number of arrivals and departures, composition of aircraft types and timewise distribution, resulting in a large and reliable data set. Sources for delays vary and thus different distributions can be drawn for various selections. In this paper, we focus on a split between arrivals and departures, which show a great difference in both mean and standard deviation. Figure 8 shows lognormal probability density functions for arrival and departure delays on a busy day. As noted before, the functions are cut at negative and positive delays of 180 minutes or three hours in order to keep the problem realistic. As can be seen, the PDF for arrival delays tend to shape more like a normal distribution, while the PDF for departure delays almost fully encompasses only positive delays. Arriving flights have the possibility to fly faster than scheduled or have a deliberately longer flight time scheduled, resulting in early arrivals, however for departing aircraft it is practically very difficult to leave before the scheduled time.

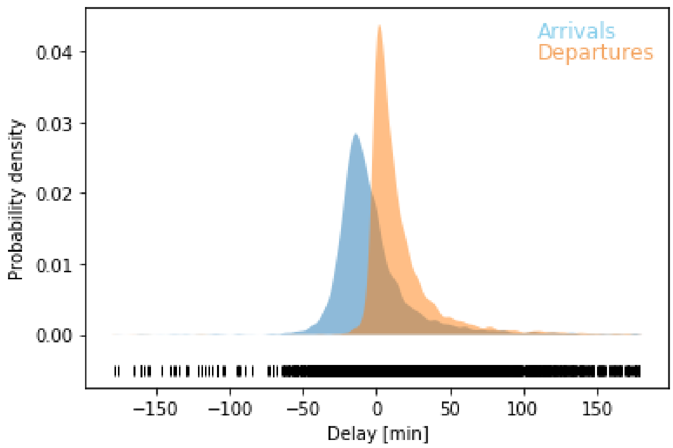

Figure 8: Lognormal probability density functions for arrival and departure delays on a busy day.

4.2 Sensitivity Analysis

A sensitivity analysis is used to evaluate the sensitivity of two model parameters. These parameters are the number of TaxiBots used and the penalty multiplication factors. The parameters and their upper and lower boundaries are shown in Table 7. As mentioned before, the number of TaxiBots necessary is dependent on multiple factors and results in different costs. For example, one might want to schedule the bare minimum number of TaxiBots required on a calm day in order to minimise personnel costs. Therefore, it is of paramount importance to find the optimum value for this parameter and the model's sensitivity to change. A range of 10 to 50 is chosen to cover all values used in the four scenarios and to use realistic values used on both calm and busy days. The results for this can be found in section 5.3.1 A second important input parameter is the relative penalties given to a schedule change. The penalty multiplication factors play an important role in determining the selected penalty option based on the thresholds. For example, the default penalty for switching of TaxiBot is 50, while the default penalty for waiting is 10 per minute. In other words, the threshold value for waiting lies on 50/10 = 5 minutes before the model decides to go for the switching option. Following on that, the threshold for taxiing without a TaxiBot lies on 100/10 = 10 minutes. Changing these relative multiplication factors will result in different decisions taken by the model. Finding the sensitivity of the model with respect to this parameter will be investigated as well, as found in section 5.3.2

Table 7: Different sensitivity analysis case studies performed. Bold values represent the base scenario.

Sensitivity Analysis Scenarios	**Parameter to Change**	**Range [-]**					
Busy day + Current set of certified aircraft + ATA A/D times	Nr of TaxiBots	10,12,...,**30**,...50					
Busy day + Current set of certified aircraft + ATA A/D times	Penalty Multiplication factors $P_{switching}/P_{no\ TB}$						
	Early aircraft	10/20	- 10/15	- **5/10**	- 5/7.5	- 2.5/7.5	- 2.5/10
	Delayed aircraft	8/16	- 8/12	- **4/8**	- 4/6	- 2/6	- 2/8
	Relative factor	2x	- 1.5x	- **2x**	- 1.5x	- 3x	- 4x

5 Results

The results can be split in three sections. First a comparison of the strategic and tactical model output will follow in section 5.1. Here, the base scenario described in Table 6 is used for the development of the results. A comparison with the other scenarios follows in section 5.2. Lastly, a parametric study on the base scenario is described in section 5.3. All GVRSP models were written in Python 3.7 and were executed with a 1.8GHz Intel Core i7 8565U processor with 16GB DDR4 RAM.

5.1 Base Scenario Analysis

A number of output metrics can be taken from both the strategic and tactical model in order to compare the resulting schedules. For this, the base scenario, corresponding with scenario #1 as explained in section 4. A comparision with the other three scenarios will follow in section 5.2. A set of taxiing metrics can be found in Table 8. As can be seen, the parameters do not differ much, which is the objective of the model. Due to the delays occurring on the day of operations, a schedule which is not optimal anymore will be followed, resulting in shorter TaxiBot towing times for the tactical schedule with respect to the strategic schedule. The total taxi time of all aircraft combined decreases from 328.5 hour to 317.9 hour. This is due to different routes taken throughout the day or different conflict and collision avoidance actions. This could be for example due to closure of runway-crossing roads, resulting in different routes. The total taxi time can be split up in total taxi time for certified (159.7 hr) and non-certified aircraft (168.9 hr). This split is necessary to compare with the total summed time of aircraft taxiing with a TaxiBot (159.5 hr). In the strategic schedule, one certified aircraft could not be towed by a TaxiBot due to too long waiting times, resulting in a coverage of 99.9% of all certified flights. In other words, almost all certified aircraft have been towed by a TaxiBot. This coverage of certified aircraft being taxied is slightly decreased in the tactical schedule. More certified aircraft have to wait too long before a TaxiBot would be available and have to taxi without a TaxiBot, resulting in a coverage of only 99.2%. Overall, for almost half of the total taxi time (48.5% and 47.9% for the strategic and tactical schedule respectively) a TaxiBot tows the aircraft from its respective starting to ending node. In other words, in almost 50% of the time, aircraft taxi without using their engines. For the strategic schedules in the two calm scenarios a coverage of 100% is reached. As these taxiing parameters do not differ much between the strategic and tactical schedule, the rest of this paper will show plots of the tactical model to analyse. The plots of the strategic schedule closely resemble the plots shown.

Table 8: Comparison of taxi parameters between the strategic and tactical schedule, based on the base scenario for one day of operations with 1427 aircraft and 30 TaxiBots.

Parameter	**Strategic Schedule**	**Tactical Schedule**
Total taxi time [hr]	328.5	317.9
Of which certified ac/non-certified ac [hr (%)]	159.7 (48.6%) / 168.9 (51.4%)	153.6 (48.3%) / 164.3 (51.7%)
Total towing time with a TaxiBot [hr]	159.5	152.3
Of certified ac taxi time/total taxi time [%]	99.9% / 48.5%	99.2 % / 47.9 %
Certified aircraft taxied without a TaxiBot [-]	1	7

Figure 9 and Figure 10 show the waiting times before a certified aircraft is picked up by a TaxiBot and the taxiing times of both certified and non-certified aircraft respectively. The waiting times are shown on a normalised logarithmic scale, taken from 100 simulation runs. The maximum waiting time for certified aircraft before they got picked up by a TaxiBot is 8 minutes, however in most cases the aircraft do not have to wait at all as the TaxiBot will be readily available. Three other peaks can be seen, at 140s, 200s and 380s. These peaks result from the three most used routes, which start at the three waiting nodes for TaxiBots. The taxiing times for certified and non-certified aircraft are shown on a normalised scale, taken from 100 simulation runs. As can be seen, the most peaks for certified aircraft can be found around seven and a half minute and eleven and a half minute, which is equivalent to the most used routes plus the coupling and uncoupling operation times for arriving and departing flights. The same observation cannot be made distinctively for non-certified aircraft, as the operation time for arriving non-certified aircraft is 60 seconds, which is not distinctively clear due to all other taxiing times in the same range. The peak around 15 minutes is quite clear, of which 12 minutes is for operations in case of departing aircraft. Logically, the taxi times of certified aircraft tend to be shorter than the ones from non-certified aircraft, which can be largely assigned to the difference in coupling and uncoupling times and the presence or absence of ESUT and ECDT.

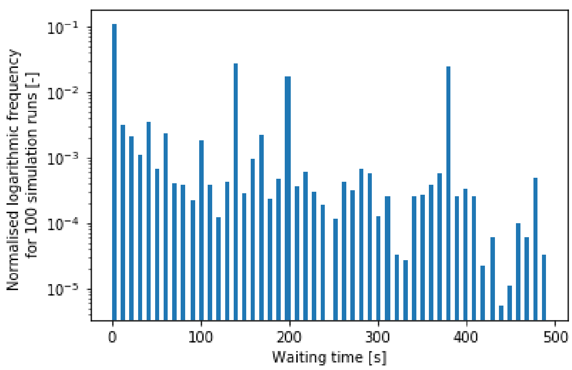

Figure 9: Normalised logarithmic histogram of the waiting times of certified aircraft before pickup in the tactical schedule, for 100 simulation runs with 30 TB.

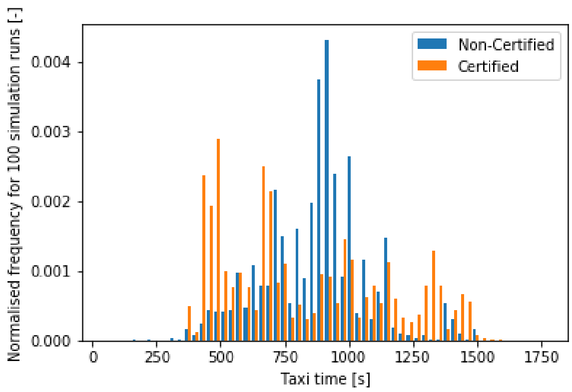

Figure 10: Normalised histogram of the taxiing times for both aircraft taxied with a TaxiBot (certified) and aircraft taxiing without one (non-certified) in the tactical schedule, for 100 simulation runs with 30 TB.

Next, the occupation of the TaxiBots is shown via Figure 11, Figure 12 & Figure 13. Figure 11 shows the occupation of each TaxiBot, divided into waiting, traversing and towing as a percentage of the total time. As can be seen, the first eight TaxiBots are in use for over 50% of the time, however the latter four TaxiBots are only used for 10% of the time. As the model searches for the TaxiBot closest by, a TaxiBot with a higher number could be called upon before a TaxiBot with a respective lower number if that former TaxiBot is closer by. However, as the model uses a greedy approach, overall the percentage of waiting increases over the sequence of TaxiBots. During a day of operations there are moments in which little or no aircraft arrive or depart, resulting in the TaxiBots being stationary at the waiting locations. This explains the fact that even the most used TaxiBot is waiting for approximately a quarter of the time. A corresponding figure, Figure 12, shows the number of TaxiBots in use, either traversing or towing, at each moment in time. 100 simulations have been plotted, with the thick black line showing the average. In 10% of the cases, this number of 30 TaxiBots was not sufficient to cover all aircraft immediately, and some aircraft had to wait for some time before being picked up. The waiting times as shown in Figure 9 hence is mostly due to on-route congestion, but als partly due to the TaxiBot fleet size. The same trend can be seen in Figure 13, which shows the number of aircraft present on the airport surface at each moment in time. This includes waiting on a TaxiBot and taxiing with or without a TaxiBot. During the first few hours, only a small number of aircraft arrive and depart, resulting in only a few TaxiBots in use. As soon as the day of operations starts to get busy, the number of aircraft on the airport surface and the number of TaxiBots in use start to grow. During lean hours, both decrease, while during peak hours, both values increase. The differences between the number of TaxiBots used for the 100 simulations is explained by the changing moments of congestion during the day caused by the different delays. The same can be said for the differences in the number of aircraft present on the airport surface.

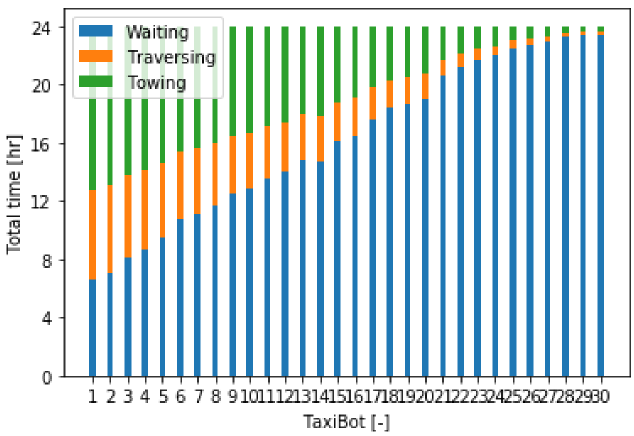

Figure 11: Occupation of each TaxiBot (30) for 100 simulation runs of the tactical schedule, divided into waiting, traversing and taxiing, split up for the 24 hours of operation.

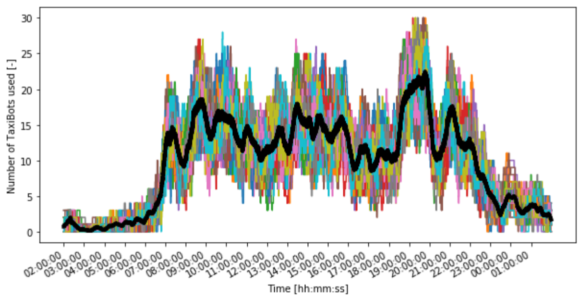

Figure 12: The number of TaxiBots that are in use at each moment in time for 100 simulation runs of the tactical schedule with 30 TB.

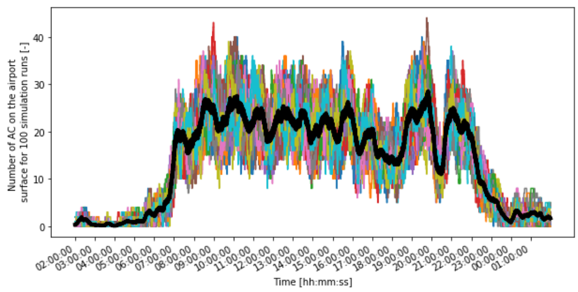

Figure 13: The number of aircraft that are present on the airport surface at each moment in time for 100 simulation runs of the tactical schedule with 30 TB.

Lastly, penalty parameters are shown in Figure 14 and Figure 15. Figure 14 shows the delays of each flight, paired with the option that was chosen for this flight in the tactical model. As can be seen, the three options show the same output, meaning that the length of the delay does not automatically decide which option will be chosen and so the three histograms look exactly the same. This choice of option is situation specific as it depends on the combination of i.e. the location of the TaxiBots, the number of TaxiBots available, the number of aircraft that just arrived or departed etc. Figure 15 shows the heights of the penalties and their relative occurrences, for each of the three options. As can be seen the heights of the penalties when option 1 is chosen are generally lower than the penalties when option 2 is chosen. Logically, option 2 has higher penalties due to the addition of the penalty multiplication factor $P_{switching}$. The highest peak occurs at 0, which means no change of TaxiBot is necessary and the aircraft does not have to wait. This peak is the most favourable one with respect to the goal of minimising deviations from the strategic schedule. For option 3, only two green peaks can be seen, at 160 ($P_{peak} \cdot P_{wait} \cdot P_{no\ TB\ -\ delayed}$) and 200 ($P_{peak} \cdot P_{wait} \cdot P_{no\ TB\ -\ early}$). As option 3 is only dependent on the multiplication factor $P_{no\ TB}$, only two peaks can occur dependent on early or late delays. For a wide range of penalty heights between approximately 40 and 180, all three options can occur, meaning that the specific penalty height does not automatically result in a specific option to be chosen.

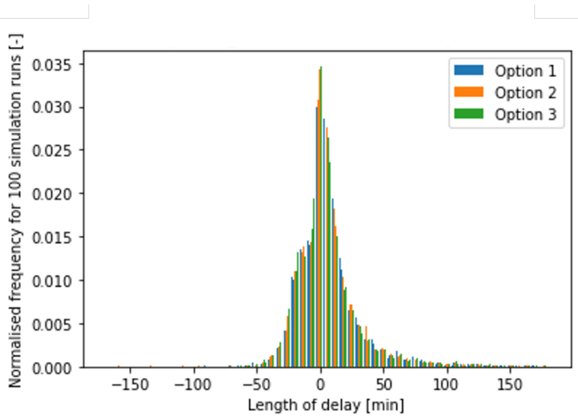

Figure 14: Normalised histogram of delays when selected penalty option 1 (scheduled TB), 2 (different TB) or 3 (no TB) over 100 simulation runs with 30 TB. All three histograms have the same shape.

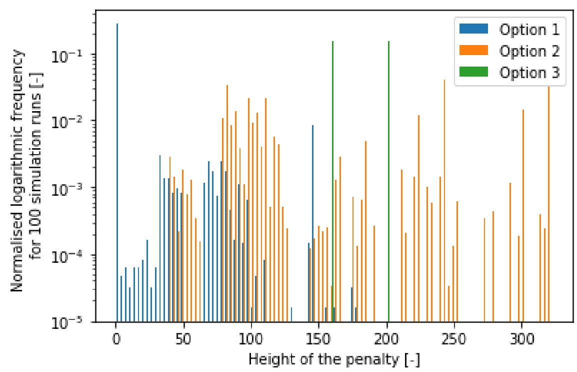

Figure 15: Normalised logarithmic histogram of the penalties when selected penalty option 1 (scheduled TB), 2 (different TB) or 3 (no TB) over 100 simulation runs with 30 TB.

5.2 Scenario Case Studies

We use the four different combinations of input data to generate the tactical schedules of the four scenarios as presented in Table 6. Table 9 provides an overview of the penalty parameters to compare. As in the calm days the number of scheduled flights is only 2/3 of the size of the busy days, fewer certified aircraft will need to be taxied, and thus the sum of penalties for scenarios 2 and 4 are approximately 2/3 of the sum of penalties for scenarios 1 and 3. Furthermore, as in the future 1.5 times more aircraft are certified, the sum of penalties are approximately 1.5 times higher for 3 and 4 when compared to 1 and 2. However, when normalising these values with respect to the total number of certified aircraft taxied by a TaxiBot, the relative differences become clear. Penalties are lower for calmer days, which is as expected, however the comparison between the *current* and *future* scenarios reveals less. This is explained by the fact that more TaxiBots are used in the future scenarios (45) as opposed to the current scenarios (30) and therefore a proper comparison is difficult. However, scenarios

1 and 2 using 30 TaxiBots resemble scenarios 3 and 4 with 45 TaxiBots.

The different options for penalties show that during calmer days, the selection of option 1 is more often possible. Again, the comparison with the future scenarios results in the same conclusions. The percentage for selected option 1 is slightly lower for scenario 3 when compared to scenario 1. Next to that on average over 12 certified aircraft had to taxi without a TaxiBot because it took too long before one was available or would arrive. This value is higher than the average of just over 7 in scenario 1. Both observations suggest that 45 TaxiBots for the future scenarios is slightly less efficient when compared to using 30 TaxiBots in the current scenarios.

The same type of conclusions can be drawn for the output parameters as shown in Figure 10, Figure 12 and Figure 15. The taxi times, use of TaxiBots and penalties for option 1 and 2 respectively tend to be generally lower for the calm scenarios, as expected. Penalties for option 3 do not even occur in the calm scenarios.

Table 9: Comparison of penalty parameters between the four scenarios with different number of (certified) aircraft and TaxiBots.

Parameter [-]	**#1**	**#2**	**#3**	**#4**
Number of Aircraft (of which certified)	1427 (53.0%)	954 (52.3%)	1427 (79.7%)	954 (75.6%)
TaxiBot fleet size	30	30	45	45
Sum of penalties	66510	40051	103731	57646
Normalised penalty per flight	88.4	80.4	91.6	80.1
Option 1 selected	51.5%	54.9%	50.6%	55.8%
Option 2 selected	48.4%	45.1%	49.4%	44.2%
Option 3 selected	0.1 %	0.0%	0.0%	0.0%
Certified aircraft taxied without a TaxiBot	7.16	3.53	12.18	6.36
Computation time [s]	59	30	55	23

5.3 Sensitivity Analysis

We have done two parametric studies on the base scenario as was described in section 4. Two parameters are of importance, the number of TaxiBots used and the penalty multiplication factors.

5.3.1 Number of TaxiBots

Ranging the number of TaxiBots from 10 to 50 and doing 100 simulation runs for each, the conclusion can be drawn that the number of TaxiBots necessary on AAS reaches an asymptote from approximately 34 TaxiBots on. This analysis can be taken from two types of output data, penalty parameters and TaxiBot occupation parameters. Starting with the former, the normalised penalties with respect to the number of certified aircraft for different sizes of the TaxiBot pool can be found in Figure 16 and the division of the penalties given to each certified aircraft can be found in Figure 17. The average minimum penalty given to a certified aircraft goes towards 80, which is in approximately half of the times coming from option 1. This same asymptote is reached from 34 TaxiBots on as well, which lies around 55%. Theoretically, this asymptote would go to 100% if sufficient TaxiBots are used, however due to the greedy approach of the strategic model, it does not take global scheduling into account. If an infinite amount of TaxiBots would be used, each certified aircraft could be scheduled to a different TaxiBot, which would always result in option 1. However, as the model does not solve to find a global optimum, this theoretical scenario would not take place with sufficient number of TaxiBots, causing this asymptote.

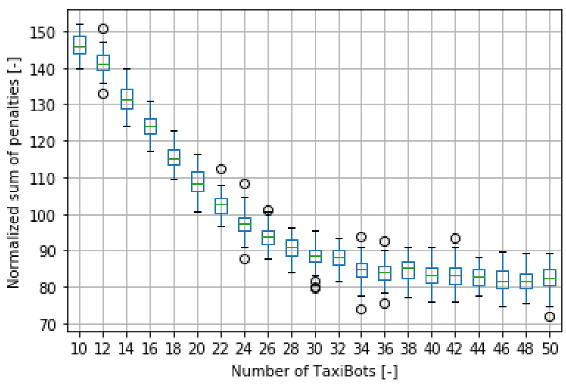

Figure 16: Normalised sum of penalties per certified aircraft for different TaxiBot pool size.

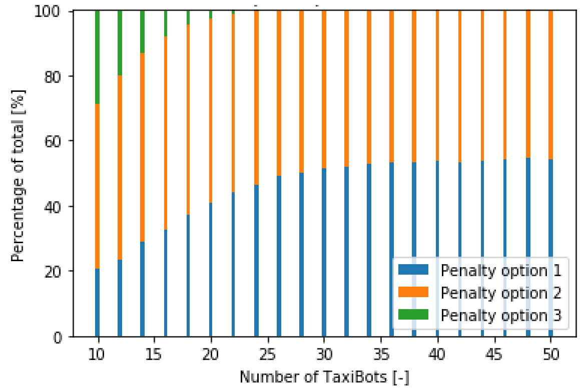

Figure 17: Division of penalty options 1 (scheduled TB), 2 (different TB) and 3 (no TB) for different TaxiBot pool size.

Combining this fact with two occupation parameters, a better insight in the number of TaxiBots necessary can be made. Figure 18 shows the percentage of the total time using TaxiBots. These percentages are based on the total taxi time of certified aircraft and the total taxi time of both certified and non-certified aircraft. Figure 19 shows the percentage of time the last TaxiBot in the pool is waiting and thus not performing any actions. This means that TaxiBot #10 only waits for approximately 45% of the time when only 10 TaxiBots are used, while TaxiBot #50 is waiting for 98% of the time in case of a TaxiBot pool size of 50. These two figures show that the same asymptote is reached, however the minimum number required can be better determined. In case of a relatively small TaxiBot pool size, some certified aircraft have to taxi without a TaxiBot, resulting in the TaxiBot towing time being smaller than the total taxi time for all certified aircraft. This percentage does not have to be 100%, but rather should be based on the expected amount of coverage of flights and the balance between busy and calm days. Related to that is the analysis drawn from Figure 19. When using a larger TaxiBot pool size, more and more TaxiBots will be solely occupied with waiting, which reduces the usage efficiency. Going for a smaller TaxiBot pool size will result in the last TaxiBot in the sequence being used more often. However, careful consideration is needed to make sure efficiency of TaxiBot usage with respect to towing and waiting and spare capacity of the number of TaxiBots is balanced.

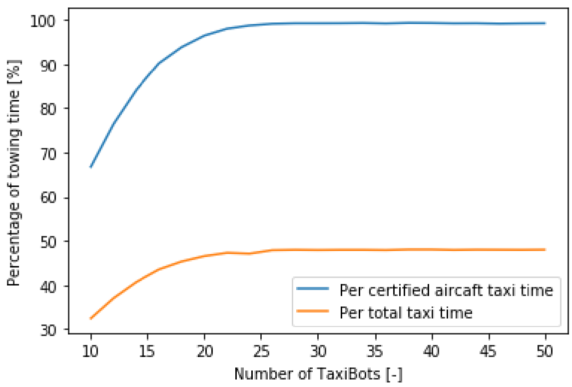

Figure 18: Percentage of time taxiing for certified aircraft has been performed by a TaxiBot (blue) and percentage of time taxiing for all aircraft has been performed by a TaxiBot (orange) for different TaxiBot pool size.

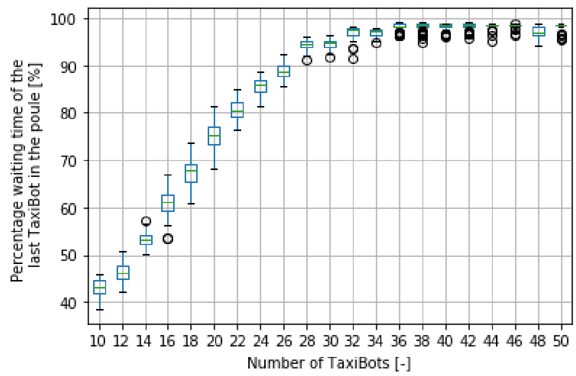

Figure 19: Percentage of time the last TaxiBot in the pool is occupied with waiting for different TaxiBot pool size.

5.3.2 Penalty Multiplication Factors

A second analysis has been performed on the penalty multiplication factors $P_{switching}$ and $P_{no\ TB}$. The default values are $5x$ and $10x$ respectively. Both the height of these factors and their relative ratio are changed, as can be seen in Table 7. The height of $P_{switching}$ affects the normalised penalty per certified aircraft most, the relative ratio has only little effect, as can be seen in Figure 20. The latter is due to the fact that only a small portion of the penalties are selected with option 3. The effect on the division of penalties, however, is very little, as can be seen in Figure 21. Both changes do not result in a different division of penalties, resulting in the conclusion that the penalty multiplication factors do not impact the model drastically.

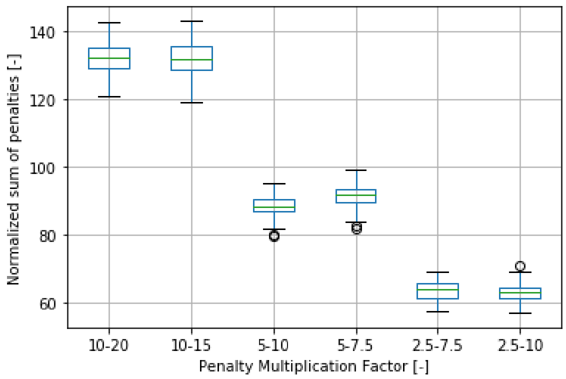

Figure 20: Normalised sum of penalties per certified aircraft for different penalty multiplication factors.

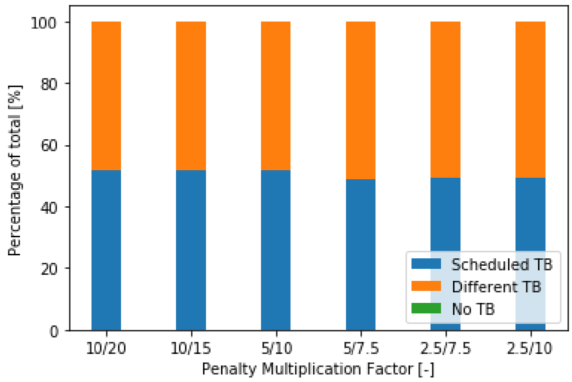

Figure 21: Division of penalties options 1 (scheduled TB), 2 (different TB) and 3 (no TB) for different penalty multiplication factors.

5.3.3 Validation

Lastly, a validation of the model and its output is ensured. A comparison with other VRP models using ETS on the other hand reveals that the number of aircraft capable to be taxied per ETS are in the range of 16 to 42 ([Kroese, 2021], [Guillaume, 2018] & [van Baaren, 2019]). Our model suggests a value of 25 certified aircraft/TaxiBot in case the default value of 30 TaxiBots is used, which is within the aforementioned range.

6 Conclusions

This paper proposes a model consisting of a strategic and tactical approach to solve the Greedy Vehicle Routing and Scheduling Problem (GVRSP). The model uses a strategic routing schedule and a stochastic flight schedule to find a tactical schedule, while minimising the deviations from the strategic routing schedule. The objective of this study was to develop a real-time airport operations planning tool which can be part of an integral planning and forecasting system.

The first main finding is the following: a comparison between the strategic and tactical schedule showed that only little had to be given in with respect to taxiing of certified aircraft when delays cause the flight schedule to change (99.9% of certified aircraft coverage in the strategic schedule with respect to 99.2% in the tactical schedule), even though a greedy approach was used, hence the efficiency of TaxiBots only decreased a little. Reasons for selecting a specific option in the tactical model were not based on the delay duration, but rather on the specific turn of events.

Next to that, a second main finding is the sizing of the number of TaxiBots. An upper limit is determined by an asymptote, which does not change any of the assessed output metrics, starting from 34 TaxiBots on. The lower limit, however, is up to the airport operations strategy to decide. This decision is dependent on the percentage of certified aircraft coverage is wanted, but also the balance between busy and calm days, which will affect the efficiency of the TaxiBots used and the spare capacity, mostly relevant for busy days.

Changing the flight schedule with respect to the number of flights showed that calmer days result in a more flexible schedule, however the selection of the favourable option 1 did not considerably increase. The reason for this is the greedy character of the model which does not take global scheduling into account. Enlarging the set of certified aircraft to be taxied by a TaxiBot did not result in a different conclusion, provided that the size of the TaxiBot pool is proportionally sized.

A parametric study was carried out to explore the features of the aircraft coverage and efficiency of the TaxiBots for a wide range of the TaxiBot pool size and penalty multiplication factors. The former resulted in an asymptote demonstrating an upper limit to the number of TaxiBots necessary on a day of operations. It is up to airport operators to determine a lower limit to the number of TaxiBots necessary based on the desired coverage of taxiing certified aircraft. The latter parametric study revealed that the size of the penalty multiplication factors can be adjusted accordingly with respect to the airport operations strategy, while their relative ratios play only a minor role. The division of the options is only slightly affected when changing this parameter.

Determining the optimal number of TaxiBots necessary on a specific day of operations needs some precaution nevertheless. As mentioned before, an asymptote is reached for the choice of option 1 at around 50%. However this is largely due to the structure of the greedy model, as theoretically eventually 100% ought to be reached. As the greedy model seeks to find a local optimum, rather than a global one, future choices are not taken into account. Further limitations in the model make this trade-off more difficult. The assumption that TaxiBots have unlimited power and can be charged throughout the night results in an underestimation of the number of TaxiBots necessary. In real-life operations, charging throughout the day might be inevitable, lowering the operational time. Furthermore, careful consideration is needed when setting the relevant thresholds in the model. For example, lowering $T_{response}$ or increasing $T_{max\ wait}$ would both result in a decreased availability of the TaxiBots as their percentage of waiting time would increase, affecting their throughpout with respect to towed aircraft again.

7 Recommendations

Further research could focus on expanding the stochasticity of the model for the different options to select. Currently, the delays in the flight schedule are used as deviations occurring throughout a day of operations, however multiple other possible changes can occur. Examples are gate alterations, TaxiBots being delayed or breaking down, the cancellation of flights or emergency flights added to the flight schedule, or errors made by the cockpit crew or operating crew.

Moreover, the penalty scoring system can be further investigated in order to go to a complete decision-support system in the future. Now, the waiting time and flight density are taken into account, however, as mentioned in section 2, decision parameters such as the follow-up missions of TaxiBots and reasons for delays could be added to the penalty decision system in order to make it more comprehensive. By weighing different types of penalties, the output decisions can be made more all-inclusive based on the different types of changes [Pei et al., 2021].

Next to that, as TaxiBots are a novel concept, future research could focus on improving the VRP in both the strategic and tactical models to better mimic real-life operations by improving the model and its parameters. Further TaxiBot operations feasibility tests will reveal challenges to be implemented in the model. Coupling and uncoupling locations of the certified aircraft are now situated at the runway or gate, however this will not be possible on the airport surface. Therefore, dedicated (un)coupling locations should be designated, at which other aircraft would not be hindered in their taxiing sequence and thus sufficient space is present. Determining the optimal locations is another recommended research direction.

8 Acknowledgements

We appreciate the helpful discussions with Smart Airport Systems (SAS) making this research more practicable. We thank Nicolas Girard and Frederic Vanoost for their valuable discussions on the model assumptions and model applicability in practice. Their experience and knowledge on the TaxiBot system and airport taxiing operations improved this paper. Their support is gratefully acknowledged.

References

[Abusalim et al., 2020] Abusalim, S., Ibrahim, R., Saringat, M. Z., Jamel, S., and Wahab, J. A. (2020). Comparative analysis between dijkstra and bellman-ford algorithms in shortest path optimization. *International Conference on Technology, Engineering and Sciences (ICTES)*, 917(012077):1–11.

[Atkin et al., 2010] Atkin, J., Burke, E., and Ravizza, S. (2010). The airport ground movement problem: Past and current research and future directions. *Research in Air Transportation*, page 131138.

[Bewoners-Aanspreekpunt-Schiphol, 2020] Bewoners-Aanspreekpunt-Schiphol (2020). Baangebruikcijfers 2019. Available at `https://bezoekbas.nl/wp-content/uploads/2020/02/Baangebruikcijfers_2019.pdf`.

[Deonandan and Balakrishnan, 2010] Deonandan, I. and Balakrishnan, H. (2010). Evaluation of strategies for reducing taxi-out emissions at airports. *Proceedings of the 10th AIAA Aviation, Technology, Integrations, and Operations Conference*, pages 13–15.

[Dorndorf, 2005] Dorndorf, U. (2005). Flight gate scheduling: State-of-the-art and recent developments. *Omega - The International Journal of Management Science*, 35:326 334.

[Dorndorf et al., 2007] Dorndorf, U., Jaehn, F., Lin, C., Ma, H., and Pesch, E. (2007). Disruption management in flight gate scheduling. *Statistica Neerlandica*, 61(1):92–114.

[Evertse and Visser, 2017] Evertse, C. and Visser, H. (2017). Real-time airport surface movement planning: Minimizing aircraft emissions. *Transportation Research Part C*, 79:224–241.

[Fordham et al., 2016] Fordham, D. et al. (2016). Deriving benefits from alternative aircraft-taxi systems - acrp report 158. Technical report, Airport Cooperative Research Program - Transportation Research Board.

[Gordijn, 2016] Gordijn, E. (2016). Milieueffectrapportage - nieuw normen- en handhavingstelsel schiphol deel 1: Hoofdrapport. Technical report, Schiphol Group.

[Gotteland et al., 2001] Gotteland, J.-B. et al. (2001). Aircraft ground traffic optimization. *ATM 2001, 4th USA/Europe Air Traffic Management Research and Development Seminar*, pages 1–9.

[Guillaume, 2018] Guillaume, N. (2018). Finding the viability of using an automated guided vehicle taxiing system for aircraft. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands.

[Guo et al., 2014] Guo, R., Zhang, Y., and Wang, Q. (2014). Comparison of emerging ground propulsion systems for electrified aircraft taxi operations. *Transportation Research Part C*, 44:89–109.

[Hospodka, 2014a] Hospodka, J. (2014a). Cost-benefit analysis of electric taxi systems for aircraft. *Journal of Air Transport Management*, 39:81–88.

[Hospodka, 2014b] Hospodka, J. (2014b). Electric taxiing taxibot system. *MAD - Magazine of Aviation Development*, 2(10):17–20.

[Ithnan et al., 2013] Ithnan, M. M. et al. (2013). Aircraft taxiing strategy optimization. *Department Maritime & Transport Technology, Technology University of Delft,*, pages 1–9.

[Jiang et al., 2013] Jiang, Y., Liao, Z., and Zhang, H. (2013). A collaborative optimization model for ground taxi based on aircraft priority. *Mathematical Problems in Engineering*, 2013:1–9.

[Kroese, 2021] Kroese, J. (2021). Electric taxiing: an optimisation study on the future of airport operations. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands.

[Kumar et al., 2008] Kumar, V., Sherry, L., and Thompson, T. (2008). Analysis of emissions inventory for "single-engine taxi-out" operations. *Third International Conference on Research in Air Transportation*, pages 445–450.

[Lan et al., 2006] Lan, S., Clarke, J.-P., and Barnhart, C. (2006). Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science*, 40(1):15–28.

[Lorscheid et al., 2012] Lorscheid, I., Heine, B.-O., and Meyer, M. (2012). Opening the 'black box' of simulations: Increased transparency and effective communication through the systematic design of experiments. *Computational and Mathematical Organization Theory*, 18(1):22–62.

[Lukic et al., 2018] Lukic, M. et al. (2018). State of the art of electric taxiing systems. *IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference*, pages 1–6.

[Lukic et al., 2019] Lukic, M. et al. (2019). Review, challenges and future developments of electric taxiing systems. *IEEE Transactions on Transportation Electrification*, 5(4):1441–1457.

[Novianingsih and Hadianti, 2014] Novianingsih, K. and Hadianti, R. (2014). Modeling flight departure delay distributions. *International Conference on Computer, Control, Informatics and Its Applications*, pages 30–34.

[Pan et al., 2017] Pan, W. et al. (2017). The impact of green taxiing systems using on airport. *3rd International Conference on Electronic Information Technology and Intellectualization*, pages 284–290.

[Pei et al., 2021] Pei, S., He, Y., Fan, Z., and Zhang, B. (2021). Decision support system for the irregular flight recovery problem. *Research in Transportation Business & Management*, 38(100501):1–9.

[Pillac et al., 2013] Pillac, V., Gendreau, M., Gueret, C., and Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–13.

[Psaraftis et al., 2016] Psaraftis, H. N., Wen, M., and Kontovas, C. A. (2016). Dynamic vehicle routing problems: Three decades and counting. *Networks*, 67(1):3–31.

[Roling, 2009] Roling, P. (2009). Airport surface traffic planning optimization: a case study of amsterdam airport schiphol. *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, (7079):1–7.

[Roling et al., 2015] Roling, P., Sillekens, P., and Curran, R. (2015). The effects of electric taxi systems on airport surface congestion. *15th AIAA Aviation Technology, Integration, and Operations Conference*, pages 1–10.

[Salihu et al., 2021] Salihu, A. L., Lloyd, S. M., and Akgunduz, A. (2021). Electrification of airport taxiway operations: A simulation framework for analyzing congestion and cost. *Transportation Research Part D*, 97(102962):1–13.

[Schiffer and Walther, 2017] Schiffer, M. and Walther, G. (2017). The electric location routing problem with time windows and partial recharging. *European Journal of Operational Research*, 260:995–1013.

[Schiphol, 2020] Schiphol (2020). Schiphol and partners to begin sustainable aircraft taxiing trial. Available at `https://news.schiphol.com/schiphol-and-partners-to-begin-sustainable-aircraft-taxiing-trial/`.

[Sirigu, 2017] Sirigu, G. (2017). *Planning and reconfigurable control of a fleet of unmanned vehicles for taxi operations in airport environment.* PhD thesis, Politecnico di Torino, Turin.

[Sirigu et al., 2018] Sirigu, G. et al. (2018). Autonomous taxi operations: algorithms for the solution of the routing problem. *AIAA Information Systems-AIAA Infotech*, pages 1–7.

[Smeltink et al., 2004] Smeltink, J. et al. (2004). An optimisation model for airport taxi scheduling. *Elsevier Science*, pages 1–25.

[Su et al., 2021] Su, Y., Xie, K., Wang, H., Liang, Z., Chaovalitwongse, W. A., and M.Pardalos, P. (2021). Airline disruption management: A review of models and solution methods. *Engineering*, 7(4):435–447.

[Tas et al., 2013] Tas, D., Dellaert, N., van Woensel, T., and de Kok, T. (2013). Vehicle routing problem with stochastic travel times including soft time windows and service costs. *Computers & Operations Research*, 40(1):214–224.

[TLD, 2019] TLD (2019). Taxibot-nb undercarriage taxiing system - specifications sheet.

[van Baaren, 2019] van Baaren, E. (2019). The feasibility of a fully electric aircraft towing system. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands.

[Yan-Du et al., 2014] Yan-Du, J., Brunner, J. O., and Kolisch, R. (2014). Planning towing processes at airports more efficiently. *Transportation Research Part E*, 70:293–304.

[Zaninotto et al., 2019] Zaninotto, S., Gauci, J., Farrugia, G., and Debattista, J. (2019). Design of a human-in-the-loop aircraft taxi optimisation system using autonomous tow trucks. *AIAA AVIATION Forum - Airport Surface Ops*, pages 1–13.

II

Literature Study
previously graded under AE4020

1
Abstract

The following literature review is focused on the implementation of TaxiBots, electric taxiing vehicles, on Amsterdam Airport Schiphol and on the development of a reactive model to cope with sudden changes in the routing and scheduling of vehicles at the apron due to aircraft delays. As emissions from aviation continue to increase, solutions need to be found to counteract this. Historically, solutions were mostly sought in the airborne phase of a flight, however among on-ground operations are areas with huge potential as well. Multiple solutions have been found to electrify the taxiing of aircraft from and to the runway, one of which being the TaxiBot solution by Smart Airport Systems (SAS). However, as this solution is quite a novel concept, lots of technical and operational challenges arise. The routing and scheduling of all aircraft on the runways and taxiways is quite a complex puzzle to solve, especially with the addition of TaxiBots, increasing the total number of vehicles driving around.

Similar research on the routing and scheduling, named the vehicle routing problem, on airports with the addition of TaxiBots and other electric taxiing solutions has been focusing on minimising any of the key performance indicators, such as fuel consumption, emissions and/or costs. One common aspect in the literature is the use of deterministic input data. However, real-life situations always bring uncertainty and sudden changes will always occur. Therefore, general recommendations suggest to incorporate disruption management into the problem. One way of doing so is the make schedules more robust to sudden changes, the other one is by dynamically modelling solutions with interactive input changes. A split can be be made in strategic (defined a set time before the actual operation) or tactical (defined during the day of operation) solutions, as is recommended by SESAR Joint Undertaking as well. Therefore, the research aim of this thesis is defined as follows:

To determine the effect of probabilistic aircraft departure and arrival delays on a vehicle routing schedule at Amsterdam Airport Schiphol which includes the use of an electric taxiing system, viz TaxiBot, to schedule the routing of aircraft and TaxiBots with near real-time updates on arrival and departure times, by creating a reactive optimisation routing and scheduling model which can reiterate the planning based on the new received non-deterministic time information while trying to minimise the deviations from the initial schedule.

In order to reach this aim, a main research question is formulated as follows:

What is the effect of probabilistic aircraft arrival and departure delays on the time-space planning of the vehicle routing problem with electric taxiing systems at Amsterdam Airport Schiphol? Sub-questions that follow are focused on defining how such a strategic and tactical optimisation should look like.

The literature study consists of a review of current taxiing methods, but also other external and on-board solutions. Examples are single-engine taxiing, electric landing gear systems and the TaxiBot. Each of these solutions is compared with the current method, based on a different set of criteria such as fuel consumption, emissions and costs. The TaxiBot itself is discussed as well. Recent tests at Schiphol show new insights that will be used in future research. The company structure, technical specifications and implementation considerations of this system are elaborated. As not all specifications of the TaxiBot vehicle are known, previous literature has designed a simplistic set of taxiing vehicles themselves, mimicking the TaxiBot.

Different methods to model such operations at an airport are discussed. One main split can be made between mixed-integer linear programming, finding an optimal solution, or meta-heuristics, trying to find a near-optimal solution in a shorter computation time. The first one is split up in the two most used problems, vehicle routing problems and fleet assignment problems. Time-space diagrams and rolling windows are the two relevant aspects taken from literature, which has been summarised extensively based on their model objectives, output and other characteristics. Fleet assignment problems result in the scheduling of TaxiBots to tasks planned. The conclusion can be drawn that both problems can be combined as well, resulting in an optimal routing and planning solution. The other type of models, meta-heuristics contains amongst others genetic algorithms or other artificial intelligence methods, which have been used to solve above problems as well. However, as an exact solution ought to be found, MILP are favoured.
A case study on the routing of vehicles at AAS follows. Different VRP modelling aspects, airport networks, flight schedules, TaxiBot specifications and aspects regarding operational time uncertainty are discussed. For each of these subjects, the most suitable option is selected to be used in the case study to be developed in the thesis.

A discussion on the scientific research gaps follows. The first one mentions the research on the TaxiBot battery. The sizing of the battery itself, the charging capacity on an airport and the number and location of charging stations are all aspects that can be optimised. Different strategies such as battery swapping should be researched as well. A second gap is to pursue a feasibility study of TaxiBots at an airport. Many different parameters come into play when determining if the implementation of TaxiBots is feasible with respect to costs and/or sustainability parameters. Such parameters could be the airport structure, local temperature or maintenance possibilities. The relative importance but also the range of parameters could be determined which will be of use when determining the implementation of TaxiBots. Each of these research gaps can also be explored for other types of electric taxiing solutions, such as on-board systems.

The model that will be built will consist of three parts, each resulting in a routing and scheduling solution. First, a vehicle routing problem will be developed containing necessary aspects and with unlimited TaxiBots and unlimited energy capacity. Then, in the second step, this number of TaxiBots is decreased up to the point where a minimal number is used while still reaching a feasible solution. This strategic solution will consist of among others a time-space diagram. Then, the third part will have randomised aircraft delays, both positive and negative, which will be incorporated dynamically. The tactical solution found can then be compared to the strategic solution. Requirements for this model have been established and a fitting experimental setup has been defined. The input, output and verification and validation aspects are covered as well. Above models and planning are visualised in a functional flow diagram and a Gantt chart respectively.

To conclude, due to the increasing need in the aviation sector, new electric solutions have been developed. In order to correctly implement these in the current operations, particular routing and scheduling solutions are needed. However, with the always present chance of sudden changes, the optimal schedule will almost never be exactly followed. In order to still find a feasible solution, new optimal tactical solutions have to be found on the spot, which should not deviate too much from the original. With the implementation of such models, vehicle routing schedulers can find new fitting solutions in a reactive way just after such a change is needed.

The reviewed literature provides insights into the TaxiBot and its implementation, but also compares this with other ETS present. Different modelling methods are used in order to research the implementation of TaxiBots, and different perspectives can be found on the aim of the research. On a higher level, research is done in vehicle routing problems, but also fleet scheduling assignments and determining the optimum number of TaxiBots to implement. On a lower level, each of these research direction has different objectives. Focus is laid on the minimisation of emissions, which are correlated to fuel consumption, however costs is an important objective as well. Looking at defining the case study, different angle of approaches come forward as well, each with their reasoning based on the aim of the research. As this specific research has a different objective, careful selection of the appropriate resources is necessary.
One general consensus found in above literature is the defining of a model in a deterministic way. This result in correct theoretical models, however lots of recommendations are focused on researching stochastic or dynamically updated models. This research tries to fill this gap. With that, TaxiBot schedules can be generated on a tactical level. Hence, a future recommendation is to incorporate such a reactive model in the tactical planning of airport operations in order to minimise deviations from the defined strategic schedule.

2 Introduction

Aviation plays a big role in the current society. Even though it brings a lot of good to the world, unfortunately, it brings quite some negative consequences with it as well. 2.5% of the total global CO2 emissions are due to aviation [19]. Next to that, the aviation industry is expected to grow the coming years. Predictions made in 2018 by IATA estimated the global number of air travellers to reach 8.2 billion by 2037 [1]. However, as the world finds itself in the midst of a global pandemic, predictions will need to altered. As it is unknown yet how this crisis and the subsequent effects will evolve in the coming years, Eurocontrol predicts a number of scenarios, of which the most plausible predicts that 2024 will be the year in which the number of flights will reach the previous 2019 levels [2]. Even in the worst-case scenario, the aviation industry would be back on track by 2029. Concluding from this, the aviation industry will continue to play a large role in the global environmental pollution and this will even continue to increase after the current dip.

In order to counteract such negative environmental impacts, each of the different parties in the aviation industry are trying to reach more sustainable solutions. Examples are alternative fuels for aircraft engines such as biofuel [23] or new aircraft concepts, such as the recently developed flying V[3]. The best case scenario would be to reduce emissions in absolute numbers, however that would mean that the percentual decrease in emissions should be larger than the increase in air travel. However, sustainable improvements in aircraft efficiency tend to be only small percentages. In other words, not enough improvement can be achieved in aircraft in order to counteract the increasing usage of jet fuel, which leads to an increase of emissions. As noted by Lukic et al. [39] improvements in fuel efficiency are mostly focused on the airborne phase of a flight. However, aircraft are not continuously in the air, but rather also use their engines on the ground for a part of the mission. At these phases of the flight, the on-ground taxiing phase more specifically, larger improvements can be gained. During these phase, the engines of the aircraft are used, which is not optimal. These engines are not designed for these settings and thus produce more pollutants than during cruise. Next to that, on the ground, many more energy options are present to make sure these aircraft are moved to their desired position. One such viable option is via electric taxiing systems (ETS). By using electrical energy for the taxiing phase instead of jet fuel, a big step can be taken towards a more sustainable solution.

Next to that, as these ground phases take place at an airport, the surrounding environment directly benefits from this. Airports which seek to improve their sustainability benefit from such ETS. Royal Schiphol Group, the owner of Amsterdam Airport Schiphol (AAS) has set its mission to become emission free on the airport by 2030[4]. All ground bound vehicles ought to be driving on electricity or hydrogen, which includes the towing vehicles used. One of the ways Schiphol tries to reach this goal is to team up with SAS, one of the developers of the TaxiBot. Tests conducted in the first half of 2020 on the feasibility of TaxiBots on Schiphol resulted in the Proof of Concept of TaxiBot at Schiphol, however further research is necessary.

[1] https://www.iata.org/en/pressroom/pr/2018-10-24-02/, accessed on 24-12-2020

[2] https://www.eurocontrol.int/publication/eurocontrol-five-year-forecast-2020-2024, accessed on 24-12-2020

[3] https://www.tudelft.nl/lr/flying-v/, accessed on 24-12-2020

[4] https://www.schiphol.nl/nl/schiphol-als-buur/pagina/emissievrij-in-2030/, accessed on 24-12-2020

This research specifically focuses on the external electric taxiing solution, TaxiBot, and the implementation of it at an airport. Developments of such ET solutions are underway, however as they are a novel research direction, practical implications on the implementation of it arise. This research tries to fill that gap by strengthening the research in the field of TaxiBot routing and scheduling.

Important to note is that "TaxiBot" is used in many settings in this report. TaxiBot is the brand of the electric taxiing solution by SAS, however taxibot is also used as generic term for such a type of solution. Continuing on that, many synonyms are used for "Electric Taxiing Solutions/Systems (ETS)". These systems represent all different types of solutions, both external and on-board. External solutions are also identified as "Electric Taxiing Vehicles" or ETV.

This report is structured as follows: First a research framework is given in chapter 3 which states what will be investigated and what will not. The research questions and research aim will be given as well. Then, a literature review follows in chapter 4. This chapter covers the different taxiing methods and compares these with respect to the current situation. Secondly, the TaxiBot in particular is discussed, providing the necessary information to model it correctly. The different modelling methods found in literature are also covered in this chapter. Literature on a case study will provide all necessary information to model a specific case. Lastly, the four scientific gaps found in literature are discussed as well. chapter 5 provides the method that will be followed in order to model the use case necessary to answer the research question. The experimental setup and expected results will be covered as well. Finally, an overview of the planning of the thesis will be given in chapter 6. All of this will be covered in the conclusion in chapter 7.

3
Research Framework

This chapter gives a framework for the research that will be conducted. The initial problem for this project will first be covered in section 3.1. After that, the research aim and questions of the thesis will be discussed in section 3.3 and section 3.4 respectively. The scope of the project will be clearly set in section 3.5. Finally, the research framework will be summarised in the contribution work in section 3.6.

3.1. Problem Statement

While there are multiple different taxiing methods assessed and even developed, there is one consensus reached on the problem statement; pollutant emissions need to be reduced.
"During idle mode, an engines performance is less efficient due to the low combustor temperature. This induces higher fuel consumption, and emissions of hydrocarbon and CO," according to Ithnan et al. [28, p. 2]. As engines are not designed for this stage, relatively high fuel consumption and emissions occur which should be reduced as much as possible. Next to that, emissions are proportionate to the taxi time and taxi times tend to increase over the past couple of years [16],[17] [22]. Furthermore, Guo et al. [22] mention the taxi time to be 10-30% of the total flight time and the excess fuel burn during taxi-out phases to be 75 kg per flight. Overall, the aviation industry is one of the fastest-growing contributors of greenhouse gas emissions and with the increase of air travel this trend will continue the coming years.
Next to these two growing problems, the price of jet fuel seems to steadily rise as well, according to Lukic et al. [39] and Guo et al. [22]. Hence, there is an urgent need to more eco-friendly and fuel-efficient solutions. These solutions have been sought after in the aircraft development, however lots of progress is to be found in the ground operations as well. Therefore, multiple solutions have been brought forward which could all lead to substantial decreases in fuel consumption and emissions while being ready in the near future. This combination of both makes this field in interesting one to explore further.

Concerns from the direct surroundings of airports have been growing and as solutions for this problem will be tackled mostly directly at the airport, the emitted pollutants in the airport neighbourhood will be decreased. Improving taxiing operations at an airport can reduce CO2 emissions up to 70% , hence much potential can be gotten out of these solutions. [44]

As TaxiBots are a novel concept on airport, lots of technical and operational challenges arise when such systems are implemented at an airport. TaxiBots are yet another type of vehicles introduced at the apron, which should all be managed in a certain way. As aircraft and TaxiBots will make use of the runways and taxiways of an airport, ATC has control over these vehicles to make sure safety is secured. Careful routing will need to be considered and strategic scheduling is needed for that. Therefore, a routing and scheduling solution needs to be found to make sure conflicts and collisions are avoided while making sure taxi time, and all its other linked parameters, such as fuel consumption, are minimised. However, sudden changes will always be present. A planning is ought to be made robust, however limitations are always present. Therefore, when for example an arriving aircraft is delayed, the old schedule might not be feasible anymore and a new tactical schedule needs to be developed. As an airport usually has to make real-time decisions, such a new schedule has to be be generated within a reasonable time. Only then, airport operations can continue with no or limited disruptions.

3.2. Similar Research

Research has been done on vehicle routing problems with electric taxiing solutions with varying output requirements. Literature on this topic starts in 2015 with the paper by Sillekens [55], who analyses the effect of on-board ETS on the VRP with respect to capacity, average taxi times and other KPI's. in 2017, Yan Du [67] continues on this topic by defining a VRP model, while focusing on low computation times. This paper focuses on planning the ET vehicles at an airport more efficiently while looking a mixed fleet with vehicles routing to multiple depots and multiple trips respecting time windows. Guillaume [21] developed a routing and scheduling model in order to optimise the number of automated guided vehicles for aircraft taxiing with respect to taxiing costs in 2018. Later on in 2019, Van Baaren [64] developed a similar VRP model which focused on optimising the number of electric taxiing systems with respect to fuel consumption, emissions and energy usage. Van Baaren defined three towing vehicle designs, mostly resembling the TaxiBot, which in turn have been used by Kroese in 2021 [31]. Kroese combined the VRP with a fleet scheduling assignment (FSA) resulting in a scheduling of the ET vehicle, taking into account charging of the batteries.

As will be thoroughly explained in section 4.3, the VRP and FSA are often combined. Dorndorf et al. (2007a) [12] review the state-of-the-art flight gate scheduling literature and conclude their findings with recommendations regarding the solution methods. These classes are called "a priori, interactive and a posteriori methods" and is based on when the applicable decision maker, being either the programmer or a random allocator, intercedes in the running program. Most of above research has been performed a priori, specifying the input characteristics beforehand. The second class, interactive interventions during the execution, is what will be done in this research. Similar research is done by Dorndorf et al. (2007b) [13], which focused on disruption management in flight gate scheduling. Each iteration of the problem will result in a non-optimal solution, as the input change will result in a different optimisation, however it is important to minimise any deviations from the previous optima [12]. Adding disruptions to the schedule is recommended by Yan Du et al. [67] as well, along with Zaninotto et al. [68] who recommended to implement a tactical planning module, adjusting input parameters, such as taxi routes and schedules, dynamically.

Lastly, the need for the development of airport operations planning with ETS is also adopted by the SESAR Joint Undertaking partnership, as projects in this topic are being developed[1]. The three research directions that will be developed are the following:

- *Overall aircraft engine-off navigation concept of operations, detailing how the three eco-friendly solutions above may combine in the airport surface management process both at strategic and tactical level in order to minimise fuel consumption and emissions without impacting arrival and departure flight schedules*
- *Business model to help airports and/or airlines evaluate their benefits in the implementing these technologies*
- *Real-time evaluation of environmental indicators to support decision-making, conflict free routing for all vehicles, reallocation of techniques to adapt to in real time.*[1]

Especially the first item, developing VRP models that focus both on the strategic and tactical level aligns with the research of this thesis.

3.3. Research Aim

In order to fill the research gap with respect to the problem stated above, the following research aim has been formulated:

To determine the effect of probabilistic aircraft departure and arrival delays on a vehicle routing schedule at Amsterdam Airport Schiphol which includes the use of an electric taxiing system, viz TaxiBot, to schedule the routing of aircraft and TaxiBots with near real-time updates on arrival and departure times, by creating a reactive optimisation routing and scheduling model which can reiterate the planning based on the new received non-deterministic time information while trying to minimise the deviations from the initial schedule.

[1] https://www.sesarju.eu/news/green-promise-aircraft-taxiing-technologies, accessed on 27-01-2021

Effect: There will be two different schedules generated, one strategic and one tactical. The effects will not only be determined by the visualisation of a time-space diagram, but other KPI's will be compared as well.
Probabilistic aircraft departure and arrival delays: Aircraft operations are tightly planned, but do not always follow that schedule. Therefore, there will be a difference between the planned and actual departures and arrivals, in which the delays are drawn from a probability density function.
Vehicle routing schedule: The aircraft and TaxiBots will move over the airport, and with such a schedule the location of each vehicle at every moment in time is known.
ETS, viz TaxiBot: There are multiple different electric taxiing solutions, such as on-board systems, but this research focuses on the external towing vehicle TaxiBot.
near real-time updates: Every couple of minutes, a new time interval is assessed in which the operations starting in this interval will be known and fixed. In real-world operations, data retrieval is continuous, as at every moment in time new information can arrive. Here, it is discretised in small time intervals.
Reactive: The model will take the new information from each time interval and use it to reiterate in order to find a new optimum schedule.
optimisation: The routing and scheduling of aircraft and TaxiBots will be done while minimising the total taxi time, which links to minimum fuel consumption, emissions and delays.
Reiterate: Every time interval, the new schedule for the coming part of the day will be scheduled, while the past time intervals are fixed for the rest of the day.
New received: Each time interval, a draw from the probability density function takes place for all aircraft that have their delay range in said time interval and thus only at this specific interval the new information is revealed.
Non-deterministic: The delays will be drawn from a probability density function and thus will differ every time the model is run.
Minimise deviations: The difference between the strategic and tactical schedule needs to be as small as possible in order to cause as little disruption as possible to the rest of the schedule.
Initial schedule: A strategic schedule will be generated before the tactical one in the same model, however in real-life such a strategic schedule could be developed a couple of weeks or months before the actual start of that period.

3.4. Research Questions

Based on the aforementioned research aim together with the problem statement, one main research questions can be formulated:

What is the effect of probabilistic aircraft arrival and departure delays on the time-space planning of the vehicle routing problem with electric taxiing systems at Amsterdam Airport Schiphol?

In order to accurately answer this question, a set of sub-questions are formulated as follows:

1. How should the strategic optimisation model of a vehicle routing problem of an electric taxiing system, viz TaxiBot, look like in order to be able to find an optimal airport routing solution which can be used as input for the tactical optimisation model?
   - (i) How does the vehicle routing problem of aircraft and TaxiBots have to be modelled at Amsterdam Airport Schiphol?
   - (ii) How should the conflict and collision avoidance of both aircraft and TaxiBots at AAS be modelled?
   - (iii) How should the airport routing nodal network of AAS look like?
   - (iv) What are the TaxiBot specifications to be used in the model?
   - (v) What is the minimum number of TaxiBots needed to attain a feasible solution?
   - (vi) What would be the best output, both in terms of visualisation and metrics for the optimal routing schedule?
2. How should the tactical optimisation model of a vehicle routing problem of an electric taxiing system, viz TaxiBot, look like in order to be able to find a new optimal schedule taking into account probabilistic aircraft arrival and departure delays?
   - (i) How are the probabilistic aircraft delays for arrivals and departures modelled?
   - (ii) How should deviations from the strategic schedule be minimised?
   - (iii) What is the best time interval for the iteration of the problem taking into account the average time aircraft delays are known beforehand by ATC and the model computation time?
   - (iv) What would, next to the time-space schedule, be the best output KPI's in order to determine the effect of the probabilistic input data?

3.5. Research Scope & Assumptions

In order to answer the research questions set, clear boundaries have to be set to define the research scope. In order to scope the framework, limitations in the research come forward, which are shaped by the following set of assumptions:

- The output of the tactical optimisation model will be a schedule for only one day. Even though aircraft operations continue around the clock, the number of night operations is substantially lower than the number of operations during the day. Therefore, it is assumed that days are independent from each other and the events and schedules from one day do not affect the next day. In reality, some personnel will continue to work at night or any other activities will continue over multiple days. However the available time span will be divided in 24hour periods. Aircraft that have a different arrival and departure day will be split up and thus it might occur that only the departure or only the arrival of one aircraft turnaround is assessed.
- As some of the specifications of the TaxiBot by SAS are confidential, some parameters will be taken from academic literature.
- In order to limit computation time, the strategic schedule will be computed for the length of one day as explained above. Usually, such a strategic schedule will be generated in terms of years, however as the tactical schedule will be compared with the strategic one, there is no need to compute a full year schedule. On the other hand, different types of days will be assessed, i.e. the busiest and calmest days throughout the year, in order to determine differences in daily schedules.

- Only aircraft delays will be taken into account as changing variables. Secondary options are delays in the TaxiBots or other sudden changes, such as runway configuration changes.
- It is assumed that all cockpit crew is able to handle TaxiBots. According to [5], cockpit crew needs training before it can operate TaxiBots, however as these TaxiBots have not been implemented widely, the number of cockpit crew members allowed to operate TaxiBots is limited to none. As aircraft from various countries and airlines arrive at AAS, it is impractical to gather data on which cockpit crew is allowed to operate with TaxiBots and which is not, hence this assumption is purely practically based.
- Next to that, it is also assumed that there will be enough operating crew able to operate TaxiBots. As operating crew is airport-based, the probability that enough crew is present to operate all TaxiBots makes this assumption more sound.
- Even though specifications of the TaxiBot can be altered for the purpose of this research, e.g. by means of a sensitivity analysis, technical alterations are not researched.
- The model will be based on VRP models from other literature as is explained in section 4.4, hence their assumptions will be taken over.
- Even though, data from 2020 will be available and would not be out-dated, data from 2019 will be used. As the current COVID-19 pandemic has altered flight schedules and aviation in general completely throughout the whole 2020, 2019 would be a more realistic scenario for the coming years, as was mentioned in chapter 2.
- As SAS is in the process of certification, a TaxiBot is not allowed to tow all aircraft types yet. Therefore, only certified aircraft will be towed by the TaxiBots.
- The number of TaxiBots is fixed per day, however variable for different days to be assessed.
- Even though only certified aircraft can be towed by the TaxiBots, all aircraft can experience delays in their operations.
- As this model focuses on the probabilistic effects of delays, the VRP model does not include an energy assessment of the TaxiBots. Therefore, an unlimited battery capacity is assumed for the TaxiBots.

3.6. Contribution

As the overall trend with making aviation more sustainable continues to intensify, new solutions tend to counteract the increasing greenhouse gas emissions emitted by the aviation sector. Electric taxiing solutions such as TaxiBot offer novel solutions for this problem, which need to be integrated into the airport operations. Vehicle routing problems with integrated TaxiBots have been researched, however all input data is deterministic. As lots of parameters tend to be not fixed in real life operations, this research tries to set a first step into combining the VRP with integrated ETS with the probabilistic aircraft delays which occur at airport operations. Below, the status quo and contributions of this research are stated.

Status Quo

- optimisation of a vehicle routing problem with ETV for costs, fuel consumption and emissions
- Determining the optimal number of ETV necessary at an airport based on costs, fuel consumption and emissions.

Contribution

- optimisation of a tactical vehicle routing problem with ETV including probabilistic aircraft delays.
- Effect of probabilistic aircraft delays on a time-space planning of a VRP with ETS.

4
Literature Review

The electric taxiing system, such as the TaxiBot application, touches many fields. Research on this topic is described in this literature review. TaxiBot is not the only eco-friendly taxiing method, many others have been investigated and developed as well. These are described in section 4.1. TaxiBot itself is described in section 4.2.
Ground operations have been modelled extensively with different optimisation methods, and in light of the research framework as described in chapter 3, the scope of the model design choice will be elaborated on in section 4.3. Above fields are combined in a literature review on a case study in section 4.4. Lastly, all open research fields will be discussed in section 4.5.

4.1. Taxiing Methods

Currently, taxiing of aircraft is done using the engines of the aircraft itself. Towing vehicles are extensively used already nowadays, however these have the sole purpose of push back. This push back is only a small part of the entire taxi phase. A number of environmental friendly taxiing methods have been described in literature, most being electrically powered. A differentiation can be made between operational and technical solutions, in which the latter can be split in on-board and on-ground systems. This categorisation can be seen in Figure 4.1 [39]. The three solutions highlighted in green present electric taxiing systems (ETS) which are further discussed below. This section is started with the reason for the extensive research in different taxiing methods.

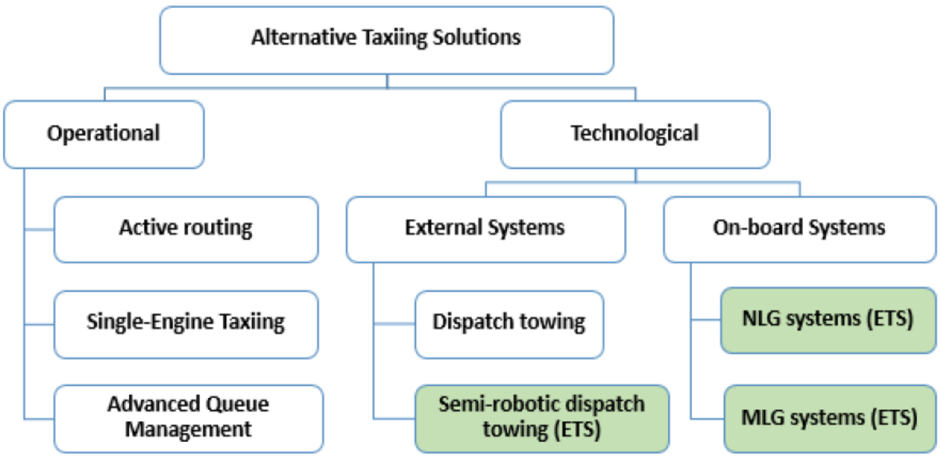

Figure 4.1: The proposed categorisation of alternative taxiing solutions [39].

4.1.1. Current Situation

Aircraft do not have a reverse gear, hence push-back vehicles are needed in order for the aircraft to leave the parking spot backwards [22]. After this procedure, aircraft are decoupled from the towing vehicle and the engines are started. Purely on engine-power, the aircraft taxi to the runway before the take-off. During taxiing, engine settings are not at 100%, but rather at lower throttle around 7%, which is the prevailing number according to the sources in the report by Ithnan [28]. As previously discussed, a high number of pollutants are emitted during taxi phase and taxi times range from only a couple of minutes to an hour, with outliers of longer taxi times. The longest average taxi-out time in the USA are experienced at New York JFK, with 37.1 minutes. The average taxi-out times was 16.7 minutes in the USA in 2007. [17] This time depends on many factors, e.g. the size of the airport hub, the efficiency of the ATC or the distance between gate and runway.

Operational measures such as the ones in Figure 4.1 are already implemented in many ways. Active routing and advanced queue management are optimisation models, which will be explained further in section 4.3. Single-engine taxiing is the method of simply using less engines during taxiing. By shutting down 1 or more engines, still enough thrust is produced to move forward, however less emissions will be generated. Even though, this seems a straightforward solution, it is not widely adopted according to Vaishnav [63]. Guo et al. [22] lists a number of obstacles of this solution. The most prominent is the added responsibility of this procedure to the pilots and its airlines. Secondly, this solution cannot be executed during special circumstances, which are uphill slopes, slippery surfaces and deicing operations and sharp turns. Hence, pilots tend to not perform a single-engine taxi operation if they are unfamiliar with the taxiing route. This could be because either they do not know the airport or there is no standard routing for aircraft at the airport [9]. A third obstacle that is brought forward is the increased risk of foreign object damage and jet blast, which is higher due to the fact that one engine now has to account for the thrust provided instead of, for example, two and thus a higher power is necessary. One last concern is of a broader scope and does not only relate to single-engine taxiing. Engines need a warm-up and cool-down period in which the engines are gradually prepared for their respective new state. General values used for these times are 2-5 minutes. [22] [33] This means that, from the total taxi time, there will always be a portion at which the aircraft engines are on. The longer the total taxi time is, the smaller this percentage is, however a 100% decrease cannot be reached. Research in a whole different direction is necessary in order to find alternative methods on how this engine warm up and cool down could be performed with shutting off the engines, as recommended by [64]. Only then, a decrease of 100% can be reached.

4.1.2. On-board Taxiing Systems

One of the solutions provided are on-board systems in which an electrical motor is installed in the landing gear, either the nose landing gear (NLG) or main landing gear (MLG). This motor is powered by the auxiliary power unit and thus the engines do not have to used anymore for taxiing. Furthermore, this makes the aircraft fully autonomous on the ground, as push back vehicles are not needed anymore as well. Lukic et al. [39] mention that the main drawback however is that these systems need to be added to the aircraft, which brings a lot of operational work as well as an increase on the aircraft weight. The former is explained as all the changes and adjustments necessary to the aircraft architecture, which is something aircraft manufacturers do not prefer. The latter means the addition of the electric motor weight, which could in the end nullify the decrease in taxi fuel by the increase of fuel flow during the airborne phase. However, another effect to be taken into account is that the reserve fuel allocated for unexpected delays during taxiing can be removed from the aircraft, reducing the total weight again [28]. Guo et al. [22] start the discussion on some studies performed on global fuel saving with on-board ETS. They mention two studies in which such a comparison analysis resulted in global fuel reductions. The first one is for mid-sized aircraft with a 500 kg on-board taxiing system, which shows savings up to 2.5% [46], the second one shows savings between 1.1% and 3.9% based on USA domestic flights in 2007 with an on-board solution weighing around 1000kg. A careful comparison will be given in subsection 4.1.4.

The trade-off between an electric motor on the NLG or MLG is based on multiple criteria. As the NLG has a larger space available, more freedom in terms of design is possible. However, as the MLG carriers around 90% of the aircraft weight, traction forces would be higher in this case, hence less strict design criteria come forward. Next to that, the number of wheels is higher on the MLG, hence more, and thus smaller, motors can be installed here. More requirements for the design of such a system are described by Lukic et al. [39]. Another important consideration is thermal management, according to Re [46], as the electric motor and the brakes are close to each other. These brakes can reach high temperatures and thermal influence to the motor is to be separated.

Two companies who are in the frontier of this solution development are Wheeltug and EGTS [28], [30] [39]. Wheeltug is an electrical solution to be placed on the nose gear of an aircraft. It has been tested thoroughly already and is certified for the B737 aircraft [28]. The systems weighs 130 kg and is able to gain speeds up to 9 knots or 16.7 km/h. Wheeltug has collaborated with the German Aerospace Centre DLR and Lufthansa Technik as well for further development of such a solution.
Another solution named Electric Green Taxiing System or EGTS is developed by Safran Landing Systems and Honeywell Aerospace. This system would be incorporated in the MLG and was designed to reach higher speeds. According to a presentation given by Messier-Bugatti-Dowty and Honeywell [41], the following requirements were stated:
"1) to achieve a maximum speed of 20 knots (37 km/h) for a time-window of 90 s;
2) to obtain a speed of 10 knots (18.5 km/h) in 20 s during active runway crossing;
3) to develop breakaway torque at full MTOW on a taxiway with 1.5% slope." [39, p. 7]
This project was terminated in 2016, however Safran continued on this project through its involvement in the Clean Sky 2 framework. [39]

The general idea behind all solutions presented is to shorten the time the engines are running, whenever this is possible. As mentioned before, the warm-up and cool-down phase result in the fact that not a full 100% electric taxiing operation can take place. The ground operation sequence as presented by Safran can be found in Figure 4.2. Even though this image is based on the EGTS system, this can be expanded to all other taxiing solutions as well.

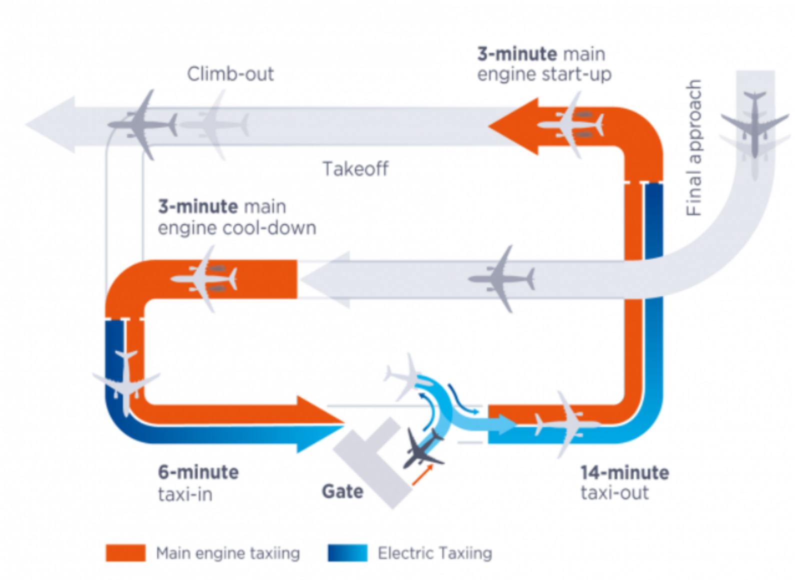

Figure 4.2: Visual representation of the different taxiing phases during ground operations, from Safran [53].

4.1.3. Electric Energy Storage Systems

In order to design such an electrical taxi system, both structural and energy systems come into play. Structural considerations for on-board systems could for example include prerotations of wheels before landing [27]. For ground solutions, the grasping mechanism to attach the nose wheel to the towing vehicle is one of such considerations. The latter will be further explained in section 4.2.

Overall design considerations that come into play are the energy mechanisms needed to actually transport the aircraft. As for on-board solutions, the amount of energy storage is relatively limited because of space and weight limitations. Placing an energy storage system next to the wheels will result in the addition of weight, while making sure this system fits in the landing gear. At the moment, electrical energy storage systems (EESS) such as batteries are still relatively heavy compared to other energy sources. Hence, other options are looked into as well. Battipede et al. [3] looked into the use of hydrogen as energy source for towing vehicles. Such autonomous tractors, called CHAT (Clean Hydrogen Autonomous Tractor), contain hydrogen storage tanks and would able to lift all types of aircraft, up to the super heavy class, including Airbus A380's. Asensio et al. [1] even looked into a combination of hydrogen and batteries and simulated such a system in order to size the electrical generation system on-board of an aircraft in an optimal way. Next to that, the operation methodology on energy recovery resulted in the sizing of the optimal system.

Lukic et al. have been investigating EESS both in sizing and optimisation of such systems in different papers. Lukic et al. first modelled an on-board electric taxiing system containing two traction motors on the main landing gear of a mid-sized commercial aircraft, such as a Boeing B737 and Airbus A320 [38]. This model included power electronics, an electrical machine and a mechanical drivetrain. After analysis, values on the acceleration performance, power and energy load profiles, as well as the energy harvesting stages were determined. This is done via a case study on an actual taxiing mission profile of a B737-400 aircraft. This resulted in the determination of the energy requirements for such an electrical system. Both taxi-out and taxi-in procedures were analysed and it was found that per one of the two motors 11.34kWh of energy is consumed. However, Lukic et al. estimated that 1.5 kWh could be regenerated, which equals to 13.2%. During the taxi-in phase, only 2.1kWh is consumed. In this case, 17% of energy could be recovered due to braking energy recovery methods. This equals to 0.35kWh.

On the other hand, Lukic et al. continued on these findings in another paper [37]. Taxi-out procedures at Amsterdam Schiphol Airport and taxi-in procedures at London Heathrow Airport are considered as a case study. From this, different energy requirements are determined for the local energy storage system (LESS), which can be seen in Figure 4.3. T-O and T-I stand for taxi-out and taxi-in respectively. the r-value is defined as the maximum potential regenerated energy E_{reg} over the total energy consumption E_{tot}. As taxi-out times are usually longer and the aircraft is heavier during this phase, the energy needed is higher compared to the taxi-in phase. Next to that, the power values are higher as well, which is explained due to the fact that the aircraft already has an initial taxiing speed during taxi-in and thus less break-away power is needed. In order to correctly size the LESS, the maximum depth of discharge is set to 80% and the efficiency is set to 90%. This results in an energy capacity of 18.7 kWh and a discharge power of 81.1 kW. One other important feature of the LESS is the charging time, which could be stringent for the planning. The LESS should be completely charged in between the itineraries in which the solution taxies one or more aircraft in and/or out.

	$P_{disch,max}$	$\lvert P_{ch,max} \rvert$	P_{avg}	E_{tot}	E_{reg}	r
T-O	73.2kW	51.2kW	22.7kW	11.3kWh	1.5kWh	13.2%
T-I	64.8kW	48kW	15.1kW	2.1kWh	0.35kWh	16.7%

Figure 4.3: Energy characteristics of the taxiing case study by Lukic et al. [37].

The second part, as discussed by Lukic et al. [37], is the type of energy storage device. Batteries and electrochemical capacitors are discussed and compared, as well as three different LESS topologies. Battery-only configurations, HLESS and a combination of LESS and APU is a third possible configuration. The first two options result in the electric taxiing power system to be completely autonomous from the rest of the power system of the aircraft. The latter option, however, results in a lower total weight as parts of the power of the APU could be used for taxiing. This last option is further analysed by Recalde et al. [47]. Here, a power distribution optimisation is determined in order to minimise fuel consumption. In the case study performed, only off-the-shelf batteries are used and three different energy management strategies were analysed.

4.1.4. Systems Comparison

Lots of research has been done on the comparison between all different electric taxiing solutions ([10] [15] [22] [25] [28] [36] [39] [43]). These analyses consist of technical; fuel consumption and emissions; and costs comparisons.
A quantitative analysis of all different solutions is done by Lukic et al. [36], [39], as can be seen in Figure 4.4. This paper has compared seven different criteria from six different ETS. Each of these solutions have been described before, however Lektro is another company focusing on electric pushback (EP) towing vehicles. As can be seen, the specifications of external solutions (especially the TaxiBot) differ from on-board solutions. The general trade-off between these two types of solutions is the choice between the most important factors,

Criteria	TaxiBot	LEKTRO	Wheel-Tug	DLR	EGTS	Safran*
System Configuration	External	External (EP)	On-board (NLG + Geared)	On-board (NLG + Geared)	On-board (MLG + Geared)	On-board (MLG + Direct Drive)
Estimated time to enter service	Operational since 2014	Operational since 1990s	2019	N/A	Stopped in 2016	2021-2022 [80]
On-board weight [kg]	-	-	130-140	N/A	400 (36 per TM)	320-380 est. (108 per TM) [109]
Max. power [kW]	500	90	N/A	50	120 (90 cont.)	120 (60 per TM) [109]
Max. speed [knots]	23	3.5	9	13.5	20	20
Towing capacity [t]	68-85 (B737)	127 (B757)	N/A	78 (A320)	78 (A320)	N/A
Cost	$1.5-3 million [87]	From $159,00 [38]	Power by hour [54]	N/A	N/A	<$1 million per aircraft [80]

*Data provided includes both e-taxiing and Safran/UoN projects

Figure 4.4: Quantitative comparison of the existing electric taxiing solutions [39].

being weight, power, costs, operations and possession. External solutions do not bring any on-board weight and have high power capabilities. However, these systems are bought by an airport for relatively high costs, and are thus only deployable on that specific airport. Airlines have to consider if adapting and licensing their aircraft is profitable, which mainly depends on the frequency of flights to this airport. On the other hand on-board solutions are cheaper to to buy and operate and are owned by airlines, while providing less power and having extra on-board weight. These systems, however, can be used at practically any airport by the airline. It is to be noted however, that different sources are given and thus the units of these specifications might differ. A proper comparison should not be read from this table one-to-one, rather a proper conversion to the same units should be done in order to quantitatively compare the different solutions.

The technical comparison done by Ithnan et al. [28] was based on two case studies performed at Amsterdam Airport Schiphol and Kuala Lumpur International Airport. Calculations done on the fuel consumption and emissions resulted in a comparison of single-engine taxiing, operational towing and electrical nose gear with respect to conventional taxiing. It should be noted however that operational towing was not fully electric and still produced pollutants.

The results for Schiphol can be found in Figure 4.5. The percentual changes for Kuala Lumpur International Airport were less, which could be explained by the fact that taxi distances and thus taxi times were shorter. In the table, the comparison is based on the same specifications and units, hence a clear comparison stands. Single-engine taxiing is relatively the worst solution, even though it reduces fuel consumption and emissions with approximately 26%, which is in line with the calculations shown in [9]. Operational towing and electrical nose gear perform better than single-engine taxiing, and electrical nose gear seems to even win this comparison. However, as mentioned before, not fully electrical operational towing is taken into consideration, hence these percentages for an electric solution can be higher than any of these three proposed solutions. Bresser and Prent [8] did calculations on sustainable operational towing and calculate the savings at ground level to be between 50-85%. Moreover, they calculate the savings per taxiing minute to be 95%.

Continuing on the fuel consumption and emissions analysis Guo et al. [22] and Pan et al. [43] both have compared different ETS. Guo et al. [22] has analysed ten different airports and four different taxiing methods. Each of these airports were located in the USA. Large reductions in fuel consumption and emissions are visible, in which either a towing solution or on-board solution performs best, depending on the evaluation criterion. The reductions, mainly in fuel burn, HC and CO emissions are very large; decreases of over 90% are present. Again, the towing solution considered is seen as a hybrid diesel-electric vehicle. In the case of fully electrical powered, reductions would be even higher and this solution is expected to outperform on-board solutions on all four evaluation criteria.

Lastly, Pan et al. [43] performed a quantitative and qualitative analysis of the different solutions. The quantitative analysis is composed of calculations based on all flights during one week at Beijing Capital International Airport. Again, reductions in mainly CO_2 and CO reach over 90% in decrease.

Strategy	A	B	(A-B)%	C	(A-C)%	D	(A-D)%
Fuel consumption aircraft (ton)	1073	793	-26,1 %	682	-36,5 %	633	-41,0 %
Fuel consumption tug/ mass (ton)	N/A	N/A	-	115	-	140	-
CO_2 emission (ton x 1000)	3,39	2,51	-26,1 %	2,51	-26,0 %	2,00	-41,0 %
HC emission (ton)	2,59	1,91	-26,2 %	1,49	-42,6 %	1,34	-48,4 %
CO emission (ton)	23,25	17,15	-26.2 %	13,97	-39,9 %	11,91	-48,8 %
NO_x emission (ton)	4,69	3,47	-26,0 %	4,80	2,5 %	3,28	-29,9 %

Note:

A : Full-engine taxiing
B : Single-engine taxiing
C : Operational towing
D : Electrical nose gear

Figure 4.5: Results of the taxiing strategy performance and percentage change compared to strategy A at Amsterdam Schiphol Airport [28].

Table 4.1: Overview of literature on ETS comparison.

Paper	**Type of comparison**	**Characteristics**
Lukic et al. [39]	Technical	Comparison of 7 ETS systems. The data is taken from various different sources, hence the units or the way of generation might differ. Absolute comparison might need conversion beforehand.
Lukic et al. [39]	Pollution, fuel consumption, time and money	These figures have been collected via the combination of studies from multiple papers. The TaxiBot considered uses diesel as energy source and thus an all-electric TaxiBot is not considered. The latter might reduce the presented criteria in percentages even more. Same note on absolute comparison as above. Reductions are given for various different pollutants, but all show high reductions.
Ithnan et al. [28]	Pollution and fuel consumption	The research is done via VRP case studies on Amsterdam Airport Schiphol and Kuala Lumpur International Airport. Here the diesel version of the TaxiBot is used as well. Fuel consumption and emissions reduction for all other scenarios from 25 to 50%.
Guo et al. [22]	Pollution and fuel consumption	10 different airports have been considered in this research, all located in the USA. The TaxiBot is considered as a hybrid diesel-electric vehicle. Decreases of over 90% for fuel consumption, HC and CO.
Pan et al. [43]	Pollution and fuel consumption	Research based on a case study on Being Capital International Airport. Reductions in CO2 and CO of over 90% compared to the conventional scenarios.
Pan et al. [43]	Pros and cons	Advantages and disadvantages of three ETS are given. Common features of ETS are given as well.

4.1.5. Cost Analysis

Hospodka [25] has made a cost and benefit analysis of electric taxiing systems. As not all of these ETS work in the same manner, it should be noted that this analysis is mostly fitted to on-board electrical solutions, however appropriate costs and benefits can be added or removed in order to perform such an analysis on ground solutions such as the TaxiBot. The benefits of such an electric taxiing solution are fuel savings, push back savings, time savings, fuel savings resulting from smaller quantities of the transported taxi fuel, engine life and maintenance savings, foreign object damage savings, lower emissions savings or any other special savings [25]. This could include better usage of space, reduced airport fees and noise reduction. On the other hand, such solutions also cause added costs. The added weight of the on-board solution will lead to an increase of fuel consumption. Other costs are additional tyre wear-out, additional maintenance costs, delay costs as well as one-time expenses. Hospodka also performed a case study in which all costs and benefits for a conservative set of parameters are put together. These savings per flight cycle are determined for different turn around times (TOT) and flight times and are plotted in Figure 4.6. As can be seen, such electric taxiing solutions result in savings if the taxi time is at least 5 minutes. As a TaxiBot does not increase the aircraft weight and is ought to use electrical energy, the savings are expected to be even higher. Lastly, Wijnterp et al. [66] have analysed the costs and benefits of electric taxiing systems in a broader sense. Via a value operation methodology the impact of such ETS on a number of other value drivers is investigated. Here it is concluded that fuel and emissions, maintenance, time benefits, ground operations are important value drivers. Operating times is seen as the most important utilization driver.

Time of taxi TOT in minutes \ Flight time in minutes	20	40	60	80	100	120	140	160	180	200
60	534,6	532,8	531	529,2	527,4	525,6	523,8	522	520,2	518,4
55	489,9	488,1	486,3	484,5	482,7	480,9	479,1	477,3	475,5	473,7
50	445,2	443,4	441,6	439,8	438	436,2	434,4	432,6	430,8	429
45	400,5	398,7	396,9	395,1	393,3	391,5	389,7	387,9	386,1	384,3
40	355,8	354	352,2	350,4	348,6	346,8	345	343,2	341,4	339,6
35	311,1	309,3	307,5	305,7	303,9	302,1	300,3	298,5	296,7	294,9
30	266,4	264,6	262,8	261	259,2	257,4	255,6	253,8	252	250,2
25	221,7	219,9	218,1	216,3	214,5	212,7	210,9	209,1	207,3	205,5
20	177	175,2	173,4	171,6	169,8	168	166,2	164,4	162,6	160,8
15	132,3	130,5	128,7	126,9	125,1	123,3	121,5	119,7	117,9	116,1
10	87,6	85,8	84	82,2	80,4	78,6	76,8	75	73,2	71,4
5	42,9	41,1	39,3	37,5	35,7	33,9	32,1	30,3	28,5	26,7
0	-1,8	-3,6	-5,4	-7,2	-9	-10,8	-12,6	-14,4	-16,2	-18

Figure 4.6: Cost savings (in euros), without any direct savings or any direct costs, per flight cycle for different turn around times (TOT) and flight times [25].

Important to note is that the cost findings by Hospodka might give a distorted picture. As the direct savings or direct costs are not taken into account, the final cost benefit might turn out remarkably higher. delay costs might be as high as 20 to 40 dollars per minute [59] and energy costs can differ as well. Kerosine, electricity or even biofuels[1] all have different costs, so result in different savings as well. As biofuels are expected to be more costly than conventional fuel, the gain for airlines and airports with using TaxiBots becomes even higher. Soltani [59] has determined these overall costs based on Montreal airport, which are in the range of tens of thousands to millions. The total costs when using two different number of TaxiBots when compared to using no TaxiBots over a time period of 7 years can be found in Figure 4.7. Defining these scenarios are based on finding the minimum total costs, as is explained in subsection 4.2.2.

[1] https://boeing.mediaroom.com/2021-01-22-Boeing-Commits-to-Deliver-Commercial-Airplanes-Ready-to-Fly-on-100-Sustainable-Fuels, accessed on 26-02-2021

	Scenario 1 (18 Tugs)	Scenario 2 (12 Tugs)	Scenario 3 (Engine On)
Fuel Cost	–	$6,210,299.00	$104,906,066.32
Energy Cost	$41,616.41	$38,813.00	–
Emission Cost	–	$6,612,340.00	$96,662,037.50
Delay cost	$8,949,886.59	$2,018,273.71	–
Purchase cost	$14,400,000.00	$9,600,000.00	–
Operation Cost	$20,158,336.80	$13,438,891.20	–
Total Cost	$43,549,839.80	$37,918,617.50	$201,568,103.82

Figure 4.7: Total costs of operation for three different scenarios over a 7 year time period for Montreal airport.

4.2. TaxiBot

Each of the electric taxiing systems have been elaborately discussed, however the most prominent solution for electric taxiing systems is the TaxiBotő, developed by Smart Airport Systems SAS[2]. The development of this product is originated after the collaboration of different companies. Israel Aerospace Industries IAI owns the concept and has developed most of the software, while Teleflex Lionel-Dupont (TLD) is the industrial partner and manufacturer. SAS is, while being the architect of this product, the joint venture of these two companies [4]. Furthermore, collaborations with aircraft manufacturers (Airbus, Boeing), airports (Frankfurt am Main Airport, Amsterdam Airport Schiphol, New Delhi International Airport) and airlines (Lufthansa Leos, KLM, Transavia, EasyJet, Corendon) lead to extensive testing[3]. The latest test that has been performed took place at Schiphol in Q1 to Q3 of 2020[4]. A photo of the TaxiBot taxiing an aircraft at Schiphol can be seen in Figure 4.8 [5]. Over 150 missions have been performed, of which nine were actual live flights. Other missions that have been performed were training missions and the testing and validation of uncoupling points. Hence, the testing consisted of determining the feasibility of TaxiBots in the whole of ground operations at Schiphol. Therefore, apart from the airlines involved, other essential parties were present as well. LVNL as Air Traffic Control party, ground handler parties such as KLM Ground Services, Dnata and Schiphol Operations, and governmental parties such as the ministry of infrastructure and water management and the human environment and transport inspectorate were all involved as well. Preliminary conclusions drawn from this proof of concept test were the following [5, p. 16]:

- *Staff required: Big effort of both airlines and ground handlers is needed;*
- *Dimensions of TaxiBot: Too wide for service roads;*
- *Infrastructure: Service roads too small for TaxiBot;*
- *Procedures: Hard to reach places throughout the airport due to one-way traffic;*
- *Procedures 2.0: TaxiBot technology is relatively new so airport operations are not adapted to its use;*
- *Procedures 3.0: Disconnection points leading to changes in airport flow* [5, p. 16]

These conclusions can be split up in two, where the first part considers the structure of the airport, which has multiple limiting factors due to the constraining layout. The second part considers the operational side, where it is concluded that the procedures for all involved parties are not working well yet and need to be improved. The final goal of Schiphol is to combine the use of TaxiBots with an automated process [11]. Vehicle routing problems come into play for efficient routing, which will be explained in section 4.3. Furthermore, based on previous tests, SAS claims to cause positive effects in fuel consumption, emissions and other beneficial effects[5]:

- *Up to 85% reduction of fuel consumption during taxiing;*
- *Up to 85% reduction of CO2 and other noxious emission during taxiing;*
- *60% reduction in noise pollution;*
- *50% reduction of FOD per takeoff;*
- *Improves gate efficiency through the reduction of wasted time during engine start-up at the gate area, which not only affects the gate used, but also the nearest airplanes to the gates.*[4]

These figures tend to comply with the general findings from literature as described in subsection 4.1.4. The recent tests conducted at Schiphol will contribute to composing above effects with more significance.

[2] https://www.smart-airport-systems.com/solutions/TaxiBot/, accessed on 03-12-2020
[3] https://www.TaxiBot-international.com, accessed on 03-12-2020
[4] https://news.schiphol.com/schiphol-and-partners-to-begin-sustainable-aircraft-taxiing-trial/, accessed on 03-12-2020
[5] https://www.smart-airport-systems.com/solutions/TaxiBot/, accessed on 03-12-2020

Figure 4.8: TaxiBot taxiing an aircraft at Schiphol during the Proof-of-Concept test in 2020 [5].

4.2.1. Technical Specifications

Currently, normal push back vehicles tow the aircraft from the gate to a position in which the aircraft can taxi to the runway on its own, however a TaxiBot can extend this towing procedure and can take the aircraft from the gate all the way to the runway or the other way around. At first, it seems that a normal push back truck could do the same, however constraints regarding the nose landing gear come into play. As Dirk en Bresser say, *these tractors can move a full plane over a very short distance, or an empty aircraft over a long distance. But a combination of the two a full aircraft over a long distance at speed is not permitted, because it would damage the aircrafts nose wheel.* [10] Such a landing gear cannot cope with high forces exerted because of continuous accelerating and braking. The reason TaxiBot is able to perform such procedures is the difference in control compared to conventional taxiing. The TaxiBot lifts the nose landing gear (NLG) slightly and clamps it onto a freely rotatable platform, as can be seen in Figure 4.9 [26]. In this way, the pilot of the aircraft is able to steer and brake its own aircraft which is translated to movements of the TaxiBot. The clamping platform senses both rotational and longitudinal motions via a set of sensors, which is then transformed into control signals. These pulling and pushing forces are continuously monitored to not exceed the maximum loads on the nose gear in order to extend its life time. [46] In other words, TaxiBot can be seen as an external propulsion system. Even though the pilot controls the TaxiBot during taxiing, a driver is present in the vehicle to control it whenever no aircraft is coupled. [10]

Two versions of the TaxiBot are developed, one for narrow-body (NB) aircraft and one for wide-body (WB) aircraft. Only the NB version has been completed yet. Van Baaren [64] has defined three different towing vehicles. The specifications of these vehicles have been determined by calculating the necessary energy requirements. These requirements are based on the power demand in order to accelerate and move the aircraft forward and the kinematic forces that come into play during these manoeuvres. Energy required for preconditioned air and engine start up is also taken into consideration [49]. Technical specifications of these three versions can be found in Table 4.2. The weight of the medium version complies with the actual specification of the TaxiBot-NB weight, according to its official specifications sheet [61]. This is logical, as the defined aircraft that can be towed for this version are aircraft up to the A320 and B737 series [49]. The NB TaxiBot already received its certification for the towing of these two aircraft types [26]. As mentioned before, SAS only developed two versions of the TaxiBot and the wide-body version is said to be able to tow heavy aircraft such as the A380 [26], which complies with the super heavy version which is designed for the B747 and A380. The heavy version vehicle thus is an intermediate version, which is designed for aircraft up to the A340 [49].

The speed with which these TaxiBot drive when coupled to an aircraft is important, as it determines for

Figure 4.9: The four steps in which the NLG enters the vehicle turret and gets clamped securely in position [26].

Table 4.2: Specifications of the three defined TaxiBot versions. Adapted from Van Baaren [64, p. 31].

	Medium	**Heavy**	**Super-heavy**
Total mass [kg]	25,000	45,000	60,000
Battery type [-]	LiFe	LiFe	LiFe
Battery mass [kg]	7,000	14,000	21,000
Battery volume [m^3]	3.82	7.64	11.45
Battery capacity [kWh]	840	1,680	2,520
Maximum power [kW]	1,400	2,800	4,200
Maximum ASU/PCA power [kW]	436	783	783
Drive train [-]	4x4	6x6	6x6

a large part the difference in throughput and delays. Various sources mention speeds between 20 knots (10.3m/s) and 23 knots (11.8m/s). Hospodka [26] mentions a design speed of 20 knots, while TLD itself [61], Lukic et al. [39] and Hospodka [26] mention a maximum speed of 23knots. In comparison, conventional taxi speeds reach 30 knots(15m/s) [39] or even 31 knots (16m/s) [50]. Therefore, it is important to have both taxi speeds clear in order to perform a qualitative comparison.

4.2.2. Implementation considerations

The strategy of implementing a TaxiBot becomes relevant as soon as the taxi times are relatively long. As the aircraft engines still need their warm-up and cool-down phase, the taxi times should at least be longer than that. Advantages of this solution are the reduction in fuel consumption, emissions, field noise, operational time and costs. This latter can be sought after in maintenance, FOD risk and jet blast risk. [28] A disadvantage however is that the towing vehicle on the other hand can still produce pollutant emissions. Therefore, TaxiBot is, next to a diesel power engine, also convertible to a hybrid or even full-electric vehicle. As discussed in subsection 4.1.4, towing vehicles are already a better solution than the conventional taxiing on aircraft engines, however fully electrical vehicles will be highly beneficial and result in being the best solution compared to all others as discussed in subsection 4.1.4. [28] One real disadvantage could be the price of a TaxiBot, which is around 1.5 or 3 million for the narrow-body and wide body versions respectively [39].

These costs are taken into account when determining the number of TaxiBots necessary on an airport. As the acquisition costs of these TaxiBots are relatively high when compared to other ETS or towing vehicles, a sound analysis on the necessary number of TaxiBots is required. As analysed by Soltani [59], the total costs of operations at an airport, both the on-time capital expenditures and the yearly operating expenditures, do not only go down when acquiring more TaxiBots. This trend would show a negative parabola, meaning that the minimum costs can be found for a specific number of TaxiBots. At first, when only a couple of TaxiBots will be acquired, not all flights can be handled and the fuel costs and delays will still be high. Every added TaxiBot will result in a decrease in costs as the marginal benefits will outweigh the marginal costs. However, up to a certain point these marginal benefits are becoming less than the costs as the purchasing and operational costs become too high. The moment the marginal benefits and costs equal each other, the optimal number of TaxiBots is reached.

Determining the necessary number of TaxiBots can also be done in different ways, rather than minimum costs. Operational-wise, the necessary number is determined by the minimum needed to attain a feasible solution. The question, however, is when such a feasible solution is reached. One can aim at providing TaxiBots to all arriving and departing aircraft on the busiest day on the year, however this number will probably be too high for any other day. So if costs are considered together with feasibility, one might dislike this number. Vice versa, aiming at the calmest day in the year will only be feasible for one day in the year, however might be cheapest option. Hence the optimal number should be found considering multiple, or all, days of a year and considering feasibility and costs. Practically, many more aspects come into play when determining the number of necessary TaxiBots, such as the number of available operating crew, airport layout etc.

Another disadvantage of the implementation of these TaxiBots is the added movements on the airport. These vehicles have to move in between tows and such additional vehicle movements lead to an increase in risk of accidents.[26] Therefore, collision and conflict avoidance needs to be considered for these vehicles as well. TaxiBots could also make use of the service roads next to the taxiways. However, this option is airport dependent, as TaxiBots are wider than conventional cars or other airport vehicles and thus might not fit on these roads.

4.3. Modelling Methods

In order to optimise electric taxiing operations at an airport, different modelling methods can be used. Two of which are generally used in literature [2]. Mixed-integer linear programming (MILP) is the most used method in literature. The second method is meta-heuristics, this will be further explained in subsection 4.3.3. A MILP problem is often formulated in a mathematical way. A generalisation of this is the following: [21, p. 8]

$$\begin{aligned} &maximise/minimise\ c^T x \\ &Subject\ to\ Ax \leq B \\ &and\ x \geq 0 \end{aligned}$$

The first line represents the objective function which either should be maximised or minimised. x represents the decision variables and c^T are the constants. The second line represents the constraints the objective function has to adhere to and the third line shows the constraints given to the decision variables to be non-negative integers.

Two distinct problems can be solved using MILP, which are both used to model airport operations. The first one is the vehicle routing problem (VRP). This problem addresses the way vehicles, such as aircraft or towing vehicles, move around a defined area. Space is an important parameter in these types of problems. The second problem is focused on time. The second problem is called a gate assignment problem (GAP) or fleet scheduling assignment (FSA). Here the scheduling of two or more entities is optimised. This could be for example the scheduling of departing aircraft to towing vehicles or coupling arriving aircraft to an empty gate or even combined. Another method is using heuristics, which does not per se search for an optimal solution, but rather tries to find a reasonable solution in a shorter computation time. Each of these modelling methods are discussed below.

4.3.1. Vehicle Routing Problem

The first type of problem is based on scheduling the routing of vehicles over an airport. Plenty of literature has been written about this subject, all with their own type of objectives and characteristics. An overview of literature on VRP can be found in Table 4.3. The general idea behind all of these problems is to find the most optimal routing schedule by minimizing any of the relevant parameters. These parameters are all connected. Often times the taxi times and/or holding times are to be minimised, which can directly be related to the taxi distances. The longer the taxi times and distances, the more fuel consumption and emissions. Overall, each of these aspects can be monetized. Hence, a minimisation of one parameter often directly causes the minimization of another parameter.

Even though each paper performs such an optimisation problem differently, some general aspects can be seen. Next to the aforementioned objective functions, the outputs generated consist of a time-space diagram (as can be seen in Figure 4.10) and the comparison of the same parameter in the objective with respect to the conventional taxiing methods. Fuel consumption, emissions, delays, costs or number of conflicts. Some papers generate some particular outputs. Schiffer and Walther [54] for example compare their developed model with the Solomon benchmark instances.[6] These benchmarks are predefined problems and are often used to compare computation times of different algorithms. Lastly, one general finding in literature is the trade-off between the advantages of the reduction in fuel consumption and emissions with the disadvantage of the decrease in throughput. Because of the lower taxi speeds of the TaxiBot, taxi times take longer and the throughput of the airport is reduced. Roling, Sillekens and Curran [51] determine these delays per flight for different scenarios at AAS which can be in the order of minutes. There are some scenarios in which the implementation of TaxiBots does not decrease the throughput. Only in the cases in which one or more conditions are met, the capacity does not decline, according to Guillaume [21]. If TaxiBots are able to use service roads at the airport in order not to hinder the other aircraft or if TaxiBots are able to make up for the extra time necessary for the additional operations, the same capacity can be guaranteed. In the extreme case when the time interval cannot be reached or the costs become higher than the benefits, mentions Guillaume, the conventional taxi procedure in which the aircraft uses its main engines can be performed.

One other important aspect noted is the comparison of different TaxiBot versions implemented. Van Baaren [64], Guillaume [21] and Kroese [31]. Here, either two or three versions of the TaxiBot are assessed. The other related aspect is the number of TaxiBots. Some use a fixed number as input, while others try to determine the optimal number via the model. A third approach, which can be seen as a combined approach,

[6] http://www.bernabe.dorronsoro.es/vrp/index.html?/results/resultsSolom.html, accessed on 08-12-2020

compares two or three scenarios in which the number of TaxiBots is varied. Soltani [59] for example compares the scenarios in which no, a part and all flights are towed by TaxiBots. With this, the advantages of TaxiBots with respect to the conventional methods can be compared, however the exact number of TaxiBots necessary cannot be determined from that.

Lastly, Atkin et al. [2] summarise the literature on VRP. Both MILP and genetic algorithm (GA) methods are described and compared qualitatively. Frequent objectives and constraints are elaborated on. The objectives mentioned do not differ from the ones above, however a new insight is given in the multi-objective problems. Next to the minimisation of taxi times, a second objective such as penalising deviations from a previous schedule is mentioned as example. Even more objective functions can be combined and weighted accordingly. Constraints include among others considerations in the route taken, where the routes can be predetermined up to a complete free map. Secondly, constraints in the separation of aircraft and aircraft moving speeds can limit the problem as well. Next to that, timing constraints for arrivals and departures are mentioned. Gates but also towing vehicles have to be vacant in order to handle these aircraft, however aircraft have to reach the runway at a certain time as well. [2]. As mentioned in Table 4.3, six future directions have been described. The first one, consistency and comparability, tries to harmonize all approaches. This is attempted by setting up a repository with data sets for these vehicle routing problems. Secondly, three additions to the current models are proposed, being integration of other airport operations, uncertainty in the input data and environmental changes such as more gradual moving without constant accelerating and decelerating. These directions do make the model more complex, however also more realistic. Lastly, two restrictions are proposed. The first one is to add restricted stopping positions to the model, based on real life operations at an airport. Secondly, last-minute changes will always occur, and even though, one can make a more robust schedule, not every aspect can be taken into account beforehand. Therefore, Atkin et al. [2] mention to consider models in which changes are limited.

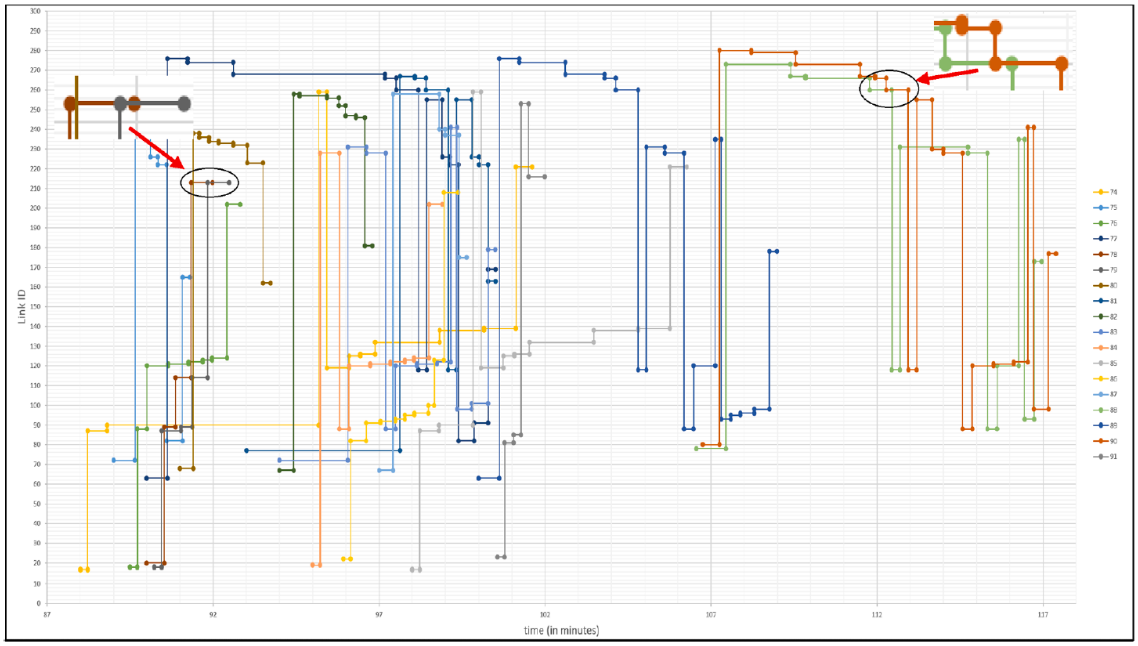

Figure 4.10: Time-space diagram output from Soltani et al. [59]. Each line represents an aircraft that moves in between the nodes and avoids conflicts.

Rolling Windows

These VRP models are quite extensive and the proposed future directions make them even more elaborated. Therefore, computation times can become relatively long and tend to increase even more with such additions. Multiple solutions have been used in order to decrease this computation time, however one algorithm has been widely used in literature. VRP problems can be solved all at once, however as these problems tend to follow a chronological timeline, they can also be split up in time windows. This reduces the large problem into a set of multiple smaller problems, which have a reduced computation time. The exact method to do so can consist of a rolling window or sliding window or any other algorithm. Smeltink et al. [58] propose three different versions of this method.

1. First, the total period is split up in a number of intervals. Each iteration, one of such intervals is solved in a chronological order. All aircraft with the starting time in the specific interval are scheduled and fixed, even if the total routing time of these aircraft overlap with the next interval. In Figure 4.11, aircraft a and c are completely fixed in the middle interval and only aircraft d would be scheduled.

2. The second variant looks like the first one, however now the aircraft are split up if they occupy multiple intervals. Hence, this breaks aircraft a into a1 and a2 for example. This means, less (parts of) of the routes of aircraft are fixed, so in Figure 4.11 aircraft parts a2, c2 and d would be rescheduled.

3. Thirdly a sliding window variant is proposed. Here, the problem is not split up in time intervals, but rather all aircraft are split up in aircraft batches, with size m. In every iteration, aircraft 1 up to m are scheduled after which the whole window slides one up, meaning that in the second iteration aircraft 2 up to $m+1$ are considered. Here, it is important to choose a batch size m that is small enough to decrease computation time considerately, however large enough in order to ensure aircraft separated by long enough period become almost independent. In other words, if the size m is too small, aircraft separated by only a small time interval can be dependent on each other, resulting in a not optimal schedule. [58]

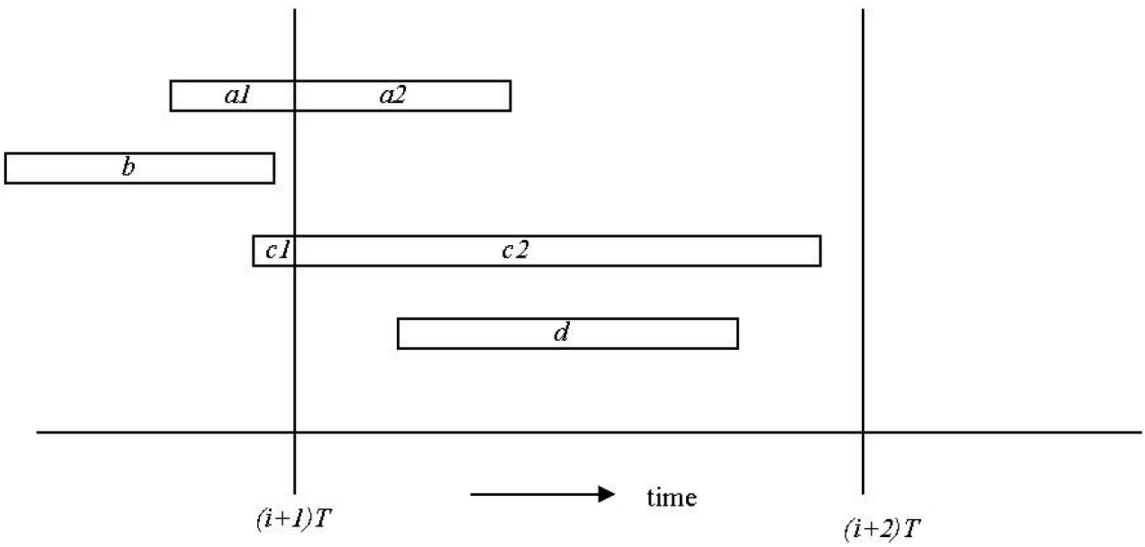

Figure 4.11: Schematic example used for the explanation of the two versions of the rolling windows, taken from Smeltink et al. [58].

Table 4.3: Overview of literature on Vehicle Routing Problems using MILP.

Source	Year	Model objective	Airport	Output
Van Baaren and Roling [49] Van Baaren [64]	2019	Minimum fuel consumption	AAS & RTM	Difference in taxi duration, fuel consumption and energy consumption
	Three types of TaxiBots have been defined for the towing tasks. The fleet size has been determined based on fuel consumption.			
Roling, Sillekens and Curran [51]	2015	minimise delay times	AAS	The delays per routed aircraft for different taxi speeds
	The effect of different maximum speeds on the taxi times and possible delays for different aircraft types. A number of different aircraft had to find the most optimal and where able to be rerouted.			
Guillaume [21]	2018	minimise taxiing costs	AAS	General taxi-out information and time-space graph
	Two full days have been analysed Two TaxiBot versions for NB and WB have been implemented Both fixed and flexible number of TaxiBots have been analysed			
Schiffer and Walther [54]	2017	minimise total driven distance	Solomon benchmark instances	A comparison with respect to the Solomon benchmark instances 6[0]
	Five different models are compared; each having one or more constraints added to the model: Electric Location Routing Problem (ELRP) and Electric Vehicle Routing Problem (EVRP) with Time Windows (TW) and Partial Recharging (PR) are introduced.			
Roling and Visser [50]	2008	minimise total taxi time and total holding time	example airport	Time-space table
	An example airport with one-way routes has been used to determine the shortest path, with the option of rerouting. A sample flight planning with limited number of aircraft has been used.			
Roling [48]	2009	minimise total taxi time and total holding time	AAS	Time-space table and number of conflicts over time
	Based on previous paper by Roling and Visser [50] as described above			
Yan Du, Brunner and Kolisch [67]	2014	minimise operating costs	unknown airport	CGH is better than a manual schedule The importance of selecting the right vehicles
	Column generation heuristic (CGH) has been used to decrease computation time. The number of TaxiBots, flights, depots and trips are fixed. The model combines time windows, mixed fleet, multiple depots and multiple trips.			
Soltani et al. [60]	2020	minimise fuel consumption and maximise the desired service quality	YUL, Montreal	Time-space diagram Optimum number of TaxiBots
	Collision of conflict avoidance has been taken into account in the model. Two cases with unlimited and limited number of TaxiBots are analysed. The optimum number of TaxiBots with respect to costs is determined as well.			
Evertse and Visser [14]	2017	minimise weighted combination of offset time, taxi time and emissions	AAS	Comparison in fuel consumption and emissions
	The model can update every 15s, hence unforeseen disturbances can be mitigated in time. Results were given for scenarios in which the weights of the objective function where changed.			
Sillekens [55]	2015	-	AAS	Visual simulation which showed a capacity decrease and fuel savings
	A predefined model named "SMARTlab" developed by Honeywell was used for this research. This model was based on the on-board ETS EGTS, which is different from the other literature described.			
Kroese [31]	2021	minimise total taxi time and maximise number of tasks and minimise battery charging	AAS	optimised routing schedule and TaxiBot fleet schedule with battery charging schedule
	The research consists of two parts, first a VRP and then a FSA. Input parameters on the TaxiBot are taken from Van Baaren [64]. Sensitivity analysis performed on the number of flights, taxi velocity, battery capacity and location of charging stations. An adaptation is proposed in which the charging time is limited			
Soltani [59]	2019	minimise operating costs	YUL, Montreal	A hybrid solution is the optimum
	Three scenarios covering null, part and all flights are towed are analysed and compared. The optimum number of towing vehicles with respect to costs is found by seeking the minimum. Multiple types of costs (fuel, energy, emissions, delay, purchasing and operation) are taken into account.			
Smeltink et al. [58]	2004	minimise waiting times while taxiing	AAS	Delay results and most suitable rolling window algorithm
	Three different types of rolling windows are addressed. One busy day with a real schedule has been used as input.			
Sirigu [56]	2017	minimise energy consumption	TRN, Turin MXP, Milan AAS, Schiphol	Comparison of different models, airports and algorithms
	The different models have been defined, one having continuous time and one having discrete time evolution. Different types of algorithms haven been used, being a particle swarm optimisation and a tree search heuristic.			
Atkin, Burke and Ravizza [2]	2010	various papers on the airport ground movement problem	-	Important future directions
	Thirteen different papers on VRP using MILP (8) and GA (5) are described, analysed and compared. Important future directions are described, being: consistency & comparability, integration of other airport operations, robustness and uncertainty, restricted stopping positions, environmental considerations in taxiing and limiting changes.			

4.3.2. Fleet Assignment Problem

The second type of problem is based on a time structure and does not per se take space into account. Lots of literature is written about the assignment problem in general, however the following literature only assess this problem on an airport level. Different types of assigning entities have been found.

The first is assigning aircraft to specific gates. This problem has been around for quite some time already, as the literature by Mangoubi [40] and Bihr [7] dates back to 1980 and 1990 respectively. Both try to minimise the average walking distance per passenger by assigning one aircraft per gate, where the aircraft and gate times are known in prior. More recent literature has been summarised by Dorndorf et al. [12], which provides four other possible objectives:

- *The number of un-gated (open) aircraft activities has to be minimised;*
- *Preferences of certain aircraft for particular gates have to be maximised;*
- *The deviation of the current schedule from a reference schedule has to be minimised in order to increase schedule attractiveness and passenger comfort;*
- *The number of expensive aircraft towing procedures (that otherwise decrease the available time for some ground service operations on the ramp as well as in the terminal) has to be minimised. [12, p. 327]*

These objectives can be labelled as comfort for passengers or convenience for airport services. Dorndorf et al. discuss various problem classifications (slot models, types of objectives and mathematical models) as well as state-of-the-art algorithms (mathematical programming techniques and rule based expert systems). Recent developments include among others multi-objective models. The authors present two models with different combinations of objectives and conclude with two open ends. The first is the development of solution techniques for such multi-objective models, the second is the investigation and development of robust or stable models that can handle uncertainty or perturbations, either deterministic or stochastic. [12] This will be further evaluated in subsection 4.4.6.

This problem can be extended with the addition of ETS. As mentioned in section 4.2, TaxiBots are allowed to tow an aircraft for a longer period of time or over a longer distance compared to conventional push back. This opens up the possibility for pit stops of aircraft which do not need to be at a gate specifically as explained by Roling et al. [52]. By towing these aircraft, which are delayed due to for example occupied departure slots, or occupied en-route and/or arrival slots, to buffer locations, the gates will be less congested and the utilization of the gates and aircraft increases. Other reasons for delay could be last-minute baggage loading or last-minute maintenance. Each of these does not require the aircraft to be at a gate and such pit stop operations could add a maximum of 25% extra flights scheduled with a pit stop [52]. This value is based on a minimum turn-around time of 170 minutes before a pit stop will take place and only 10% extra flights are planned on top of the normal planning. This maximum value of 25% goes down when more extra flights are scheduled, due to saturation of the gate schedule. One extra advantage of such operations is the possible decrease of push back time, which could be up to 1:50 minutes, according to Roling et al. [52]. Van Lingen [65] also concludes that pit stops increase the gate utilization, however at the costs of increased delays. Here, the average delay was determined for three different number of gates (4, 6 and 8) and three different turn around times (80, 100 and 120 minutes). Only for the highest turn around time, an increase in utilization was noticed.

A second problem is assigning towing vehicles to arriving aircraft. Van Baaren [64] assessed this problem after the optimisation of a vehicle routing problem. The objective set is to minimise the total fuel consumption of towed and non-towed vehicles and outputs are the differences in travel time, fuel consumption and energy used between conventional taxiing and fully electric towing. This problem resulted in the determination of the number of towing vehicles necessary as the marginal fuel reduction and power consumption was plotted per towing vehicle added. This analysis has been performed for two different cases; Amsterdam Airport Schiphol and Rotterdam-The Hague airport. Furthermore, analysis at AAS is done for three aircraft types; medium, heavy and super heavy. One day with full operations has been selected for the optimisation model, this day consisted of respectively 1273, 173 and 27 flights for the three different aircraft types. No clear breaking point was visible in the plot to determine the optimal number of electrical towing vehicles for the medium category as the lines gradually reach an asymptote, however the number ranges between 32-36. For the heavy and super heavy category this optimal number lies between 10-12 and 4-5 respectively.

Kroese [31] also researched this problem and combined the two MILP problems as well. First, a VRP model is produced based on Schiphol airport, which defines the optimal taxi routes of all towed aircraft within a flight schedule. Two days have been selected for this optimisation problem, one of the busiest and one of the calmest days of 2019. The total taxi times are minimised while the assumption states that there is an indefinite number of TaxiBots available for all operations for the first part. The output of this optimisation problem provides, per time stamps of 10 seconds, the location of each TaxiBot version. Here, the same version definitions as described by Van Baaren [64] are used. The second problem consists of a fleet scheduling assignment in which the number of necessary TaxiBots is determined. All tasks in the schedule need some energy to perform this, hence the battery level is considered as well. With this, any recharging periods are scheduled as well. A visualisation of such a schedule can be found in Figure 4.12. In total 77, 17 and 3 TaxiBots of the medium, heavy and super-heavy category respectively are necessary for these operations on one of the busiest days of 2019. 45, 13 and 2 TaxiBots are necessary on one of the calmest days of 2019. The charging stations necessary for the recharging of the batteries are placed at three tactically chosen locations on Schiphol, however the number and location of these charging stations has not been found to be optimised. This finding is more elaborately discussed in section 4.5. One other sensitivity test performed by Kroese is the size of the batteries. Logically, if the battery size is decreased, charging goes a lot faster, and due to less battery volume and thus weight, the TaxiBot might be able to reach higher speeds as well. However, the drawback is that less towing tasks can be performed per itinerary and more charging periods are necessary. The other way around, increasing the battery size reduces the number of times charging is necessary and might even extend the battery size such that only charging at night is necessary. By this, the towing tasks would consist out of just one itinerary in a day. However, taxi speeds might go down and charging times might become too long. Hence a trade-off is necessary to find the optimal battery size. This is discussed in the research gaps in section 4.5 as well.

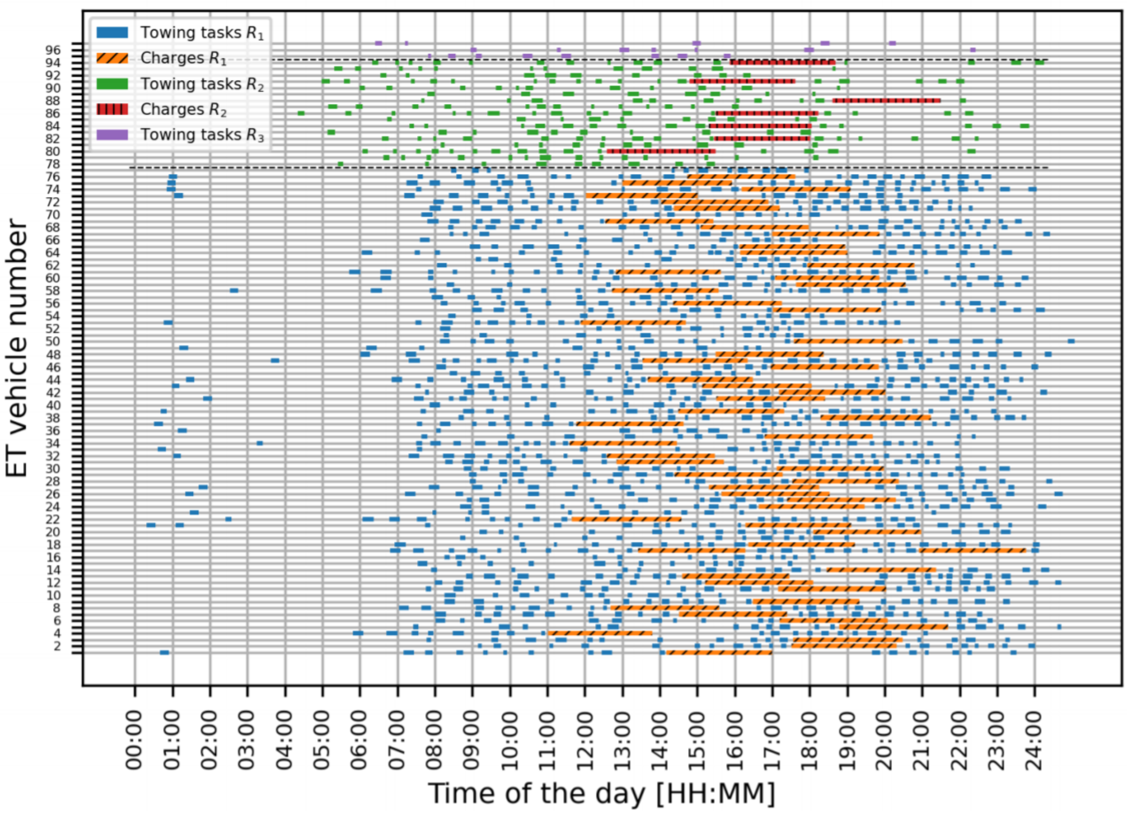

Figure 4.12: Schedule from the Fleet Assignment Problem developed by Kroese. [31]

As can be seen from the previous subsections on the two problems, a clear distinction is not always present, and often times both problems are combined, either in two separate steps or as a combined optimisation problem. Hiermann et al. [24] introduces the Electric Fleet Size and Mix Vehicle Routing Problem with Time Windows and Recharging Stations (E-FSMFTW), which is a combination of multiple optimisation problems based on a VRP. Each of these variants add new constraints to the VRP problem, the electric vehicles need charging, hence time and power constraints are added to the problem. Subsequently, the recharging stations necessary need to be placed strategically in the network. The location and number of stations is constraining the problem as well. A mixed fleet size changes the parameters of the problem set as well, resulting in different analyses for the mixed fleet. Lastly, time windows will make sure the vehicles are on the right place on the right time, ensuring even more time constraints.

One last subject that comes forward in literature on these two problems is the size of the fleet used in the models. The number of TaxiBots necessary for such operations is either set beforehand [21], or determined in the optimisation model ([20], [31], [59], [64]). The latter does this optimisation based on either maximum utilization ([31]), minimum fuel consumption ([64]) or minimum total costs ([59], [20]). How to find the optimal when looking at the total costs has been described already in subsection 4.2.2.

4.3.3. Meta-Heuristics

A second type of models that could be found in literature is the heuristic approach. While MILP tries to find the optimal model, heuristics does not necessarily end up with the global optimum as a solution, but rather finds a local optimum or an approximation. Different approaches used within this type are variable neighbourhood search (VNS), iterated local search (ILS) or adaptive large neighbourhood search (ALNS) [24]. This latter approach is used by Hiermann [24] in order to solve the electric fleet size and mix vehicle routing problem with time windows and recharging stations. Moreover, this approach was a hybrid solution method as it was combined with an embedded intensification mechanism. The results showed a solution which deviation from the best known solution with a gap of around 1%.

These aforementioned search methods can be subdivided into specific fields as well, as can be found in lots of literature as well, as is summarised by Dorndorf et al. [12]. Branch and bound and Tabu search are the two most used fields of search algorithms.

Genetic Algorithms

One other specific field of algorithms within meta-heuristics is genetic algorithms (GA). This approach is based on the process of natural selection, in which the population evolves and gradually becomes better over the course of multiple generations. Atkin et al. [2] summarise literature in which three approaches for GA are proposed, as summarised by Guillaume [21, p. 10]:

1. *GA determines for each aircraft initial delay/hold time prior to push-back.*
2. *GA determines a delay during movement, which is not restricted to a delay/hold time at the start of taxiing. It determines when and where delay should be applied.*
3. *GA investigates the possibility to prioritize aircraft instead of directly hold the aircraft. Priority determines the sequence of aircraft movement when there are conflicts.* [21, p. 10]

For each of these three approaches the GA delays and/or routes are allocated to each aircraft. The first approach lets the GA assign both a route and delay to the aircraft, while the second method is more freely with respect to when the delay will be assigned. The third approach focuses on placing the aircraft in the right order rather than allocating delays and is the most used approach in literature. [2] One other aspect Atkin et al. write about is the comparison of MILP and GA. GA do not guarantee to give the optimal solution, moreover an approximation of the solution is not always guaranteed. However, as MILP usually have a longer computation time, this brings problems to airports as they are usually seeking for a solution with real-time decisions.

Multiple papers apply GA on a vehicle routing problem, as is combined in the literature review by Kroese [31]. One paper that applies GA on a vehicle routing problem and also compares it to other algorithms is Gotteland et al. [18]. The defined model tries to minimise both the total rolling time and the extra time spent for rerouted trajectories. This combination is used so that the extra time for longer routes is taken twice, giving this a penalty which is twice as high than waiting in queues. The GA is compared to the A* algorithm and a third version combines these two methods. Then this model is run for two airports in France, Roissy Charles

De Gaulle and Orly airport and both airports result in the mixed solution being the best option. Furthermore, it could be found that GA outperform the A* algorithm in this model. Important to note is that this model only schedules aircraft without any ETS solution. Just like previous examples, Jian et al. [29] use genetic algorithms as well on a scheduling optimisation model. This VRP model is focused on aircraft priority and resulted in a decrease in taxi time. One can thus conclude that aircraft routing can be optimised with GA, however it needs to be determined if the implementation of this specific VRP with TaxiBots and probabilistic sudden changes can be optimised as well resulting in the global optimum or if a near-optimal solution would be sufficient as well.

Artificial intelligence

One other solving approach is the use of artificial intelligence. In order to solve a VRP for autonomous taxi operations, Sirigu et al. [57] propose four different solution mechanisms in order to determine which performs better. Two neural networks are proposed, the Hopfield neural network and a modified version of it. Next to that, two graph theory approaches are presented, Dijkstra's algorithm and A* algorithm. The different routes at the airport of Turin, Italy are used for 'on the fly'-generation of towing vehicles routings. The results show that all methods approach to the same optimum shortest path, except for the modified Hopfield neural network, which is slightly longer. However, the graph theory methods only need 0.01 s computation time, while the neural networks need a computation time in the order of 100s or even 1000s of seconds, making them several orders less efficient. [57]

4.3.4. Conclusion on Modelling Methods

The first approach is the mathematical optimisation via MILP, which has been split up in the VRP and FAP. Both problems have been modeled numerous times in order to find optimal solutions for electrical taxi systems. Often times, both problems are combined and even extended with other constraints based on time, charging etc. As said before, heuristics does not necessarily reach the optimal solution, but rather finds an approximation of the problem. However, when comparing the computation time of both approaches, big differences can be found. MILP usually has a lot of variables, in the order of thousands up to even billions, with a lot of constraints as well, in the order of hundreds to hundred thousands in general. As the solver has to go over each of these, or parts if special methods such as column generation based algorithms are used, the computation time can get impractically high. On the other hand, meta-heuristic methods do not cover all possible options, but rather explore interesting areas or directions. Therefore, computation times can go drastically down.
Concluding, the trade-off between the two approaches is based on the choice between reaching an optimal solution or having a low computation time. The model design choice can be found in section 5.2.

4.4. Defining the Case Study

In order to model the vehicle routing problem as defined in chapter 3, different information sources regarding daily apron operations need to be reviewed. This information will be used as input data for the model to be developed, focusing on a case study on Amsterdam Schiphol Airport. Therefore, first literature on each of the following topics will be reviewed, after which the most suitable option will be assessed for the specific case at AAS. First, different VRP modelling choices will be discussed in subsection 4.4.1, after which the input data for the nodal network, flight schedule and TaxiBot specifications will be assessed in subsection 4.4.2, subsection 4.4.3 and subsection 4.4.4 respectively. Lastly, literature on operational time uncertainty and scheduling robustness are two topics that come forward in the case study in subsection 4.4.5 and subsection 4.4.6.

4.4.1. VRP modelling

As is described in subsection 4.3.1 objective functions of a VRP tend to differ with respect to the decision variable to be assessed, while each of them are connected in some way. The most fundamental objective is to minimise travelling times, which implies the minimisation of fuel consumption, [49], emissions [14] and costs [67] [21] in turn. However, as the focus of this research is on the differences of the strategic and tactical solutions and the minimum deviations between these, the minimisation of taxi times is the most suitable objective to focus on. Next to that, the minimisation of deviations is an important objective function to take into account for the tactical VRP model. This can be done by minimising the costs received as penalties for deviations [60]. However, other non-cost parameters would be suitable as well. Other metrics that come forward in a time-space planning are for example the number of rerouting via different nodes or the increase in taxi time. Penalties could be set for such changes in metrics.
Using penalties to minimise is one way of doing so, however this brings one inherent problem with it. Even though penalties are to be minimised, a number of changes might still be present, depending on which metrics are used. If some small rerouting with a limited delay causes another rerouting with another limited delay, the overall delay and thus penalties is little, however two changes were needed. If the model focuses on the specific route to be rerouted only, only one change is necessary causing no other effects to the rest of the schedule, albeit that this change might result in a large delay for this specific vehicle. Therefore, a trade-off is necessary that focuses on minimising the total penalties or minimising the total number of changes necessary.

Next to that, the constraints necessary for such a VRP can be split up in their respective type as follows amongst others. [49]

1. **Flow constraints:** make sure that vehicles move from node to node and and use only one node at the time. Next to that, following travel directions are addressed here.

2. **Conflict and collision avoidance constraints:** make sure that vehicles keep a safe distance from each other and do not use the same node at moment in time. Next to that, they make sure vehicles do not travel towards each other on an edge.

3. **End node constraints:** make sure that vehicles start and end at respective nodes, such as gates and runways.

4. **Time constraints:** ensure that vehicles can only move in their respective time window.

5. **TaxiBoting constraints:** make sure that TaxiBots can only tow one vehicle at the time and one time maximally during a task.

6. **Aircraft specific constraints:** include among others the prohibited driving on service roads and taxiways.

Yan Du et al. [67] added the column generation algorithm to the model, which results in shorter computation times. However, in order to do so, a split is needed between a master problem and a sub-problem. This leads to the creation of another objective function and appropriate constraints. A pricing problem needs to be solved in the sub-problem which determines if another iteration is necessary in order to find the optimum in the master problem.

4.4.2. Airport Network

As can be seen from Table 4.3, many different airports have been used to model a VRP. AAS will be analysed in the case study, which reduces the literature slightly. Different nodal networks in terms of accuracy and comprehensiveness come forward. The airport can be split up in different layout parts. The runways and the apron and terminal with its gates are usually the start and ending positions of the routing. Runways contain multiple entering and leaving positions which all require a node. However, at the apron, gates from a specific service area or terminal can also be combined into one starting node. Taxiways are the connecting ways between these end points for taxiing aircraft and service roads can be used by other airport vehicles. Each edge connecting two nodes has a certain length and a certain maximum speed. Moreover, it might be the case that certain edges only allow one-way traffic, as can be seen in Figure 4.14. The network used by Smeltink et al. [58] has a simplistic taxiway structure, connecting the apron and terminal via 9 nodes, as can be seen in Figure 4.13. Both contain around 100 nodes.

Kroese [31] uses in his model the taxiway nodal network, but makes its model more complex by also incorporating another nodal network for the service roads. This network, containing around 50 nodes, can be used by ET vehicles in his approach. The validity of the assumption that the ET vehicles can drive on the service roads can be questioned, as the preliminary results of the TaxiBot feasibility test at AAS show that these service roads might be too small for the TaxiBot [5]. Schiphol has announced to expand on and build new service roads suitable for TaxiBots. Using the taxiways for TaxiBots as well as aircraft might not be feasible as well. Increasing the traffic on taxiways might be undesirable as ATC controls these roads, making the work load unbearable. However, as mentioned in section 4.2, Schiphol is aiming to have a full autonomous system in the end. With a dedicated fleet management system all TaxiBots can be controlled, i.e. allocated to an aircraft, control the speed and communication between the TaxiBots themselves. Concluding this, on the short term, service roads are used as well as taxiways in case the service roads do not suffice. This will increase the workload for ATC, however this will be decreased in the long term due to the addition of new service roads as well as the implementation of a fleet management system.

Van Baaren [64] has a nodal network of approximately 120 nodes. This model is slightly more extensive as it has nodes for multiple entering and leaving locations of the runways. All other literature assumes one entering or leaving node at each end of the runway, however the model by Van Baaren tends more to reality as runways have multiple entries and exits which are used dependent on the necessary runway length for take-off and departure. Adding these extra runway entries makes the model more complex as more nodes are necessary and information regarding the runway length needed per aircraft is needed. Therefore, the assumption is made that aircraft enter and exit the runway via one entrance/exit, i.e. one node.

Guillaume [21] uses a model that fits in between the previous discussed networks, as its nodal network contains 120 nodes of which approximately 20 refer to service road nodes. Moreover, this nodal network contains one more unexplained parameter. Some nodes are characterised as "holding nodes", at which aircraft and TaxiBots can wait without disturbing other vehicles. As this nodal network is the most suitable one regarding all aforementioned characteristics, it will be used in the case study. A visualisation of this network can be seen in Figure 4.15.

AAS consists of six runways in total, whereas the nodal network consists of only 5 runways. The Oostbaan 04-22 at the east part of the airport, as can be seen in Figure 4.14, is not taken into account. The reason for that is that this runway is used merely by general aviation and private jets and only in emergencies international flights are allowed here.[7] The number of international flight operations at the Oostbaan was in 2019 6058 flights (6023 landings and only 26 take-offs), while the overall number of flight operations concluded 497.303 in that year, resulting in a percentage of 1.2%.[8] Therefore, considering this slim percentage of use of this runway, it it not taken into account in the nodal network and therefore avoiding an unnecessarily more complex problem.

All parameters of the nodes in the nodal network are listed below:

- **Node ID:** Unique ID number per node
- **Position** The relative position in x- and y-coordinates, where the reference point lies approximately in the middle below the airport.

[7] https://www.schiphol.nl/nl/schiphol-als-buur/pagina/vliegroutes-en-baangebruik/, accessed on 19-02-2021
[8] https://bezoekbas.nl/wp-content/uploads/2020/02/Baangebruikcijfers_2019.pdf, accessed on 19-02-2021

- **Connecting Node:** Each node can have up to a maximum of four connecting nodes via the edges.
- **Distance Node:** Each of these edges from a node has a certain distance.
- **Speed Node:** Next to that, each of these edges from a node has a certain maximum speed.
- **Gate:** Each node has a binary value stating if the node is connecting to a gate or not.
- **Holding Node:** Each node has a binary value stating if the node is a holding node or not. At these holding notes, vehicles can wait while not interrupting other vehicles passing by.
- **Runway:** Each node has a binary value stating if the node is connecting to a runway.
- **Service Road:** Each node has a binary value stating if the node is a service road node or not. Each edge connected to a service road node is a service road.

Next to that, two other parameters necessary for the nodal network are the following:

- Each gate at the airport is connected to a certain gate node.
- Each runway entry and exit is connected to a certain runway.

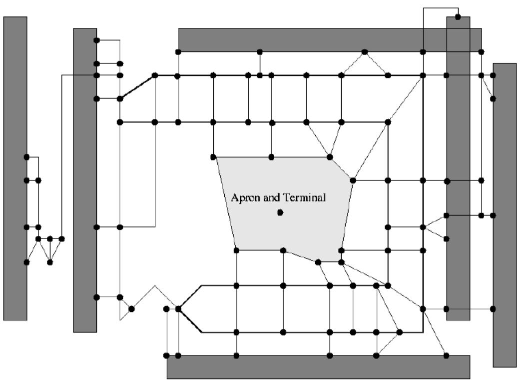

Figure 4.13: Simplistic nodal network used by Smeltink et al. [58]

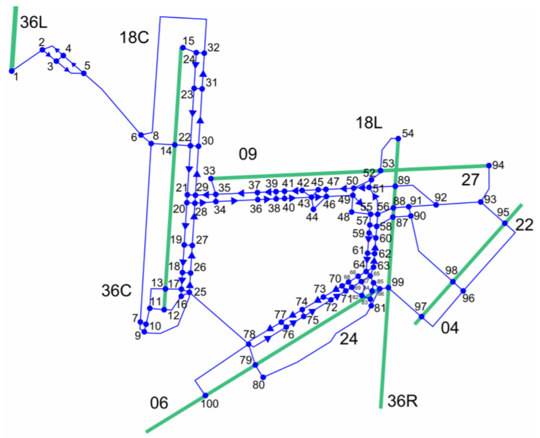

Figure 4.14: Detailed taxiway nodal network used by Kroese [31].

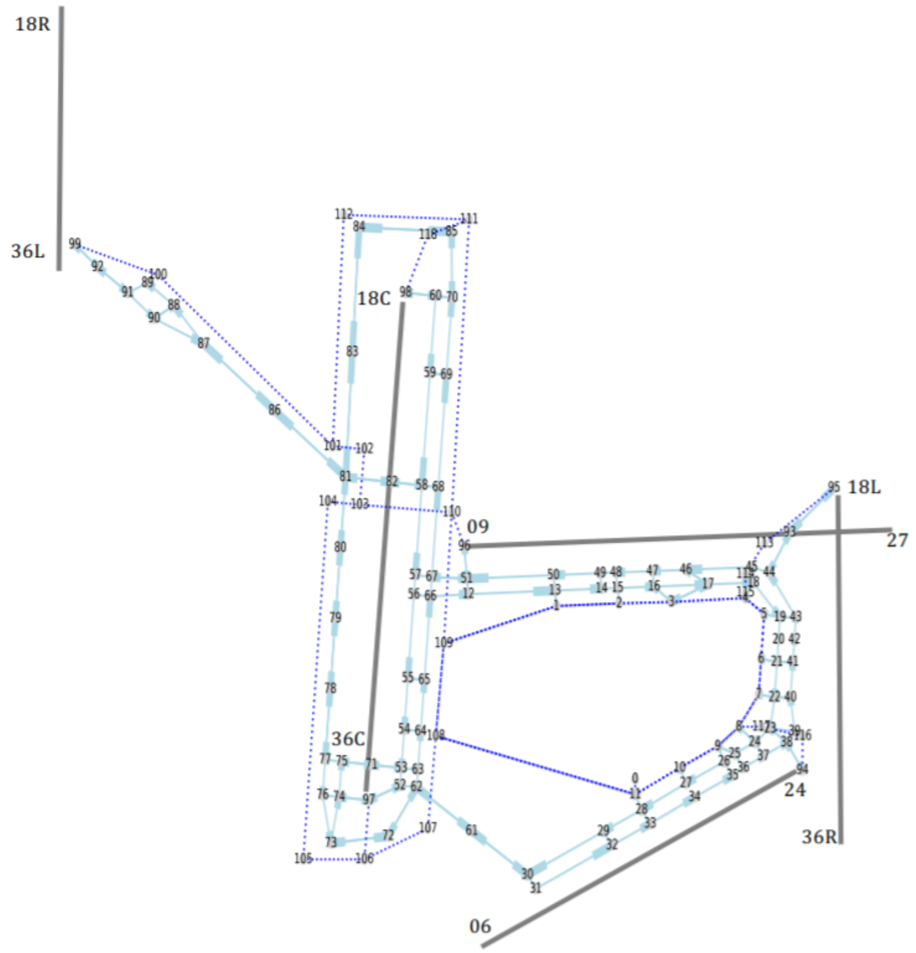

Figure 4.15: Visualisation of the chosen nodal network containing taxiways (green lines) and service roads (dotted blue lines), as well as travel direction (towards the thicker green line) by Guillaume [21].

4.4.3. Flight Schedule

Literature on VRP uses different sources for collecting the flight data. Data sources used are OAG, a global travel data provider[9] ([21]), FlightStats by FlightGlobal[10] ([64]), the website DutchPlaneSpotters[11] ([28]) or directly from AAS Flight API[12] ([31]). The parameters that are shown per source range from only the flight number and arrival time and other important parameters, up to a set of 35 parameters in the Flight API containing for example baggage belt or even check-in information. As the latter is the most complete and also directly available from Schiphol, it is the most suitable source to use. The most important parameters to be used from this source will be the following:

- **Flight Direction:** States if the aircraft is arriving or departing.
- **Flight Name:** Gives an unique ID to each flight.
- **Scheduled Time:** Gives the predefined time of operation.
- **Actual Time:** Gives the actual time of operation. In the model, this parameter will be determined via a probability density function, however in order to verify the model, fixed actual operating times will be used.
- **Service Type:** States the type of service of a flight, such as passengers, cargo or charter.

[9] https://www.oag.com/historical-flight-data, accessed on 10-02-2021
[10] https://www.flightstats.com/v2/flight-tracker/search, accessed on 10-02-2021
[11] https://schiphol.dutchplanespotters.nl, accessed on 10-02-2021
[12] https://developer.schiphol.nl/applications, accessed on 10-02-2021

- **Aircraft Type:** refers to the the aircraft brand and type such as B737. From this, the weight and certification can be distilled.
- **Gate:** Refers to which gate the aircraft goes and leaves from.
- **Airline:** This information might be useful when looking at the operability of the TaxiBot for the type of aircraft.
- **Status:** Gives the status of the flight. Cancelled or redirected flights to other airports are not taken into account.

Related to this data is the runway usage over time. By keeping track of which runways are used for arrival and departure, the arriving and departure runway and thus adjacent node of each flight can be derived.

Decisions on which day of the year should be analysed differ in literature. Kroese [31] focuses on one of the busiest and one of the calmest days of the year. Multiple reasons are give for this selection. A busy day shows the maximum capacity of the TaxiBots needed, in order to cover all other less busy days. A relatively calm day will be used as most certainly, TaxiBots will be introduced on such a quiet day. An important aspect to take into account could be the year assessment of such operations. The busiest and calmest days will reflect the minimum and maximum capacity needed, however the actual number of TaxiBots needed for a full year of operations to minimise the total costs might most certainly differ from the actual number of TaxiBots needed to minimise fuel consumption and emissions. Moreover, the average might not be exactly in the middle, but the weight of calm days might be higher than that of busy days. Therefore, an analyses of multiple days throughout the year can determine the capacity in a better way. Guillaume [21] analysed two different days in the month May, and two different time intervals of an hour. The reason why these specific days and time intervals have choosen is not made clear in the paper. Van Baaren [64] focuses on one day in October, being described as an average day. Smeltink et al. [58] on their turn focus on half a day in the month of September, characterised as a busy day as well. One other aspect Kroese looked at was the runway availability. The two days were specifically choosen as multiple runways had been in use.

To conclude assessing multiple days with different parameters will result in a broader analysis of operations. Both busy and calm days will be assessed as well as other days throughout the year will be taken into account.

4.4.4. TaxiBot Specifications

As not all detailed specifications of the TaxiBot are publicly known, past literature has tried to mimic such a vehicle design. Van Baaren [64] has designed three versions of the TaxiBot by taking into account operational and technical design requirements, such as design weight, power demand, energy demand and kinematic performance. This design has been taken over by Kroese [31], which in turn improved the design by combining parameters from multiple sources. Kroese also compared parameters of electric taxiing with those of conventional taxiing. Lastly, the recent test of the TaxiBot at AAS resulted in new insights on relevant parameters [5]. Each of these sources will be used to model the vehicles in the VRP.

4.4.5. Operational Time Uncertainty

In previous literature on VRP, deterministic data is used in solving the problem, however this does not reflect real-life operations. Deterministic input is used both for the sake of keeping the problem away from unnecessary complex methods as well as the large uncertainty in operational times. Arrival and departure delays are not uncommon and aircraft taxi times are difficult to predict as well. Novianingsih and Hadianti [42] try to model flight departure delay into probability distribution functions, while Ravizza et al. [45] try to model statistically predict aircraft taxi times for A/D without the inclusion of delays. The latter proposes to include this prediction of taxi times into a VRP. If a conflict occurs in the routing of the aircraft, some delay or rerouting with a foreknown taxi time can be added to the problem, in order to accurately predict the new travelling time.

The third part of the model will take into account the uncertainty in arrival and departure times. This will be done by letting the operational times follow a probability density function. Novianingsih and Hadianti try to find such a PDF for aircraft departures, however aircraft arrivals do not necessarily have to follow the same PDF. As different operations occur before the arrival and departure, the PDF tend to look differently. One

other difference that is noted in the paper is the distribution of departure delay duration and the distribution of departure delay-time. The first one assesses the delays per aircraft, however the latter focuses at the distribution of delays occurring over a period of time, e.g. one day. [42] Other parameters to filter on could be for example airline, aircraft type or even airport specific. In the paper, the optimal PDF is found by letting a GA maximise for the log-likelihood function and minimise for the sum of the squared error. It is concluded that of the three best options, Log-normal, Rayleigh and Gamma, Log-normal is the best suitable one. This conclusion is drawn as well by Lan et al. [34], who statistically tested gamma, log-normal and Weibull distributions. However, as previously noted, this PDF depends on many different parameters. Therefore, it would be most realistic if empirical data from AAS is used. Moreover, the presented paper only addressed departures from Garuda Indonesia Airline. By getting all scheduled and actual operational times, the number of delays can be plotted per small time interval of eg. 2 minutes and converted into a PDF. A split can be made between arrival and departure and investigation is necessary if even more different PDF's with more detailed divisions are worthwhile.
An important note to make is that above PDF derived from Novianingsih and Hadianti are on an infinite range, which makes the problem impractical. Therefore, the assumption will have to be made to truncate such PDF into a finite range. Extreme delay values that are located in the tail of a PDF, e.g. outside the 95% confidence range will be regarded as outliers and assumed to have a probability of zero instead of a very slim probability.

4.4.6. Scheduling Robustness

In order to minimise any deviations from the strategic schedule in the tactical one, the first one can be made more robust. By this, small deviations would not affect the schedule that much and operations can continue as planned. Lots of literature is available regarding robustness methods, which is summarised by Kumar and Bierlaire [32] to include flexible buffer times between operations, use stochastic delays as explained before in subsection 4.4.5 or use time windows. The buffer times in their paper are set by maximising the gate rest in between operations. This is done by accounting the kth percentile delay of empirical delay data and adding this to the gate rest. The maximum of both arrival and departure at the specific gate is chosen and added if feasible. [32] Dorndorf et al. [13] proposed two other methods. The first one is making sure that there are overlapping operation slots which can be swapped if necessary, the second method makes use of a fuzzy set approach.

4.5. Scientific Research Gaps

Following from the literature review, a number of scientific gaps can be pointed out. Previous literature can always be improved, however new research directions can also be determined. Some of the most important ones are discussed below.

The first research direction refers to the battery of the TaxiBot. Next to ongoing research on battery improvements, making TaxiBots more powerful and capable of carrying more tasks, research in the battery size should also be considered. Many different aspects come into play when sizing the battery. The total energy capacity, charging time and charging capacity are some of the most stringent one. The first two aspects are related. One can size the battery of the TaxiBot relatively large, which makes sure that a large extend of the operations at an airport can be carried out, however the downside to it are the long charging times. Vice versa, one can opt for short charging times, however only a limited number of operations can be carried out in such a scenario. Following on that, charging capacity at an airport may be limited, which can obstruct the possibility to charge all TaxiBots at once, for example during calmer night times. A more spread out charging schedule might be necessary, resulting in another constraint for the battery sizing.

Continuing on battery capacity, charging stations needed for the charging of TaxiBots can be a limiting factor as well. Too few stations can result in waiting times for the TaxiBot before they can be charged or in too long distances for the TaxiBot to realize charging in time. Hence, charging stations location placement is a different optimisation problem which has to be solved. A solution for the stringent battery constraints could be battery swapping. A concept in which the TaxiBots and the batteries are separated, a higher number of batteries can be acquired, while a lesser number of TaxiBots is necessary. Spare batteries can be charged and swapped with empty batteries if necessary. optimisation in the planning of these operations is a different research direction to pursue.

One different research direction to pursue is the feasibility study of TaxiBots at an airport. As determined before, large airport hubs benefit from the addition of ETS, looking at cost and sustainability parameters. However, smaller airports or structurally complicated airports might not obtain any advantages. Airports which are located in relatively cold regions might have temperatures too low for a TaxiBot to properly function. Hence, a study regarding the feasibility with respect to all relevant parameters is an interesting direction. Not only will the determined which parameters are important to take into account when considering implementing ETS, the range and relative importance are important aspects to look at as well. These parameters might be structural considerations, such as the layout of service roads or the distance to runways, but also operational ones, such as the availability of certified operating crew, maintenance possibilities or even the aircraft types that usually land or take off from the specific airport. Such a feasibility study can be done in a generic way, but also specifically tailored on an airport to be considered.

Each one of these scientific gaps can be explored with the view focused on TaxiBots as these type of ETS turned out to be the best, according to the comparisons mentioned in subsection 4.1.4. However, TaxiBots do have its limitations as well, therefore exploring further into the implementation of other electric taxiing systems is a final scientific gap that can be concluded.

5 Methodology

In order to adhere to the research aim set, a computer model will be built which will consist of three parts, containing two types of optimisation models. First, an explanation of the model is given in section 5.1, which is visualised with a functional flow diagram in section 6.1. Thereafter, the model design choice, experimental setup and expected results are discussed in section 5.2, section 5.3 and section 5.4 respectively.

5.1. Model Explanation

The first part will be a vehicle routing problem on the routing of taxibots on Amsterdam Airport Schiphol. The VRP will have only fixed inputs as are shown in Table 5.1 and a set of requirements as specified in section 5.2. The objective of the model will be to minimise the taxi time between the origin and destination, e.g. from gate to runway or vice versa. The taxi time depends on the velocity of the taxibots and on the distance between nodes. Only certified aircraft can be towed by the TaxiBot, other aircraft will taxi in the conventional way. The input of the problem will consist of a fixed nodal network of AAS, in which the taxiways, gates and runways of AAS will be configured. Next to that, a fixed flight schedule, based on real-world data will be used for the planning of one day of operations. The TaxiBots present will not be of any limiting factor in the first part of the problem, as a sufficient number of TaxiBots will be present, which are assumed to have unlimited battery power. The second part of the problem is focused on finding the lowest number of TaxiBots necessary to be still able to find a feasible solution. This output will be reached by lowering the number of TaxiBots through iteration. Then, in the third part, whenever the amount of TaxiBots has been defined a strategic and deterministic solution has been found. Then, on the day of operation, a tactical VRP will be run. Aircraft arrival and departure times become probabilistic and the expected arrival time will have a normal distribution around the scheduled arrival time. The shape of the probability distribution will be the split up as was explained in subsection 4.4.5. The day is split up in time intervals of e.g. 5 minutes. The optimal time interval has to be determined based on a suitable iteration time and the amount of time such arrival/departure information is normally known in advance by ATC. The model iterates through these time intervals and reruns the problem for every next interval. In each interval, the input flight schedule might be altered as follows. As soon as a part of the probability distribution falls in this 5 minute interval, the actual delay is determined by a random draw and fixed. Deviations in the arrival or departure time will occur and the updated flight schedule will result in a new vehicle routing solution. This actual arrival time might be in the current 5 minute interval but might also be in the next or second next interval. After this fixation of the exact time arrival, the new schedule is calculated for the rest of the day in which deviations are minimised. Hence, if all aircraft arrive at the expected time of arrival, the schedule will not change, but a deviation can alter the schedule slightly or severely, based on the added positive or negative delay. These schedule changes are made on a tactical decision level. A visual example is given in section 6.1.

5.2. Model Design Choice

As is mentioned in subsection 4.3.4, the choice on the model design being MILP or GA lies on a trade-off between reaching an optimal solution or having a low computation time. Preferably, both advantages would have been implemented, however as that is not possible, another solution has to be found. Inherent to this specific problem, it is of utter importance to have a model which can generate a new model which is exactly the same, or as close as possible to the previous schedule in order for it to be practically useful. In other words, the schedule should be reproducible and thus finding the global optimum every run. As GA might find only a near-optimal solution, this is unwanted. This can be resolved by letting the GA run the problem multiple times, which would eventually lead to the global optimum. However then computation time constraints come into play again. MILP allows for easier implementation and will find a global optimum and is therefore the preferred method. This only holds if the computation time constraints are met.

Last but not least, one argument regarding the computation time needs to be addressed. Because VRP problems tend to be NP-hard, computation times tend to rise quickly [35]. Therefore, even though GA cannot be used, other methods for decreasing the computation time should be used. The aforementioned rolling windows in Figure 4.3.1 are such an algorithm to be used in the model. As a rolling window algorithm might result in a near-optimal solution instead of an optimal solution when computing a whole day of operations at once, this method can be justified as follows. As in the tactical model deviations in the schedule are to occur, there is no need to compute the rest of the daily schedule. As relatively long-term schedules tend to change, it is logical to focus on short-term deviations with the rolling windows. Hence, optimising for the whole day is not necessarily needed for this tactical model.

Therefore, the best way to answer the research questions is to model an optimisation problem and try to find the optimal solution. As such a research problem has thousands of variables and should adhere to a number of constraints, a vehicle routing problem is the most suitable MILP. The goal is to make a general model that can be applied to many various airports, however in order to test its validity, a case study will be performed in which data and information from one specific airport will be used, which will be Amsterdam Airport Schiphol. The reason why this airport has been chosen is because of its complexity. AAS has a lot of variables to take into account, making this an interesting airport to consider. There are multiple runways in use simultaneously, it is an important hub airport in Europe and the fact that tests with the TaxiBot have been performed here before make this a reasonable choice. Other literature has considered this airport in their models as well [21] [31] [64].

The initial strategic optimisation model and the reactive tactical optimisation model are part of one program, however their methodologies differ and thus their functional requirements are to be separated. First the requirements for the initial model are given. For this, state-of-the-art literature on MILP problems described by Dorndorf [12] are used as reference.

- The flight schedule used in the model shall match a real flight schedule of AAS.
- Conflict and collision avoidance of both vehicle types shall be in place.
- The airport nodal network model shall match the real airport structure of AAS.
- The TaxiBot specifications shall be as close to reality as possible.
- TaxiBots will have a central resting location and are always charged enough to perform the next task.
- The model's objective shall be a fitting optimisation function.
- The output of the model shall be given in visual graphs and textual tables .
- The output of the model shall have such a format so that it can be used as input for the tactical optimisation model.

Secondly, the requirements for the tactical optimisation model are the following:

- The model's objective shall be to minimise any deviations from the initial schedule.
- the input variable delays shall represent real-world occurrences.
- The model shall be able to generate results within the length of the set time interval of the iterations.
- The output of the model shall ensure the possibility of a proper comparison with the input of the model.

Concluding, the model will be designed to cover a lot of aspects. The random aspect makes this research different from deterministic optimisation problems in previous works [21] [31] [64]. On the other hand, the case study on AAS will try to cover the depth of the experiment as well by applying the model on an existing and complex airport. As an MILP optimisation algorithm will be used, the problem will be a deterministic one. The sudden changes in the variables, however, can be predefined or randomly generated. By making the initial schedule generated by the initial VRP more robust, random probability distributions on one or more input variables will have less effect on schedule changes.

5.3. Experimental Setup

The model will be written with Python 3.7 as programming language and Spyder as integrated development environment. Next to that, the commercial optimisation solver Gurobi version 9.1 will be added as software package.[1] This package is able to solve linear programming (LP), quadratic programming (QP), mixed-integer linear programming (MILP), mixed-integer quadratic programming (MIQP) and mixed-integer quadratically-constrained programming (MIQCP).[1] The reason why these three options have been chosen is the level of experience in each of these three software programs. The advantage of python and spyder is that both are free and open-source, making them easily available to use. Gurobi however is an commercial software package, but offers free academic licenses if used for research or educational purposes.
The model will need a number of input data and give a set of output results. The set of input data can be found in Table 5.1. Specifically for the tactical optimisation model, the bottom right input comes into play. This data will be gathered from their respective sources, e.g. AAS, SAS or LVNL[2], bas[3]. The optimal routing schedule will be referenced from previous VRP models. This will be either one or more outcomes from Table 4.3.

Table 5.1: Input parameters with their respective sources.

Parameter	Source	Parameter	Source
Airport structure information	AAS, online open sources	Gate scheduling over time	AAS, online open sources
Runway configuration over time	LVNL[2], bas[3]	Aircraft flight schedule over time	AAS API
Airport routing network	Guillaume [21]	Certified aircraft information	SAS
Aircraft taxiing specifications	VRP literature	Taxibot specifications	SAS, TaxiBot literature, VRP literature
Aircraft specifications	Online open sources	Aircraft A/D delays	AAS API

5.4. Expected Results

As the program consists of two optimisation models, two sets of outcomes will be generated as well. The strategic optimisation model will have a vehicle routing schedule as output. All vehicles are tracked over the airport at each moment in time. This does include the aircraft, the TaxiBots towing any aircraft but also the movement of TaxiBots between gates and runways when not taxiing. This schedule will both be visual in the form of a time-allocation graph and a quantitative output in the form of a textual schedule. This latter is important as this output in turn should be used as input for the next model. Then, the second set of outcomes generated by the tactical optimisation model should consist of multiple output types. The first and foremost one is a new vehicle routing schedule which should have exactly the same format as the previous schedule. By this, a proper comparison can take place. Next to that, descriptive statistics of both schedules will be needed in order to quantitatively compare the results and find the effects of probabilistic delays on the schedule. Such descriptive statistics will consist of the following aspects:

- Efficiency of TaxiBot utilisation
- Average taxi time per vehicle type
- Number of rerouted vehicles
- Quantitative total number of changes

[1] https://www.gurobi.com/products/gurobi-optimiser/, accessed on 28-12-2020
[2] https://www.lvnl.nl/omgeving/baangebruik, accessed on 22-01-2021
[3] https://bezoekbas.nl, accessed on 22-01-2021

- Quantitative range of changes
- Range of delays causing no/limited effects or problems
- Number of iterations with schedule changes necessary
- Differences in above metrics between different days

Such descriptive statistics will be used to verify the models. First of all, the input schedule given to the tactical optimisation model should give the exact same output schedule if no sudden change is occurring. Next to that, if a sudden change is occurring, the relative changes on the new schedule will be in line with the relative change of the sudden occurrence. In other words, if the changing parameter only concerns a small effect, such as a delay of a couple of minutes, the output schedule might only show a small change with respect to the initial schedule. However, when large alterations will be seen, such as a change in the runway configuration, large deviations are expected in the output schedule as well.
Other ways of verifying the model are the standard unit and block tests, and the use of a simplified data set of which the outcome can be verified by hand. Furthermore, checks can be added to the code to visualize errors when they might occur. Lastly, all constraints in the problem can be individually added or removed from the program, from which the effects of this can be analysed and should be in line with expected correlations. In other words, removing apparent limiting constraints would result in a more relaxed schedule and vice versa.

A thorough sensitivity analysis on critical input parameters will be used to validate the model. The two sets of parameters, the ones used as input for the initial optimisation model and the variables resulting in a sudden change will have a range of possible attributes, based on real-life circumstances. By covering the whole range of these parameters in a sensitivity analysis, both the elasticity and sensitivity of the model parameters will be tested. So, both conclusion validity and internal validity will be used [62]. Conclusion validity refers to the question if there is a relationship. This will be tested by looking at the difference between the strategic and tactical solutions. Secondly, internal validity aims to answer the question what type of relationship is present. This will be determined via the proposed sensitivity analysis.

6
Project Planning

This chapter provides an overview of the planning of the thesis. The first part discusses the functional flow diagram of the programming model that will build, the second part contains the Gantt chart, elaborating the precise time-wise planning of this project.

6.1. Functional Flow Diagram

Figure 6.1 gives an overview of how the model will be build up. Figure 6.2 below gives a simplified version of this functional flow diagram. The connection and direction of flow can be seen here. Input data is represented by parallelograms, output is visualised via circles. All the functions are put in a rectangle box. A description of each dashed block is given below:

As explained in section 5.2, the model consists of three parts, which is visualised by the three dashed blocks. Each part has an aircraft routing solution as output, visualised with the circles, which can be labelled as preliminary, strategic and tactical optimal solutions. Each of these solutions has a time-space diagram as output, however the last part also contains KPI's on the effects of probabilistic aircraft delays, after comparing the strategic and tactical solutions.

The first part collects all necessary raw input to create an airport routing nodal network, an aircraft flight schedule and all necessary parameters of all vehicles, being aircraft and TaxiBots. These three sets of structured input data will be used to create a vehicle routing problem, together with the notion that a sufficient amount of TaxiBots will be at hand for this preliminary VRP. Practically, this could be defined as an equal number of TaxiBots as the number of arriving and departing aircraft. This VRP will be ran in order to optimise for a minimum total taxi duration resulting in the aforementioned optimal solution.

Then, this output will be used as input for the second block. The same VRP problem is used, however now an iteration takes place to determine the minimum number of TaxiBots necessary while maintaining a feasible solution. The box "Decrease the Number of TaxiBots" makes sure that at every iteration this number is lowered. Theoretically, this number should be lowered by one at every iteration, however for the sake of computation time, large steps can be taken in the beginning. Only whenever a predefined threshold will be reached, iterations should decrease as slowly as possible. This threshold will be taken from literature in which the optimal number of TaxiBots has been defined for other goals. Whenever a non-feasible solution will be encountered for the first time, the number of TaxiBots used at that iteration will be increased by 1 in the "Add 1 for the Minimum Number of TaxiBots" box in order to find the minimum number necessary. This results in a strategic solution.

Lastly, the third part of the model tries to find an optimal solution after probabilistic delays occur. These delays are modelled as follows:

- **Probability Distribution Function Aircraft Delays:** Aircraft delays will follow a probability distribution function (PDF), in which the probability of positive delays (late arrivals/departures) and negative delays (early arrivals/departures) are both possible. The function will be constrained to a finite range in order to practically work with it.

- **Randomly Select Actual A/D Time:** Each aircraft, whether arriving or departing, will be randomly allocated a delay from this PDF, resulting in a new operation time.
- **Random Aircraft Arrival/Departure Time:** This new operation time will be different from the original input time, however this difference might be negative, zero or positive. This is then used as input to the VRP.

The last iteration travels through time, discretised with time intervals. The iteration process works as follows:

- **Move Through Time Intervals:** A day of operations is discretised from its continuous process into time intervals of to be defined lengths. Preliminary analysis shows an time span of 5 to 10 minutes. Practically, the calmest point of the day in the middle of the night is the boundary between two days of operation, so the first interval would start e.g. at 4:00 am. to 4:05 am.
- **Fix all Aircraft A/D Times Starting in this Specific Time Interval:** All aircraft that have the beginning of the range of the probability of A/D in the specific time interval to be assessed will fix their particular operational time, which might be within the current time interval but also in e.g. the third next time interval. So, if flight F001 has an expected arrival time of 4:15, but the probability of arrival will lie between 4:03 and 4:30, then in this specific time interval of 4:00-4:05, this flight's F101 actual arrival time will be drawn from the PDF. This might be 4:04, but can be 4:28 just as well. The latter, meaning that the aircraft will be scheduled in the time interval 4:25-4:30, makes sure that aircraft can be too early as well, and the model wouldn't be "too late" scheduling this flight as well.
- **VRP** The next two blocks represent the aforementioned VRP modelling boxes. One alteration is the optimisation objective which, next to the minimisation of total taxi duration, has been combined with the minimisation of deviations from the original schedule.
- **Are all A/D Times Fixed?** This decision box determines if the iteration is terminated (if all aircraft have been fixed and scheduled in the same round) or continued (if there are still flights to be fixed).
- **Slide to Next Time Interval:** If the decision is no, the next interval is considered and the process starts over again. In aforementioned example, this new time interval would become 4:05-4:10.

Only when the decision is yes, the tactical optimal aircraft routing solution is found. This solution consists of a time-space diagram showing the position of all vehicles at each moment in time, but also a comparison of the tactical solution with the strategic solution. KPI's on the effect of probabilistic delays will be output.

This method is visualised in Figure 6.3. The top figure shows the fixed schedule where the vehicles, taxibots and aircraft move over time. Two vehicles colored red and blue are visualised here. Three flights have a PDF for their arrival time. The grey time period is the current period to be assessed. F101 arrived exactly at its planned time, so no deviations in the schedule occurred. F102 and F103 have an arrival time later than the expected time so there might be a change in the schedule needed. Firstly, the PDF (i.e. range of arrival times) of F103 ranges four time spans and as the beginning of the PDF is located in the selected time span, the exact arrival time will be selected now. If it turns out this arrival time will lay ahead of the current time span, it is not considered yet and will only be in a later time span, the next one in this case. If it turns out the drawn arrival time will be in the current time span, such as is the case for F102, the new schedule will be generated, taking into account this change from the expected time. In this case, F102 had been fixed already one time interval before. As the actual arrival time of F102 is different from the expected time, a change in the schedule had to be made. Therefore, the new routing for the red line is displayed with the dashed red line.

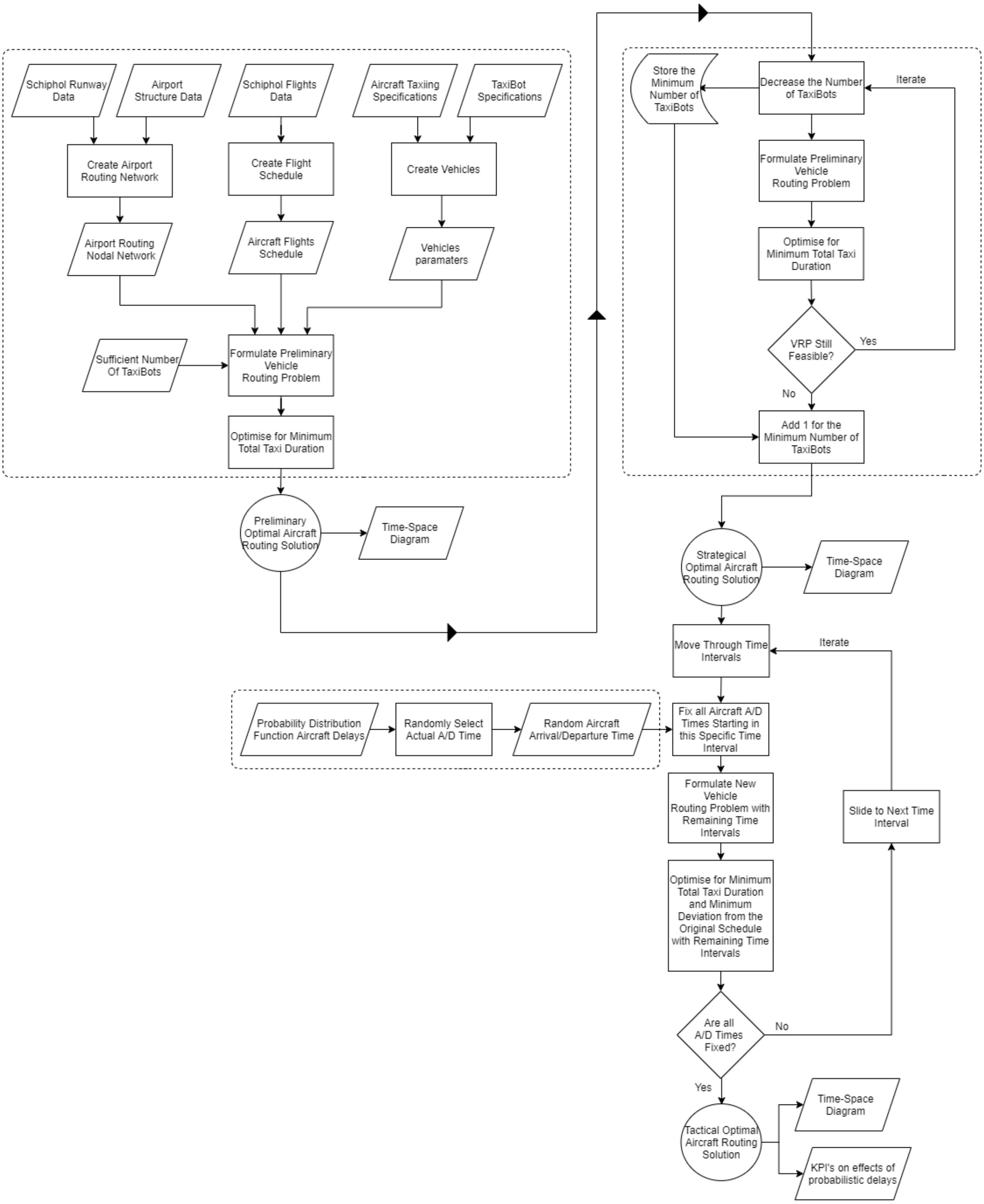

Figure 6.1: Functional flow diagram of the model to be built.

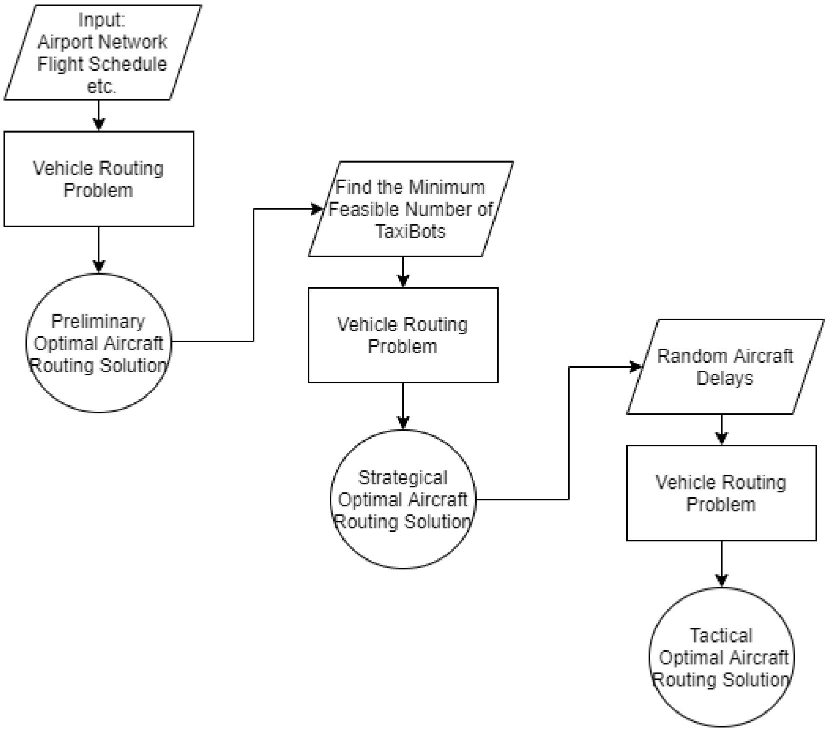

Figure 6.2: Simplified version of the functional flow diagram of the model to be built.

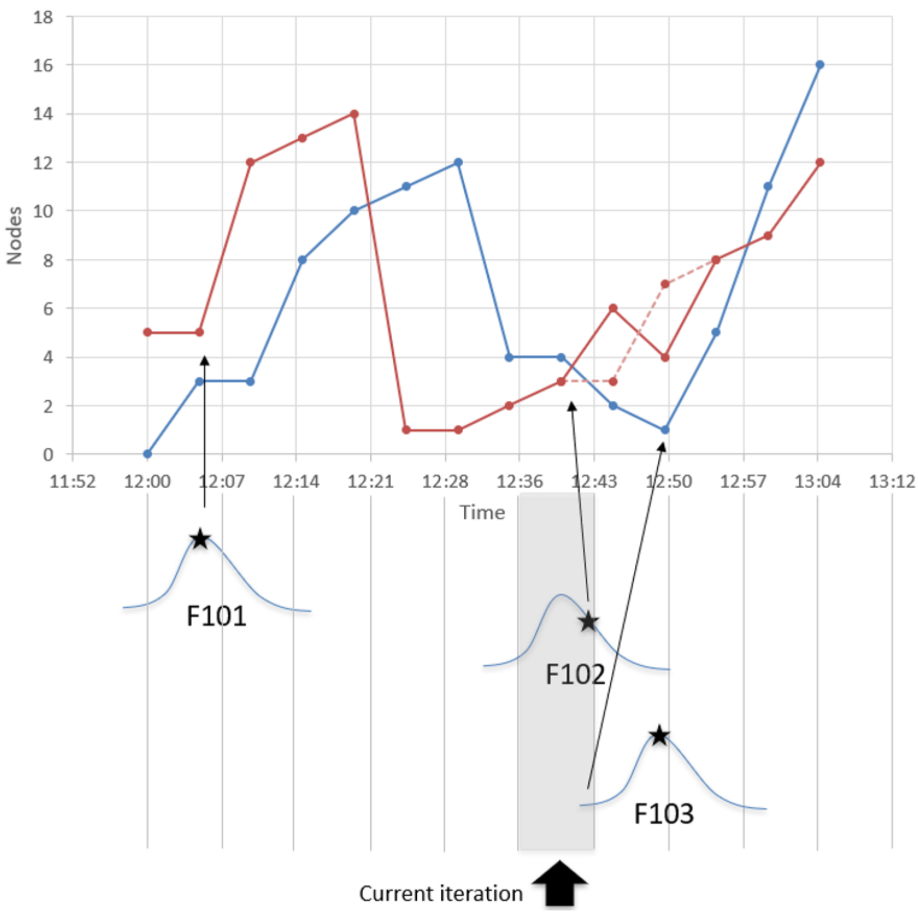

Figure 6.3: Visual simplified example of one iteration of the reactive model.

6.2. Gantt Chart

The project planning of the thesis is visually represented via a Gantt chart, as can be seen in Figure 6.4. The tasks on the left have been separated in their respective phase of the project. Green diamonds represent deliverables, grey tasks are holidays, arrows show the dependencies of some tasks.

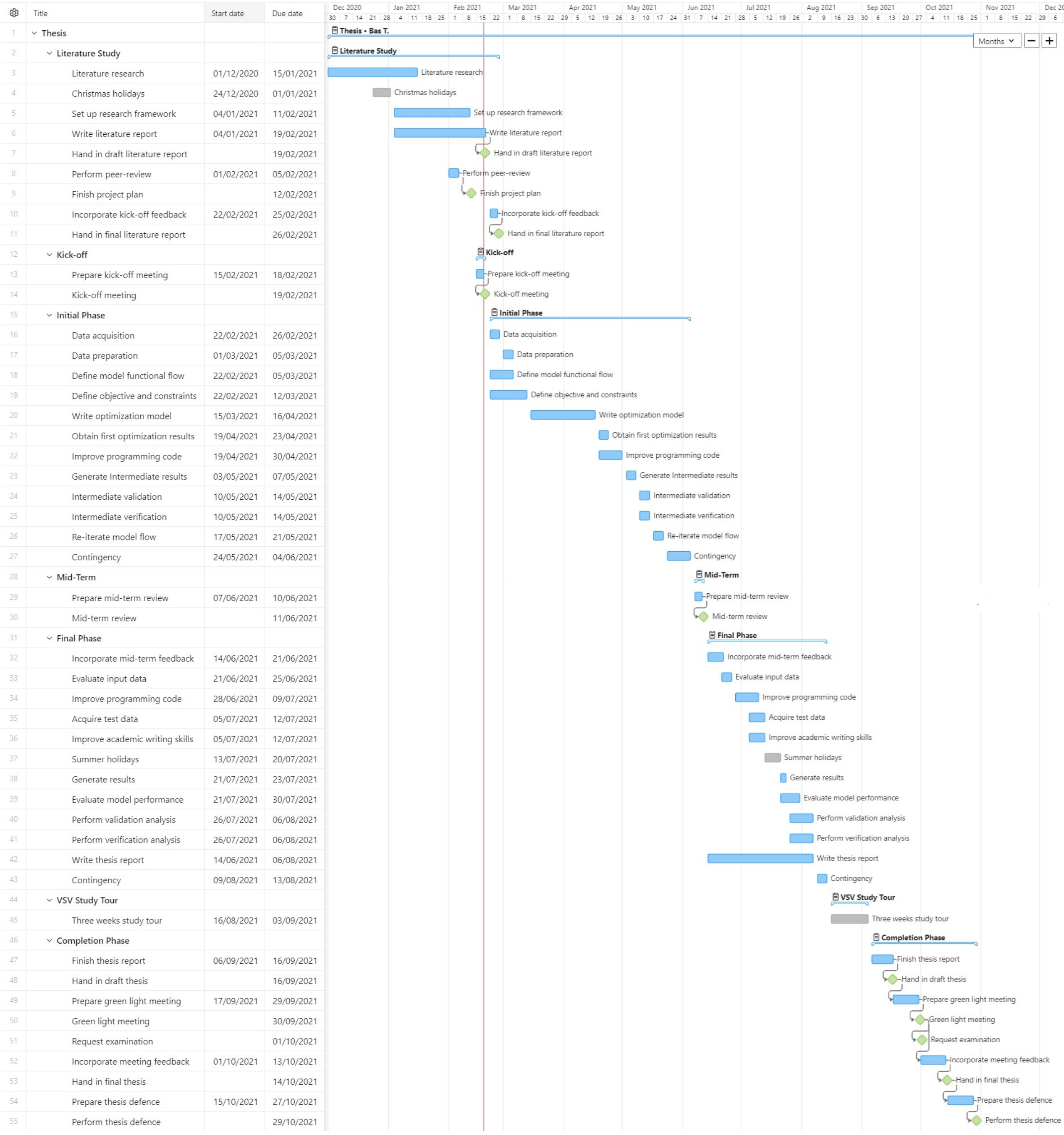

Figure 6.4: Gantt chart visualising the project planning.

7 Conclusion

TaxiBots are a promising solution to help defeat the continuously increasing pollutants emitted by the aviation sector. Even though multiple other electric taxiing solutions have been researched and even developed, TaxiBot has been found to be the most promising. The industry has an increasing interest in this novel concept and thus Amsterdam Airport Schiphol has performed a Proof of Concept test. From this test, among others, it has become clear that implementation brings some practical concerns, which have to be resolved. One of these problems is the scheduling of TaxiBots and taxiing aircraft at the airport apron. Such a problem has to be solved exactly, hence from the various methods found in literature, a vehicle routing problem, a type of mixed-integer linear programming, is found to be the most suitable. Therefore, this research tries to fill that gap by strengthening the research in the field of TaxiBot routing and scheduling.

The second research aspect that arose is the need to be able to regenerate a new schedule in the case of sudden changes. As a lot of parameters are involved in the planning, one small change can easily occur, however can disrupt the schedule entirely. Therefore, the tactical optimisation model should be able to solve this problem in order to generate a new feasible solution. Algorithms found in literature, such as rolling windows or column generation heuristic, can help decreasing computation time. In order to answer the research question set a model performing a case study will be build. Literature on the case study to be performed at AAS is used to determine the most suitable VRP modelling aspects as well as most fitting nodal networks and flight schedules. The latter will also be used to define the most suiting probability density functions of aircraft delays. TaxiBot specifications from various sources will be combined and lastly literature on robust scheduling will be used in the comparison as well.

All in all, with this model, the research will be able to **determine the effect of probabilistic aircraft departure and arrival delays on a vehicle routing schedule at Amsterdam Airport Schiphol which includes the use of an electric taxiing system, viz TaxiBot, to schedule the routing of aircraft and TaxiBots with near real-time updates on arrival and departure times, by creating a reactive optimisation routing and scheduling model which can reiterate the planning based on the new received non-deterministic time information while trying to minimise the deviations from the initial schedule.**

With such a model, TaxiBot schedules can be generated on a tactical level. Hence, a future recommendation is to incorporate such a reactive model in the tactical planning of airport operations in order to minimise deviations from the defined strategic schedule.

III

Supporting work

1

Input Data

Table 1.1 gives an overview of the runways used throughout the day of operations of the busy day, 8 August 2019. A similar table is used on the calm day of operations, 26 January 2019, as can be seen in Table 1.2. *A* represents the runway(s) used for arrivals, *D* is for departures. Table 1.3 and Table 1.4 provide the nodal network nodes, including x and y positions, connected nodes, distances and speeds on these edges with the connected nodes and its attributes with respect to node functions. Table 1.5 shows all gates that are connected with the gate nodes and Table 1.6 show all the runways that are connected with the runway nodes. Table 1.7 lists all types of aircraft that are certified now and in the near future.

Table 1.1: Runway configuration, 8 August 2019.

From	Till	A	D
02:00:00	06:49:00	27	24
06:50:00	07:04:00	18R	24, 18L
07:05:00	07:49:00	18R, 18C	24, 18L
07:50:00	09:24:00	18R, 18C	24
09:25:00	10:59:00	18R	24, 18L
11:00:00	11:39:00	18R, 18C	24
11:40:00	11:54:00	18R	24
11:55:00	12:44:00	18R	24, 18L
12:45:00	13:29:00	18R, 18C	24, 18L
13:30:00	14:14:00	18R	24
14:15:00	15:14:00	18R	24, 18L
15:15:00	16:14:00	18R, 18C	24
16:15:00	16:34:00	18R, 18C	24
16:35:00	17:19:00	18R	24, 18L
17:20:00	18:44:00	18R	24
18:45:00	19:34:00	18R, 18C	18L
19:35:00	20:34:00	18R	24
20:35:00	21:29:00	18R	24, 18L
21:30:00	23:59:00	18R	24
00:00:00	01:59:00	27	24

Table 1.2: Runway configuration, 26 January 2019.

From	Till	A	D
02:00:00	06:39:00	18R	24
06:40:00	07:44:00	18R, 18C	24, 18L
07:45:00	09:24:00	18R, 18C	24
09:25:00	10:29:00	18R	24, 18L
10:30:00	11:54:00	18R, 18C	24, 18L
11:55:00	12:19:00	18R	24, 18L
12:20:00	13:44:00	18R, 18C	24, 18L
13:45:00	14:29:00	18R	24, 18L
14:30:00	15:24:00	18R, 18C	24, 18L
15:25:00	16:14:00	18R, 18C	24
16:15:00	17:39:00	18R	24, 18L
17:40:00	17:49:00	18R	24
17:50:00	19:39:00	18R, 18C	24
19:40:00	20:19:00	18R, 27	24
20:20:00	22:19:00	27	36L, 24
22:20:00	23:04:00	18R	24
23:05:00	23:59:00	18C	24
00:00:00	01:04:00	18C	24
01:05:00	01:59:00	18R	24

Table 1.3: Nodal Network input data, adapted from Guillaume [21], Part 1.

Node ID	posx	posy	Connecting nodes				Distance (m)				Speed (m/s)				Gate	Runway	Service Road
			1	2	3	4	1	2	3	4	1	2	3	4			
0	1906	1721	11	0	0	0	100	0	0	0	8	0	0	0	TRUE	FALSE	FALSE
1	1410	2843	13	0	109	2	100	0	1000	220	4	0	14	8	TRUE	FALSE	FALSE
2	1805	2860	15	0	1	3	100	0	220	300	4	0	8	8	TRUE	FALSE	FALSE
3	2135	2869	16	17	2	4	149	220	300	180	4	4	8	8	TRUE	FALSE	FALSE
4	2600	2895	18	115	3	5	112	45	180	50	4	8	8	8	TRUE	FALSE	FALSE
5	2718	2790	19	0	4	6	102	0	50	70	4	0	8	8	TRUE	FALSE	FALSE
6	2699	2500	21	0	5	7	102	0	70	50	4	0	8	8	TRUE	FALSE	FALSE
7	2684	2270	22	0	6	8	102	0	50	50	4	0	8	8	TRUE	FALSE	FALSE
8	2559	2060	24	117	7	9	141	500	50	70	4	8	8	8	TRUE	FALSE	FALSE
9	2426	1934	26	25	8	10	108	118	70	200	4	4	8	8	TRUE	FALSE	FALSE
10	2186	1793	27	0	9	11	108	0	200	320	4	0	8	8	TRUE	FALSE	FALSE
11	1906	1621	28	0	10	108	108	0	320	1518	4	0	8	14	TRUE	FALSE	FALSE
12	855	2920	0	51	13	0	0	100	545	0	0	10	10	0	FALSE	FALSE	FALSE
13	1400	2943	0	50	14	1	0	100	295	100	0	10	10	4	FALSE	FALSE	FALSE
14	1695	2956	0	49	15	0	0	100	100	0	0	10	10	0	FALSE	FALSE	FALSE
15	1795	2960	0	48	16	2	0	100	230	100	0	10	10	4	FALSE	FALSE	FALSE
16	2025	2969	0	47	17	3	0	100	340	149	0	10	10	4	FALSE	FALSE	FALSE
17	2365	2983	0	46	18	3	0	169	285	220	0	10	10	4	FALSE	FALSE	FALSE
18	2650	2995	0	45	19	4	0	100	281	112	0	9	9	4	FALSE	FALSE	FALSE
19	2818	2770	20	43	0	5	145	100	0	102	7	7	0	4	FALSE	FALSE	FALSE
20	2809	2625	21	0	0	0	145	0	0	0	7	0	0	0	FALSE	FALSE	FALSE
21	2799	2480	22	41	0	6	230	100	0	102	7	7	0	4	FALSE	FALSE	FALSE
22	2784	2250	23	40	0	7	204	100	0	102	7	7	0	4	FALSE	FALSE	FALSE
23	2756	2048	24	38	0	0	131	138	0	0	7	7	0	0	FALSE	FALSE	FALSE
24	2659	1960	25	37	0	8	147	105	0	141	7	7	0	4	FALSE	FALSE	FALSE
25	2533	1884	26	36	0	9	84	105	0	118	10	10	0	4	FALSE	FALSE	FALSE
26	2466	1834	27	35	0	9	278	105	0	108	10	10	0	4	FALSE	FALSE	FALSE
27	2226	1693	28	0	34	10	329	0	105	108	10	0	10	4	FALSE	FALSE	FALSE
28	1946	1521	29	33	0	11	278	105	0	108	10	10	0	4	FALSE	FALSE	FALSE
29	1706	1381	30	32	0	0	557	105	0	0	10	10	0	0	FALSE	FALSE	FALSE
30	1226	1099	31	61	0	0	105	450	0	0	10	10	0	0	FALSE	FALSE	FALSE
31	1280	1009	0	30	32	0	758	105	557	206	0	10	10	0	FALSE	FALSE	FALSE
32	1760	1291	0	29	33	0	0	105	278	320	0	10	10	0	FALSE	FALSE	FALSE
33	2000	1431	0	28	34	0	0	105	329	0	0	10	10	0	FALSE	FALSE	FALSE
34	2280	1603	0	35	0	27	0	278	320	105	0	10	10	10	FALSE	FALSE	FALSE
35	2520	1744	0	26	36	0	0	105	84	0	0	10	7	0	FALSE	FALSE	FALSE
36	2587	1794	0	25	37	0	0	105	147	216	0	10	7	0	FALSE	FALSE	FALSE
37	2713	1870	0	24	38	0	0	105	171	221	0	7	7	0	FALSE	FALSE	FALSE
38	2860	1958	0	23	39	94	0	138	88	206	0	7	7	7	FALSE	FALSE	FALSE
39	2910	2030	0	23	40	0	0	155	221	242	0	7	7	0	FALSE	FALSE	FALSE
40	2884	2250	0	22	41	0	0	100	230	0	0	7	7	0	FALSE	FALSE	FALSE
41	2899	2480	0	21	42	0	0	100	145	0	0	7	7	0	FALSE	FALSE	FALSE
42	2909	2625	0	0	43	0	0	0	145	0	0	0	7	0	FALSE	FALSE	FALSE
43	2918	2770	19	0	44	0	100	0	335	0	7	0	7	0	FALSE	FALSE	FALSE
44	2750	3060	45	0	93	0	115	0	347	0	9	0	9	0	FALSE	FALSE	FALSE
45	2640	3095	46	18	0	0	415	100	0	360	10	10	0	0	FALSE	FALSE	FALSE
46	2225	3078	47	17	0	0	210	169	0	244	10	10	0	0	FALSE	FALSE	FALSE
47	2015	3069	48	16	0	0	230	100	0	0	10	10	0	0	FALSE	FALSE	FALSE
48	1785	3060	49	15	0	0	100	100	0	279	10	10	0	0	FALSE	FALSE	FALSE
49	1685	3056	50	14	0	0	295	100	0	0	10	10	0	0	FALSE	FALSE	FALSE
50	1390	3043	51	13	0	0	545	100	0	0	10	10	0	0	FALSE	FALSE	FALSE
51	845	3020	67	96	0	12	220	211	0	100	10	10	0	10	FALSE	FALSE	FALSE
52	425	1679	97	0	62	0	218	0	105	0	10	0	10	0	FALSE	FALSE	FALSE
53	433	1792	52	71	0	63	113	189	0	105	10	10	0	10	FALSE	FALSE	FALSE
54	451	2042	53	0	0	64	251	220	0	105	11	0	0	11	FALSE	FALSE	FALSE
55	474	2375	54	0	65	0	334	409	105	0	11	0	11	0	FALSE	FALSE	FALSE
56	512	2917	55	0	66	0	543	0	105	0	11	0	11	0	FALSE	FALSE	FALSE
57	521	3043	56	0	0	0	126	0	336	0	11	0	0	0	FALSE	FALSE	FALSE
58	562	3625	57	82	0	68	583	189	0	105	11	10	0	11	FALSE	FALSE	FALSE
59	613	4354	58	0	0	69	731	0	247	105	11	0	0	11	FALSE	FALSE	FALSE

Table 1.4: Nodal Network input data, adapted from Guillaume [21], Part 2.

Node ID	posx	posy	Connecting nodes				Distance (m)				Speed (m/s)				Gate	Runway	Service Road
			1	2	3	4	1	2	3	4	1	2	3	4			
60	648	4854	59	98	70	0	502	189	105	0	11	11	11	0	FALSE	FALSE	FALSE
61	877.5	1383	62	30	0	0	449.5	450	0	0	10	10	0	10	FALSE	FALSE	FALSE
62	529	1667	52	72	63	61	105	364	113	449.5	10	10	10	10	FALSE	FALSE	FALSE
63	537	1780	0	64	53	0	0	251	105	0	0	10	10	0	FALSE	FALSE	FALSE
64	555	2030	0	65	54	0	0	334	105	0	0	11	11	0	FALSE	FALSE	FALSE
65	578	2363	0	66	55	0	0	543	105	0	0	11	11	0	FALSE	FALSE	FALSE
66	616	2905	0	67	0	12	0	126	0	239	0	11	0	11	FALSE	FALSE	FALSE
67	625	3031	0	68	57	51	0	583	105	220	0	11	11	11	FALSE	FALSE	FALSE
68	666	3613	0	69	58	0	0	731	105	0	0	11	11	0	FALSE	FALSE	FALSE
69	717	4342	0	70	59	0	0	501	105	0	0	11	11	0	FALSE	FALSE	FALSE
70	752	4842	0	85	60	0	0	428	105	0	0	12	12	0	FALSE	FALSE	FALSE
71	245	1813	0	75	0	53	0	189	0	189	0	10	0	10	FALSE	FALSE	FALSE
72	350	1350	62	73	0	0	364	364	0	0	10	10	0	0	FALSE	FALSE	FALSE
73	-11	1304	72	74	0	0	364	304	0	0	10	10	0	0	FALSE	FALSE	FALSE
74	41	1604	0	0	97	75	71	0	189	231	0	0	10	10	FALSE	FALSE	FALSE
75	57	1834	0	77	71	0	0	105	189	0	0	10	10	0	FALSE	FALSE	FALSE
76	-63	1616	73	74	0	0	316	105	0	0	10	10	0	0	FALSE	FALSE	FALSE
77	-47	1846	76	0	0	0	231	0	0	0	10	0	0	0	FALSE	FALSE	FALSE
78	-15	2304	0	77	0	0	0	500	0	0	0	14	0	0	FALSE	FALSE	FALSE
79	17	2762	78	0	0	0	410	0	0	0	14	0	0	0	FALSE	FALSE	FALSE
80	49.5	3220.5	0	79	0	0	0	500	0	0	0	14	14	0	FALSE	FALSE	FALSE
81	82	3679	80	82	86	0	410	294	621	0	14	10	12	0	FALSE	FALSE	FALSE
82	374	3646	0	0	81	58	0	0	294	189	0	0	10	10	FALSE	FALSE	FALSE
83	124	4489	81	0	0	0	812	0	0	0	12	0	0	0	FALSE	FALSE	FALSE
84	167	5300	83	0	0	0	812	0	0	0	12	0	0	0	FALSE	FALSE	FALSE
85	750	5270	0	84	0	0	0	584	0	0	0	12	0	0	FALSE	FALSE	FALSE
86	-365	4110	81	87	0	0	621	621	0	0	12	12	0	0	FALSE	FALSE	FALSE
87	-812	4542	88	0	86	0	313	0	621	0	12	0	12	0	FALSE	FALSE	FALSE
88	-1000	4792	89	90	0	0	221	151	0	0	10	10	0	0	FALSE	FALSE	FALSE
89	-1166	4938	91	0	0	0	140	0	0	0	10	0	0	0	FALSE	FALSE	FALSE
90	-1125	4708	0	88	87	0	0	151	354	0	0	10	10	0	FALSE	FALSE	FALSE
91	-1291	4875	0	92	0	90	486	278	0	235	0	10	0	10	FALSE	FALSE	FALSE
92	-1480	5041	99	91	0	0	217	278	460	0	10	10	0	0	FALSE	FALSE	FALSE
93	2880	3320	0	44	0	95	0	347	0	443	0	9	0	9	FALSE	FALSE	FALSE
94	2960	1778	0	38	0	116	0	206	0	200	0	7	0	10	FALSE	TRUE	FALSE
95	3160	3610	93	113	0	0	443	670	0	0	9	14	0	0	FALSE	TRUE	FALSE
96	830	3230	51	110	0	0	211	150	0	0	10	10	0	0	FALSE	TRUE	FALSE
97	229	1583	74	0	52	106	189	0	218	380	10	0	10	10	FALSE	TRUE	FALSE
98	460	4875	118	60	0	0	400	189	0	0	10	11	0	0	FALSE	TRUE	FALSE
99	-1624	5188	100	0	92	0	500	0	217	0	14	0	10	0	FALSE	TRUE	FALSE
100	-1100	4990	101	0	99	0	1500	0	500	0	14	0	14	0	FALSE	FALSE	TRUE
101	0	3879	102	112	100	0	250	1470	1500	0	10	14	14	0	FALSE	FALSE	TRUE
102	200	3860	103	0	101	0	250	0	250	0	10	0	10	0	FALSE	FALSE	TRUE
103	170	3500	104	110	102	0	250	580	250	0	10	10	10	0	FALSE	FALSE	TRUE
104	-30	3520	105	0	103	0	2300	0	250	0	14	0	10	0	FALSE	FALSE	TRUE
105	-185	1200	106	0	104	0	350	0	2300	0	10	0	14	0	FALSE	FALSE	TRUE
106	200	1200	107	105	97	0	700	350	380	0	14	10	10	0	FALSE	FALSE	TRUE
107	600	1400	108	0	106	0	460	0	700	0	10	0	14	0	FALSE	FALSE	TRUE
108	650	2000	109	11	107	0	380	1518	460	0	10	14	10	0	FALSE	FALSE	TRUE
109	700	2600	108	1	110	0	380	1000	1100	0	10	14	14	0	FALSE	FALSE	TRUE
110	750	3450	109	103	96	111	1100	580	150	1870	14	10	10	14	FALSE	FALSE	TRUE
111	860	5350	110	112	118	0	1870	650	160	0	14	14	10	0	FALSE	FALSE	TRUE
112	70	5380	111	0	101	0	650	0	1470	0	14	0	14	0	FALSE	FALSE	TRUE
113	2720	3250	114	0	95	0	570	0	670	0	10	0	14	0	FALSE	FALSE	TRUE
114	2600	3050	115	0	113	0	85	0	570	0	10	0	10	0	FALSE	FALSE	TRUE
115	2600	2930	4	0	114	0	45	0	85	0	8	0	10	0	FALSE	FALSE	TRUE
116	2960	2000	117	0	94	0	200	0	200	0	10	0	10	0	FALSE	FALSE	TRUE
117	2700	2060	8	0	116	0	500	0	200	0	8	0	10	0	FALSE	FALSE	TRUE
118	600	5250	111	0	98	0	160	0	400	0	10	0	10	0	FALSE	FALSE	TRUE

Table 1.5: Gates per node.

Node	Gates
1	H1,H2,H3,H4,H5,H6,H7,M1,M2,M3,M4,M5,M6,M7,G1,G3,G5,G7,G9,G11,G13,G15,G17
2	G2,G4,G6,G8,G10,G12,G14,G16,F1,F3,F5,F7,F9
3	E1,E3,E5,E7,E9,E11,E13,E15,E17,E19,E21,E23,E25,F2,F4,F6,F8,F10,F12
4	E2,E4,E6,E8,E10,E12,E14,E16,E18,E20,E22,E24
5	E2,E4,E6,E8,E10,E12,E14,E16,E18,E20,E22,E24
6	D3,D5,D7,D41,D43,D45,D47,D49,D51,D53,D55,D57,D59,D61,D63,D71,D73,D75,D77,D79, D81,D83,D85,D87,D89,E2,E4,E6,E8,E10,E12,E14,E16,E18,E20,E22,E24
7	D23,D25,D27,D29,D31,D42,D44,D46,D48,D50,D52,D54,D56,D74,D76,D78,D80,D82,D84,D86
8	D2,D4,D6,D8,D10,D12,D14,D16,D18,D20,D22,D24,D26,D28,D60,D62,D64,D66,D68
9	C1,C3,C5,C7,C9,C11,C13,C15,C17,C19,C21,C23,C25
10	B1,B3,B5,B7,B9,B11,B13,B15,B17,B19,B21,B23,B25,B27,B29,B31,B33,B35,B37,B39,,C2,C4, C6,C8,C10,C12,C14,C16,C18,C20,C22,C24
11	B2,B4,B6,B8,B10,B12,B14,B16,B18,B20,B22,B24,B26,B28,B30,B32,B34,B36

Table 1.6: Runways per node.

Nodes	Runways
94	06,24
95	18L,36R
96	09,27
97	36C
98	18C
99	18R,36L

Table 1.7: List of certified aircraft and future certified aircraft.

Certified ac	Certified ac	Future Certified ac
AIRBUS A319-111	BOEING 737MAX-8	BOEING B757-300 WINGLETS
AIRBUS A320-200	BOEING 737-500	BOEING 757-200 Winglets
AIRBUS A320-100 (Sharklets)	BOEING 737-400	BOEING B757-300
AIRBUS A320 NEO	BOEING 737-700	BOEING 757-200 PASSENGER
AIRBUS A318	BOEING 737MAX-9	BOEING 757-200PF FREIGHTER
AIRBUS A319 NEO	BOEING 737-500 Winglets	BOEING 757-200 MIXED CONFIGURATION
AIRBUS A321 NEO	BOEING 737-300	AIRBUS A220-300
AIRBUS A320 Passenger	BOEING 737-800 WINGLETS	AIRBUS A220-100
AIRBUS A321-100/200	BOEING 737-900/Winglets	EMBRAER170
BOEING 737-700 Winglets	BOEING 737-600	EMBRAER175(170-200 STD)
BOEING 737MAX-7	BOEING 737-800 Freighter (Winglets)	EMBRAER 190 (IGW)
BOEING 737-800 PASSENGER	BOEING 737-400 FREIGHTER	EMBRAER 195 ERJ 190-200
BOEING 737-300 Winglets	BOEING 737- 300 FREIGHTER	EMBRAER 175-E2
BOEING 737-200/200C ADVANCED PASS.	BOEING 737-200/200C/200QC PASSENGER	EMBRAER 190-E2
		EMBRAER E190-E2 (ERJ190-300)
		EMBRAER ERJ-195-E2 (190-400STD)
		COMAC ARJ21
		COMAC C919
		COMAC C929

2

Descriptive Statistics

Table 2.1 gives an overview of the number of arrivals and departures at each runway in the year 2019. As can be seen, some runways are very little used for specific operations, which resulted in the nodal network being adapted to this runway use. Figure 2.1 shows the aerodrome of Amsterdam Airport Schiphol used to develop and validate the nodal network.

Table 2.1: Number of arrivals and departures per runway in 2019 [6].

Runway	**Arrivals**	**Departures**
Schiphol-Oostbaan (04)	3	11
Schiphol-Oostbaan (22)	6029	15
Kaagbaan (06)	39174	46
Kaagbaan (24)	553	79325
Buitenveldertbaan (09)	0	13392
Buitenveldertbaan (27)	22484	1993
Polderbaan (18R)	95759	0
Polderbaan (36L)	1	61293
Zwanenburgbaan (18C)	38891	10427
Zwanenburgbaan (36C)	14609	19565
Aalsmeerbaan (18L)	0	62575
Aalsmeerbaan (36C)	31158	0
Total Arrival/Departure	248661	248642
Total	497303	

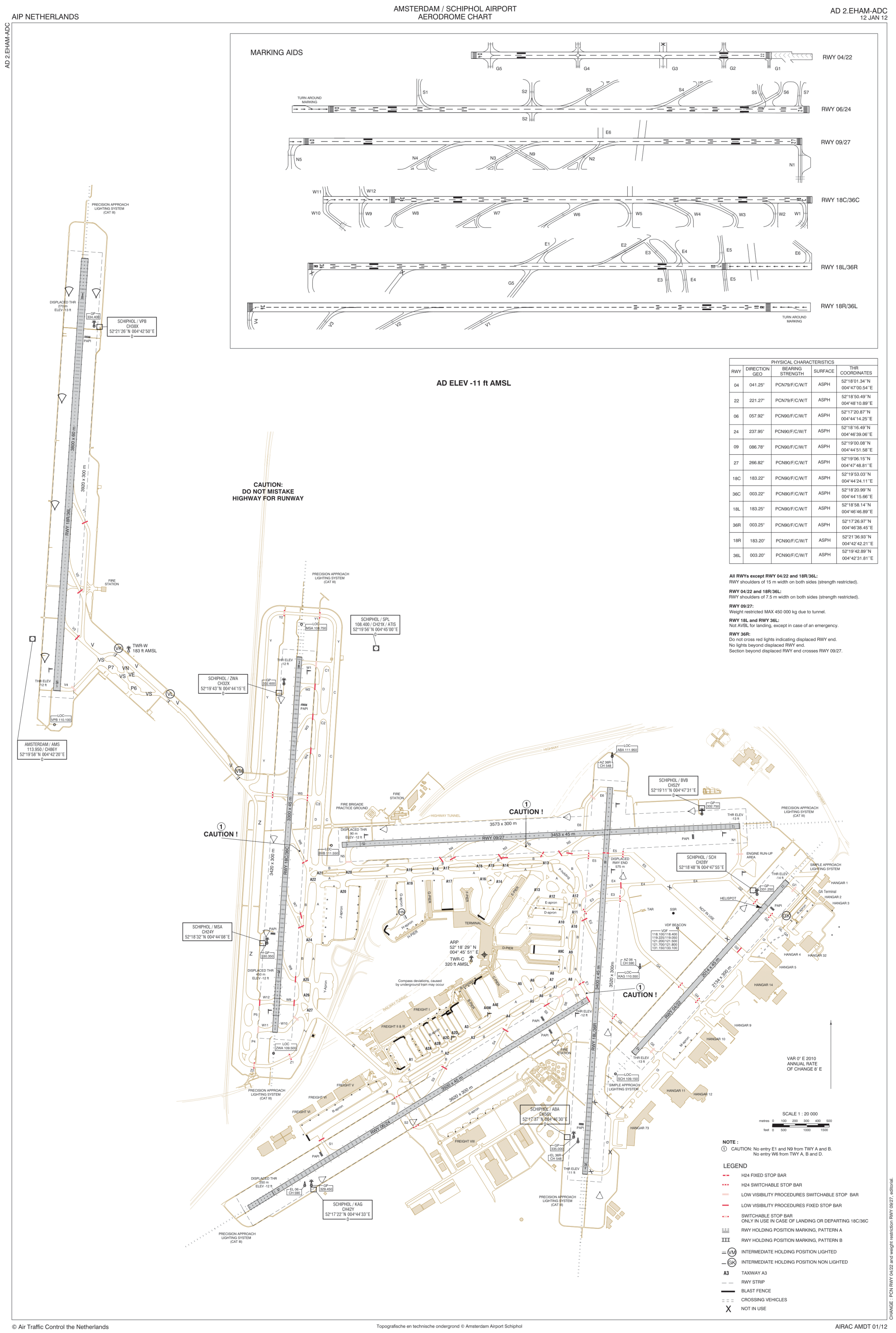

Figure 2.1: Aerodrome Chart of Amsterdam Airport Schiphol, which is the basis of the nodal network.

Figure 2.2 and Figure 2.3 show the number of ac arriving and departing on the busy and calm days assessed respectively. From this, the peak and lean hours throughout the day become clear.

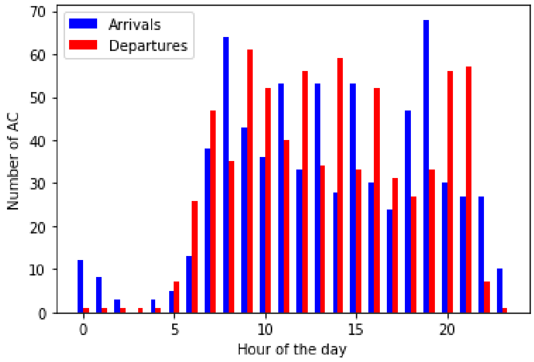

Figure 2.2: Number of aircraft arriving and departing per hour; 08-08-2019, busy scenario.

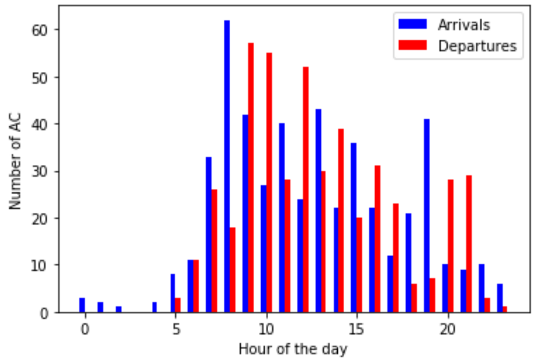

Figure 2.3: Number of aircraft arriving and departing per hour; 26-01-2019, calm scenario.

Figure 2.4 and Figure 2.5 represent the delays including both arrival and departure delays for the busy and calm day respectively. The delays on the busy day are on average around 0 minutes, however delays on the calm day tend to average out in negative delays, meaning that aircraft are earlier than planned. Figure 2.6 and Figure 2.7 show the delays for arriving flights only on both busy and calm days respectively. As can be seen, almost all delays, except for the outliers, are negative, i.e. earlier than expected. Most outliers can be found to have positive delays, i.e. are later than expected. On the other hand, Figure 2.8 and Figure 2.9 show the delays for aircraft that depart from Schiphol. Here, almost all delays are larger than 0, with very few outliers below 0. This means that almost all aircraft departed later than scheduled. Important to note is that the delays from these six figures are measured from the actual times of arrival with respect to the scheduled times of arrival. The probability density function is not used here yet.

This probability density function is determined from the delays from the busy and calm day respectively, including all delays from flights from one week before to week after this actual date. The PDFs for arrivals and departures for both busy and calm days can be seen in Figure 2.10 and Figure 2.11 respectively. As concluded before, most delays for arriving flights tend to be negative, while departing flights tend to be positive. The departure PDF is higher and narrower, meaning less variance. Figure 2.12 and Figure 2.13 show the same arrival delays as before in above two figures, however with this, a better visual comparision can be given. As can be seen, the probability for high delays is lower on calm days, while the general shapes of the PDFs look alike. The same can be said for the PDFs of departures in Figure 2.14 and Figure 2.15 for busy and calm days. The probability of high delays is lower on calm days, while the probability of 0 delay is higher as this peak is higher on calm days with respect to busy days.

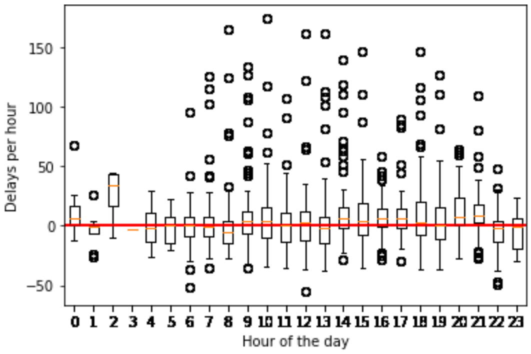

Figure 2.4: Delays [min] with respect to STA, both arrival and departure flights; 08-08-2019, busy scenario.

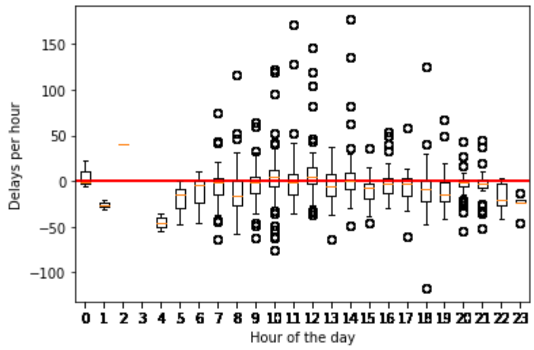

Figure 2.5: Delays [min] with respect to STA, both arrival and departure flights; 26-01-2019, calm scenario.

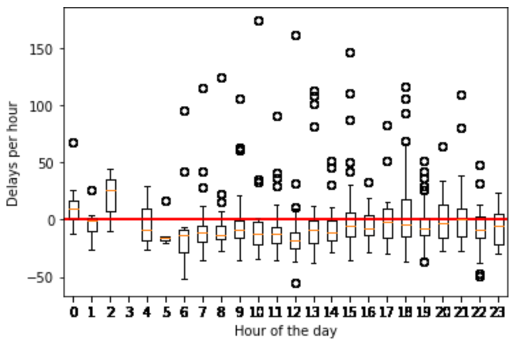

Figure 2.6: Delays [min] with respect to STA, for arriving flights; 08-08-2019, busy scenario.

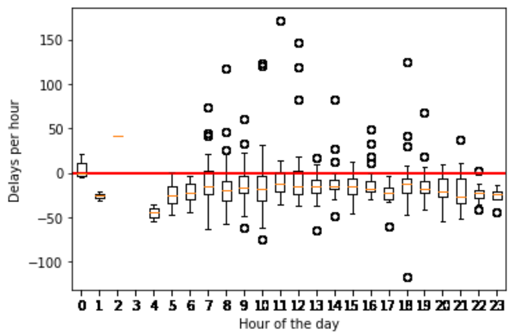

Figure 2.7: Delays [min] with respect to STA, for arriving flights; 26-01-2019, calm scenario.

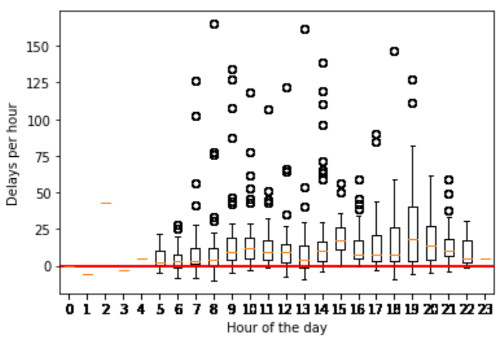

Figure 2.8: Delays [min] with respect to STA, for departing flights; 08-08-2019, busy scenario.

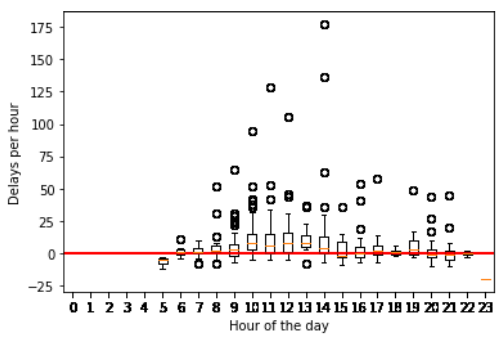

Figure 2.9: Delays [min] with respect to STA, for departing flights; 26-01-2019, calm scenario.

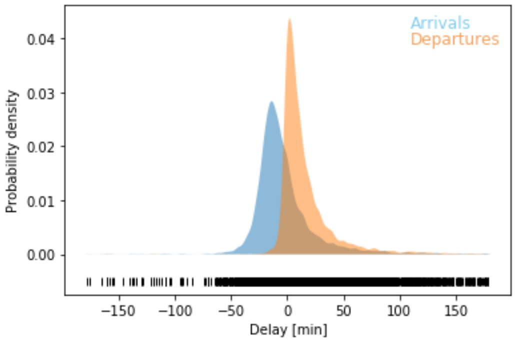

Figure 2.10: Lognormal probability density function for delays for both arriving and departing flights; 08-08-2019, busy scenario.

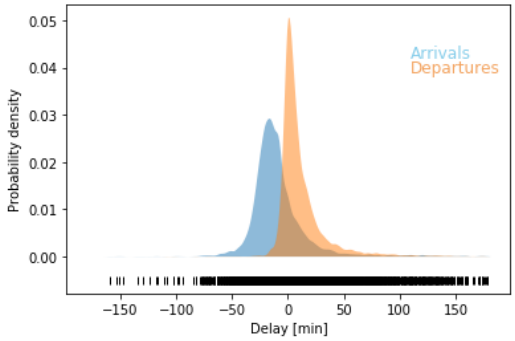

Figure 2.11: Lognormal probability density function for delays for both arriving and departing flights; 26-01-2019, calm scenario.

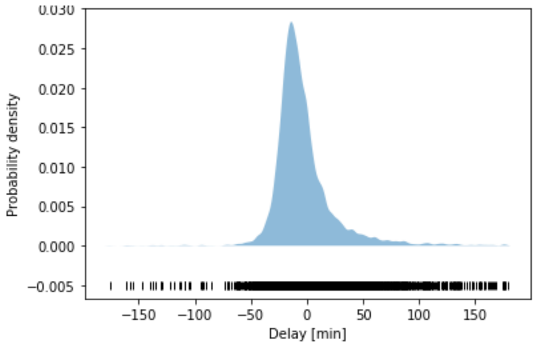

Figure 2.12: Lognormal probability density function for delays for arriving flights; 08-08-2019, busy scenario.

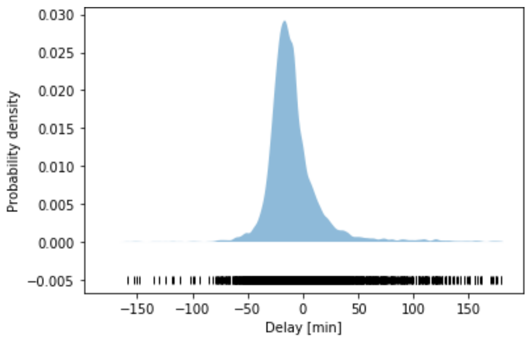

Figure 2.13: Lognormal probability density function for delays for arriving flights; 26-01-2019, calm scenario.

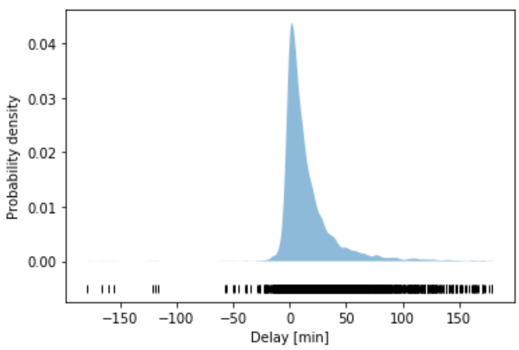

Figure 2.14: Lognormal probability density function for delays for departing flights; 08-08-2019, busy scenario.

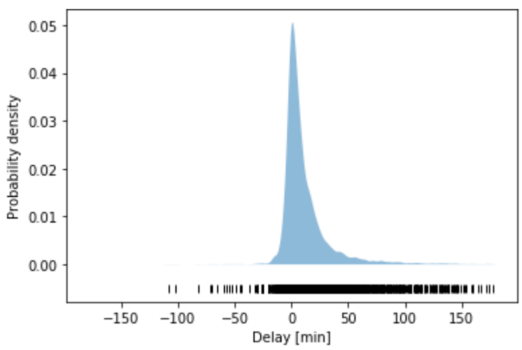

Figure 2.15: Lognormal probability density function for delays for departing flights; 26-01-2019, calm scenario.

The coefficient of variation c_v is the fraction $\frac{\sigma(penalty_{sum})}{\mu(penalty_{sum})}$ in which $\sigma(penalty_{sum})$ is the standard deviation of the sum of penalties and $\mu(penalty_{sum})$ is the mean value of the sum of penalties. A stabilization of this coefficient means that sufficient number of iterations have taken place in order to draw reliable conclusions.

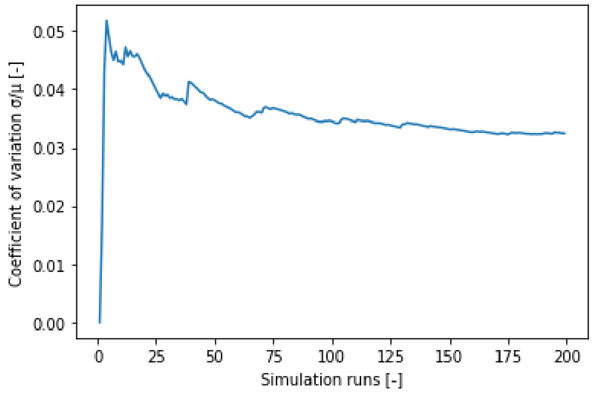

Figure 2.16: Coefficient of variation c_v of the sum of penalties over 200 simulation runs.

The time before the actual emerging time of aircraft to start up TaxiBots to move towards the starting node, $T_{response}$, is set to 5 minutes. This value is based on the average traversing time for TaxiBots. A histogram of all traversing times for TaxiBots to move from their respective location towards the starting node, either gate or runway, of the aircraft they are going to pick up can be seen in Figure 2.17. This figure shows that most of the traversing times lie around 3 minutes with some outliers around 6.5, 7.5 and 11 minutes. Five minutes is chosen to incorporate the high peak around 3 minutes while also making sure the TaxiBots do not wait too long after arriving at the starting node. If a higher value, i.e. around 12 was chosen, then most of the time the TaxiBot would have traversed all the way to the starting node within $T_{response}$, however then the TaxiBot would have to wait at this starting node for $11-3=8$ minutes in most times, which is a waste of time. Therefore, 5 minutes is a good balance between being on time and not waiting too long.

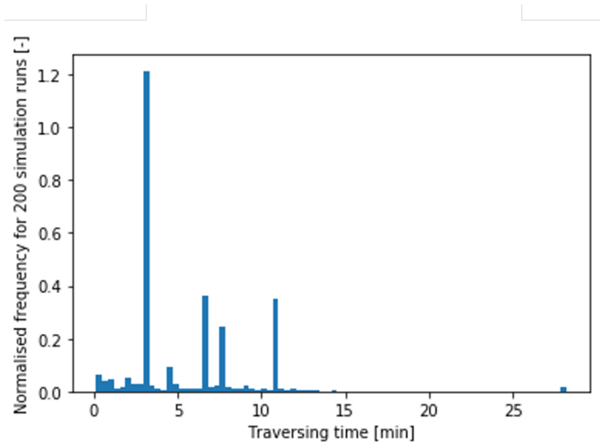

Figure 2.17: Histogram of traversing times for TaxiBots over 200 simulation runs.

3

Output Figures

First, section 3.1 shows visualisations of the strategic schedule from the base scenario. Followed by that, section 3.2 shows supplementary information regarding the comparison of the scenarios.

3.1. Base Scenario Analysis

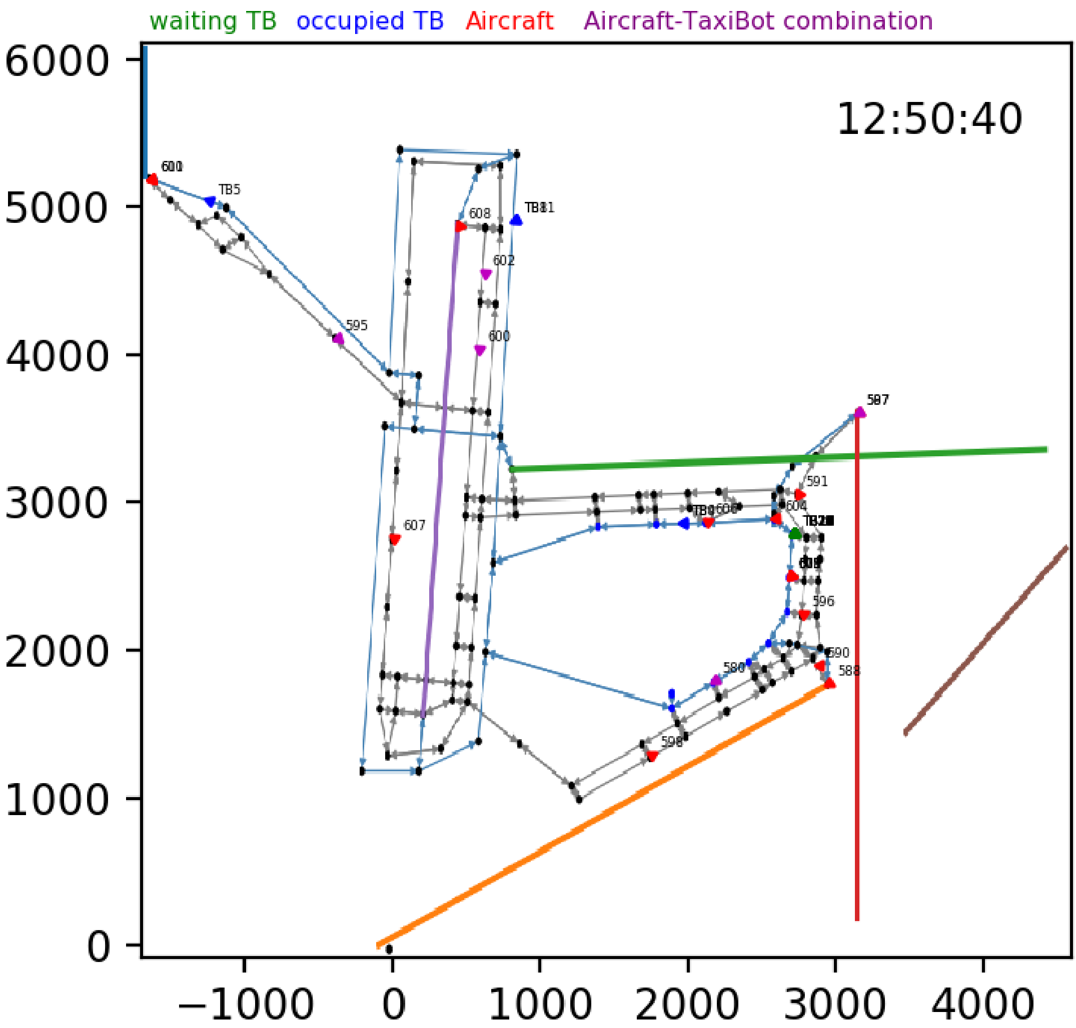

Figure 3.1: Screenshot of the animation video showing all vehicles on the airport.

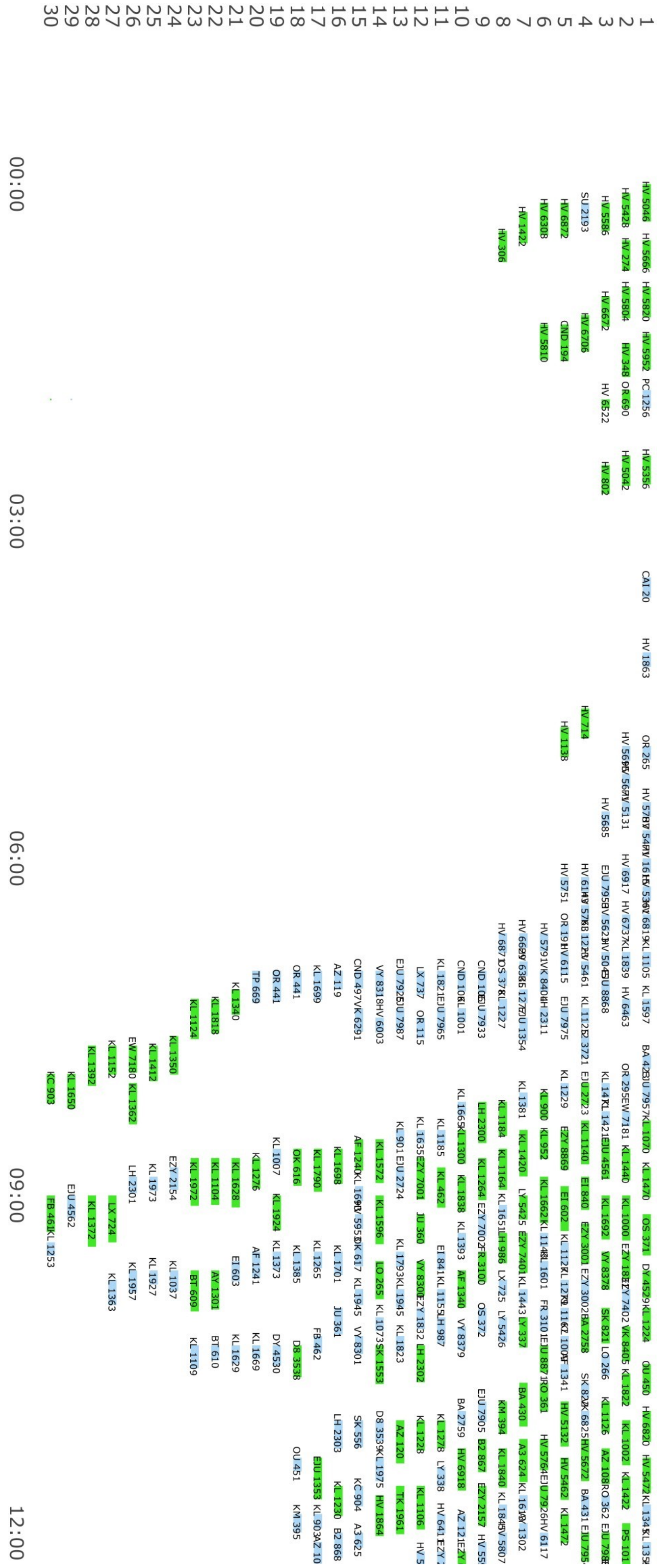

Figure 3.2: Visual representation of the scheduling of each certified aircraft to the pool of TaxiBots for the base scenario, part 1.

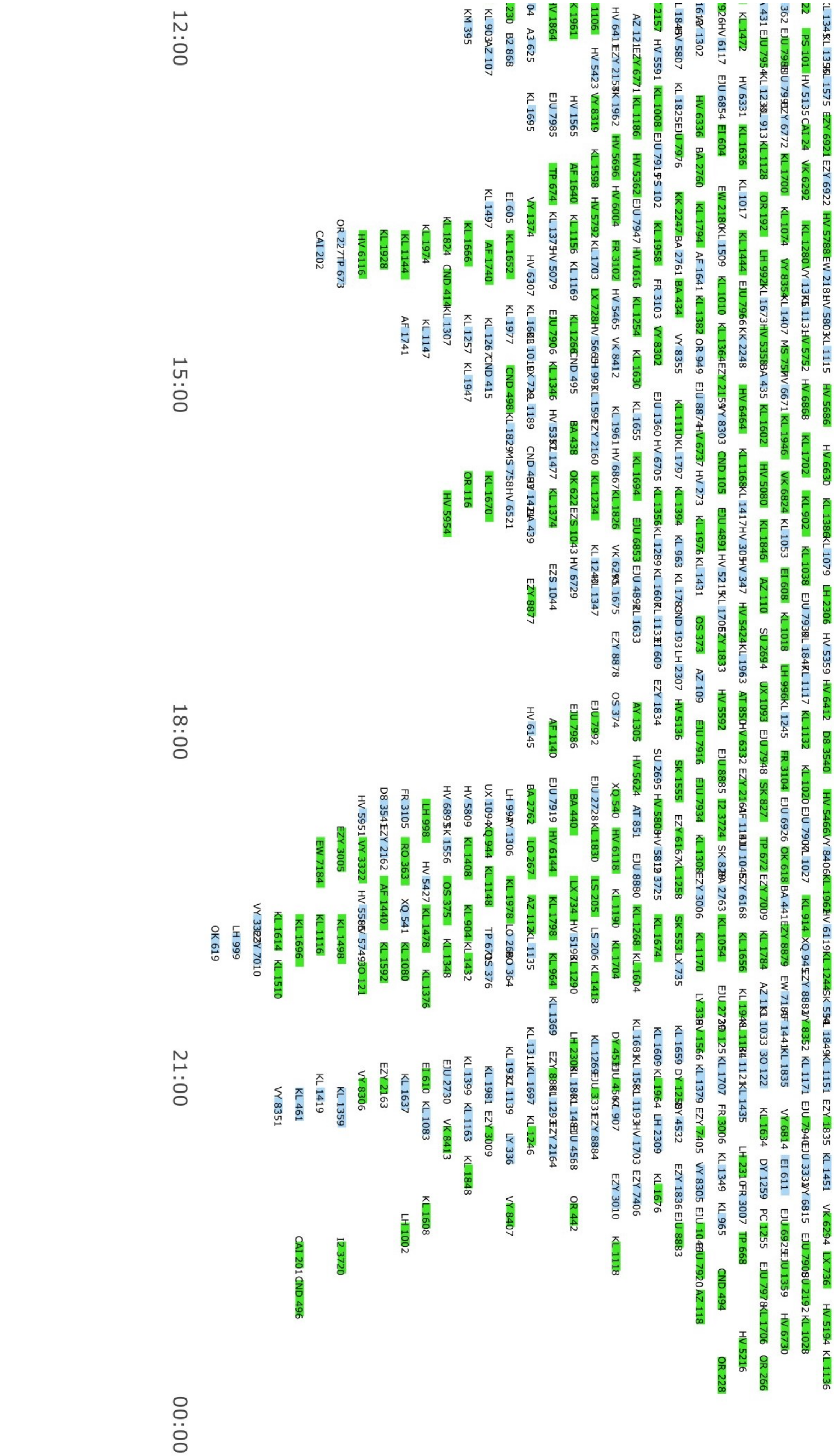

Figure 3.3: Visual representation of the scheduling of each certified aircraft to the pool of TaxiBots for the base scenario, part 2.

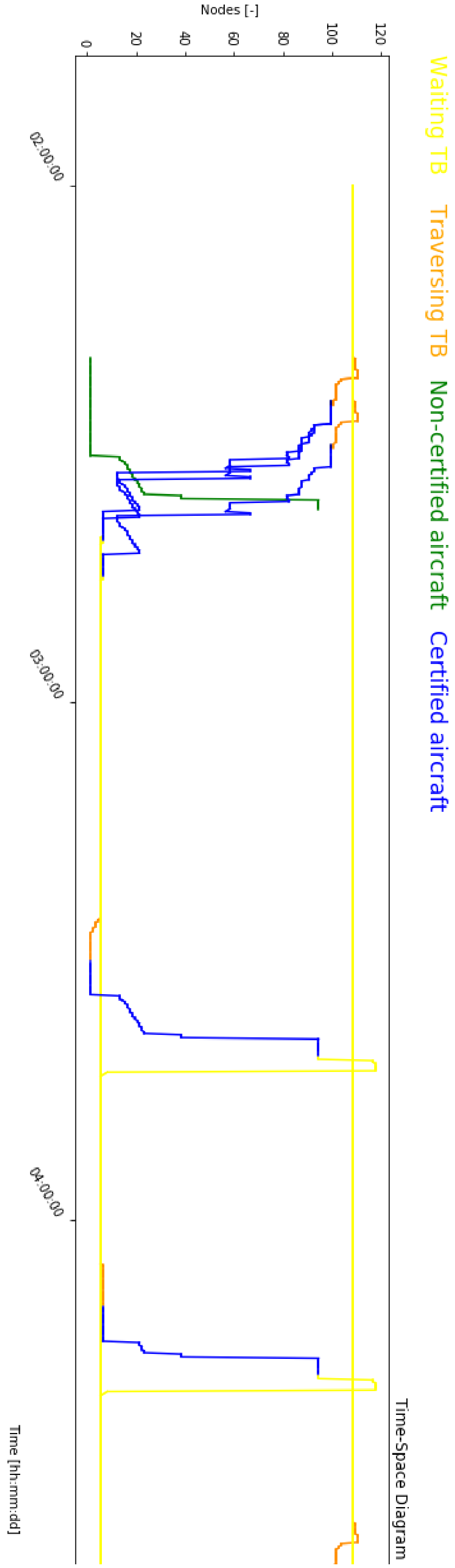

Figure 3.4: Part of the time-space diagram showing the routes of the vehicles, via the nodes over time, part 1.

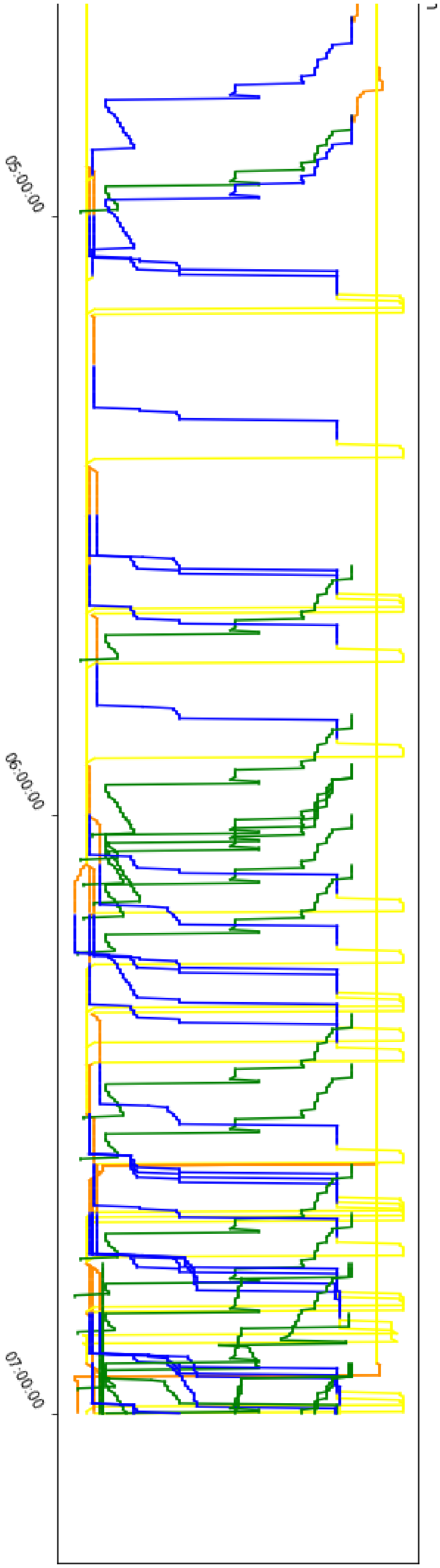

Figure 3.5: Part of the time-space diagram showing the routes of the vehicles, via the nodes over time, part 2.

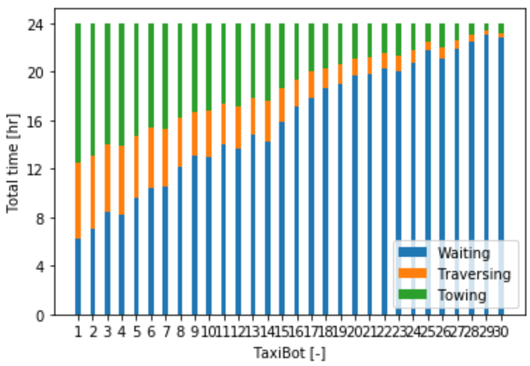

Figure 3.6: Occupation of each TaxiBot in the strategic schedule, divided into waiting, traversing and taxiing.

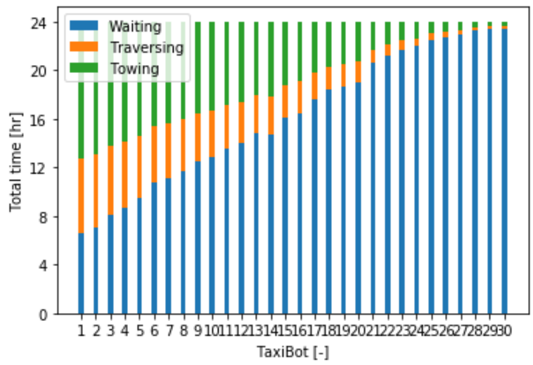

Figure 3.7: Occupation of each TaxiBot for 100 simulation runs of the tactical schedule, divided into waiting, traversing and taxiing.

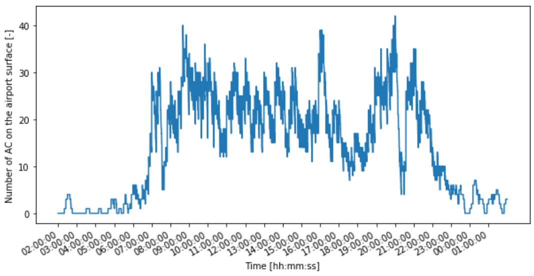

Figure 3.8: The number of aircraft that are present on the airport at each moment in time for the strategic schedule.

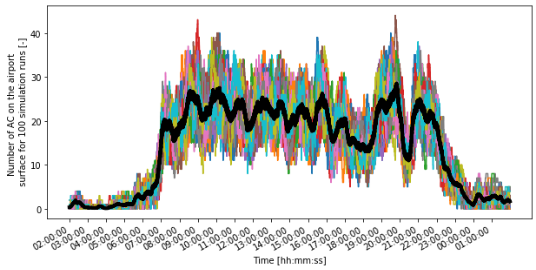

Figure 3.9: The number of aircraft that are present on the airport at each moment in time for 100 simulation runs of the tactical schedule.

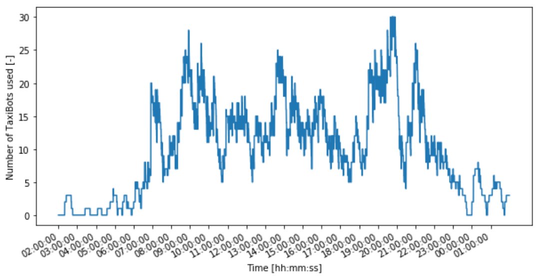

Figure 3.10: The number of TaxiBots that are in use at each moment in time for the strategic schedule.

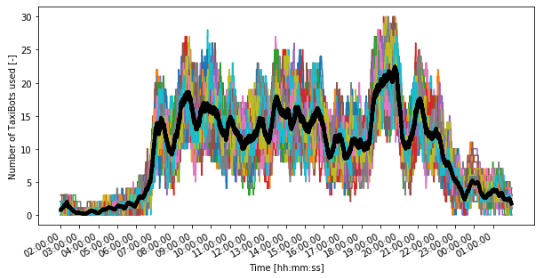

Figure 3.11: The number of TaxiBots that are in use at each moment in time for 100 simulation runs of the tactical schedule.

3.2. Scenario Case Studies

Table 3.1: Comparison of taxi parameters between strategic and tactical schedule, based on the base scenario #1.

Parameter	Strategic Schedule	Tactical Schedule
Total taxi time [hr]	328.5	317.9
Of which certified/non-certified [hr]	159.7 (48.6%) / 168.9 (51.4%)	153.6 (48.3%) / 164.3 (51.7%)
Total towing time [hr]	159.5	152.3
Of certified/total taxi time [-]	99.9% / 48.5%	99.2 % / 47.9 %
Aircraft taxied without a TaxiBot	1	7

Table 3.2: Comparison of taxi parameters between strategic and tactical schedule, based on scenario #2.

Parameter	Strategic Schedule	Tactical Schedule
Total taxi time [hr]	215.4	209.2
Of which certified/non-certified [hr]	105.5 (49.0%) / 109.9 (51%)	102.6 (49.1%) / 106.6 (50.9%)
Total towing time [hr]	105.5	102.0
Of certified/total taxi time [-]	100.0% / 49.0%	99.4 % / 48.8 %
Aircraft taxied without a TaxiBot	0	4

Table 3.3: Comparison of taxi parameters between strategic and tactical schedule, based on scenario #3.

Parameter	Strategic Schedule	Tactical Schedule
Total taxi time [hr]	318.4	304.8
Of which certified/non-certified [hr]	248.1 (77.9%) / 70.3 (22.1%)	234.9 (77.1%) / 69.8 (22.9%)
Total towing time [hr]	246.4	232.8
Of certified/total taxi time [-]	99.3% / 77.4%	99.1 % / 76.4 %
Aircraft taxied without a TaxiBot		12

Table 3.4: Comparison of taxi parameters between strategic and tactical schedule, based on scenario #4.

Parameter	Strategic Schedule	Tactical Schedule
Total taxi time [hr]	210.1	202.6
Of which certified/non-certified [hr]	156.6 (74.5%) / 53.5 (25.5%)	149.7 (73.9%) / 53.0 (26.1%)
Total towing time [hr]	156.6	148.6
Of certified/total taxi time [-]	100.0% / 74.6%	99.3 % / 73.3 %
Aircraft taxied without a TaxiBot	0	6

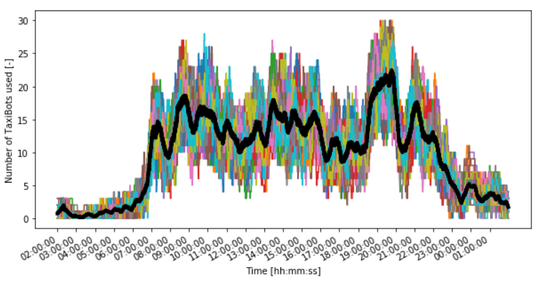

Figure 3.12: The number of TaxiBots that are in use at each moment in time for 100 simulation runs of scenario 1.

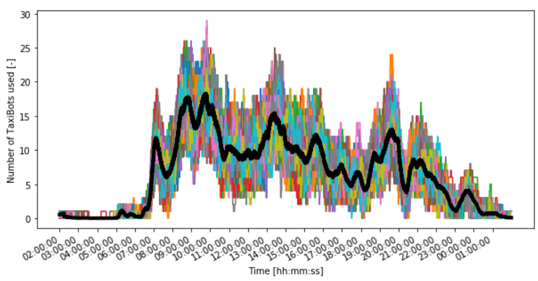

Figure 3.13: The number of TaxiBots that are in use at each moment in time for 100 simulation runs of scenario 2.

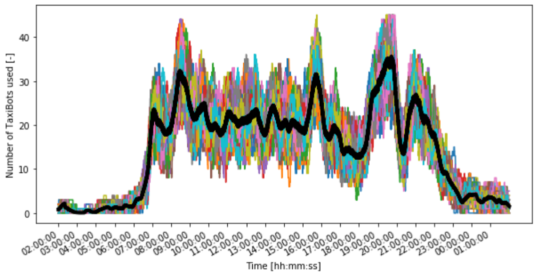

Figure 3.14: The number of TaxiBots that are in use at each moment in time for 100 simulation runs of scenario 3.

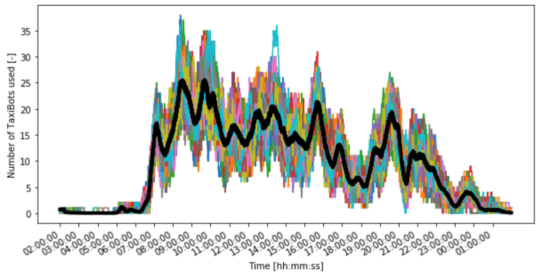

Figure 3.15: The number of TaxiBots that are in use at each moment in time for 100 simulation runs of scenario 4.

Bibliography

[1] F.J. Asensio et al. Optimal sizing and operation methodology for the on-board electrical generation and energy recovery system of an aircraft. *46th Annual Conference of the IEEE Industrial Electronics Society*, pages 4277–4282, 2020.

[2] J Atkin, E Burke, and S Ravizza. The airport ground movement problem: Past and current research and future directions. *Research in Air Transportation*, page 131138, 2010.

[3] M. Battipede et al. Innovative airplane ground handling system for green operations. *27th International Congress of the Auronautical Sciences*, pages 1–7, 2010.

[4] Kevin Beelen. - operational supervisor taxibot test at schiphol. Personal Conversation, November 2020.

[5] Kevin Beelen. - operational supervisor taxibot test at schiphol. sustainable aviation. University promotion presentation, September 2020.

[6] Bewoners-Aanspreekpunt-Schiphol. Baangebruikcijfers 2019. Available at https://bezoekbas.nl/wp-content/uploads/2020/02/Baangebruikcijfers_2019.pdf, February 2020.

[7] Richard Bihr. A conceptual solution to the aircraft gate assignment problem using 0,1 linear programming. *12th Annual Conference on Computers & Industrial Engineering*, 18(1-4):280–284, 1990.

[8] Dirk Bresser and Simon Prent. The benefits of sustainable taxiing. Available at https://www.schiphol.nl/en/innovation/page/the-benefits-of- sustainable-taxiing/ (10/12/2020), August 2020.

[9] Dirk Bresser and Simon Prent. Single-engine taxiing; wouldn't it be better? Available at https://www.schiphol.nl/en/innovation/page/single-engine- taxiing/ (10/12/2020), September 2020.

[10] Dirk Bresser and Simon Prent. What is sustainable taxiing? (part 1). Available at https://www.schiphol.nl/en/innovation/page/what-is-sustainable- taxiing-part-1/ (10/12/2020), October 2020.

[11] Dirk Bresser and Simon Prent. What is sustainable taxiing? (part 2). Available at https://www.schiphol.nl/en/innovation/page/what-is-sustainable- taxiing-part-2/ (10/12/2020), November 2020.

[12] Ulrich Dorndorf. Flight gate scheduling: State-of-the-art and recent developments. *Omega - The International Journal of Management Science*, 35:326 334, 2005.

[13] Ulrich Dorndorf, Florian Jaehn, Chen Lin, Hui Ma, and Erwin Pesch. Disruption management in flight gate scheduling. *Statistica Neerlandica*, 61(1):92–114, 2007.

[14] C. Evertse and H.G. Visser. Real-time airport surface movement planning: Minimizing aircraft emissions. *Transportation Research Part C*, 79:224–241, 2017.

[15] Damon Fordham et al. Deriving benefits from alternative aircraft-taxi systems - acrp report 158. Technical report, Airport Cooperative Research Program - Transportation Research Board, 2016.

[16] Charles N. Glover and Michael O. Ball. Stochastic optimization models for ground delay program planning with equityefficiency tradeoffs. *Transportation Research Part C*, (39):196–202, 2011.

[17] Bruce Goldberg and David Chesser. Sitting on the runway: Current aircraft taxi times now exceed pre-9/11 experience. Available at https://www.bts.gov/archive/publications/special_reports_and_issue_briefs/special_report/2008_008/entire (09/12/2020), May 2017.

[18] Jean-Baptiste Gotteland et al. Aircraft ground traffic optimization. *ATM 2001, 4th USA/Europe Air Traffic Management Research and Development Seminar*, pages 1–9, 2001.

[19] Volker Grewe et al. Transport impacts on atmosphere and climate: Aviation. *Atmospheric environment*, 44:4678–4734, 2010.

[20] Merlijn Groot. The impact of electric aircraft taxiing. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands, June 2020.

[21] Nico Guillaume. Finding the viability of using an automated guided vehicle taxiing system for aircraft. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands, April 2018.

[22] Rui Guo, Yu Zhang, and Qing Wang. Comparison of emerging ground propulsion systems for electrified aircraft taxi operations. *Transportation Research Part C*, (44):89–109, 2014.

[23] Thushara Kandaramath Hari, Zahira Yaakob, and Narayanan N Binitha. Aviation biofuel from renewable resources: Routes, opportunities and challenges. *Renewable and Sustainable Energy Reviews*, 42:1234–1244, 2015.

[24] Gerhard Hiermann et al. The electric fleet size and mix vehicle routing problem with time windows and recharging stations. *European Journal of Operational Research*, 252:995–1018, 2016.

[25] Jakub Hospodka. Cost-benefit analysis of electric taxi systems for aircraft. *Journal of Air Transport Management*, (39):81–88, 2014.

[26] Jakub Hospodka. Electric taxiing taxibot system. *MAD - Magazine of Aviation Development*, 2(10):17–20, 2014.

[27] M. Huang et al. Main wheel prerotation and ground taxi driven by electric taxi system. *Journal of Aerospace Engineering*, 32(6), 2019.

[28] M.I. MD Ithnan et al. Aircraft taxiing strategy optimization. *Department Maritime & Transport Technology, Technology University of Delft,*, pages 1–9, 2013.

[29] Yu Jiang, Zhihua Liao, and Honghai Zhang. A collaborative optimization model for ground taxi based on aircraft priority. *Mathematical Problems in Engineering*, 2013:1–9, 2013.

[30] Thomas F. Johnson. Electric green taxiing system (egts) for aircraft. Available at https://tec.ieee.org/newsletter/march-april-2014/electric-green-taxiing-system-egts-for-aircraft (3/12/2020), April 2014.

[31] Jurjen Kroese. Electric taxiing: an optimisation study on the future of airport operations. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands, January 2021.

[32] Prem Kumar and Michel Bierlaire. Multi-objective airport gate assignment problem in planning and operations. Technical report, Transport and Mobility Laboratory (TRANSP-OR) School of Architecture, Civil and Environmental Engineering (ENAC) École Polytechnique Fédérale de Lausanne (EPFL), 2012.

[33] Vivek Kumar, Lance Sherry, and Terrence Thompson. Analysis of emissions inventory for "single-engine taxi-out" operations. *Third International Conference on Research in Air Transportation*, pages 445–450, 2008.

[34] Shan Lan, John-Paul Clarke, and Cynthia Barnhart. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science*, 40(1):15–28, 2006.

[35] J. K. Lenstra and A. H. G. Rinnooy Kan. Complexity of vehicle routing and scheduling problems. *Wiley Periodicals Inc.*, 11(2):221–227, 1981.

[36] M. Lukic et al. State of the art of electric taxiing systems. *IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference*, pages 1–6, 2018.

[37] Milos Lukic et al. Analysis of energy storage system requirements for aircraft electric taxiing operations. *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal*, pages 2616–2621, 2019.

[38] Milos Lukic et al. Modelling and analysis of an aircraft on-board electric taxiing system. *22nd International Conference on Electrical Machines and Systems (ICEMS), Harbin, China*, pages 1–6, 2019.

[39] Milos Lukic et al. Review, challenges and future developments of electric taxiing systems. *IEEE Transactions on Transportation Electrification*, 5(4):1441–1457, 2019.

[40] Rami Sabet Mangoubi. *A Linear Programming Solution to the Gate Assignment Problem at Airport Terminals*, volume FTL REPORT R80-1. Flight Transportation Laboratory Massachusetts Institute of Technology, 1980.

[41] Messier-Bugatti-Dowti and Honeywell. electric green taxiing system presentation to arts et métiers. Available at https://www.arts-et-metiers.asso.fr/manifestation_cr/678_compte_rendu.pdf (11/12/2020), June 2013.

[42] Khusnul Novianingsih and Rieske Hadianti. Modeling flight departure delay distributions. *International Conference on Computer, Control, Informatics and Its Applications*, pages 30–34, 2014.

[43] Weijun Pan et al. The impact of green taxiing systems using on airport. *3rd International Conference on Electronic Information Technology and Intellectualization*, pages 284–290, 2017.

[44] Maria Nadia Postorino, Luca Mantecchini, and Filippo Paganelli. Improving taxi-out operations at city airports to reduce co2 emissions. *Transport Policy*, 80:167–176, 2019.

[45] S Ravizza, JAD Atkin, MH Maathuis, and EK Burke. A combined statistical approach and ground movement model for improving taxi time estimations at airports. *Journal of the Operational Research Society*, 64:1347–1360, 2013.

[46] F Re. Viability and state of the art of environmentally friendly aircraft taxiing systems. *Electrical Systems for Aircraft, Railway and Ship Propulsion*, pages 1–6, 2012.

[47] A Recalde et al. Energy storage system selection for optimal fuel consumption of aircraft hybrid electric taxiing systems. *IEEE Transactions on Transportation Electrification*, 2020.

[48] Paul Roling. Airport surface traffic planning optimization: a case study of amsterdam airport schiphol. *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, (7079):1–7, 2009.

[49] Paul Roling and Edzard Van Baaren. Design of a zero emission aircraft towing system. *AIAA AVIATION Forum*, pages 1–11, 2019.

[50] Paul Roling and H Visser. Optimal airport surface traffic planning using mixed-integer linear programming. *International Journal of Aerospace Engineering*, pages 1–11, 2008.

[51] Paul Roling, Pjotr Sillekens, and Richard Curran. The effects of electric taxi systems on airport surface congestion. *15th AIAA Aviation Technology, Integration, and Operations Conference*, pages 1–10, 2015.

[52] Paul Roling et al. Impact of electric taxi systems on airport apron operations and gate congestion. *17th AIAA Aviation Technology, Integration, and Operations Conference*, pages 1–9, 2017.

[53] Safran. Aircraft taxiing along with their engines shut down: Safran believes this is a winner! Available at https://www.safran-landing-systems.com/systems-equipment/electric-taxiing-0 (3/12/2020), na.

[54] Maximilian Schiffer and Grit Walther. The electric location routing problem with time windows and partial recharging. *European Journal of Operational Research*, 260:995–1013, 2017.

[55] P.J.A. Sillekens. Effect of egts on airport taxi movements at aas. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands, April 2015.

[56] Giuseppe Sirigu. *Planning and reconfigurable control of a fleet of unmanned vehicles for taxi operations in airport environment*. PhD thesis, Politecnico di Torino, Turin, 2017.

[57] Giuseppe Sirigu et al. Autonomous taxi operations: algorithms for the solution of the routing problem. *AIAA Information Systems-AIAA Infotech*, pages 1–7, 2018.

[58] J.W. Smeltink et al. An optimisation model for airport taxi scheduling. *Elsevier Science*, pages 1–25, 2004.

[59] Mojdeh Soltani. Electrification of airport operations: Electric powered tow-truck utilization in taxiing operations. Master's thesis, Concordia University - Mechanical and Industrial Engineering, Montreal, Canada, December 2019.

[60] Mojdeh Soltani et al. An eco-friendly aircraft taxiing approach with collision and conflict avoidance. *Transportation Research Part C*, 121:1–15, 202.

[61] TLD. Taxibot-nb undercarriage taxiing system - specifications sheet, October 2019.

[62] W.M.K. Trochim and J.P. Donnelly. *Research methods knowledge base*, volume 2. Atomic Dog Pub, 3 edition, March 2001.

[63] Parth Vaishnav. Costs and benefits of reducing fuel burn and emissions from taxiing aircraft. *Journal of the Transportation Research Board*, (2400):65–77, 2014.

[64] Edzard van Baaren. The feasibility of a fully electric aircraft towing system. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands, May 2019.

[65] Wouter van Lingen. Modelling the effects of gate pit-stops on apron congestion. Master's thesis, Delft University of Technology - Aerospace Engineering, Delft, The Netherlands, June 2019.

[66] Chris Wijnterp et al. Electric taxi systems: An operations and value estimation. *AIAA 2014-3266 Session: Airline Operations II*, (3266), 2014.

[67] J. Yan-Du, Jens O. Brunner, and Rainer Kolisch. Planning towing processes at airports more efficiently. *Transportation Research Part E*, 70:293–304, 2014.

[68] Stefano Zaninotto, Jason Gauci, Geoffrey Farrugia, and Johan Debattista. Design of a human-in-the-loop aircraft taxi optimisation system using autonomous tow trucks. *AIAA AVIATION Forum - Airport Surface Ops*, pages 1–13, 2019.