Agent-Based Modelling and Analysis of Non-Autonomous Airport Ground Surface Operations

Master of Science Thesis

B. Benda

Technische Universiteit Delft

TU Delft

to70

Agent-Based Modelling and Analysis of Non-Autonomous Airport Ground Surface Operations

Master of Science Thesis

by

Bram Benda

To obtain the degree of

Master of Science
in Aerospace Engineering

at the Delft University of Technology.

To be publicly defended on November 16th 2020 at 14:30 LT.

Student number	4583744	
Thesis committee:	Dr. B.F. Santos	TU Delft, Chairman
	Dr. O.A. Sharpanskykh	TU Delft, Supervisor
	Ir. X.R.I. Mobertz	To70, Company Supervisor
	Dr. J. Ellerbroek	TU Delft, External Examiner

Cover image is taken from [1] and modified.

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

Preface

Dear reader,

This report describes the research I conducted in the domain of agent-based modelling of airport ground surface movements and concludes my studies at the Delft University of Technology within the Aerospace Engineering department. This document consists of three separate parts of which the main work is discussed in the first part consisting of the MSc thesis paper. The second part details the literature study, as conducted in the first months of the research. At last, a set of appendices are included that are supportive of the thesis paper.

My interest in both the agent-based modelling paradigm and the developments in alternative ground propulsion systems for aircraft led to the idea for this study. I certainly believe that there is still a lot to come within this specific domain and I am more than curious to see where it will go for the aviation industry, especially during these difficult times. This study has certainly put an initial step in the direction to study the implementation of alternative means for aircraft taxiing.

First of all, I want to massively thank my supervisors Dr Alexei Sharpanskykh and Xander Mobertz for their support throughout the entire project. Their guidance within this project, both from an academic and industry perspective, was exceptional and certainly kept me on the right track.
Secondly, I would like to thank To70 for allowing me to conduct my research at an aviation-related company. This has unquestionably helped me in acquiring operational knowledge for this research but has also broadened my knowledge of the aviation industry and its related aspects. Aside from the thesis work and the changing times the last couple of months, I have had a great time within the company and would certainly recommend it to future students.
Furthermore, I want to express my gratitude to my family and friends for their never-ending support. The endless talks I have had with them allowed me to disconnect from my work whenever it was necessary and they were there to put all my concerns into perspective. A special thanks to my girlfriend Nathalie, who was and still is always there to support me to fulfil my ambitions and has supported me from the moment I started studying. Thanks to Yuri for not forgetting to exercise, cry with laughter now and then and being of huge support throughout the entire thesis period.

Now that this thesis has concluded my study period of over 8 years, I am very curious to see what the future holds. Now more than ever, we need to be ready to challenge the future!
Until we meet again,

B. Benda
November 2020, Delft.

Contents

List of Figures

List of Tables

List of Acronyms

A-CDM Airport Collaborative Decision Making.

A-SMGCS Advanced Surface Movement Guidance & Control System.

A-VDGS Automated Activation of Advanced-Visual Docking Guidance Systems.

AAS Amsterdam Airport Schiphol.

ABM Agent-Based Modelling.

ACISP Airport CDM Information Sharing Platform.

ADS-B Automatic Dependent Surveillance - Broadcast.

AGL Airfield Ground Lighting.

AGPS Advanced Ground Propulsion System.

ANSP Air Navigation Service Provider.

APU Auxiliary Power Unit.

ATC Air Traffic Control.

ATIS Automatic Terminal Information Service.

B&B Branch & Bound.

BIP Binary Integer Programming.

CBS Conflict-Based Search.

CPA Closest Point of Approach.

DV Decision Variable.

EGTS Electric Green Taxiing System.

FtG Follow-the-Greens.

GA Genetic Algorithm.

GC Ground Controller.

HMI Human Machine Interface.

IP Integer Programming.

ISMS Integral Safety Management System.

LP Linear Programming.

LVC Low Visibility Conditions.

LVNL Luchtverkeersleiding Nederland - Air Traffic Control Netherlands.

MILP Mixed Integer Linear Programming.

MIP Mixed Integer Programming.

MLG Main Landing Gear.

MMEL Master Minimum Equipment List.

MSL Mean Sea Level.

MTOM Maximum Take-Off Mass.

MTOW Maximum Take-Off Weight.

NB Narrow-Body.

NLG Nose Landing Gear.

RET Rapid-Exit Taxiway(s).

RL Reinforcement Learning.

RMO Runway Mode of Operation.

SESAR Single European Sky ATM Research.

SID Standard Instrument Departure.

SSR Secondary Surveillance Radar.

TAT Turn-Around Time.

TCL Taxiway Centreline Lights.

TRACC Taxi Routing for Aircraft: Creation & Controlling.

WB Wide-Body.

WTC Wake Turbulence Category.

List of Symbols

Symbol	Description	Units
AC_{acc}	Maximum acceleration level of Aircraft agent	m/s^2
AC_{cd}	Aircraft waiting time for Taxibot attachment after ready-signal	s
$AC_{dec,max}$	Maximum deceleration level Aircraft agents	m/s^2
$AC_{dec,comfort}$	Comfort deceleration level aircraft agents	m/s^2
AC_{td}	Aircraft taxi-distance	km
AC_{ts}	Aircraft average taxi-speed	m/s^2
AC_{tt}	Aircraft taxi-time	s
$AC_{v,max}$	Maximum taxi-speed Aircraft agent	m/s^2
$AC_{v,turn}$	Maximum taxi-speed through turns larger than turn-speed angle for Aircraft agents	m/s^2
D_{ind}	Individual distance travelled by Taxibot	km
$D_{ind,sr}$	Individual distance travelled by Taxibot on service-roads	km
$D_{ind,tw}$	Individual distance travelled by Taxibot on taxiways	km
D_{tow}	Towing distance of Taxibot	km
μ	Sample mean	-
R_c	Number of reassignments of a single task	-
$\rho_{airport}$	Average number of active vehicles on the airport surface	-
R_{max}	Maximum number of times a task can be reassigned to another agent	-
σ	Sample standard deviation	-
t	Time-point in simulator	s
$\Delta t_{arrival}$	Arrival time-window of Taxibot agent at allocated task	s
t_{couple}	Couple time of Taxibot	s
TA_{rate}	Rate at which Taxibot coordinator considers task allocation	s
$TA_{sorted,list}$	Boolean to indicate whether the list of Taxibot routes is sorted for nearest Taxibots first	-
τ	List for Taxibot coordinator of tasks to allocate	-
$TA_{active,TB}$	Boolean whether active Taxibots are included in task allocation	-
TB_{acc}	Maximum acceleration level of Taxibot agents	m/s^2
TB_{dec}	Maximum deceleration level Taxibot agents	m/s^2
TB_{td}	Taxibot taxi-distance	km
TB_{ts}	Taxibot average taxi-speed	m/s^2
TB_{tt}	Taxibot taxi-time	s
TB_{util}	Taxibot utilisation, i.e. percentage of operational time over total time	-
$TB_{v,max}$	Maximum taxi-speed Taxibot agent	m/s^2
$TB_{v,turn}$	Maximum taxi-speed through turns larger than turn-speed angle for Taxibot agents	m/s^2
$TB_{waiting,gate}$	Taxibot waiting time at the gate when aircraft is not yet ready for attachment	s
Δt_{future}	Time-window for upcoming task consideration of Taxibot coordinator agent	min
Θ_{turn}	Turn angle for segment	$^\circ$

I

MSc Thesis Paper

Agent-Based Modelling and Analysis of Non-Autonomous Airport Ground Surface Operations

B. Benda, *MSc Student, Delft University of Technology*
Dr. O.A. Sharpanskykh, *Assistant Professor, Delft University of Technology*
Ir. X.R.I. Mobertz, *Aviation Consultant, To70 Aviation The Hague*

***Abstract*—The highly fuel-inefficient aircraft taxi-phase causes the industry to consider alternative ground propulsion systems. External systems like the Taxibot are preferred over onboard systems as they do not require aircraft structural additions and recertifications. However, the operational implementation of external taxi-systems is expected to increase both traffic complexity and human workload for Air Traffic Control (ATC). Systematic assessment of operational safety and efficiency consequences of novel taxi-concepts employing automated external systems within a hub airport's ground surface operation has not yet been considered in existing research. This study has taken the first steps into the design and evaluation of a novel taxi-concept for outbound aircraft enabled by autonomous Taxibots. An Agent-Based model is created to explore the novel concept of operations within the ground surface operations of Amsterdam Airport Schiphol in the Netherlands. Four days of real-world data from ADS-B ground tracks have been used to simulate the novel taxi-concept for different operational scenarios. The aircraft taxi-time was found to significantly increase for outbound aircraft in the novel taxi-concept compared to real-world taxi operations, especially for large taxi-distances. The number of necessary Taxibots in the novel taxi-concept and the aircraft engine-off times significantly depended on the considered operational scenario. Distributed coordination and planning has shown to allow for safe and efficient guidance of heterogeneous vehicle types within an airport ground surface operation for increasing traffic complexity.**

Index Terms* — *Agent-Based Modelling | Aircraft Towing | Sustainable Airport Operations | Distributed Control | Multi-Agent Path Finding | Multi-Agent Pickup-and-Delivery

I. Introduction

Aircraft taxi operations are responsible for significant amounts of noise and pollution, having adverse effects on both the environment and airport surroundings [1], [2], [3]. Aircraft are calculated to spend 10-30% of their operational time taxiing on the airport ground surface, where an Airbus A320 is estimated to burn around 5-10% of its total fuel during the taxiing phase [4], [5]. In current taxi operations, an aircraft's main engines power the aircraft at a low power setting, resulting in lower fuel efficiency compared to cruise settings and significant noise at and around the airport [6], [7], [8]. Although the COVID-19 pandemic has had a significant impact on the aviation industry and significantly lowered the global number of flights [9], the total number of flights and corresponding airport taxi-times are expected to rise again in the coming years [9], [10]. Therefore, research focuses on methods to reduce the environmental impact of aircraft taxi operations.

Existing research considers the implementation of alternative ground propulsion systems to remove or reduce the need for aircraft to be powered by their main engines during the fuel-inefficient taxiing phase. Existing research distinguishes between two types of systems: systems installed on the aircraft itself, so-called onboard systems, and external systems that do not affect the aircraft's structure [11], [12]. Onboard systems are capable of reducing the fuel burn for mid-size aircraft by up to 2.5% [13]. The usage of onboard systems lowers aircraft pollution and improves aircraft mobility near the gates [11]. Major drawbacks of onboard systems are that they increase the aircraft structural weight and that the systems depend on the power capabilities of the Auxiliary Power Unit (APU), making these types of systems less suitable for both long-haul flights and large aircraft [11], [13], [14]. Besides, existing onboard systems allow for relatively low taxi-speeds [5]. Alternatively, external systems can be connected to tow an aircraft from the gate to its designated runway and vice-versa [7]. An external taxi-system is preferred over onboard systems, as it does not require structural modifications and renewed certifications of the aircraft [7] and can operate at higher speeds [5]. However, the implementation of external systems is expected to increase the airport traffic complexity and ATC workload, requiring the need for extra guidance and control of such vehicles [15], [16]. On one hand, the existing research focuses on the automation of external taxi-systems and the consequences of the implementation of such systems for conventional ATC procedures. Morris et al. [3] study the implementation of self-driving tugs for aircraft towing, planned and assisted by ATC controllers. Chua et al. [17] consider priority cues for automated tow tugs within an airport's ground surface operation, based on ATC expert judgement, and the impact of external taxi-systems on current airport operations and ATC workload [16]. Furthermore, existing research considers case studies to assess the implementation of external taxi-systems within a specific airport's ground surface operations. These case studies primarily focus on the quantification of potential fuel- and emission savings using pre-determined ICAO factors [15], [18]. The case studies often consider a small airport with only a single [15] or two runways [18] and a limited amount of external taxi-systems. Other studies solve for an optimal distribution of operational towing and conventional taxi-operations from a central perspective [19], [20]. However, these approaches do not take into account the interactions between different parties and the operational consequences of the implementation of automated external taxi systems within a large hub airport's ground surface operation. Systematic assessment of operational safety and efficiency consequences of novel concepts of airport surface movement operations, based on automated external taxi-systems, has not yet been done.

The objective of this study is *to design and evaluate a novel taxi-concept for outbound traffic enabled by autonomous Taxibots and controlled through distributed coordination and planning*. The Taxibot system is chosen as a reference system for this study due to its operational readiness and because it is currently being tested in the ground surface operations of Amsterdam Airport Schiphol (AAS) [21], [22]. The novel taxi-concept is considered for outbound traffic only, as outbound flights already require the allocation of an external truck for push-back at the gate in conventional taxi-operations.

The Agent-Based Modelling and Simulation (ABMS) paradigm is found to be most suitable to reach the research goal for the following reasons. First, the bottom-up approach of ABMS allows for analysis and understanding of global system performance emerging from local properties and interactions among autonomous agents [23], [24]. Agent-based models can capture emergent behaviour and are particularly suitable for systems with dynamics originating from flexible and local interactions [24], [25], as is expected for autonomous Taxibots. Furthermore, agent technology has proven to be well applicable in the traffic and transportation domain due to its spatially distributed nature, diversity of actors and interactions between them [24], [26]. Applications in the ground surface domain of airports have shown the ability of agent-based models to be easily expanded to larger systems [27], [28], [29]. Besides, it suggests a means to control complex flows of traffic without dependency on human capabilities.

An agent-based model is created in this study to implement the novel taxi-concept and assess the operational impact of the novel concept on the airport's ground surface operations. An agent-based model is specified by three parts [24], [30]. The first part constitutes a model of the environment, consisting of all objects without autonomous behaviour. Secondly, a set of agents with their local properties needs to be specified. At last, the mutual interactions between agents and agents and their shared environment has to be defined. The lay-out and ground surface movement operations of AAS have been adopted for this study. The agent-based model employs distributed coordination and planning for the guidance of traffic by locating an ATC agent at each intersection of the airport taxiway system. Furthermore, the conflict-based search algorithm is implemented to ensure conflict-free operations. The novel taxi-concept is implemented according to the Multi-Agent Pickup-and-Delivery paradigm, in which a set of agents (Taxibots) is responsible for carrying out a set of tasks (aircraft towing). Experiments are carried out to analyse the novel taxi-concept regarding aircraft-, Taxibot- and airport performance. Four operational days of input data, derived from real-world Automatic Dependent Surveillance-Broadcast (ADS-B) ground-tracks of aircraft taxi operations, are used to assess the novel taxi-concept performance for different operational scenarios. A comparison with real-world taxi operations allowed for an analysis of the effects of the novel taxi-concept on individual aircraft taxi performance. Furthermore, the operational consequences of the novel taxi-concept are discussed regarding the increase in total vehicle movements and how to accommodate the autonomous Taxibots within AAS' ground surface operations.

The remainder of this paper is build-up as follows. Section II proposes the methodological approach for this study. Section III details the agent-based model; a specification of the environment, agents and corresponding interactions is given. Section IV discusses the steps for model verification and validation. The experimental set-up is provided in Section V. Thereafter, Section VI analyses the results obtained from the experimental analyses. A discussion of the proposed methodology, model and results is given in Section VII. At last, the conclusions and recommended future work are detailed in Section VIII.

II. Methodological Approach

A step-wise methodological approach is proposed to meet the research objective of this study and consists of the following phases: Phase 1 resembles a preparatory activity to gather the necessary information to construct a novel concept for taxi operations and can be found in the appendices. The novel taxi-concept is formalised in phase 2 of the research and is delineated in Section II-A. The section finishes with a set of modelling assumptions that are made to translate the operational concept to an agent-based model. The formal concept is based on multiple modelling components and methods of which two are elaborated below; the Multi-Agent Path Finding component in Section II-B and the component of Multi-Agent Pick-up-and-Delivery in Section II-C. A clear understanding of the modelling components allowed for a formalisation of an agent-based model, which is translated into a computer model for simulation and analysis. These phases will be discussed from Section III onwards.

A. A Concept For Autonomous Towing of Outbound Aircraft at Amsterdam Schiphol Airport

The novel concept of operations is developed in collaboration with Amsterdam Airport Schiphol (AAS) experts. AAS is one of Europe's main hub airports and is located between densely populated urban areas [31]. The complaints of local residencies on AAS' air traffic keep increasing, requiring further measurements to reduce nuisance. With taxi-times of up to 20 minutes, the implementation of alternative ground propulsion systems could achieve significant fuel-burn and emission reductions. The implementation of a novel taxi-concept provides significant challenges due to the complex infrastructural network of AAS. Therefore, some general requirements for the novel concept are proposed first. Thereafter, a global overview of the novel taxi-concept is given and the differences with conventional taxi operations are detailed.

A novel taxi-concept should adhere to a set of requirements to minimise adverse effects on current ground surface operations due to changes in the turn-around process of outbound aircraft. These requirements are partly based on work conducted by Morris et al. [3]. First, the concept should have as little impact as possible on individual aircraft taxi performance. This foremost relates to taxi-time, as taxi-delay is costly for airlines and is to be kept at a minimum. Secondly, the Taxibot movements should cause as little nuisance as possible to taxiing traffic. This relates to potential traffic congestion at specific parts on the airport, due to extra vehicle movements caused by individual Taxibots. At last, safe operations must be guaranteed at all times; Taxibots should maintain a minimum separation with each other and other traffic at all times.

A global overview of the proposed taxi-concept and its relevant actors and interactions for an example outbound flight at AAS is depicted in Figure 1 and will be discussed accordingly. A Taxibot coordinator is in charge of allocating Taxibots to flights considered for operational towing. The Taxibots autonomously move from their parking facility towards the allocated gate via the service-roads[1]. After arrival at the gate, the Taxibot informs ATC and waits for the aircraft to signal ready. Whenever the aircraft's flight-crew communicates to be ready, the Taxibot starts attaching to the aircraft and locks in following-mode after completion of the procedure; the flight-crew is responsible for the aircraft's taxi-procedure as usual. After push-back at the gate, the Taxibot remains attached to the aircraft and serves as the aircraft's ground propulsion system during the remainder of the taxi procedure, whereas for conventional taxi-operations the push-back truck would detach from the aircraft and return to the parking facility. Near the runway, ATC commands the aircraft-Taxibot combination to stand still at a designated point for Taxibot detachment. After detachment, ATC instructs the autonomous Taxibot back to its parking facility, preferably via service-roads to minimise nuisance to taxiing traffic.

A set of modelling assumptions is made to translate the novel taxi-concept to an agent-based model (Section III) and are discussed here as they follow from the operational concept. First, the Taxibots are located at one of two parking facilities currently in use at AAS; one at the B-pier and one at the H-pier. The Taxibot decoupling points are modelled as close as possible to the locations obtained from AAS experts. These points are primarily located at the stopbar of each runway entry, except for departures from runway 36C. Departures for runway 36C use the P4 and P5 parking locations as decoupling points, which are the second to last nodes before the runway-entry as depicted in Figure 2[2]. Furthermore, some modelling assumptions are formalised:

[1]The service-roads resemble all roads between the Piers and gates on Schiphol Center and around the runways and are solely accessible for ground support vehicles. The service-roads are indicated in red in Figure 2.

[2]Appendix B visualises the infrastructural layout of AAS and the locations of P4 and P5. The stopbar locations are indicated by red lines.

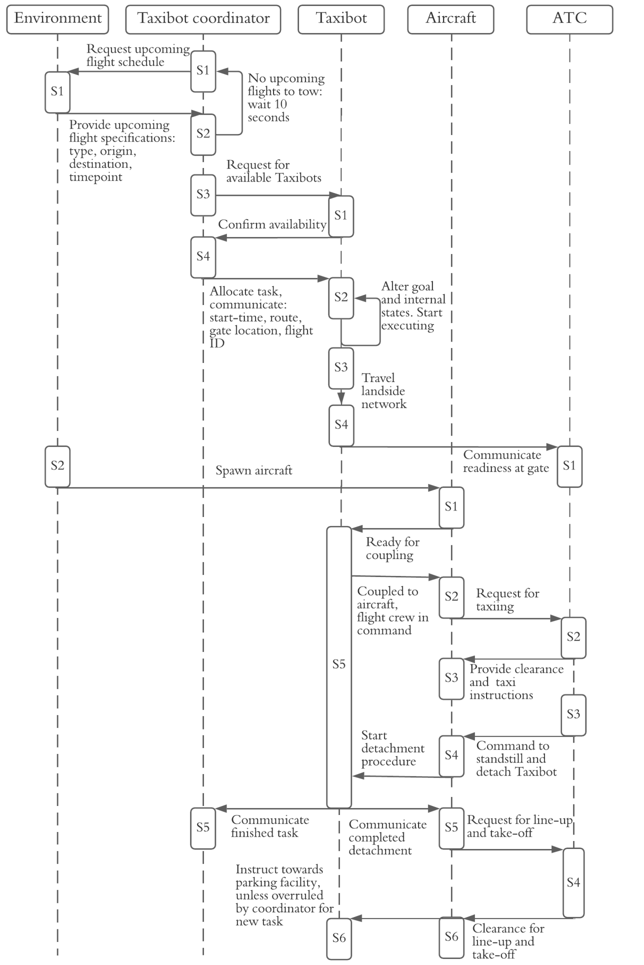


Fig. 1. Actor states and interactions in the novel taxi-concept for outbound flights. Section II-A discusses where the novel taxi-concept differs from conventional taxi-operations.

- The operational concept focuses on the most frequently used departure runways; 36L, 36C, 24 and 18L [32].
- Engine warm-up is assumed to be carried out during operational towing.
- Apron operations are excluded from the model, i.e. no operations at the gate are considered.

B. Multi-Agent Path Finding

To perform distributed coordination and planning in the agent-based model, a Multi-Agent Path Finding (MAPF) approach was used. MAPF resolves the problem of finding paths for all agents while avoiding conflicts [33]. The goal of a MAPF algorithm is to find the shortest conflict-free path for each agent. A path in our simulation model constitutes an aircraft route from its gate to the runway, or vice-versa, or a Taxibot route to or from its parking facility. The Conflict-Based Search (CBS) algorithm is adopted as the MAPF algorithm for this study due to its scalability with an increasing number of agents and its performance in airport-like maps [33]. The CBS algorithm has the goal to find the shortest conflict-free path for each agent and consists of two phases [33], [34]. The high-level phase considers all constraints, referring to unavailable segments within a graph, and the cost of all paths. The high-level search algorithm invokes the lower-level search algorithm for each path calculation. The low-level phase determines the actual path using the A* algorithm [35], given the set of constraints.

C. Multi-Agent Pickup-and-Delivery

Another important component of the agent-based model addresses the Multi-Agent Pickup-and-Delivery (MAPD) problem, where a set of tasks must be allocated to a set of agents in a specific environment and collision-free paths must be planned for these agents [36]. The representation of MAPD is adopted as it closely resembles the novel taxi-concept for aircraft from the perspective of the Taxibot; a single task is characterised by a pick-up location, drop-off location and a corresponding time-point [36]. In our implementation, this relates to an aircraft's gate (pick-up), the Taxibot detachment point near the runway (drop-off) and the starting time of the taxi-operations. Research distinguishes between centralised and decoupled, or decentralised, MAPD algorithms. In a decoupled MAPD algorithm, each agent assigns itself to a task and determines a collision-free path given the available information [37]. A decoupled approach is based on the local information of an agent, requires only limited communication and is more efficient for increasing model size [37]. A centralised algorithm assigns tasks to available agents from a central perspective and thus requires information on all respective agents regarding e.g. actual status, location and speed. For this study, a centralised algorithm is adopted to allocate tasks to individual Taxibots. It is assumed that the Taxibot technology is not yet sufficiently developed to allow for decentralised control, especially regarding necessary communications between individual Taxibots (e.g. via data-link). Besides, the number of Taxibots is expected to be relatively low such that a centralised approach allows for a valid initial consideration of the novel taxi-concept [38]. Furthermore, centralised approaches have been adopted in previous studies on ground surface operations [3]. The implemented algorithm will be elaborated upon in the next section.

III. Agent-Based Model

The aforementioned conceptual framework and modelling assumptions allowed for the construction of the agent-based model. A baseline agent-based model and simulator for conventional aircraft ground surface movements at AAS were used from previous studies [27], [28], [29]. The model and simulator were expanded and modified to implement and evaluate the novel taxi-concept. First, a specification of the environment is provided. Thereafter, the local properties of the agents and their corresponding interactions are discussed.

A. Environment specification

Figure 2 depicts the graph of the abstracted infrastructural network of Schiphol Airport as used in the agent-based model. The model distinguishes between taxiways (indicated by a thin black line), runways (indicated by a thick black line) and service-roads (indicated in red). The piers are modelled as meta-gates (blue dot in between red- and black lines) due to the exclusion of apron operations in the model, connecting the airside and landside part of the airport. Two purple dots can be seen in the middle of the airport layout, indicating the parking locations for the Taxibots.

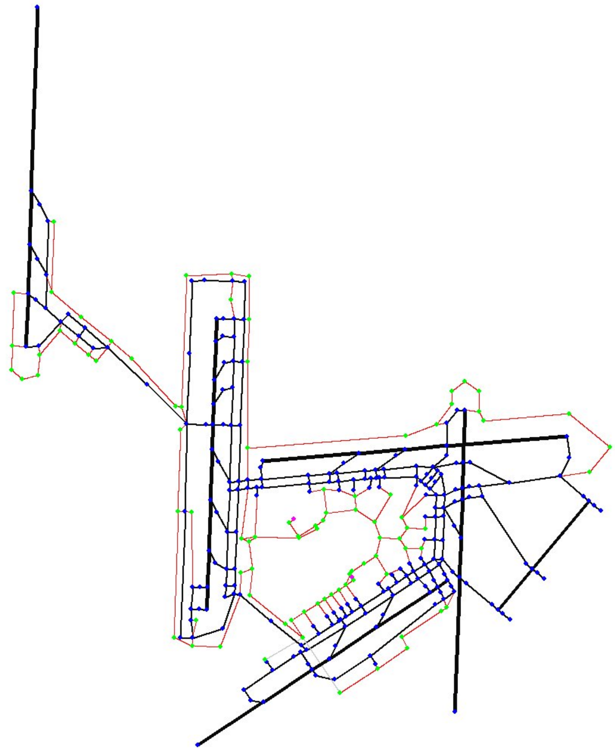


Fig. 2. Environment of the agent-based model with representations of the taxiways, runways and service-roads at Schiphol Airport.

The airside part of the environment with taxiways and runways is controlled by ATC agents located at each intersection. It comprises a dynamic environment due to the possibility for closure and re-opening of specific taxiway segments. All taxiway segments uni-directional roads and, therefore, allow for a single direction of traffic. The environment is fully accessible to ATC agents due to their ability to communicate and coordinate with each other. The landside part of the environment, i.e. the service-roads, is not controlled by ATC and thus separation between Taxibots must be maintained independently. The landside part of the airport resembles a static environment. All service-roads are dual-lane roads.

Each edge in the graph has a corresponding weight, denoting the expected traversal time of that segment. This weight depends on the traversal speeds of previous vehicle crossings of the segment and the type of vehicle.

B. Agent specifications

This section details the agent specifications of the agent-based model. First, the characteristics of each agent type are given. Thereafter, the local properties and corresponding interactions with other agents are discussed.

*1) **Source/Sink agents**:* The source and sink agents are responsible for insertion and removal of aircraft agents from the simulator and are located at the gates and runway nodes. The goal for the source agents is to release an aircraft as close to its planned spawn-time as possible. The actual spawn-times of aircraft are retrieved from a flight schedule. Occupancy times for runways and gates are implemented from previous studies [27], [28]. After runway/gate usage, a source or sink agent toggles an occupancy time during which the runway entry or gate cannot be used by any other vehicle.

*2) **Aircraft agents**:* The aircraft agents' goal is to travel from the gate to the runway, or vice-versa, as fast as possible within the operational limits of the aircraft. They receive speed and/or route commands from ATC and must adhere to these commands. The aircraft agents can observe their surroundings up to 250 meters and can act accordingly to maintain a safe distance from other traffic. Aircraft agents are based on an A320/B737 type of aircraft and are bounded by a maximum level of acceleration, deceleration, taxi-speed and turn-speed [27]. The aircraft agents store several dynamic aspects of their movements in their internal states, like taxi-speeds, taxi-time, delays, taxi-distances and (de-)coupling times of Taxibots.

Process Command Property: This property consists of interactions between aircraft agents and ATC agents. Aircraft agents are guided over the airport surface by ATC agents. ATC agents can communicate two types of commands to an aircraft agent: a speed command or a route command. Speed commands consist of a commanded speed at a specific distance. The aircraft agent determines what acceleration or deceleration level is needed for this particular speed and at which time it should start accelerating or decelerating. Without a speed command, the aircraft tries to taxi at its maximum speed. Route and heading commands require a specific heading change for an aircraft at a specific point in the network. The turn angle (θ_{turn}) is communicated in the specific ATC command, whereas the corresponding speed is decided for by the aircraft agent as depicted in Equation 1. *The Taxibot agents feature this same property.*

$$V_{taxi} = \begin{cases} V_{max}, & \text{if } \theta_{turn} < 30^\circ. \\ V_{turn}, & \text{if } \theta_{turn} \geq 30^\circ. \end{cases} \tag{1}$$

Conflict Avoidance Property: This property considers interactions between aircraft agents and aircraft agents and aircraft agents and Taxibot agents. The *Conflict Avoidance Property* is initiated whenever an aircraft observes another vehicle within its radar range. First, the intentions of the other vehicle are determined to decide whether this aircraft is being followed or the vehicle's path will be crossed. This happens through visual observation of the other vehicle. The aircraft agent decides what speed alteration is necessary to adhere to the minimum separation distances on the airport, based on conventional prioritisation rules. This means that the following vehicle is responsible for maintaining separation. For crossing conflicts, the nearest vehicle is prioritised if not solved by ATC. The *Conflict Avoidance Property* determines a necessary deceleration level to ensure a safe separation between the two vehicles. If a conflict resolution command had been given by an ATC agent, either a speed or route command, an aircraft agent primarily adheres to this and the *Conflict Avoidance Property* is made redundant. *The Taxibot agents feature this same property when travelling individually.*

*3) **Taxibot agents**:* The Taxibot agents have the goal to carry out their allocated tasks accurately and safely. The Taxibot agents are parked at one of the two parking facilities as discussed in Section II-A and visualised by the purple dots in Figure 2. No distinction is made between the two parking facilities. However, a Taxibot agent cannot change its parking location throughout the day to prevent unequal distributions of Taxibots. The Taxibot agents are based on the real-life Narrow-Body Taxibot [21] and are bounded by a maximum value for their acceleration, deceleration, turn-speed and taxi-speed. They maintain an internal model regarding their operational performance that can be communicated to the Taxibot coordinator agent. It is assumed that unlimited Taxibots are available, such that all flights can be covered and the amount of Taxibots necessary to facilitate operational towing for all outbound flights can be determined.

Attachment/Detachment Property: This property considers interactions between Taxibot agents and aircraft agents. Whenever an aircraft signals to be ready for attachment at the gate, it interacts with the Taxibot agent. If the Taxibot agent confirms its readiness, it begins the attachment procedure to the aircraft's Nose Landing Gear (NLG). Whenever attachment is completed, the Taxibot agent confirms with the aircraft agent and locks into towing mode; the aircraft agent is responsible for taxiing and the Taxibot control input is determined by the aircraft flight controls. The detachment procedure is initiated by the aircraft agent by communicating the need for detachment to the Taxibot agent. After completion of the detachment procedure, the Taxibot confirms with the aircraft agent the safe detachment from the aircraft.

Communicate Task Execution Property: This property involves interactions between Taxibot agents and the Taxibot coordinator agent. A Taxibot agent confirms task completion through a message to the Taxibot coordinator agent. The Taxibot coordinator internally processes this message and keeps track of completed tasks through the *Process Task Execution Messages Property*.

Communicate Mission Specifications Property: This property considers interactions between the Taxibot agent and the Taxibot coordinator agent. When returning at the parking facility, a Taxibot agent contacts the Taxibot coordinator agent that it is ready to park. The Taxibot agent reports its performance of the mission and, if approved, autonomously parks at a free parking spot.

*4) **Taxibot coordinator agent:*** The Taxibot coordinator agent is responsible for allocating Taxibots to upcoming outbound flights and resembles the paradigm of Multi-Agent Pickup-and-Delivery as discussed in Section II-C. It has the goal to allocate the available Taxibots such that aircraft do not incur any extra delay due to waiting. The Taxibot coordinator agent constantly communicates with ATC regarding upcoming flights and requests for vehicle routes and with Taxibots regarding their status and task performance.

Obtain Future Tasks Property: This property involves interactions between the Taxibot coordinator agent and ATC agents. Before the *Assign Task Property* can be carried out, the Taxibot coordinator agent retrieves the flight schedule for the upcoming 10 minutes from ATC. ATC communicates the expected flights within the time-window to the Taxibot coordinator, detailing for each respective flight:

$$F_i = \begin{bmatrix} Origin_i & Goal_i & Time-point_i & Arr/Dep_i & ID_i \end{bmatrix}$$

The $Origin_i$, $Goal_i$ and $Time-point_i$ properties denote the start- and goal location and the corresponding starting time of operations for $flight_i$. Arr/Dep_i indicates whether the flight concerns an arrival or departure flight. At last, ID_i indicates the unique flight-id of $flight_i$.

Unassign Task Property: This property involves interactions between the Taxibot coordinator agent and the Taxibot agents and the Taxibot coordinator agent and the ATC agents. The property is initiated whenever a task needs to be unassigned from a Taxibot agent. The Taxibot coordinator agent may find a more suitable (active) Taxibot agent for a task than the one currently assigned (see *Assign Task Property* for re-allocation reasoning). If this task must be allocated from one Taxibot agent to another, the currently responsible Taxibot agent must receive an update to dissolve its responsibility from the task, such that this task can be allocated to another Taxibot agent. The *Unassign Task Property* ensures the correct handling for task unassignment for the former responsible Taxibot agent, to prevent two Taxibot agents from travelling towards the same allocated task. Algorithm 1 presents the algorithmic implementation of the *Unassign Task Property*. The algorithm determines whether the considered $task_j$ can be unassigned from the currently responsible $agent_i$. If $agent_i$ is still at its parking location, the Taxibot coordinator commands $agent_i$ to remain parked and unassigns $agent_i$ from the task. If $agent_i$ is active, the Taxibot coordinator agent communicates with ATC_i whether a route towards the parking facility can be determined for this Taxibot agent, preferably via service-roads to minimise nuisance to other taxiing traffic. Successful route determination allows the Taxibot coordinator agent to communicate the altered route-plan to both ATC_i and $agent_i$. The algorithm returns a boolean indicating the successful unassignment of $task_j$, which is used in the *Assign Task Property*.

Update Property: At a pre-defined task allocation rate (TA_{rate}), the Taxibot coordinator agent requests all upcoming tasks within t+10 minutes from ATC and iteratively tries to allocate a Taxibot agent to each task by running the *Assign Task Property*. The algorithmic implementation is presented in Algorithm 2 and describes the connections between the Taxibot coordinator properties.

Algorithm 1 Unassign Task Property

$agent_i \leftarrow$ agent to unassign from $task_j$
$loc_i \leftarrow$ location of $agent_i$
$park_i \leftarrow$ parking facility for $agent_i$
$ATC_i \leftarrow$ ATC agent in command of $agent_i$
$unassign_j \leftarrow$ True
if $agent_i$ is at $park_i$ **then**
    $agent_i \leftarrow$ instructions from coordinator agent to remain parked
else if $agent_i$ is active **then**
    $R_i \leftarrow$ request ATC_i for route from loc_i to $park_i$, preferably via service-roads
    if R_i is found **then**
        $ATC_i \leftarrow$ communicate destination change $agent_i$
        $agent_i \leftarrow R_i$ instructions from ATC_i
        $agent_i \leftarrow$ available to new tasks
    else
        $unassign_j \leftarrow$ False
    end if
end if
return $unassign_j$

Algorithm 2 Taxibot Coordinator Update

$t \leftarrow$ current time-point
$TA_{rate} \leftarrow$ interval task allocation rate
$\Delta t_{arrival} \leftarrow$ Taxibot arrival time-window at task
if t is multiple of TA_{rate} **then**
    $\tau \leftarrow$ *future tasks within t+10 minutes*
end if
while task in τ **do**
    $\tau_{new} \leftarrow$ task
    $a_i \leftarrow$ *available Taxibot agents*
    $AssignTask \leftarrow \tau_{new}, a_i, \Delta t_{arrival}$
    Remove τ_{new} from τ
end while
Process Task Execution Messages

Assign Task Property: The *Assign Task Property* consists of interactions between the Taxibot coordinator agent and the Taxibot agents and the Taxibot coordinator agent and the ATC agents and represents the main property for task allocation in the agent-based model. The algorithm is depicted in Algorithm 3 and can be divided into three parts: initialisation, locating available Taxibot agents and task allocation. First, the algorithm initialises all necessary parameters (*lines 1-9*). Thereafter, the algorithm checks how many times the considered task has been re-allocated before. The re-allocation of tasks provides the Taxibot coordinator agent with a means to reconsider its initial task-allocation and re-allocate a task from one Taxibot agent to another. A task can be considered for re-allocation if the responsible Taxibot agent is still at its parking facility and the 'new' Taxibot agent is already active. The argument for re-allocation is to have less simultaneously active Taxibot agents on the airport and consequently a lower traffic density. In case of an arrival time-window, a new Taxibot agent must be expected to arrive within this time-window at the considered task. The absence of an arrival time-window allows the Taxibot coordinator agent to re-assign the task to any active Taxibot that is closer to the task. If the task is allocated to another Taxibot agent, the previously responsible Taxibot agent receives updated instructions to remain parked through the *Unassign Task Property*. A task can be re-allocated to another Taxibot agent at a maximum of one time. This has been implemented to prevent the Taxibot coordinator agent from constantly re-allocating a task when the algorithm is run, which would significantly increase computational times.

If a task has not been allocated too many times, the Taxibot coordinator agent loops over all available Taxibots and, if possible, calculates a path ($path$) and estimated time duration of travel ($path_{length}$) for each Taxibot (A_t) to the considered task location (*lines 10-25*). The Taxibot coordinator agent communicates with ATC and tries to minimise the Taxibot agent paths for taxiway crossings. Taxibot agents that cross more than three airside nodes are not considered in task allocation to prevent nuisance for taxiing traffic. The list of available Taxibots and their expected travel-time towards the task location is sorted on closest Taxibot agents first and is used for task allocation. In allocating the task (*lines 26-49*), the Taxibot coordinator checks if an arrival time-window is requested. If not, it allocates the nearest Taxibot agent to the task. Otherwise, the Taxibot coordinator starts iterating from the nearest Taxibot agent in the available Taxibots list and selects the first Taxibot agent that is estimated to arrive at the considered task-location within the requested time-window.

Process Task Execution Messages Property: This property consists of interactions between the Taxibot coordinator agent and Taxibot agents. A Taxibot agent communicates the task specifications after completion of its task to the Taxibot coordinator agent, as described in the *Communicate Task Execution Property*. Upon receipt of the message, the Taxibot coordinator agent processes the message and alters its internal state. The corresponding flight is checked and removed from the list of unexecuted-tasks.

5) ***ATC agents:*** The ATC Agents are responsible for safe and efficient operations on the airport's airside. Their goal is to handle as many vehicles as possible while maintaining safe and efficient operations. The ATC agents ensure traffic guidance and surveillance via distributed coordination and planning, with an ATC agent located at each airside node on the airport (Figure 2). An ATC agent is in command of all aircraft approaching the node and they determine the direction of the usage of each taxiway. Communication and coordination are required between the ATC agents to ensure safe operations and solve for potential conflicts. A total of three different ATC agent types and corresponding properties are implemented in the model. ATC endpoint agents are

Algorithm 3 Assign Task Property

1: $t_{task} \leftarrow$ start-time of task
2: $t \leftarrow$ current time-point
3: $loc_{task} \leftarrow$ location of task
4: $flight_{id} \leftarrow$ flight-ID of task
5: $R_c \leftarrow$ number of task reassignments
6: $R_{max} \leftarrow$ maximum number of task reassignments
7: $a_j \leftarrow$ responsible Taxibot agent for $task_j$
8: $TB \leftarrow$ available Taxibot agents
9: $\Delta t_{arrival} \leftarrow$ time-window arrival at task
10: **if** $R_c \leq R_{max}$ **then**
11:     $T_{to-task} \leftarrow$ empty list
12:     **for** A_t in TB **do**
13:         $loc_{A_t} \leftarrow$ current location A_t
14:         $success, path, path_{length} \leftarrow$ boolean if path is found, calculated path and time-duration from loc_{A_t} to loc_{task}, prefer service-roads
15:         **if** success is True **then**
16:             **if** A_t is active **then**
17:                 $nodes_{airside} \leftarrow$ total airside nodes in $path$
18:                 **if** $nodes_{airside} \leq 3$ **then**
19:                     $T_{to-task} \leftarrow path, path_{length}, A_t$
20:                 **end if**
21:             **else**
22:                 $T_{to-task} \leftarrow path, path_{length}, A_t$
23:             **end if**
24:         **end if**
25:     **end for**
26:     $T_{to-task} \leftarrow$ sort on shortest $path_{length}$ first
27:     **for** $path, path_{length}, a_{new}$ in $T_{to-task}$ **do**
28:         **if** $\Delta t_{arrival}$ is not None **then**
29:             **if** $t_{task} - \Delta t_{arrival} \leq path_{length} + t \leq t_{task}$ **then**
30:                 **if** a_j is not None **then**
31:                     **if** a_j is parked and a_{new} is active **then**
32:                         Unassign a_j from $task_j$ and instruct to parking facility
33:                         $a_{new} \leftarrow$ instructions for $task_j$
34:                         **return**
35:                     **end if**
36:                 **else**
37:                     $a_{new} \leftarrow$ instructions for $task_j$
38:                     **return**
39:                 **end if**
40:             **end if**
41:         **else**
42:             **if** a_j is not None **then**
43:                 **if** $path_{length,a_{new}} < path_{length,a_j}$ **then**
44:                     Unassign a_j from $task_j$ and instruct to parking facility
45:                     $a_{new} \leftarrow$ instructions for $task_j$
46:                     **return**
47:                 **end if**
48:             **else**
49:                 $a_{new} \leftarrow task_j$
50:                 **return**
51:             **end if**
52:         **end if**
53:     **end for**
54: **end if**

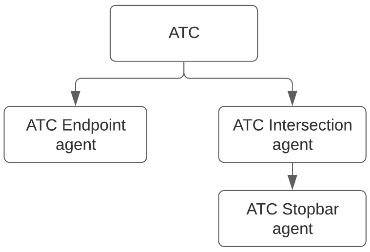


Fig. 3. Types of ATC Agents in ABM. All agents contain general ATC properties and type specific properties. ATC Stopbar agents inherit all properties from ATC intersection agents.

located at the endpoints of aircraft travels, i.e. at the gate and the runway entries. ATC intersection agents are located at each intersection in the airside network, whereas ATC stopbar agents are located at the second to last node towards the runway entry i.e. at the stopbar line. Some properties are type-specific, while others are general ATC agent properties; this is indicated below in between brackets.

Runway Crossing Closure/Opening Property [ATC]: This property consists of interactions between the ATC agents and the environment. Particular runway crossing segments can be closed or re-opened again based on the active Runway Mode of Operations (RMO). An example of such procedure is found for crossing W5 of runway 18C/36C (see Appendix B), which is closed during any direction of operations on this runway.

Conflict Based Search Property [Intersection]: This property involves interactions between ATC intersection agents. The CBS algorithm ensures conflict-free path-finding for aircraft- and Taxibot agents on the airport airside. The algorithmic implementation has been adapted from a previous study [28]. At each time-point, an ATC agent that has vehicles under its command forward simulates the vehicles' routes towards their endpoints. The forward simulation determines the time-point of passing at each future node of the vehicle's route and can be found in [28]. The expected time-passings of each vehicle are communicated with the other ATC agents, from which it is determined whether any two vehicles cross the same node within a time-window of 15 seconds; these vehicles are denoted as a potential conflict pair. The responsible ATC agent determines what type of conflict resolution command is needed and the *Determine Conflict Resolution Property* is run.

Determine Conflict Resolution Property [Intersection]: This property constitutes an internal reasoning property of ATC intersection agents and is initiated after an ATC agent received conflict data between any two vehicles. The obtained conflict data from the *Conflict Based Search Property* is internally processed by an ATC intersection agent to determine the necessary conflict resolution command. A speed command is given in case two vehicles cross nodes but are not expected to cause a gridlock, i.e. a head-to-head situation. In this case, the *Speed Command Conflict Resolution Property* is run. A head-to-head situation occurs when two vehicles are directed towards each other on a unidirectional taxiway segment. A gridlock causes the taxiway system to be blocked, as neither of the two vehicles can continue its travel and a reversed direction of travel is needed to resolve the gridlock. This procedure is very time consuming and, therefore, is to be avoided at all cost. To prevent this from happening, the *Route Command Conflict Resolution Property* is run to solve for potential head-to-head conflicts.

Prioritise Hand-off Queue Property [Intersection]: This property details a local property of ATC intersection agents. Each ATC intersection agent holds a hand-off queue which determines the priority of vehicle handover, i.e. which vehicle will be handed over to its next segment first. Due to the heterogeneous agent types (aircraft/Taxibot), the ATC intersection agents follow a set of priority rules to determine their hand-off queue. Aircraft agents receive priority over individual Taxibots at all times. This follows the concept requirements from Section II-A that individual Taxibots should cause as little nuisance as possible to other taxiing traffic. Priority between two aircraft is given to the aircraft that has been under command for the longest time. The prioritisation rules from [17] have been slightly modified to fit the goals for this study and were adopted, to determine priority between two or more Taxibot agents. The following order of priority rules is used, in case of no discrimination of the former:

1) Closest Taxibot;
2) Most number of trailing aircraft;
3) Taxibots on an active mission.

The latter cue prioritises Taxibots that are on active duty over Taxibot agents moving towards their parking facility. Each time a new vehicle becomes under command, the ATC agent checks whether the hand-off queue needs to be re-prioritised again. The queue is only reordered if safety can be maintained, i.e. the ATC agent checks if the previously prioritised vehicle(s) can safely stop at separation distance from the intersection.

Handover Property [ATC]: This property involves interactions between ATC agents. The *Handover Property* is initiated whenever an aircraft- or Taxibot agent reaches the ATC agent location. The property ensures that the agent under command is correctly handed over to the next ATC agent. Besides, the responsible ATC agent updates the weight of the segment in the graph with the vehicle's traversal time. The controlled vehicle agent receives final heading instructions towards its next ATC agent. The ATC agent contacts the next ATC agent and hands over the responsibility for the aircraft or Taxibot agent.

Issue Active Runway Crossing Property [Stopbar]: This property consists of interactions between ATC agents. In addition to the *Handover Property*, the *Issue Active Runway Crossing Property* is implemented for ATC stopbar agents. Vehicle handover from an ATC stopbar agent often requires a vehicle to enter or cross a runway. An ATC stopbar agent contacts the responsible ATC endpoint agent to request a vehicle handover towards the runway. Whenever the runway is available, this request is granted and the corresponding vehicle can be safely handed over to its next segment. A special situation occurs for the ATC stopbar segments to the north and south of runway 18C/36C, i.e. the Yankee and Zulu taxiways. In case of a vehicle handover to one of these segments, the ATC Stopbar agent determines the currently used RMO from its internal state. Whenever 18C is used for departures or 36C is used for arrivals, taxiing traffic via the Zulu taxiway must be issued an active runway crossing request from the responsible ATC endpoint agent. Similarly, inbound usage of 18C or outbound usage of 36C requires an issue for runway crossing via the Yankee taxiway. A runway crossing toggles an occupancy time during which the runway cannot be used by any other vehicle. Sequencing happens on a first-come-first-serve basis.

Initiate Taxibot Detachment Property [ATC]: This property involves interactions between ATC agents and aircraft agents that are powered by Taxibots. As indicated in Section II, a set of pre-determined detachment points for the Taxibot are used. Whenever an ATC agent notices that an aircraft agent is approaching the designated Taxibot detachment point, it commands the aircraft agent to stop at the corresponding location. Upon standstill, the aircraft agent is cleared by the ATC agent to start the Taxibot detachment procedure. After completion of the detachment procedure, the ATC agent hands-off respectively the Taxibot and aircraft agent towards their next segments.

Speed Command Conflict Resolution Property [Intersection] This property involves interactions between ATC agents and aircraft agents and ATC agents and Taxibot agents. This property determines the value of the speed command necessary to resolve an anticipated conflict between any two vehicles. The vehicle that is furthest away from the conflict node receives the speed command in case of no discrimination between the two vehicle types. The speed command is given to a Taxibot agent in the case of heterogeneous agent types. The responsible ATC intersection agent internally calculates the required value for the speed command. The ATC intersection agent uses the estimated arrival time of both vehicles at the conflict node, their current speeds, the remaining distance towards the conflict node for the vehicle to slow down and the required time-separation between the two vehicles. The ATC intersection agent determines the required speed over the remainder of the vehicle's taxi distance towards the anticipated conflict node and commands the vehicle to alter its speed.

Route Command Conflict Resolution Property [Intersection] This property consists of interactions between ATC agents and aircraft agents and ATC agents and Taxibot agents. This property decides which vehicle must receive a re-route command and determines a new route to resolve for an anticipated head-on conflict. The vehicle that is furthest away from the conflict node receives the route command, due to this vehicle being more flexible for alternative routes. A new path is found by the ATC intersection agent in which the conflict segment is removed from the graph; the vehicle is not allowed to travel this specific segment within its currently planned travel. The new route is communicated to the vehicle agent and the ATC agent currently in command of the vehicle is informed regarding the route change. The latter ATC agent is responsible for communicating a new heading command upon vehicle handover.

IV. Verification and Validation

Verification and validation of individual agent performance and the model implementation are carried out applying the procedural steps and techniques from Klügl and Sargent [39], [40]. The conceptual model was validated with experts and subsequently translated into a computer model. A new set of properties was implemented at each iteration and tested accordingly. Operational validation of the computer model and the agent performances was first carried out on a less complex airport layout, consisting of a single runway (06/24), before considering the entire AAS layout. Face validation of the computer model animation was carried out to analyse the model functioning for both the less complex- and expanded model. Furthermore, individual traces were utilised to check for agent behaviour and their interactions with other agents and the environment. Computerised model verification was iteratively carried out throughout the validation procedure by solving compile errors and through unit testing. Some model parameters have been obtained from interviews or via abstracted product information of the Taxibot and could have influenced model uncertainty. Therefore, local and global sensitivity analyses are carried out to determine the sensitivity of the model output to particular inputs and model assumptions. Section V describes the experimental set-up used for obtaining simulation results and the set-up for sensitivity analyses.

V. Experimental Set-up

This section presents the experimental set-up for analysis of the novel taxi-concept. Section V-A proposes a simulation plan for analysis of the model performance. The set of Key Performance Indicators (KPIs) for analysis of the model results are described in Section V-B. The statistical evaluation of simulation results is discussed in Section V-C. The model performance and validity are analysed for varying inputs and assumptions through sensitivity analyses. The set-up for these sensitivity analyses is elaborated in Section V-D.

A. Simulation Plan

Four days of real-world aircraft taxi-operations at AAS are used as input to test the model performance for various operational scenarios. The input data details the origin, destination, time-point and type per flight and follows a wave-like pattern of inbound and outbound flights, similar to flight operations at AAS (Appendix A). Table I indicates the data specifications. The specific days have been chosen due to their different RMOs throughout the day, with an emphasis on outbound traffic via runways 36L, 36C, 24 and 18L as assumed in Section II-A. The various RMOs and relatively high number of runway reconfigurations allow for analysis of the novel taxi-concept performance for varying circumstances per day. The availability of real-world data regarding aircraft performance provides a means for one-to-one comparison of aircraft performance in the novel taxi-concept and conventional scenario. The set of KPIs that are used is elaborated next.

TABLE I
SPECIFICATIONS OF OPERATIONAL DAYS FOR INPUT. THE DATA-SETS ARE OBTAINED FROM A PREVIOUS STUDY AND ORIGINATE FROM EXTENSIVE FILTERING OF RAW ADS-B DATA [27].

Input day	Flights	Outbound	Inbound	Rwy reconfigs
1 May	778	36C, 36L	06, 36R	19
2 May	869	18L, 24, 36L	06, 18C, 18R	18
7 May	843	18L, 24, 36L	06, 18C, 18R	20
13 May	803	36C, 36L	06, 36R	19

B. Key Performance Indicators

Analysis of the novel taxi-concept is carried out using a set of pre-determined KPIs. The focus has been put on the operational actors within the conceptual model to test for the taxi-concept performance; the performances of aircraft- and Taxibot agents are primarily analysed, as their operations significantly differ from conventional taxi operations. The number of vehicle crossings per segment is included to visualise the difference in traffic distribution due to the novel taxi-concept. The aircraft-related KPIs include:

- **Aircraft taxi-time** (AC_{tt}): the time of actual taxi-operation;
- **Aircraft taxi-distance** (AC_{td}): the distance covered during taxi-operation;
- **Aircraft average taxi-speed** (AC_{ts}): the average speed of the taxi-operation;
- **Aircraft couple-delay** (AC_{cd}): the waiting-time of an aircraft at the gate before a Taxibot starts attaching.

The first three KPIs allow for comparison in aircraft performance between the novel taxi-concept and real-world data of conventional aircraft taxi operations. The remainder of the KPIs cannot be compared to real-world situations due to the absence of data on the novel taxi-concept.
The second branch of KPIs focuses on Taxibot performance:

- **Active Taxibots**: the number of simultaneously active Taxibots throughout the day;
- **Taxibot utilisation** (TB_{util}): the percentage of operational time of an individual Taxibot over the day;
- **Taxibot taxi-distance** (TB_{td}): the distance covered by a Taxibot during operations. A distinction is made between individual- and towing distances and the distances covered on service-roads and taxiways;
- **Taxibot waiting time** (TB_{wait}): the total amount of minutes a Taxibot has to wait at the gate before its assigned aircraft signals to be ready for attachment.

At last, airport infrastructural performance is measured by the:

- **Traffic density** ($\rho_{airport}$): the traffic density throughout the airport network, either for aircraft, Taxibots or all vehicle movements.

C. Statistical Evaluation of Results

The Shapiro-Wilk test, D'Agostino's K^2 test and a visual inspection of quantile-quantile plots showed that the simulation outcomes did not follow a normal distribution (Appendix A). Therefore, the non-parametric Vargha-Delaney A-test is adopted for statistical evaluation of the simulation results. The A-test tests for stochastic equality between two samples. It compares two samples and determines a value between 0 and 1.0, where a value of 0.5 covers the hypothesis of stochastic equality between the two samples [41]. Values above 0.5 indicate a higher probability for a randomly selected observation from sample one to be a higher value than a randomly selected observation from the other sample. Values over 0.56, 0.64 or 0.71 indicate respectively a small, medium or a large difference between the two samples [42]. The same intervals apply for values below 0.5.

D. Sensitivity analysis set-up

The emphasis for the sensitivity analyses was put on the uncertainty in two agent types: the Taxibot agents and the Taxibot coordinator agent. The reason for this focus area is two-fold. First, the novel taxi-concept is centred around these two agent types. They are the responsible agents for both the planning and execution of the novel taxi-concept. Furthermore, no real-world data is available for neither of the two agent types. This absence required specific parameter assumptions, which could have significantly affected the model results. A previous study carried out an extensive sensitivity analysis on the performances of aircraft- and ATC agents [27] and is, therefore, excluded in this study.
A local sensitivity analysis is proposed for the analysis of the Taxibot coordinator agent properties and assumptions. Several assumptions were made, which makes the use of global methods impractical regarding time constraints. First, the model output sensitivity for varying values of the future task window (Δt_{future}) was tested. This value determines the time-window within which the Taxibot coordinator can allocate tasks and forms a key parameter of the Taxibot coordinator's internal update and task allocation consideration (Algorithm 2). Secondly, the maximum number of times a task can be re-allocated was considered (R_{max}). The task allocation algorithm (Algorithm 3) is dependent upon this maximum number of task allocations, and, it was expected that allowing the Taxibot coordinator agent to reassign a task multiple times lowers the Taxibot waiting times at the gate due to more accurate route predictions. $\Delta t_{arrival}$ was implemented based on the assumption that it is not desirable from an airport perspective to have Taxibots park at the gate for long periods, due to limited parking spots. Multiple variations in arrival time-windows are tested for their output sensitivity, as well as the absence of an arrival time-window. Furthermore, three intrinsic properties of the task-allocation algorithm are tested for output sensitivity. The TA_{rate} considers the rate of task-allocation and was expected to majorly influence computational times as it determines the update rate of the Taxibot coordinator's internal states. The baseline value of 10 seconds has been adopted from a previous study [3]. The Taxibot agent

TABLE II
TAXIBOT COORDINATOR AGENT LOCAL SENSITIVITY VALUES. NONE INDICATES THE ABSENCE OF A TRUCK-ARRIVAL TIME-WINDOW, I.E. EXCLUDING THE ASSUMPTION FROM THE MODEL.

Parameter	Units	Base	Local values
Δt_{future}	min	10	6, 8, 12
R_{max}	-	1	0, 2, 3
$\Delta t_{arrival}$	sec	60	30, 45, 70, 90, None
TA_{rate}	sec	10	20, 30, 40, 45, 50, 55, 60
$TA_{withactiveTBs}$	-	Yes	No
$TA_{sortedlistID}$	-	Yes	No

dynamical properties determine the capabilities of individual Taxibots, as well as the aircraft capabilities when powered by a Taxibot. The dynamical properties for the Taxibot agents were estimated from technical specifications of the Taxibot [21] and interviews with company representatives (Appendix C). Therefore, the model sensitivity is tested for variations in all dynamical properties. The Taxibot dynamical properties constitute the acceleration- and deceleration level (TB_{acc} and TB_{dec}), maximum taxi-speed ($TB_{v,max}$) and maximum turn-speed ($TB_{v,turn}$). These properties are varied within a range of +/- 20% of their baseline value. Furthermore, it was found from the interview that the Taxibot couple-times extremely vary in practice. Therefore, the couple-time t_{couple} is varied within a larger range of +/- 50%.

TABLE III
TAXIBOT AGENT LOCAL SENSITIVITY VALUES.

Parameter	Units	Base	Local values
TB_{acc}	m/s^2	0.41	0.33, 0.49
TB_{dec}	m/s^2	1.23	0.99, 1.48
$TB_{v,max}$	m/s	11.8	9.5, 14.2
$TB_{v,turn}$	m/s	5.9	4.7, 7.1
t_{couple}	sec	60	30, 90

Simulations are run for days 7 and 13 until 12:00 LT. This reduced flight schedule decreased computational effort significantly by a factor of 8, allowing the multiple sensitivity analyses to be run within a reasonable time. The flight schedule reduction does not affect simulation accuracy, as the reduced flight schedule consists of multiple RMO changes and the morning inbound and outbound peaks in traffic. The results are presented in Section VI-B.

VI. MODEL RESULTS AND ANALYSIS

The aforementioned simulation plan was implemented in the computer model and each of the four days of operational data were used as input to the simulator. The computer model is implemented in Python 3.8. The results are presented in Section VI-A. The results of the sensitivity analyses are presented in Section VI-B.

A. Simulation results and analysis

It was found that for all four input days the model was able to successfully simulate all aircraft and Taxibot movements safely without violating minimum separation distances. The performances on aircraft, Taxibot and airport level are discussed next.

1) Aircraft performance : The aircraft performance in the novel taxi-concept is compared with real-world taxi-data for each input day of operations. Table IV depicts the mean and standard deviation of the aircraft KPIs while Table V shows the statistical test values for comparing the two scenarios. The table distinguishes in inbound (A), outbound (D) and all flights (all). Table IV shows that the average AC_{tt} is largest for input days 1 and 13 in both scenarios A (real-world data conventional operations) and B (novel taxi-concept). Besides, the absolute differences between the two scenarios are most prominent for these days. This can be explained by two factors. First, during both days the 36L runway was active for the majority of the time for outbound traffic, requiring large taxi-distances. Furthermore, outbound traffic is towed by the Taxibot in the B-scenarios; the lower maximum speed of an aircraft due to Taxibot towing lowers the AC_{ts} and increases the AC_{tt}. Due to the relatively large taxi-distances towards runway 36L, the differences in AC_{tt} are largest for days 1 and 13 between the A- and B-scenarios; AC_{tt} is significantly higher for outbound traffic in the B-scenarios. The lower AC_{ts} for outbound traffic are confirmed by the A-test values, indicating a low probability for a higher taxi-speed in the B-scenarios for all days.

TABLE IV
SIMULATION RESULTS FOR AIRCRAFT TAXI-TIME (AC_{tt}), TAXI-DISTANCE (AC_{td}), AVERAGE TAXI-SPEED (AC_{ts}) AND COUPLE-DELAY (AC_{cd}). EACH DAY IS INDICATED BY THE DATE NUMBER AND A LETTER; A: REAL-WORLD- AND B: TOWING-SCENARIO.

		1A	1B	2A	2B	7A	7B	13A	13B
AC_{tt}	μ	8.20	9.78	7.50	7.77	7.58	7.75	8.03	9.38
[min]	σ	4.93	6.54	3.88	3.28	3.80	3.12	4.85	6.16
AC_{td}	μ	4.09	4.50	3.95	3.96	3.92	3.97	4.02	4.39
[km]	σ	2.49	2.85	2.79	2.78	2.58	2.64	2.54	2.86
AC_{ts}	μ	8.67	8.48	8.53	8.01	8.58	8.10	8.63	8.40
[m/s]	σ	2.38	2.11	3.48	3.62	3.22	3.33	2.45	1.90
AC_{cd}	μ	-	18.7	-	18.0	-	17.9	-	17.9
[sec]	σ		2.8		1.3		1.0		1.0

The A-test values for inbound traffic indicate a probability for a significantly higher AC_{ts} for inbound traffic in the B-scenarios on input days 1 and 13. Visual observation of the simulator showed that the exclusion of inbound-holding in the simulation model caused this higher AC_{ts}. In the real-world scenario, aircraft can be held at the P-platform to wait for gate-availability (inbound) or their departure slot (outbound). This forces an aircraft to a standstill, leading to a significantly lower AC_{ts}. The concept of holding is not implemented in the simulation model, allowing inbound traffic to directly travel towards their allocated gate and explaining the higher average taxi-speed of inbound traffic.

The average taxi-distance AC_{td} does not significantly differ for input days 2 and 7 when comparing the two scenarios. This was expected, as the implementation of the novel taxi-concept does not change routing strategies. However, a difference in average taxi-distance can be seen for days 1 and 13 in the B-scenarios, primarily caused by outbound traffic. This difference can be explained by a lack of information regarding future runway usage in the ATC agents model implementation. The ATC agent considering a runway crossing for an aircraft under its command checks if the corresponding runway is active. If the runway is active, the runway crossing segment cannot be used and the ATC agent reroutes the aircraft, often increasing the taxi-distance. This situation occurred for outbound traffic for runway 36L, which can either cross runway 18C/36C at the middle via runway crossing W5 for the shortest route (see Appendix B) or taxi for significantly longer via the northern Yankee taxiway surrounding 18C/36C. In the B-scenarios, an ATC agent immediately reroutes an aircraft via the Yankee taxiway whenever the 18C/36C runway is active. In scenario A, an ATCo often takes into account a future or temporary re-opening of the W5 crossing and commands the aircraft to wait before using the W5 crossing [27]. This lack of future information on runway usage caused the average taxi-distance to be slightly higher for outbound traffic in the B-scenarios of days 1 and 13, compared to the values in the A-scenarios.

At last, the AC_{cd} is indicated for the B-scenarios for each day of operations and shows to be nearly equal for each day. Together with low σ values, this indicates that the Taxibot coordinator agent is consistent in its time-estimation for Taxibots at their allocated tasks. The stability of AC_{cd} values around 18 seconds can be explained by the time it takes for a Taxibot to move from its parking location at the gate to the aircraft NLG. If the Taxibot arrives before the aircraft signals to be ready, the responsible ATC agent instructs a Taxibot agent to park at a safe distance of 30 meters from the communicated task location. Whenever the aircraft signals to be ready, the Taxibot agent moves from standstill to the aircraft's NLG, taking approximately 18 seconds before actual attachment starts.

TABLE V
A-TEST VALUES FOR AIRCRAFT PERFORMANCE. NOVEL TAXI-CONCEPT COMPARED TO REAL-WORLD DATA. DISTINCTION IN ARRIVALS (A), DEPARTURES (D) AND ALL FLIGHTS (ALL) FOR THE FOUR INPUT DAYS.

	f_{type}	1	2	7	13	Total
AC_{tt}	A	0.37	0.47	0.47	0.36	0.44
	D	0.69	0.63	0.63	0.69	0.62
	All	0.56	0.53	0.53	0.55	0.54
AC_{td}	A	0.49	0.50	0.51	0.47	0.49
	D	0.58	0.50	0.50	0.58	0.54
	All	0.54	0.50	0.50	0.54	0.52
AC_{ts}	A	0.76	0.53	0.54	0.71	0.61
	D	0.30	0.34	0.33	0.32	0.34
	All	0.47	0.47	0.46	0.46	0.47

2) Taxibot performance : Figure 4 depicts the total amount of simultaneously active Taxibots throughout each day of operations. It indicates the amount of Taxibots required to facilitate the novel taxi-concept. The wave-pattern of the AAS flight schedule is visible, showing clear peaks in active Taxibots around 10:00 and 21:00 for the 1st and 13th of May. Similar patterns for both the 1st and 13th of May and the 2nd and 7th of May are visible in the amount of Taxibots, explained by the corresponding RMOs for both days. It can be seen that the peak-difference is highest for May 1 and 13. This has to do with the average length of an outbound mission; an average Taxibot mission takes significantly more time on these two days, limiting the number of tasks an individual Taxibot can carry out. These days require up to 31 Taxibots to facilitate aircraft towing. The consequences of the added vehicle movements are discussed in Section VI-A3.

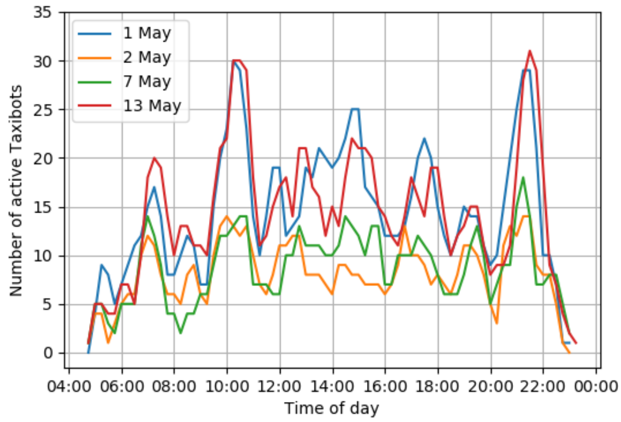


Fig. 4. Number of active Taxibots per day of operations. The number of active Taxibots provides an indication of the number of Taxibots needed to facilitate operational towing on each day.

Figure 5 depicts the Taxibot utilisation and the average number of completed tasks per Taxibot. The average Taxibot utilisation shows to be constant around 25-30%, with the lowest value for the 7th of May with 25 % utilisation on average. The high fluctuations between individual Taxibot utilisation can be appointed to the fluctuating demand for Taxibot towing. A significant amount of Taxibots is needed to facilitate aircraft towing during an outbound traffic peak. Outside of these outbound-peaks, a relatively low number of Taxibots is operational. This causes some Taxibots to be utilised only during periods of high-demand i.e. in outbound peaks. This is reflected by the minimum utilisation of around 1-5% for some days; some Taxibots are only used for a single mission throughout the day. From the right part of Figure 5 it can be seen that especially on 2 and 7 May the average Taxibot completed a significant amount of tasks during the day. This can be appointed to the outbound RMOs for these days; the average mission time is shorter in comparison to the other days, allowing Taxibots to carry out more tasks. A single Taxibot towed up to 52 aircraft on the 2nd of May. From an operational perspective, it could be interesting to further investigate what the environmental benefits could be of implementing a single (or few) Taxibot(s) into the daily operations. From a Taxibot operational perspective, procedures like utilisation levelling could be investigated to ensure equal depreciation of the vehicles.

Table VI depicts the total distances covered by Taxibots on the airport ground surface. In line with the discussion above, it can be seen that the Taxibots covered significantly more distances on days at which the 36L runway is in use for outbound traffic (i.e. 1 and 13 May). For both days, the towing distances of over 2800 km indicate that the aircraft engines can remain off for a significant amount of time in the novel taxi-concept compared to conventional taxi-operations. Besides, the fraction of towing distance over total distance is around 20% higher for May 1 and 13. This suggests a careful consideration for Taxibot

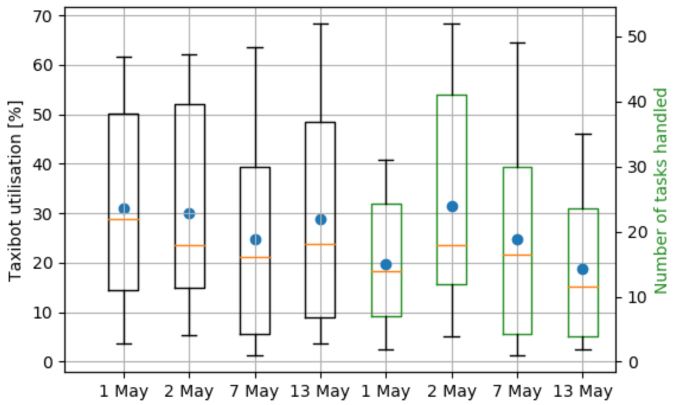


Fig. 5. Taxibot utilisation (black) and the average number of tasks completed (green) for Taxibot performance. The Taxibot utilisation is determined as the operational time over an entire day.

implementation with specific RMOs, as for some cases only a slight benefit in aircraft engine-off time may be achieved at the cost of significant distances covered by Taxibots and increments in total vehicle movements on the airport.

A requirement for the novel taxi-concept was to minimise nuisance to other taxiing traffic. The total amount of airport airside distance covered by Taxibots is relatively small for all four days of operations, as depicted in Table VI. This shows that ATC agents can efficiently guide Taxibots from the decoupling location to a service-road, without requiring to travel large distances on taxiways. The distances are primarily dependent on the location of the Taxibot decoupling point. A drawback for take-offs from 18L and 36L, at different points than the runway head, and take-offs from runway 24, is the need for the Taxibot to travel a short distance over the taxiway system and/or cross the runway. Although the total taxiway distance is small, this runway crossing requires the aircraft to wait for the runway to be cleared again. Future studies should consider the implementation of a planning mechanism to take decoupling and runway crossing into account in the aircraft departure schedule.

TABLE VI
TOTAL DISTANCES COVERED PER DAY IN KM: TOWING DISTANCE (D_{tow}), INDIVIDUAL DISTANCE TAXIBOT (D_{ind}) AND THE FRACTION OF TOWING DISTANCE OVER TOTAL DISTANCE, WITH D_{ind} FURTHER DIVIDED IN: ON SERVICE-ROADS ($D_{ind,sr}$) AND ON TAXIWAYS ($D_{ind,tw}$).

Day	D_{tow}	D_{ind}	$\frac{D_{tow}}{\sum(D_{tow}+D_{ind})}$	$D_{ind,sr}$	$D_{ind,tw}$
1	2864	4257	42.5	4198	58
2	650	2732	19.2	2617	115
7	777	2947	20.9	2807	139
13	2854	4297	39.9	4230	65

Figure 6 presents the average waiting time for Taxibots at the gate per day of operations. Although the Taxibot coordinator uses a 60-second time-window for Taxibot arrival at its allocated task, the waiting time shows to be a little higher, averaging 1 minute and 12 seconds. The values are clustered in between 1 and 2 minutes, which can be explained by the following reasoning. The Taxibot coordinator agent solely has access to the upcoming flight schedule within 10 minutes, excluding any gate-delay of the aircraft. If two aircraft are scheduled from the same meta-gate within a short time-window, it frequently occurs that the second aircraft is slightly delayed. The attachment procedure and corresponding gate occupancy of the release of the first aircraft cause the second aircraft to be delayed. Therefore, the Taxibot allocated to the second aircraft has to wait for a longer period. However, this waiting time is shown to be relatively short compared to the turn-around time of an aircraft. Thus it can be concluded that the Taxibot coordinator successfully allocates Taxibot agents to upcoming tasks without increasing aircraft delay at the gate.

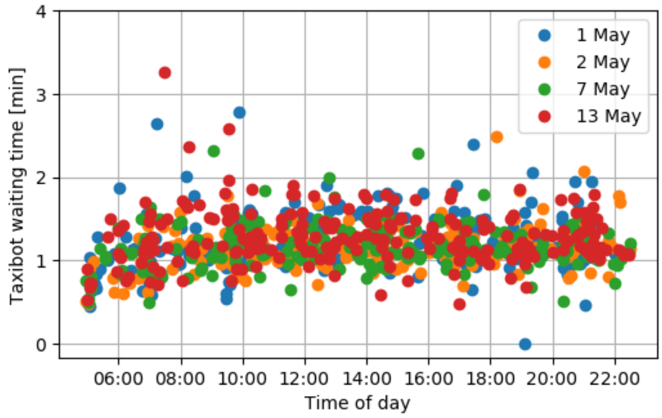
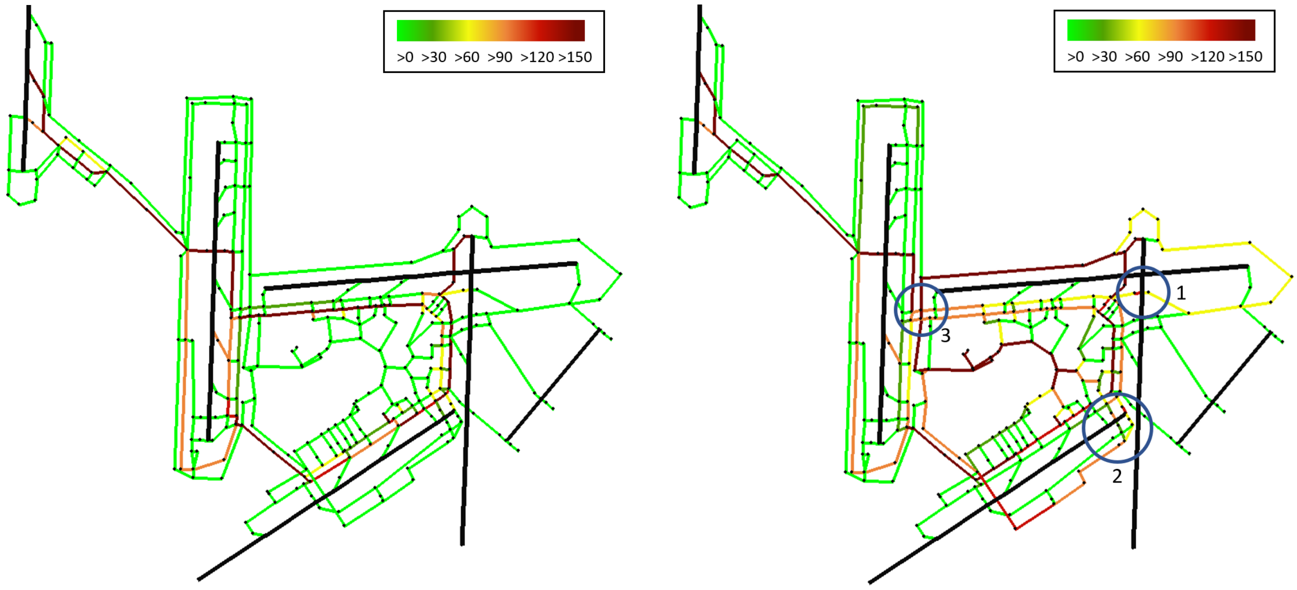


Fig. 6. Taxibot waiting time in minutes per day of operations.

3) Airport performance : Figure 7 visualises the total amount of vehicle crossings per airport segment, i.e. the traffic density, for the real-world and novel taxi-concept scenarios on May 7th (Appendix A for other days). The implementation of autonomous Taxibots can be seen in Figure 7(b); the novel taxi-concept significantly increased the number of movements on the service-roads, solely accessible to Taxibots. Three cases are discussed that required an alteration of the model assumptions or should be considered in future operational procedures for the novel taxi-concept. First, consider the areas circled 1 and 2. For these runway entries, AAS experts indicated that Taxibot decoupling should happen at the Alfa or Bravo taxiways, i.e. the circular taxiways surrounding Schiphol Centre. After decoupling, the Taxibot would return to the parking facility or a next task via the A/B taxiways and consequently via a gate. Due to the opposite directions of travel of outbound traffic and individual Taxibots, system grid-locks often occurred in the simulator. Therefore, the decoupling points for these runway entries had been moved one node further down the aircraft route, i.e. the last node before the runway entry point. The altered decoupling location caused Taxibots to cross the active runway to travel towards the service-road network, indicated by the increased number of movements crossing the runways. Although this requires aircraft to wait longer for take-off, safety and efficiency are not affected by system grid-locks. This conceptual alteration, however, requires careful consideration at a runway entry point



(a) Real-world scenario

(b) Novel taxi-concept

Fig. 7. Total number of vehicle passings per segment for input day 7.

with outbound traffic from two directions. Such a situation can be expected to occur at the runway head of runway 24; one aircraft intended to start Taxibot detachment and take-off from runway 24, while another aircraft (originating from gate 2, the lower part of the airport) wishes to cross runway 24 to travel towards runway 18L. Future research should focus on how to accommodate both decoupling and outbound traffic from two directions at the same runway entry point.

Another operational consideration is the fact that for some return routes the Taxibots **need** to cross active taxiways. An example situation is indicated for Taxibots returning from decoupling at the 18L runway head (*3*, the dark-red line from 18L above 09/27). The priority rules used by ATC allow for a safe crossing of the Taxibots, prioritising taxiing aircraft in all cases to minimise the nuisance of the crossing Taxibots. However, the significant amount of Taxibot movements on this particular crossing (*>150*) suggests the need for infrastructural alterations within the network. Any accident within this part of the airport network could lead to major problems.

B. Results Sensitivity Analysis

The parameter variations as described for the sensitivity analysis set-up in Section V-D have been implemented in the simulator. All sensitivity results are compared to the baseline-scenario with the novel taxi-concept for each consecutive day. The statistical results are presented in Appendix A. First, the results of variations in the Taxibot coordinator agent parameters are presented. Thereafter, the sensitivity results of varying parameters of the Taxibot dynamics are elaborated upon.

1) Variations in Taxibot Coordinator Agent Assumptions: Varying the input values for respectively Δt_{future}, TR_{max} and $\Delta t_{arrival}$ resulted in negligible changes to aircraft, Taxibot and airport performances (Appendix A). Neither of the parameter variations presented in Table II was proven to cause significant changes in the model output for the considered days of operations. It can be concluded that the task-allocation algorithm is insensitive to an altered future task time-window and that more reassignments of tasks are not deemed necessary for the current task-allocation implementation.

Increasing the TA_{rate} significantly benefited the overall computational time, as was expected; increasing the TA_{rate} from 10 to 20 seconds reduced the computational time by almost a third. Variations in the TA_{rate} affected the TB_{ts}, TB_{util} and $TB_{waiting,gate}$. It was found that for increasing values of the TA_{rate}, the TB_{ts} slightly increased while the $TB_{waiting,gate}$ decreased. This can be explained by the fact that the Taxibot coordinator agent is less successful in ensuring the timely arrival of the Taxibot at its allocated task. Due to the TA_{rate} being nearly equal to $\Delta t_{arrival}$, the task allocation algorithm is iterated at a rate nearly equal to the arrival time-window of Taxibots at their task, suggesting the deviation from the baseline KPI performances. Removal of the $\Delta t_{arrival}$ assumption confirmed that variations in the TA_{rate} do not change simulation results; further increasing the TA_{rate} without the need for a specific $\Delta t_{arrival}$ time-window stabilised the Taxibot performance for both input days (Appendix A). However, compared to the baseline performance, TB_{ts} greatly reduced due to significant more waiting time for Taxibots at the gate. Also, $\rho_{airport}$ was found to increase due to more active Taxibots. Therefore, the requirement for an arrival time-window can be justified; removal of this assumption leads to the unwanted consequence of an increase in airport density.

Allocating upcoming tasks solely to parked Taxibots, i.e. turning $TA_{with,active,TBs}$ off, led to a lower TB_{η} and a significantly lower $TB_{waiting,gate}$. This suggests that the Taxibot coordinator agent is more accurate in a correct time estimation of Taxibots from the parking facility compared to time estimation of active Taxibots. However, it requires more

Taxibots and hence slightly increases the average airport density. Therefore, employing only parked Taxibots is concluded to be not desirable.

The $TA_{sorted,dist}$ parameter considers a sorting of the list of available Taxibots based on their travel-distance from the considered task. The model output was shown to be insensitive to $TA_{sorted,dist}$ and thus concludes to be an unnecessary model assumption.

2) Variations in Taxibot Dynamics: Both the TB_{acc} and TB_{dec} parameters had little effect on the model performance. With regards to aircraft performance, the only KPI that is significantly affected on both input days is the AC_{cd}. This has to do with the time-duration of the Taxibot movement between the parking location at the gate and the NLG of the allocated aircraft, as discussed in Section VI-A; a higher acceleration or deceleration level shortens the time-duration of this travel. TB_{ts} is found to be most affected by variations in TB_{acc} and TB_{dec}, especially evident for input day 7. This is related to the RMO's on this input day, requiring Taxibots to travel shorter segments with relatively many turns, increasing the effect of a higher acceleration and deceleration value.

$TB_{v,max}$ is found to have a small effect on aircraft performance, which is more visible for lower values affecting $AC_{tt,d}$ and $AC_{ts,d}$. This relates to larger taxi-distances for which a lower $TB_{v,max}$ has more impact on the average aircraft taxi-speed and taxi-time. This is visible for input day 13, for which a significant rise of $AC_{tt,d}$ occurred due to a lower $TB_{v,max}$ and large outbound taxi-distances. Increasing $TB_{v,max}$ significantly increased $TB_{v,avg}$, with an even more pronounced effect for May 7th; the individual part of the total distance covered by Taxibots was larger for this day, explaining the larger effect of parameter variation. Increasing $TB_{v,max}$ was expected to cause Taxibots to return to their parking facility faster, lowering the airport density. However, it is found that the changes in airport density due to $TB_{v,max}$ variations were small.

$TB_{v,turn}$ is found to significantly affect $TB_{v,avg}$; higher turn speeds allowed Taxibots to travel their individual routes faster. This effect is most prominent for routes with frequent turns, as confirmed by the results of input day 7. The aircraft performances showed unaffected by varying $TB_{v,turn}$.

At last, variations in t_{couple} were found to significantly affect overall model results. An increase of 50% of t_{couple} significantly increased AC_{tt} and decreased AC_{ts} for outbound traffic on both considered days. Besides, the Taxibot performance is significantly dependent on t_{couple}; increasing the couple time significantly lowered TB_{ts}. It can be concluded that the uncertainty in t_{couple} is of high importance to the model results; accurate values from real-world testing are necessary to increase the model accuracy [22].

The sensitivity of the model output to variations in $TB_{v,max}$ and t_{couple} was further investigated for simultaneous variations of these parameters. A variation of $TB_{v,max}$ was expected to also influence the value for TB_{acc}, due to both depending on the Taxibot power levels. Therefore, a total of three parameters were simultaneously varied: TB_{acc}, $TB_{v,max}$ and t_{couple}. The aircraft performance is found to primarily depend on variations in a single Taxibot parameter. It can be seen from Figure 8 that the isolines are nearly straight for both considered days; there are no parameter interactions between TB_{acc} and $TB_{v,max}$ and variations in TB_{acc} have negligible effect on the model performance at all. The model output is influenced by varying either $TB_{v,max}$ or t_{couple}. Also, the specific day of operations is found to influence the overall model results; varying $TB_{v,max}$ had significantly more impact on the outbound aircraft taxi-time for day 13 compared to day 7, as can be seen from the higher density in isolines. This can be explained by the active RMOs for both days; outbound traffic travelled significantly larger distances on input day 13, due to runway 36L being used most of the time. The larger taxi-distances cause a varying $TB_{v,max}$ to have a direct effect on the average taxi-time, explaining the higher density of isolines. The active runways on day 7 (18L and 24) required relatively short taxi-distances, causing variations in $TB_{v,max}$ to have less effect on outbound aircraft taxi-time and consequently a lower density of the isolines. These conclusions are also found for AC_{ts} and suggest the need for careful parameter calibration for different operational scenarios when using the model (Appendix A).

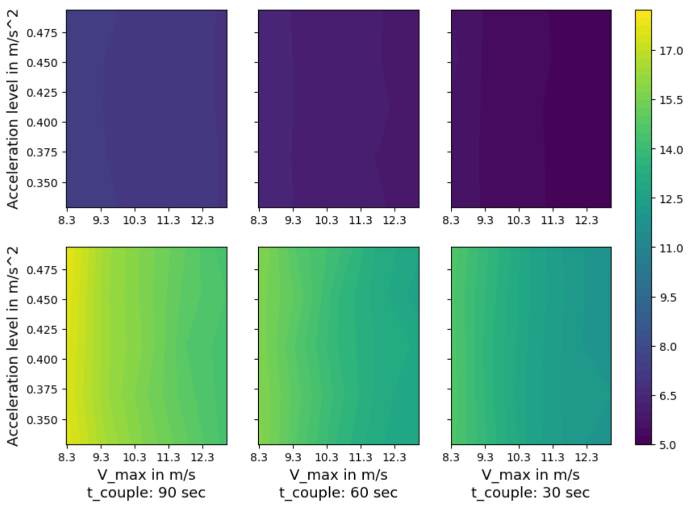


Fig. 8. Interaction plot visualising the aircraft taxi-time for outbound traffic under varying values for: $TB_{v,max}$ (x-axis), TB_{acc} (y-axis) and t_{couple} (columns of plots), for input days 7 (upper) and 13 (lower).

VII. Discussion

A. Obtained Results

This study shows that distributed coordination and planning allows for safe and efficient control and guidance of increasing vehicle movements when implementing a novel taxi-concept for towing of outbound aircraft employing autonomous Taxibots. The increasing number of vehicle movements in the novel taxi-concept due to the autonomous Taxibots, as anticipated in [3], is confirmed but can be accommodated in our model without affecting safety of other traffic. This study identified specific new 'hot-spots' on the airport in terms of increasing numbers of vehicle movements for the novel taxi-concept compared to conventional taxi operations. These points specifically relate to locations where Taxibots originating from the service-roads are required to cross active taxiways. These points require specific attention when considering autonomous Taxibots in real-life taxi-operations. Priority rules for heterogeneous vehicle types allowed for safe guidance of vehicles approaching such hot-spots and are proven to be applicable for AAS [17]. However, the applicability needs confirmation in future studies to be generalised to other airports.

Previous studies expressed the concern for increasing human workload when implementing autonomous taxi tugs [16]. The results indicate that this concern is valid as specific RMOs required a significant amount of active Taxibots on the airport to accommodate outbound aircraft towing. The applicability of distributed coordination and planning for altered taxi-operations suggests the potential for automation of ATC for such novel airport ground surface operations. Distributed coordination and planning would make the airport traffic guidance independent of human capabilities and allows for increasing numbers of vehicle movements on an airport's ground surface. The results show that the novel taxi-concept for outbound flights does not negatively affect taxi-operations of inbound flights, with inbound aircraft showing equal, and sometimes even better, performance within the simulation model compared to real-world traffic. The increase in vehicle movements does not necessarily need to affect the performance of other traffic within the ground surface operations. It also confirms the ability for AAS to allow conventional aircraft taxi-operations and novel taxi-operations simultaneously in the ground surface operations. Testing the novel taxi-concept on other airports and layouts is needed to confirm this applicability for different airport layouts. The effects of the novel taxi-concept on departure throughput is not extensively analysed in this study. However, the departure throughput is expected to be significantly affected, mainly due to the relatively large decouple-time of Taxibots. The relation between the Taxibot decouple time and the choice for Taxibot decouple locations suggests an interesting operational problem to consider in future research on autonomous Taxibots.

The model output showed most sensitive to variations in $TB_{v,max}$ and t_{couple} of the Taxibot dynamics, providing an initial direction for model calibration in future studies. The simulation model considers a single aircraft and Taxibot type based on a narrow-body aircraft, having relatively higher dynamical properties in comparison to wide-body aircraft and corresponding Taxibots. In actual ground surface operations, different aircraft types and corresponding dynamics are present and, therefore, also require different types of Taxibots. The influence of the Taxibots on outbound aircraft performance is expected to be less for larger aircraft when considering operational towing, due to the slower taxi operations compared to narrow-body aircraft. Different aircraft and Taxibot types could be simulated in specifically designed case-studies, as the simulation model allows for easy addition of different vehicle dynamics.

B. Methodological Approach

The proposed methodology has shown its capability for the initial design, exploration and evaluation of a novel taxi-concept within an airport ground surface operation. The methodology adheres to the research direction proposed by SESAR for solutions to incorporate non-autonomous engine-off taxiing into airport ground surface operations [43]. Due to the novelty of the taxi-concept for AAS, an iterative modelling procedure was necessary for accurate analysis of emergent behaviour. The absence of real-world data on the implementation of autonomous Taxibots necessitated the need for the abstraction of agent specifications and the model environment to explore the taxi-concept. The proposed methodology allows for further expansion of the model and more realistic modelling of all ground surface operations within a hub airport.

The model showed the capabilities of operational towing to significantly reduce aircraft engine-on time during taxiing. An important limitation of the proposed methodology is the exclusion of apron operations. A modelled aircraft is, therefore, spawned as close as possible to its start-time and is required to commence its taxi operations directly afterwards. Consequently, runway scheduling must be facilitated either via taxiway commands or by conventional waiting at the runway. Considering individual gates instead of meta-gates would allow for the inclusion of runway scheduling via gate holding, more realistic ground surface operations and more accurate time-estimations for Taxibot allocation. The inclusion of apron operations would also allow for the testing of new concepts, e.g. allocating a single Taxibot to a specific aircraft from landing to take-off, instead of a single outbound movement. Such a concept requires the consideration of inbound towing, but allows a single Taxibot to tow an aircraft for almost its entire ground surface operation. From an environmental perspective, this would be even more beneficial to the airport.

A concurrent study conducted by AAS allowed for the implementation of Taxibot decouple points determined by experts. Although this added practical relevance to the study, it introduced a limitation to the methodology as not more possible decouple locations were considered. From an optimal perspective, it would be difficult to consider all factors in choosing such a decouple location in the model. A large amount of decoupling locations is expected to create more chaotic operations due to a clash with current ground operations at AAS. Besides, it is expected that operational towing is not feasible for all specific gate-runway combinations due to infrastructural limitations. Therefore, consideration of new decouple locations should happen from a Pareto-efficient perspective and with a specific need for a certain decouple location, providing more leverage for changes in current procedures for all parties involved.

The centralised task-allocation algorithm has proven to allocate available Taxibots to upcoming tasks efficiently. The choice for a centralised task-allocation algorithm can be questioned, due to the distributed nature of the model. It makes the system less robust, as an error within the Taxibot coordinator agent could directly shut down the task allocation of Taxibots. Decentralised task allocation algorithms could be considered in future studies. Aside from the technical challenges in such a decentralised operation, e.g. concerning data-links between individual Taxibots, the operational consequences of a decentralised system should be carefully considered.

The interactions between autonomous Taxibots and human operators in the turn-around process (e.g. fuel-services or catering services) certainly requires a thorough discussion with all parties involved in the conceptual implementation of such decentralised and automated operations.
Aircraft and Taxibot performance have shown to be primarily dependent on $TB_{v,max}$ and t_{couple}. The Taxibot is currently tested at AAS to gather data on operational performance. Initial results indicate an even larger couple time than considered in this study. This would significantly affect the way decoupling is arranged in the model and requires reconsidering of the decouple locations. A combination of other decouple locations and the aforementioned runway scheduling paradigm could allow for a thorough exploration of the feasibility of the taxi-concept for significantly increasing Taxibot couple times.

C. Practical Relevance and Future Work

This study provided a first exploration of the applicability of operational towing employing autonomous Taxibots within the ground surface operations at AAS and, possibly, a consideration for other hub airports to study the implementation of operational towing. A few decouple locations, obtained from experts, have been proven infeasible in the model due to the increased chance for system grid-locks and add to the practical relevance of this study. From an operational perspective, future work should consider further increasing the realism of the model. Calibration of the model parameters and the inclusion of apron operations suggest the first steps in this direction. The results from the operational tests of the Taxibot at AAS allow for calibration of the Taxibot dynamics parameters. Furthermore, runway scheduling could be implemented in the model to study the effects of the novel taxi-concept on the departure throughput. A combination of runway scheduling and inclusion of apron operations would allow for the modelling of aircraft holding. This would increase the accuracy of the model and allows for better decision making regarding the feasibility of the novel taxi-concept. Moreover, the performance of task-allocation algorithms in the Multi-Agent Pickup-and-Delivery problem could be further investigated. The choice for either a centralised or decentralised algorithm, and how such an algorithm could be used for real-time planning of Taxibots within an operational environment, could be further studied. Operational considerations in decentralised algorithms, like the communication range of Taxibots or agent negotiations, could also be considered.

VIII. Conclusions

This study has taken the first steps into the design and evaluation of operational towing of outbound aircraft using distributed control and planning. A novel taxi-concept was developed that employs autonomous Taxibots within the ground surface operations of Amsterdam Airport Schiphol. These Taxibots tow outbound traffic from their gate to a designated decoupling location near the runway. A centralised task-allocation algorithm was implemented to facilitate the allocation of individual Taxibots to upcoming flights. Multi-Agent Path Finding was used for conflict-free planning of aircraft and Taxibot paths using the conflict-based search algorithm. ATC agents are located at each intersection in the airport taxiway system and guide traffic approaching the intersection. Potential conflicts are solved via speed or route commands and prioritise aircraft over individual Taxibots, being one of the proposed requirements for the novel taxi-concept.
The model performance was analysed by simulating four days of operational data retrieved from aircraft ADS-B ground tracks. It was found that for all operational scenarios the model could safely route all traffic and tow outbound aircraft without violating minimum separation distances between vehicles. The novel taxi-concept for outbound traffic was found to significantly affect the performance of outbound aircraft, compared to conventional taxi-operations, with an increase in total taxi-time and a lower average taxi-speed caused by the lower maximum taxi-speed of the Taxibot. The operational scenarios per day of operations were also found to influence aircraft performance and the total amount of Taxibots necessary; the differences with conventional operations increased for larger outbound taxi-distances. This research showed that the implementation of a novel taxi-concept significantly depends on the considered airport layout, e.g. runway, taxiway and service-road layout, and airport operational modes, e.g., active runways, amount of traffic and consequently the choice for decoupling locations. The proposed concept of operation showed promising results for novel taxi-operations within a hub airport ground surface operation, however, more research is needed to study the consequences for airport capacity and assess the applicability of the novel taxi-concept for other operational modes and airport layouts.

References

[1] F. Re, "Viability and State of the Art of Environmentally Friendly Aircraft Taxiing Systems," *2012 Electrical Systems for Aircraft, Railway and Ship Propulsion*, pp. 1–6, 2012.

[2] J. Li, H. Yang, X. Liu, N. Yu, Y. Tian, X. Zhou, P. Zhang, and K. Wang, "Aircraft emission inventory and characteristics of the airport cluster in the Guangdong-Hong Kong-Macao Greater Bay area, China," *Atmosphere*, vol. 11, no. 4, 2020.

[3] R. e. a. R. Morris, M. Chang, "Self-Driving Aircraft Towing Vehicles : A Preliminary Report The Case for Autonomous Towing," *Artificial Intelligence for Transportation: Advice, Interactivity and Actor Modeling*, pp. 41–48, 2015.

[4] I. Deonandan and H. Balakrishnan, "Evaluation of strategies for reducing taxi-out emissions at airports," *10th AIAA Aviation Technology, Integration and Operations Conference 2010, ATIO 2010*, vol. 3, pp. 0–14, 2010.

[5] I. Edem, O. Ikechukwu, I. A. E, and E. E. Ikpe, "The Future of Conventional Aircraft Ground Propulsion Systems in Relation to Fuel Consumption and CO 2 Emission," vol. 13, no. 2, pp. 91–100, 2016.

[6] M. Ithnan, T. Selderbeek, W. Beelaerts van Blokland, and G. Lodewijks, "Aircraft Taxiing Strategy Optimization," pp. 1–9, 2015. [Online]. Available: http://rstrail.nl/new/wp-content/uploads/2015/02/izzudin_ithnan.pdf

[7] R. Guo, Y. Zhang, and Q. Wang, "Comparison of emerging ground propulsion systems for electrified aircraft taxi operations," *Transportation Research Part C: Emerging Technologies*, vol. 44, pp. 98–109, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.trc.2014.03.006

[8] M. Lukic, A. Hebala, P. Giangrande, C. Klumpner, S. Nuzzo, G. Chen, C. Gerada, C. Eastwick, and M. Galea, "State of the Art of Electric Taxiing Systems," *2018 IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles and International Transportation Electrification Conference, ESARS-ITEC 2018*, no. November, 2019.

[9] Eurocontrol, “COVID19 Impact on European Air Traffic,” no. September, 2020.

[10] ——, “European Aviation in 2040,” pp. 1–91, 2018. [Online]. Available: https://www.eurocontrol.int/publication/long-term-forecast-annual-numbers-ifr-flights-2040

[11] J. Hospodka, “Cost-benefit analysis of electric taxi systems for aircraft,” *Journal of Air Transport Management*, vol. 39, pp. 81–88, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.jairtraman.2014.05.002

[12] P. Vaishnav, “Costs and benefits of reducing fuel burn and emissions from taxiing aircraft,” *Transportation Research Record*, no. 2400, pp. 65–77, 2014.

[13] N. Okuniek and D. Beckmann, “Towards higher level of A-SMGCS: Handshake of electric taxi and trajectory-based taxi operations,” *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, vol. 2017-September, 2017.

[14] A. L. N. Dzikus, J. Fuchte and V. Gollnick, “Potential for fuel reduction through electric taxiing,” *11th AIAA Aviation Technology, Integration,and Operations (ATIO) Conference, including the AIAA Balloon Systems Conference and 19th AIAA Lighter-Than-Air Technology Conference*, no. September, pp. 1–9, 2011.

[15] M. Nadia Postorino, L. Mantecchini, and E. Gualandi, “Integration between aircraft and handling vehicles during taxiing procedures to improve airport sustainability,” *International Journal of Transport Development and Integration*, vol. 1, no. 1, pp. 28–42, 2017.

[16] Z. Chua, M. Cousy, M. Causse, and F. Lancelot, “Initial assessment of the impact of Modern Taxiing techniques on airport ground control,” *Proceedings of the International Conference on Human-Computer Interaction in Aerospace, HCI-Aero 2016*, 2016.

[17] Z. K. Chua, “Self-Managing Conflict Resolution for Autonomous Taxiing Tugs : An Initial Survey,” *SESARWPE*, no. December, pp. 1–8, 2015.

[18] L. Khammash, L. Mantecchini, and V. Reis, “Micro-simulation of airport taxiing procedures to improve operation sustainability: Application of semi-robotic towing tractor,” *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017 - Proceedings*, pp. 616–621, 2017.

[19] E. van Baaren, “The feasibility of a fully electric aircraft towing system,” *TU Delft repository*, 2019.

[20] N. J. F. P. Guillaume, “Finding the viability of us-ing an automated guided vehicle taxiing system for aircraft,” *TU Delft repository*, 2018. [Online]. Available: http://repository.tudelft.nl/.

[21] Taxibot, 2013, the world’s only certified and operational taxiing alternative.

[22] Luchtvaartnieuws, “In beeld: Schiphol test duurzaam taxiËn met taxibot,” 2020, retrieved on 16/09/2020 from : https://www.luchtvaartnieuws.nl/nieuws/categorie/3/airports/in-beeld-schiphol-test-duurzaam-taxien-met-taxibot.

[23] A. J. Heppenstall, A. T. Crooks, L. M. See, and M. Batty, “Introduction to Agent-Based Modelling,” *Agent-Based Models of Geographical Systems*, no. January, pp. 1–759, 2012.

[24] F. Klügl and A. L. Bazzan, “Agent-based modeling and simulation,” *AI Magazine*, vol. 33, no. 3, pp. 29–40, 2012.

[25] S. Bouarfa, H. A. Blom, R. Curran, and M. H. Everdij, “Agent-based modeling and simulation of emergent behavior in air transportation,” *Complex Adaptive Systems Modeling*, vol. 1, no. 1, pp. 1–26, 2013.

[26] F. Chen, “Aircraft taxiing route planning based on multi-agent system,” *Proceedings of 2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference, IMCEC 2016*, pp. 1421–1425, 2017.

[27] T. Noortman, “”Agent-Based Modelling of an Airport’s Ground Surface Movement Operation”,” 25-5-2018, understanding the principles and mechanisms of decentralised control.

[28] K. Fines, “”Decentralized Control for Resilient Airport Surface Movement Operations”,” 2019.

[29] H. Udluft, *Decentralization in air transportation*, 2017.

[30] A. Sharpanskyk’h, “Week 1: Agent-based Modelling and Simulation in Air Transport,” 2018.

[31] Dutch Safety Board, “Schiphol air traffic safety,” 2017.

[32] W. Dalmeijer, “Gebruikersprognose 2020,” September 2019, retrieved 14 July 2020 from https://www.schiphol.nl/nl/jij-en-schiphol/pagina/gebruiksprognose-alles-wat-je-maar-wilt-weten/.

[33] G. Sharon, R. Stern, A. Felner, and N. R. Sturtevant, “Conflict-based search for optimal multi-agent pathfinding,” *Artificial Intelligence*, vol. 219, pp. 40–66, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.artint.2014.11.006

[34] W. Honig, S. Kiesel, A. Tinka, J. W. Durham, and N. Ayanian, “Conflict-based search with optimal task assignment,” *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 1, no. 1, pp. 757–765, 2018.

[35] A. V. Goldberg and C. Harrelson, “Computing the shortest path: A* search meets graph theory,” *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 156–165, 2005.

[36] M. Liu, H. Ma, J. Li, and S. Koenig, “Task and path planning for multi-agent pickup and delivery,” *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 2, pp. 1152–1160, 2019.

[37] J. Ma, D. Delahaye, M. Sbihi, P. Scala, and M. M. Mota, “A Study of Tradeoffs in Airport Coordinated Surface Operations,” *EIWAC 2017, 5th ENRI international workshop on ATM/CNS*, 2017.

[38] J. Van Der Horst and J. Noble, “Distributed and centralized task allocation: When and where to use them,” *Proceedings - 2010 4th IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshop, SASOW 2010*, pp. 1–8, 2010.

[39] F. Klügl, “A validation methodology for agent-based simulations,” *Proceedings of the ACM Symposium on Applied Computing*, no. June, pp. 39–43, 2008.

[40] R. G. Sargent, “Verification and validation of simulation models,” *Journal of Simulation*, vol. 7, no. 1, pp. 12–24, 2013. [Online]. Available: http://dx.doi.org/10.1057/jos.2012.20

[41] A. Vargha and H. D. Delaney, “A critique and improvement of the CL common language effect size statistics of McGraw and Wong,” *Journal of Educational and Behavioral Statistics*, vol. 25, no. 2, pp. 101–132, 2000.

[42] A. Sharpanskyk’h, “Week 3: Agent-Based simulation; Analysis of results,” pp. 1–19, 2020.

[43] Eurocontrol, “Technical Specification of SESAR 2020 Exploratory Research Call,” Tech. Rep., 2019.

II

Literature study

1
Introduction

The aviation industry is faced with several challenges for the coming years. With the number of flights continuing to grow annually, both airport capacity and Air Traffic Control (ATC) capacity need to be reconsidered to be able to accommodate this growth. One of the consequences of this growth are increasing delays during the airport taxiing phase. Eurocontrol presented that the additional taxi-time at the top 30 European airports in 2018 averaged 4.2 minutes: an increase of 0.3 minutes compared to 2017 [16]. To keep these delays under control countermeasures are required by both ATC and airports for the airspace, as well as on the airport surface. On the other hand, emissions and pollutants from the aviation industry are put under pressure by citizens and environmental organisations. Especially in the vicinity of airports, local residents are protesting against further airport growth. Airports like AAS try to respond to this calls by aiming for an emission neutral airport by 2030 [17]. Airports and ATC are therefore forced to both look at possibilities to lower aviation emissions, while still being able to accommodate the (expected) growth in flight movements.

In order to increase capacity at airports and reduce airport congestion, a first consideration would be to add more infrastructure on and around the airport to accommodate extra vehicles. However, this is both difficult and expensive as large hub airports are often located in densely populated areas and expansion thus causes resistance from citizens. Therefore, operational improvements are considered as they are less prone to resistance from citizens. Currently, ATC resembles a centralised operation: an ATCo is responsible for the guidance of aircraft in the air and on the ground to ensure safe and efficient operations. The amount of aircraft that can be guided simultaneously is related to the amount of ATCOs and the workload they can manage. Especially in Low Visibility Conditions (LVC), more radio communication is necessary leading to a severe degradation in airport capacity [18]. A means to increase airport ground surface capacity that is currently being researched is the concept of distributed control. Distributed control shifts the responsibility for aircraft guidance form a central position to a local level: interacting virtual agents are responsible for aircraft guidance and solve conflicts according to local information while serving a global goal. This concept allows higher levels of autonomy and could increase capacity due to it being less dependent on human workload. The role of ATC changes from guiding aircraft to supervising operations. A concept that follows this reasoning is the Follow the Greens system, in which aircraft follow an illuminated path of lights on the taxiway system indicating their direction of travel. The system makes flight crew less dependent on ATC communication and instructions and allows more autonomous taxiing.

To further expand upon the concept of distributed control and the need to lower emissions on airports, this research focuses on the implementation of autonomous engine-off taxiing in the ground operations of a hub airport. Engine-off taxiing can be considered by means of external towing systems that move an aircraft over the airport surface while its engines are off. It can be compared to current towing operations and has the benefit that aircraft taxi emissions can be lowered. However, it increases the number of vehicles on the airport ground surface. In current centralised ATC, implementing (manned) towing trucks would require an ATCo to guide even more vehicles over the airport surface, possibly increasing workload and airport congestion. This research will investigate the feasibility of implementing autonomous unmanned towing trucks that are guided locally by means of distributed control. Similar to the concept of Follow the Greens, conflicts are being solved at intersections by means of local information, while ensuring safe and efficient operations throughout the network. For this research, it has been chosen to use the layout and airport operations of AAS as a case-study.

This literature study is build-up as follows. First, in Chapter **2** a description is given of the ground operations at AAS. The agents involved in ground operations and their interactions will be described by means of a socio-technical system representation. Chapter **3** will describe the developments in ATC regarding the Follow the Greens concept and autonomous engine-off taxiing, as initiated by Single European Sky ATM Research (SESAR). Chapter **4** continues on the aforementioned concept of SESAR and describes the current state-of-the-art in engine-off taxiing. This chapter will describe the currently available engine-off taxi systems as well as research that has been done regarding implementation in airport operations. In order to consider modelling of engine-off taxiing operations, several modelling techniques are reviewed in Chapter **5**. This gives the ability to gain knowledge on the current state in airport ground surface modelling and choose a suitable modelling technique for this study. After having reviewed the most relevant literature, a research proposal is provided in Chapter **6**. The research methodology in Chapter **??** will conclude the literature study.

2
Ground Surface Operations at Schiphol

AAS is one of the world's largest airports by passenger numbers and has some outstanding specifications like its complex layout and the large variations in aircraft taxi times. Before an in-depth analysis of current research on ground surface modelling can be done, the ground operations at AAS should be well understood. Therefore, this chapter contains a general description of the ground surface operations at AAS. First, a high-level description of the ground surface operations at AAS will be given in Section 2.1. Thereafter, a socio-technical system representation will be given in Section 2.2. This representation will cover the relations between the relevant parties involved in an airport surface operation, as well as the interactions between those parties.

2.1. General description

This section gives a general overview of AAS and its corresponding ground surface operations. Section 2.1.1 will describe the history of AAS and shall give a thorough description of the layout of the runways, taxiways and other specific areas on the airport. In Section 2.1.2 a description is given of the ground operations at AAS regarding the turn-around, as well as the procedures used by ATC to guide vehicles over the airport surface. At last, A-CDM will be described in Section 2.1.3 as it has a major influence on ground operations and improving its efficiency.

2.1.1. History and overview of AAS

As AAS is by far the largest airport in the Netherlands by passenger numbers, even one of the largest airports in the world [19] with over 71 million passengers in 2018, it is hard to imagine that it all started on a piece of grass with less than 500 passengers per year. It was in April 1916 that the former minister of War had given his approval for the purchase of a piece of grass of 16,5 hectares in the Haarlemmermeer polder [20]. Later in August that year, the ground was suitable to serve as an airport and air transport operations commenced on September 19th: the start of AAS.

During the first World War, it was already noticed that the surface was not big enough. Therefore, surrounding pieces of land were confiscated and AAS had reached a size of 76 hectares. The real development of AAS started in the late '30s when the municipality of Amsterdam purchased the airfield. This resulted in the construction of drainage, supply roads to the airport, a railway station, an apron and a traffic control tower. In the second World War, when AAS had reached an overall size of 180 hectares and 4 runways, it was bombed and taken over by the Germans. After the Germans left in May 1945, there was very little left of AAS and it had to be rebuilt all over again.

A few months after the Germans left, on the 8th of July, it was already possible again to operate on AAS. This was the start of rapid development of AAS in the years after, of which some significant developments are:

- Construction and opening of the terminal at AAS centre;
- The ability to handle jumbo-jets due to reconstruction of the A-pier;
- A new D-Pier;

At the start of the '90s, a new master-plan had been carried-out resulting in further enlargement of the terminal, a new railway station and more hotels and office spaces at the airport. In 2003 the sixth runway was opened, the Polderbaan, while in 2005 a new pier for low cost carries had been put into use: the H-pier.

An overview of AAS in its current state can be seen from Figure 2.4. AAS has a total of 6 runways: 5 runways that are over 3000m and can accommodate all aircraft types whereas the east runway (04-22) is shorter and therefore mainly used for smaller- or landing aircraft. As shown in Figure 2.1, all runways except 04-22 are provided with Rapid-Exit Taxiway(s) (RET) (red circled runway exits) [2]. These taxiways have an acute intersection angle of 30 degrees with the runway. RET allows a landing aircraft to exit the runway at higher speeds (max. 50 kts), compared to ordinary right-angled runway exits (normal taxi speed, 20-30 kts), reducing runway occupancy time [21]. This has the beneficial effect that runway capacity can be increased. Suitable locations of RET could reduce taxi distances for arriving aircraft.

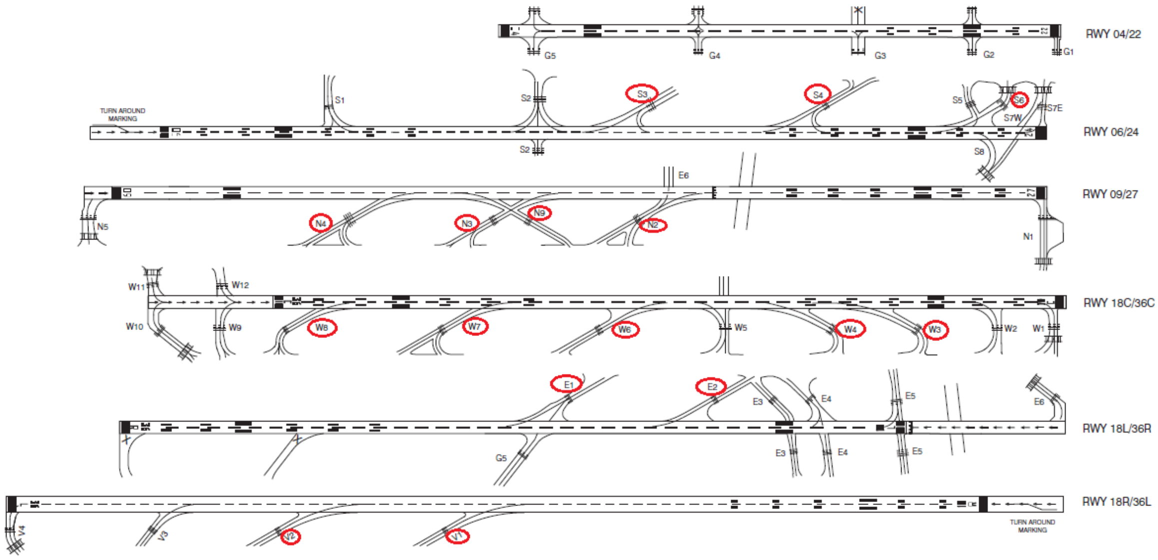


Figure 2.1: Runway specifications AAS [2]

Figure 2.2: Piers at AAS [3]

An overview of the piers at AAS is given in Figure 2.2. AAS has a total of 94 passenger-oriented gates for aircraft, divided over 7 piers. Currently, constructions are in progress for a new pier and terminal 'A', in the figure located below pier B. This pier will be able to accommodate 5 Narrow-Body (NB) aircraft and 3 Wide-Body (WB) aircraft. It must be noted that all gates except the B-platform (Bravo, left of the B-pier) are nose-in gates. This means that aircraft are parked perpendicular to the pier, with their nose forward. Push-back by means of an external truck is necessary to turn the aircraft onto the taxi-way and let it taxi autonomously.

AAS handles over 70 million passengers and almost 500.000 aircraft movements annually [22]. An overview of the total amount of departures and arrivals in 2018, with a distinction between flights within Europe and intercontinental flights, can be found in Figure 2.3. From this figure the increment in movements during summer season, especially for flights within Europe, is clearly visible for the period between May and October. Besides, it can be seen that intercontinental flights take a significant part in the number of movements at AAS: 20% of all movements correspond to intercontinental traffic. In 2018, a total of 327 direct destinations were available from AAS.

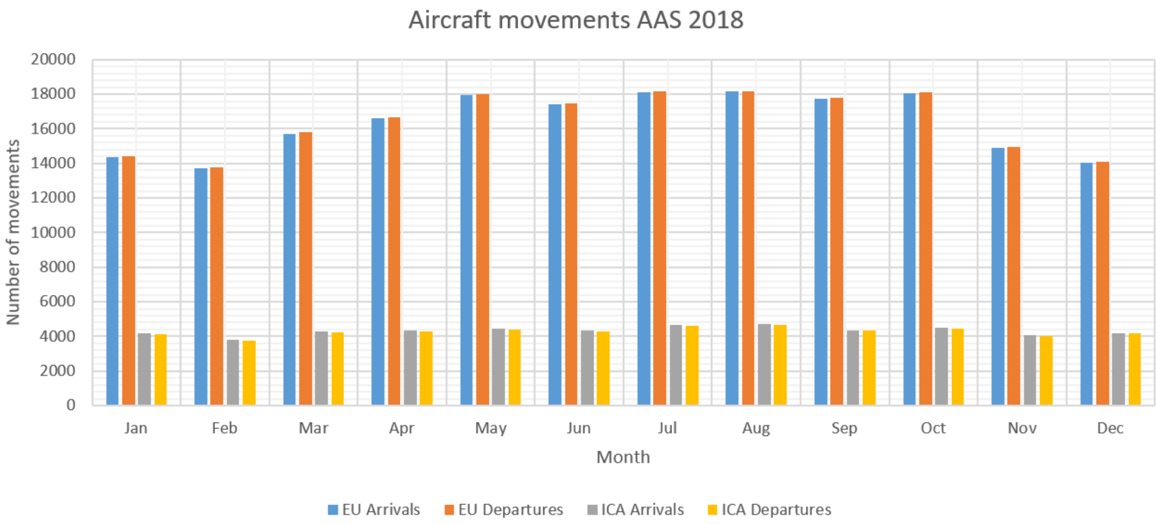


Figure 2.3: Aircraft movements per month at AAS in 2018

Figure 2.4: Departures per runway with >1% share of total movements [4], figure taken from [5]

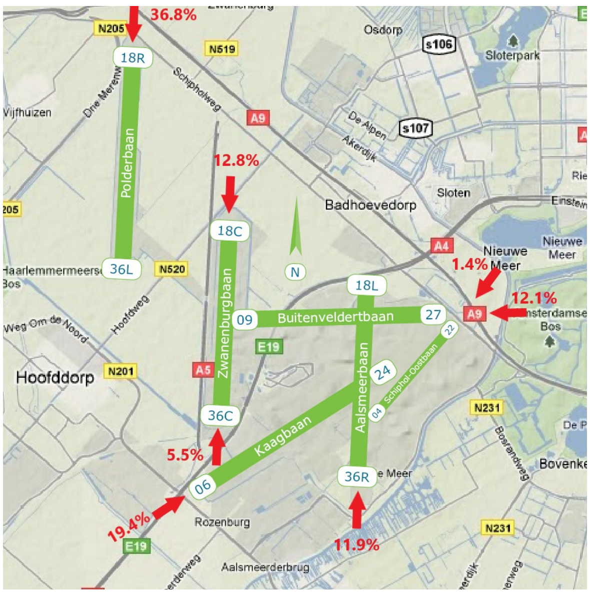


Figure 2.5: Arrivals per runway with >1% share of total movements [4]

Both arrival and departure traffic are divided over the available runways at AAS to ensure efficient operations, keeping in mind the nuisance for people living in the vicinity of the airport. The increasing number of flights at AAS has the consequence that the number of complaints from residents living near the airport is growing.

To reduce annoyance from air traffic as much as possible, AAS tries to utilise its runways as flexible as possible. Preferred runway configurations are used by ATC for both day and night operations, while the active Runway Mode of Operation (RMO) is made public by Luchtverkeersleiding Nederland - Air Traffic Control Netherlands (LVNL) in order to inform the residents.
The expected distributions of departures and arrivals over the runways for the current year (2019) are depicted in Figure 2.4 and Figure 2.5. As can be seen, a large part of both arrivals (36.8%) and departures (27.6%) use the Polderbaan. The taxi-times from the piers at AAS to the Polderbaan take up to 15 minutes and are the largest taxi-times compared to the other runways. Due to the large share of traffic using the Polderbaan, it can be assumed that significant fuel is burnt during the taxiing phase. Overall, the traffic is quite distributed over the available runways.

2.1.2. Ground operations at AAS

In between touchdown and take-off of an aircraft, ground operations take place on the airport surface. These ground operations roughly consist of five phases: landing, taxi-in, turn-around at the gate, taxi-out and take-off. A thorough description of the ground operations at AAS has been given in [6, 8, 23], which are used as a basis for this section.
A general overview of the ground operations at an airport can be found in Figure 2.6. An arriving aircraft on final approach receives landing approval from the airports ATC and thereafter lands on its designated runway. As mentioned before in Section 2.1.1, an aircraft can vacate the runway quickly by using a RET. This depends on the situation, as ATC gives instructions to the pilots regarding runway exits as well as the taxi-path to take to its assigned gate. Both the runway exit and taxi-path are usually assigned already before the aircraft enters the taxiway system [6]. If the designated gate is not available yet, the aircraft can be guided to a holding area where it can wait for its gate to be free. After the aircraft arrives at the gate, the taxiing procedure is finished and the turn-around commences.

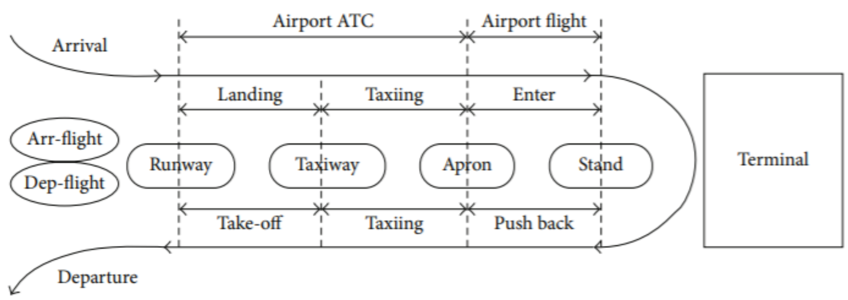


Figure 2.6: Ground surface operations breakdown [6]

The turn-around process covers all operations done on and in the aircraft while occupying a stand or gate at an airport [24]. At standstill, wheel chocks are placed around the aircraft landing gears and the pax stairs or air-bridge is connected to the aircraft. Thereafter, several processes are carried out (some simultaneously) in and surrounding the aircraft:

- (De-)boarding of passengers;
- Aircraft servicing (fuelling, freshwater, waste handling, interior and/or exterior cleaning);
- Maintenance and pre-flight checks;
- Loading of cargo/bags;

After the above processes have been successfully carried out, the aircraft awaits ATC instructions. When given start-up approval, the engines can be started. After removal of the wheel chocks and pax stairs or air-bridge, and push-back approval, the aircraft can be pushed back. The aircraft receives push-back from a towing vehicle and follows the instructions from ATC ground control regarding its taxi-path, take-off runway and entrance. Guidance during taxiing can be provided by ATC, considering taxiway intersections or runway crossings. At the runway, the aircraft receives clearances for line-up and take-off from ATC.

Aircraft are guided by ATC during taxiing to prevent the occurrence of conflicts. A distinction can be made between 3 types of conflicts that can occur in the airport taxiway system. At the left of Figure 2.7, a node-based conflict is depicted. This happens when two aircraft travel through a common intersection without keeping the minimum level of separation. ATC can solve this conflict in several ways: the most common rule in these situations is that aircraft from the right get priority. There are however two exceptions to the rule. First, it can be communicated by ATC that another aircraft gets priority. Secondly, arriving aircraft get priority due to their higher taxi speed and the need to vacate the runway as quickly as possible.
The second type of conflict occurs when an aircraft is trailing another aircraft using the same taxiway. The pilots of the trailing aircraft are responsible for maintaining a safe distance with the leading aircraft, as depicted in Figure 2.7.
The last conflict that could happen is depicted on the right in Figure 2.7, called an edge-conflict. An edge-conflict occurs when two aircraft taxi over the same (one-directional) taxiway in opposite direction. This results in the aircraft standing nose to nose, which is called a deadlock. These types of conflicts must be solved using towing vehicles, as aircraft cannot move backwards independently. Therefore, deadlock situations must be prevented at all times, as they cause major inefficiencies in the taxiway network.

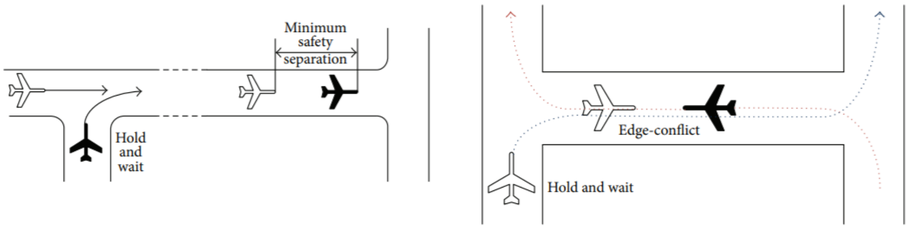


Figure 2.7: Possible taxiway conflicts [6]

In order to prevent deadlocks from happening and be able to guide all aircraft safely over the airport surface, several procedures are used by ATC to structure taxi path allocation. As can be seen in Figure 2.8, two taxiways surround the piers at AAS: taxiways Alfa (pink) and Bravo (green). ATC uses prescribed directions of travel for both taxiways to structure connections between the runways and the gates. The prescribed direction of travel for taxiway Alfa is clockwise, while it is counterclockwise for Bravo. It may, however, occur that ATC chooses to deviate from these procedures [25]. This must be clearly communicated to the pilots of taxiing aircraft, in order to avoid errors. ATC must carefully monitor the positions of aircraft to be able to intervene if necessary. At last, taxiway Quebec is depicted in blue in Figure 2.8. This is a single-way taxiway and its direction depends on the runway mode of operations. As deadlocks could occur on taxiway Quebec, ATC must carefully determine its direction of usage. AAS has started the constructions for a double taxiway Quebec in order to complete the double taxiway system around AAS' central area [26]. This will increase safety around this area, as well as a decrease in aircraft waiting time. AAS aims to finish the project in 2023.

It can occur that the gate of an arriving aircraft is still occupied due to a delayed turn-around of the departing aircraft at the gate. In this situation, ATC can put an aircraft on hold at specific areas around the airport. These areas are indicated in red in Figure 2.8. Three holding areas are present on AAS. One is located on the top right of the figure, the P-holding, capable of accommodating one aircraft with a max wingspan of 69m and one of max 36 m. In the bottom left one can find the R-apron, which can accommodate a maximum of 2 aircraft with a max wingspan of 36 m. At last, there are two holding positions east of the Polderbaan.
De-icing is done at the J-apron, indicated in orange in Figure 2.8. It can be expected that special communication procedures are carried out by ATC during de-icing conditions.
Besides aircraft there are other vehicles using the airside ground surface, like for example authority vehicles or towing trucks. A brief description of towing operations and related information is given in Section 2.1.4, after introduction of the concept of Airport Collaborative Decision Making (A-CDM) in Section 2.1.3. An overview of all parties involved in an airports ground surface operation will be given in Section 2.2.

2.1.3. A-CDM at AAS

In March 2018, AAS became the 28th full A-CDM airport and the last hub airport to be connected to the Eurocontrol Network Manager systems [27]. A-CDM is a joint initiative between airlines, ground handlers, ATC and the airport to facilitate sharing of operational information and data to achieve better informed decisions

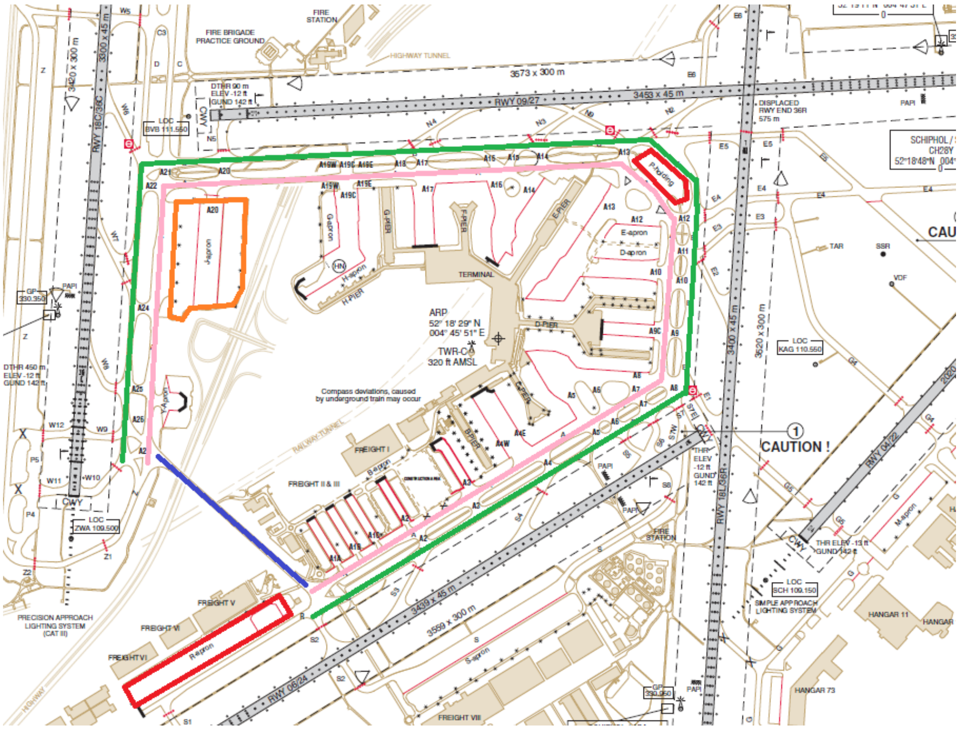


Figure 2.8: Important taxiways and aprons AAS [2]

to be made [7]. Implementation of A-CDM consists of six operational processes:

1. (Airport CDM) information Sharing;
2. The Milestones Approach (Turn-Round Process);
3. Variable Taxi Time;
4. (Collaborative) Pre-departure Sequence;
5. (CDM in) Adverse Conditions;
6. Collaborative Management of Flight Updates.

The foundation of A-CDM lies in the concept of information sharing: '*the sharing of accurate and timely information between the Airport CDM Partners in order to achieve common situational awareness and to improve traffic event predictability*' [28]. This information is used in step 1: the Milestone Approach, as shown below in Figure 2.9. When a milestone (significant event) is successfully completed, decision making for follow-up events is triggered and progress accuracy of the flight is increased. As this has consequences for all partners related to the operation of an aircraft, it is important to clarify some of the milestones in A-CDM. This study focuses on ground surface movements at an airport and therefore milestones 5 to 16 are of main importance. It must be noted that Eurocontrol does not mandate specific milestones, as it may be dependent on local procedures. As AAS is used as a case study, this section focuses on the milestones that are implemented at AAS. [7, 28] are used as a basis for this section.
During inbound, milestone 5 indicates the final approach into the airport. ATC updates for this arrival flight the Estimated Landing Time (ELDT) and Estimated In-Block Time (EIBT), and marks the flight-state as final. This is done in the Airport CDM Information Sharing Platform (ACISP), which can be accessed by all A-CDM partners to retrieve specific flight times. The ELDT update can change both the Target Off-Blocks Time (TOBT) and Target Take-Off Time (TTOT): if a TTOT changes more than a pre-defined tolerance, the airlines' network operator is informed.
To calculate the change in TTOT due to an updated TOBT, the Collaborative Pre-Departure Sequence Planning (CPDSP) system is used. A schematic overview of this calculation can be found in Figure 2.10. The TOBT and Estimated Taxi-Out Time (EXOT) are added to determine the Earliest Possible Take-Off Time (TTOT').

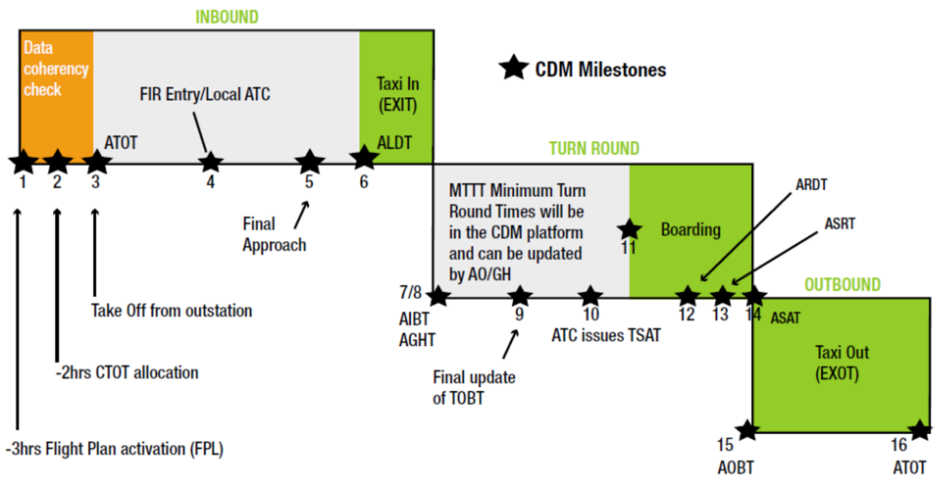


Figure 2.9: The Milestones Approach[7]

The TTOT' of all aircraft within the same time-span are used by an algorithm to determine a pre-departure sequence. This algorithm proposes a departure sequence based on:

- Calculated Take-Off Time (CTOT): This is a take-off time issued by the Central Management Unit of Eurocontrol (NMOC). At this time, the aircraft is expected to become airborne in order to fit the airspace flow. As it is an issued time, it limits the flexibility of ATC in the determination of the pre-departure sequence. An assigned CTOT must be adhered to within a time-window of +5/-10 minutes. If this window is not met, a new time must be assigned, possibly delaying the aircraft (in some cases extensively).
- the Standard Instrument Departure (SID) an aircraft will use.
- WTC of an aircraft. As can be seen in Figure 2.1, this is dependent upon the aircraft Maximum Take-Off Mass (MTOM) and has consequences for the minimum time separation between subsequent arrivals and/or departures.
- the runway capacity.

The departure sequence from the CPDSP determines the TTOT, from which the Target Start-Up Approval Time (TSAT) can be obtained by subtracting the EXOT. However, the departure sequence is vulnerable to changes like runway reconfigurations, changing weather conditions or CTOT changes [7]. EXOT and Estimated Taxi-In Time (EXIT) are both determined by ATC using the Variable Taxi-Times (VTT). This is dependent upon the specific airport and the state of the ground surface network.

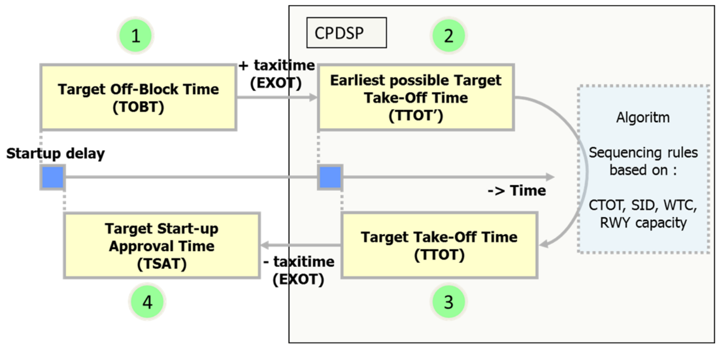


Figure 2.10: TSAT calculation [7]

Upon landing of the aircraft (milestone 6), the Actual Landing Time (ALDT) is updated and the aircraft status is set to 'landed'. The aircraft gets guidance from ATC regarding runway exit and taxi route to its gate. The Estimated In-Blocks Time (EIBT) is calculated by adding the EXIT to the ALDT. An automatic update for the

Leader / Follower (Arr/Dep)	S	H	M	L
Super (A388)	3/1	3/2	3/3	4/3
Heavy (H) - MTOM ≥ 136 tons	-/-	-/-	2/2	3/2
Medium (M) - 7 < MTOM < 136 tons	-/-	-/-	-/-	3/2
Light (L) - MTOM ≤ 7 tons	-/-	-/-	-/-	-/-

Table 2.1: WTC separation minima for arrivals / departures in minutes at AAS [8]

TOBT and TTOT of the corresponding departure flight is done in a similar manner as described above.
After taxiing, the aircraft comes to a standstill at its gate (milestone 7): EIBT changes to AIBT and consequently triggers an update of the next flight's TOBT and TTOT. This initiates milestone 8: the start of ground handling (ACGT). As mentioned in Section 2.1.2, ground handling consists of several processes on and around the aircraft. Approximately 10 minutes before the TOBT, the ground handler and aircraft operator need to provide their most accurate TOBT (9). The TOBT is constantly updated in time, as mentioned at previous milestones. However, this specific milestone is special in the sense that it checks the quality of the TOBT before the TSAT is issed by ATC (10). TSAT indicates the time an aircraft can expect engine start-up and push-back approval, and it is given as a time window: +/- 5 min of TSAT.
The start of the boarding process (ASBT, 11) gives a good indication whether the TOBT and/or TSAT will be respected. Milestones 12 and 13 are integrated at AAS: start-up is requested when the flight is ready. If the flight is in its TSAT window, ATC enters the ASRT in the ACISP and the aircraft is kept in the pre-departure sequence. After start-up has been approved by ATC (milestone 14), the engines are ignited and the aircraft will be pushed back by ground handlers. Especially the ASAT and its approval are relevant to push-back operators, in order to be present at the gate at the right time.
When the aircraft gets pushed back, AOBT is recorded and TTOT is updated by consideration of the EXOT. When reaching its designated runway after ATC guidance, the aircraft takes off and ATOT is recorded (milestone 16). The aircraft is now removed from the departure sequence and it is indicated as 'airborne'.

2.1.4. Towing operations at AAS

Aircraft towing provides a means to push-back aircraft from their parking position onto the taxiway or move aircraft over the airport surface without having to turn on the engines. A clear distinction can be made between two kinds of towing operations at AAS:

- Push-back operations at gates or stands.
- Aircraft towing over the airport surface, i.e. from AAS center to AAS East or from a parking apron to the gate and vice versa.

The operations related to push-back of aircraft are already touched upon in the section regarding A-CDM. From interviews with tow-truck drivers it is known that a tow truck driver can access the TSAT of a flight from his/her handheld computer. Based on the TSAT, the tow truck driver who is assigned to that specific flight knows when he/she has to be at the gate. The speed of a push-back maneuver is around 15-20 km/h, while the type of push-back truck depends on the specific aircraft type. Two types of push-back trucks are operated at AAS: larger tugs that can tow WB aircraft and smaller tugs that are used for NB aircraft. KLM is one of the largest ground handlers at AAS and has its own towing department. It has two designated parking locations for the towing trucks: Figure 2.11 shows the parking spot for the WB trucks at the G-Pier in red. This parking spot is located just in front of the Hotel pier (Figure 2.2). The second parking location is located near the B-pier and accommodates the NB tow trucks. It can be found at the root of the B-pier and is shown in Figure 2.12 in yellow. For other ground handlers at AAS, like Aviapartner, it was found that there are no real designated parking locations for towing trucks. Aviapartner uses the ground equipment parking spots at the D-gates to park its towing trucks.

The second operation carried out by towing trucks consists of towing aircraft from their gate to a parking spot or vice versa, or from AAS Center to AAS East or vice versa. The maintenance hangars are located at AAS East and therefore aircraft need to be moved over the airport surface to receive maintenance at the hangar. This is done with towing trucks so that the aircraft engines do not have to be used. The towing trucks, and thus also the entire towing operation, are limited to a speed of 30 km/h. The actual speed depends on the aircraft weight and is usually between 20-30 km/h.

Towing trucks are fully allowed to use the taxiway network of the airport. They are, just like aircraft, guided and cleared by ATC in their movements. Besides, towing trucks are allowed to move over the service roads surrounding the airport. These roads are also used by baggage handling, maintenance personnel and other support vehicles. From experience, it is noticed that towing trucks mainly use these roads when not towing an aircraft.

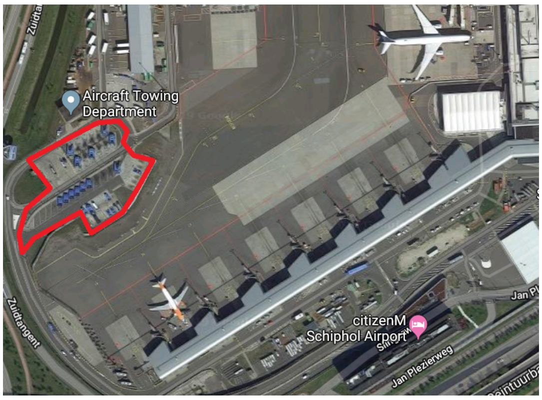


Figure 2.11: Parking location for WB towing trucks

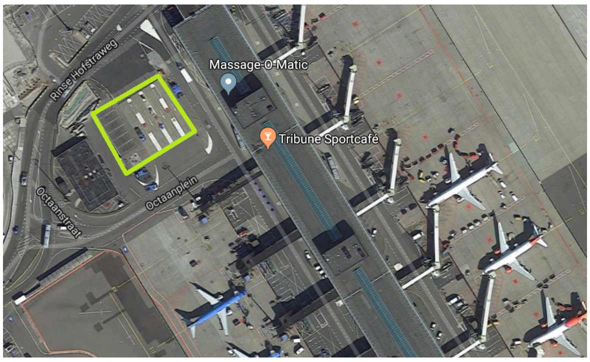


Figure 2.12: Parking location for NB towing trucks

2.2. Socio-technical system of airport ground operation

In the last section, some light was shed on the different parties involved in an airport ground surface operation. To clarify and further elaborate upon the parties involved in the airport ground operations, as well as the interactions between them, this section provides a socio-technical system representation [1] of the airport ground surface operations. Such a representation provides a better understanding of how human, social and organisational factors affect how work is done and supporting systems are used. At first, a description of all parties involved in the ground operations is given in Section 2.2.1. Thereafter, in Section 2.2.2, the interactions between those parties are elaborated upon. The system representation in this section is partially based on previous literature studies regarding ground surface operations [8, 23].

2.2.1. Parties involved in airport ground operations

Figure 2.13 gives an overview of all parties involved in an airport ground surface operation, as well as the interaction links between them. It was mentioned already in Section 2.1.3 that A-CDM provides a central way to retrieve information regarding a flight. This information can be used to achieve better informed decisions. However, there are also other interactions between aforementioned parties. This section contains a description of each actor in the system, whereas Section 2.2.2 considers the mutual interactions and their link with A-CDM.

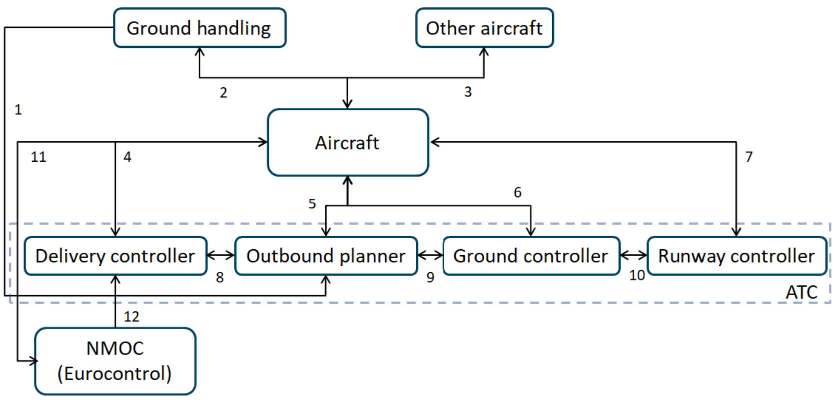


Figure 2.13: Socio-technical system representation of airport ground operations [8]

[1]Systems that involve complex interaction(s) between humans, machines and the environment of the system [29]

Aircraft

The aircraft is central in the system representation in the sense that all related parties are supportive to it. An aircraft is controlled by a flight crew and its characteristics are dependent on the aircraft type. An aircraft is for example characterised by its weight, wingspan, operational speeds, amount of passengers and more. The goal for the flight crew is to operate the aircraft according to its assigned flight schedule. It is the job of the airline management to determine the flight schedule, which usually covers three steps [30]:

- Fleet planning - What type of aircraft to acquire, when and how many?
- Route planning - Where to fly the aircraft?
- Schedule development - How frequently, at what time and which aircraft should be assigned to the schedule?

Besides the operational aspect, it is the duty of the flight crew to operate the aircraft safely on ground and in the air. To be able to do this, the flight crew must be aware of the aircraft surroundings, while it also gets guided by ATC.

Ground handling

As discussed already in Section 2.1.2, ground handling covers all aspects related to the turn-around of an aircraft at the gate. This includes several processes in and surrounding the aircraft, including but not limited to: (de)boarding of passengers, loading of cargo and bags, aircraft servicing (includes for example fuelling, cleaning, waste handling) and maintenance. Ground handling does not cover just a single person in the system: it represents maintenance technicians, cleaners, baggage handlers, fuelling services and tow truck operators.

As mentioned in Section 2.1.3, both the ground handlers and the aircraft operator are responsible for an accurate update of the TOBT, 10 minutes before current TOBT. This decision happens in consultation with all responsible parties. Goal of the ground handling parties is to service and prepare the aircraft at the right time to have it fully operational for its next flight in time. This implies that delay incurred due to ground handling should be prevented as much as possible. If ground handling is done by third parties, different (commercially related) goals could apply.

Delivery Controller

A delivery controller, abbreviated as DEL, checks the flight-plan and gives clearance to the corresponding aircraft. The delivery controller checks at least: the provided SID, height and speed of the respective flight-plan. If the flight plan is approved, the delivery controller gives clearance and communicates this with the call-sign of the flight-crew, destination, assigned runway, SID, squawk code[2] and initial climb [32]. Clearance is usually available 25-50 minutes in advance. After confirmation of the flight-crew, the aircraft is put in the pre-departure sequence. Communication can be done either via radio communication or data link. If a departing aircraft gets assigned a CTOT from NMOC, this is communicated by the delivery controller.

Outbound planner

The outbound planner is responsible for giving, or forwarding the decision to provide, clearance to aircraft that are ready for start-up. As mentioned in Section 2.1.3, milestones 12 and 13 indicate the point where the flight crew requests ATC for start-up. This can be done if the aircraft is within its TSAT window of +/- 5 minutes [2]. If the flight crew requests for clearance before this time window, a new TOBT must be issued by ground handling as noticed from live ATC communications [33]. If clearance is requested after the TSAT window, a new TOBT must be entered by ground handling. It is the responsibility of the outbound planner to determine if the start-up request can be granted and/or must be forwarded to the ground controller. At AAS, if an aircraft has to receive push-back from a tow truck, start-up clearance is given by the Ground Controller and therefore the request will be forwarded by the outbound planner. Aircraft that can taxi-out on their own get clearance from the outbound planner.

In deciding whether start-up clearance can be given/forwarded, an outbound planner uses the TSAT windows of all aircraft as shown in the CPDSP system. As mentioned before, the CPDSP system uses sequencing rules (dependent on CTOT, SID, WTC and runway capacity) for departing aircraft. With the use of this system, an outbound planner can check previous clearances and decide whether new clearances could potentially

[2] A code given by ATC to identify each aircraft. It is entered into the transponder by the flight crew [31]

cause queues at the runway or conflicts when pushing back at the gate. Arriving aircraft are also taken into consideration as they can affect traffic flows.
If clearance is granted by the outbound planner, it communicates the atmospheric pressure adjusted to Mean Sea Level (MSL) and Automatic Terminal Information Service (ATIS)[3] to the flight crew. It is noted that at AAS, the positions of delivery controller and outbound planner are (sometimes) combined [33].

Ground controller
The ground controller is responsible for guidance of all ground surface operations at the airport. At small airports the ground controller is also responsible for delivery clearance, as described above. A ground controller provides instructions related to push-back and taxiing for both aircraft and other ground vehicles. At AAS the ground controller also provides start-up approval[4] for aircraft that are ready for push-back. All movements on or crossing an active runway are controlled by the Runway controller. The ground controller is fully responsible for the taxiing operation and must ensure that aircraft arrive at the correct time at the runway to comply to the (pre-) departure sequence. As mentioned in Section 2.1.2, the ground controller may assign priority to specific aircraft at intersections in the taxiway network. This way, it is able to ensure correct arrival times for all aircraft. Its main goal is to ensure safety, i.e. prevent conflicts from happening, and provide efficient guidance to all ground surface vehicles.

Runway controller
The runway controller is responsible for safe operations on, and close to, active runways. It does so by using time-based separation between departures and arrivals, as depicted in Table 2.1. It also gives clearances for departing and arriving aircraft to take-off and land. The separation times are dependent upon the type of flight (arrival or departure) and the MTOM of the corresponding aircraft. If no time is provided, there is no separation necessary due to wake turbulence restrictions. In that case, separation can be determined by the runway controller.
The runway controller is also responsible for guidance when crossing an active runway. This is done by means of stop-bars around the runways: a line of illuminated red-lights over the taxi-way indicating that crossing is prohibited. The stopbar lights are turned off when crossing is allowed.

NMOC
The NMOC is part of Eurocontrol and optimises traffic flows by constantly balancing supply and demand while keeping in mind safety and efficiency in the European network [34]. In contrast to the aforementioned parties, it is not part of local ATC. NMOC is connected to A-CDM systems around Europe and receives accurate updates on the TOBT and TTOT of all flights. NMOC uses this information to create an overall picture of the network, which gives it the ability to monitor traffic load against airspace capacity. NMOC can assign a CTOT to a flight (as mentioned in Section 2.1.3), which is communicated to the flight crew via the Delivery controller.

2.2.2. Interactions between relevant parties

The interaction links as numbered in Figure 2.13 will be described in this section in chronological order.

1 The ground handler communicates the TOBT to the outbound planner. It provides an accurate estimate of the TOBT 10 minutes in advance of the current TOBT, or enters a new TOBT in case the current window has not been met. These TOBT updates are all communicated via the A-CDM system.

2 Interaction between ground handling and the aircraft (flight-crew) happens in several ways, dependent on the role of the ground handler. Ground handling and the flight-crew communicate the progress of the turn-around (fuelling, maintenance, servicing) and discuss whether the time-schedule will be met or need to be changed. Also, the ground handler is responsible for issuing a new TOBT. Therefore, the flight-crew must communicate their new TOBT to ATC via the ground handler.

3 The flight-crew is responsible for safe operations by maintaining a safe separation with other aircraft and give priority (for example to aircraft coming from the right) when moving over the airport surface, being supported by ATC.

[3]Automatic message containing at least local weather conditions at the airport and active runways
[4]Start-up approval indicates that the aircraft engines may be started.

4 As described above, the delivery controller provides clearance by checking the specific flight plan. An aircraft requests clearance to the delivery controller by communication via radio or data-link. The delivery controller communicates the clearance to the aircraft, together with: assigned runway, SID, squawk code and other relevant flight route clearances. Also, the delivery controller requests the flight crew to contact the outbound planner. The flight crew confirms the received information by repeating it.

5 After contact with the delivery controller and having obtained clearance, the flight-crew contacts the outbound planner via radio communication. If the flight is within its TSAT window, the outbound planner provides start-up clearance, or forwards the flight-crew to contact the ground controller for start-up clearance in case it must receive push-back. The outbound planner communicates the atmospheric pressure (MSL) and ATIS information to the flight-crew via radio communication.

6 A flight-crew contacts ground control for push-back clearance via radio communication. The ground controller checks if push-back clearance can be given, looking at other aircraft in the vicinity and their push-back procedures. Clearance is communicated by the ground controller together with taxiing instructions via radio communication. All statements are repeated by the flight-crew to confirm receiving the message. When the aircraft is ready for taxiing, the ground controller is notified.
During taxiing, the ground controller can provide the flight crew with taxiing instructions, e.g. prioritise aircraft to solve conflicts. Arriving aircraft are also guided by the ground controller. Possible holding instructions, if a gate is still occupied, are given by the ground controller.

7 The runway controller communicates clearances to line-up and take-off, to land or cross an active runway via radio communication. As mentioned before, these instructions are visualised to the flight crew by means of stop-bars on taxiways crossing a runway.

8 The delivery controller forwards the flight-strip[5] and associated responsibility to the outbound planner after confirmation of the flight-crew. The flight-strip can be returned by the outbound planner if changes occur such that the clearance must be revised (runway reconfiguration, changes in departure instructions).

9 The outbound controller and ground controller have a similar interaction as described for the delivery controller and the outbound controller. Responsibility of the flight strip is carried over from outbound controller to ground controller if it fits the workload of the ground controller. This can be visually assessed by the outbound controller by looking at the amount of flight strips. Responsibility can be handed back to the outbound controller if it turns out that an aircraft is not ready yet.

10 A flight-strip is handed over from the ground controller to the runway controller or vice-versa if: a departing aircraft arrives at its assigned runway, an arriving flight enters the taxiway system or an aircraft needs to cross an active runway.

11 For every flight in or over Europe, the flight crew must submit a flight plan to the NMOC. This is used to check and verify the flight plan to ensure safety and security [34], by ensuring that there are no conflicts in the submitted flight plans.

12 As described above, the NMOC ensures balance between capacity and demand in the European network. It does so by assigning a CTOT to specific aircraft, as described in Section 2.1.3. This can be communicated via local A-CDM, so information can be accessed by all parties. However, in most cases a (updated) CTOT is communicated to the flight-crew via the delivery controller.

This Chapter provided an overview of AAS and the ground operations carried out on the airport. As mentioned in the introduction, AAS has the ambition to have a climate-neutral airport operation in 2030. Some developments in air traffic ground control relate to this same goal to lower emissions and further automate ground operations. One concept covers the Follow-the-Greens concept, as currently investigated for implementation at AAS. The other part relates to the concept of autonomous engine-off taxiing, to (further) lower fuel use and emissions from the aircraft taxiing phase. Both concepts will be elaborated in Chapter 3.

[5] Electronic or paper strip containing the data from one specific flight plan, used by ATC for the display of flight data [35]

3

Developments in Air Traffic Control

With the number of aircraft movements expected to continue growing in the coming years, more and more possibilities to increase ATC capacity are being investigated. Regarding ground control, some developments (can) have significant consequences for current ATC procedures. Development regarding airport surface operations will be briefly discussed in this chapter, starting with the SESAR concept of FtG in Section 3.1. Thereafter another SESAR area of research will be described in Section 3.2, covering (non-) autonomous engine-off taxiing.

3.1. SESAR Follow the Greens

FtG is based on guidance assistance for the flight crew through airfield ground lightning. Section 3.1.1 will elaborate on the concept of FtG, as well as both the relevant parties involved in operation and the current status of implementation at AAS. FtG is a sub-part of the overarching A-SMGCS, which will be discussed in Section 3.1.2.

3.1.1. General description

Already in 2008, SESAR and NextGen put the emphasis on operation and guidance of aircraft on a 4D trajectory basis [36]: three space dimensions plus time, for the entire operation of an aircraft from gate to gate. Part of the entire gate to gate operation of an aircraft is the ground surface operations, like the taxiing procedure from gate to runway and vice-versa. From this desire to guide aircraft on a 4D trajectory basis over the airport surface originates the SESAR concept of Follow the Greens.

The operational concept of FtG is best explained using a visual example, as provided in Figure 3.1. On the left, a representation is given of a regular taxiway system with its corresponding taxiway centre lines in green. The FtG system provides the (flight) crew with visual navigation support by means of the Airfield Ground Lighting (AGL), and guides the aircraft, or other ground vehicles, over the taxiway system. Aircraft are used in this section, but all vehicle drivers can be guided by FtG. As can be seen on the right of the figure, guidance is visualized as a defined stretch of illuminated centre line lights in front of the aircraft [37]. The flight crew subsequently needs to follow the illuminated lights, hence the concept: Follow the Greens. It can be seen that all centre line lights not needed for the taxi path are deactivated. This way, the only lights that are illuminated indicate the path to be followed by the flight crew. By moving over the illuminated path, the aircraft 'pushes' the illuminated part of the segment up front, while all lights below and behind the aircraft are automatically switched-off again while driving over them.

To ensure safety, the system is provided with a logic to assure longitudinal separation and wingtip clearances throughout the taxiway system. This can be visualized to the flight-crew by smart use of the AGL. In the vicinity of intersections, individual lights can be switched on and off. This is done to indicate the taxi route as clear as possible to the crew. On the other hand, acceleration or deceleration commands are provided by illuminating more or less centre line lights. In case the aircraft must decelerate because it is closing in on a leading aircraft, the length of activated lights in front of the aircraft is reduced. In case the aircraft can accelerate, a longer path is illuminated in front of the aircraft [38].

Aircraft can also be mandated to stop when taxiing over the airport surface. This is indicated by illuminating a red line of stopbar lights, as mentioned in Section 2.2.1. Red lights always indicate stopping instructions, while yellow- or flashing lights indicate caution. Additional information or guidance can be provided by the ground controller via radio communication.

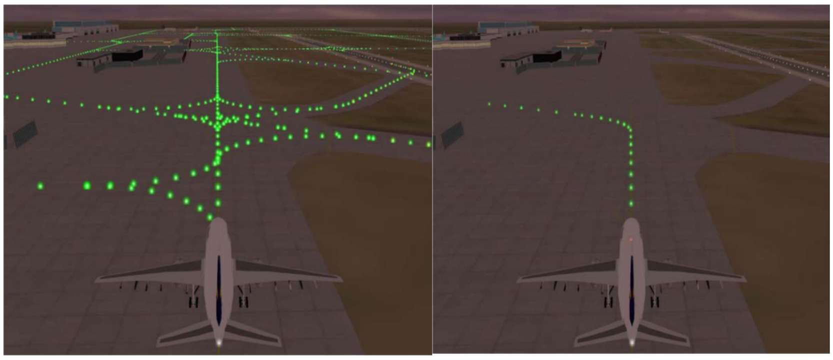


Figure 3.1: Follow the Greens concept [9]

The first signs of an FtG-like system originated 10-20 years ago at London Heathrow and Munich Airport. This system, however, relied on the manual or semi-automatic switching of lights by an operator [37]. The current implementation of FtG is more automated and is based on the concept of distributed control. This means that, for example, priorities at intersections are managed locally, instead of centrally by means of a ground controller. This has the benefit that the ground controller can now supervise the operation via the activated lights on its radar display, instead of having to actively provide guidance to all aircraft. Figure 3.2 gives an example of the HMI of a controller. As can be seen, each aircraft is labelled with its identification and its FtG path. Planned stops (e.g. to give priority to another aircraft) along the taxi route are visualised by a red dot, as can be seen for example in front of flight AFL2683.
After engine start-up, the controller can select a route proposed by the system and, if needed, make adjustments manually [39]. The controller can still overrule the FtG system during operations, by means of radio communication.

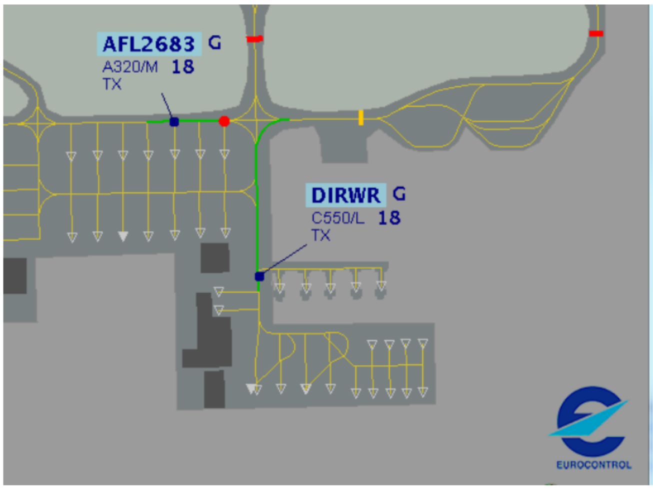


Figure 3.2: HMI for controller awareness of FtG [10]

The FtG system has several benefits for airports and the corresponding actors involved in operations [9]. These benefits have been confirmed and validated through real-time simulations at Frankfurt, Paris and Munich, as well as live trials at Riga airport [38, 40]. A list of benefits of implementation of FtG:

- Increased flight crew performance and reduced workload, due to visual guidance instead of complex radio communication clearances and instructions. This causes less radio communication between the ground controller and flight crew.

- Adverse effects in LVC are eliminated, as guidance is not dependent on the line of sight of the ground controller.

- Increased controller performance, due to their change of role from guiding to supervising. Also, ground controller capacity is increased as taxi routes are not dependent on reading instructions anymore (reduced workload and increased situational awareness also apply to the controllers).
- Less stop and go's and more regular taxi speeds, resulting in less emissions during taxiing and more accurate taxi time predictions.
- Increased safety due to fewer route deviations.

FtG at AAS

Together with Lisbon, Riga and Zurich, AAS is one of the first airports to plan for implementation of FtG. The concept is currently being investigated and researched by AAS and ATC Netherlands. A plan for the project is expected in June 2020, while actual implementation of the system is to be expected around mid 2026 [41]. The project of FtG is a sub-task of the joint sector Integral Safety Management System (ISMS): a cooperation between AAS group, ATC Netherlands, airlines, ground handlers and fuelling service providers with the goal to continuously improve safety on and around AAS.

3.1.2. A-SMGCS

As described by Eurocontrol and ICAO [10, 42], an A-SMGCS is a system that provides support for surface movement operations at an airport during all weather conditions, based on defined operational procedures and ensuring the required level of safety. The main service it contains is surveillance, in order to provide positions, identification and tracking of mobiles on the ground.

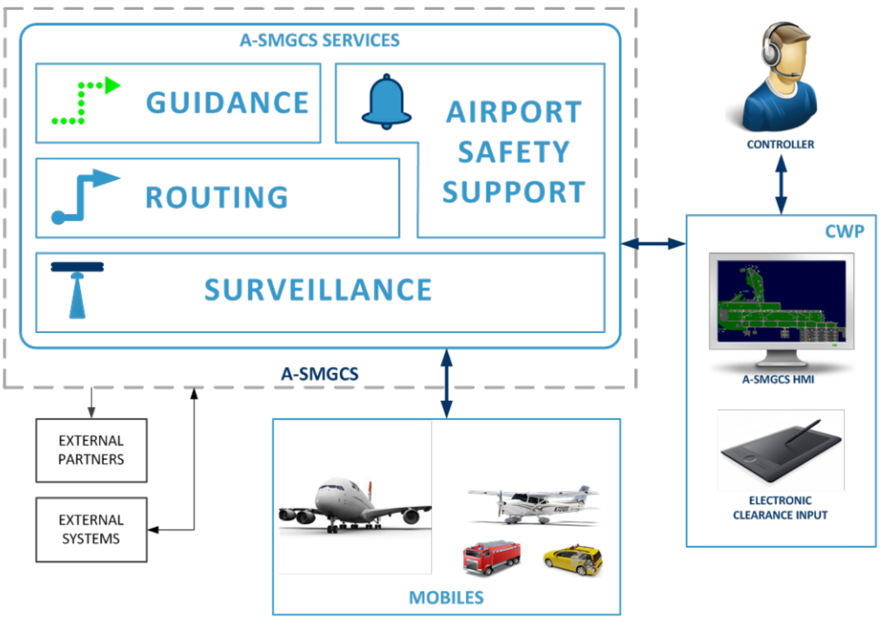


Figure 3.3: Overview of A-SMGCS [10]

Besides surveillance, as can be seen in Figure 3.3, A-SMGCS can include a combination of services like airport safety support, routing and guidance. A controller can access the A-SMGCS via a HMI (Figure 3.2). The controller can provide clearances via the electronic clearance input. The HMI consists of a visualisation of the airport with identification and positions of all relevant traffic. Furthermore, external partners can access the A-SMGCS and external systems can be linked with/to the system.

As indicated the A-SMGCS is divided into four named services. The surveillance, airport safety support and routing services are mainly supportive of ATC. The guidance service includes direct support to mobile operators[1] and includes the FtG system. The four services will be briefly discussed below.

[1] Flight crew operating an aircraft, but also includes ground handler vehicle drivers, airport operator vehicle drivers, emergency services vehicle drivers, security services vehicle drivers, Air Navigation Service Provider (ANSP) vehicle drivers and occasional airside vehicle drivers [10]

Surveillance Service

The core of an A-SMGCS implementation consists of the surveillance service. The surveillance service provides situational awareness of traffic at an airport, within a pre-determined coverage volume. It provides identification, position and tracking of aircraft and other vehicles on the airport surface. A distinction can be made between cooperative mobiles and non-cooperative mobiles (or intruders). Traffic is presented by means of a synthetic representation with the usage of sensors. It gives the controller an extra tool to guide all vehicles safe and efficiently over the airport surface, independent of the controllers local line of sight.
The surveillance service can be used in cooperation with A-CDM, as it can, for example, provide accurate landing times of aircraft (ALDT).

Airport Safety Support Service

The Airport Safety Support Service enables controllers to prevent hazards and/or incidents resulting from operational errors, or deviations of actors involved in ground operations. It depends on an operational surveillance service and supports controllers by [10]:

- Anticipating potential conflicts and detecting conflicts and incursions;
- Detection of vehicles that are not following their given clearance;
- Providing alerts.

Alerts from the safety support service are visually presented to controllers via the HMI and contain: stage of alert (information, or if more critical an alarm with an audible warning), type of alert situation (three different functions, see below) and identification of the vehicle(s) concerned. The system requires the designation of an RPA; an area including runways and sensitive areas on the airport surface. If a vehicle enters, or is expected to enter, this area, an RPA alarm is provided. The airport safety support service includes three functions that provide alerts [10]:

- **RMCA** is the only short-term conflict alerting tool, monitoring movements on and near the runway(s) by use of surveillance data. It takes into account: runway configuration, runway procedures, different types of operators on the airport with their corresponding speeds and locations and meteorological conditions.
- **CATC** is a more predictive tool and provides alerts if the clearance given by a controller is not permitted in terms of operational- or safety point of view, compared to clearances given before. It gives the controller the opportunity to immediately fix the clearance.
- **CMAC** provides alerts to controllers when the A-SMGCS detects vehicles or aircraft deviating from procedures or clearances. It is a means to (early) detect potentially hazardous situations.

Routing Service

The routing service generates routes for all vehicles and aircraft on the airport surface. A controller is allowed to modify, adjust or create a custom route. The routing service uses information from the mobile information database[2] (stand, expected holding point, pre-defined runway and predicted runway exit) to automatically generate routes. It receives information from the airport operations status database regarding airport layout, runways in use and other operational specifications of the airport. It also uses the surveillance service to identify and localise all vehicles and aircraft on the airport surface. After route planning, the routes are stored back in the mobile information database with their corresponding status: planned, cleared and pending. This also allows other services to access the most up-to-date routing of a vehicle. If a new restriction occurs, all affected routes will be updated by the routing service.
A planned route is indicated in white on the HMI, as can be seen in Figure 3.4. It can represent a route from gate to the runway for departing aircraft as shown in the figure, or a taxi route from the runway to the gate for arriving aircraft. The planned route is based on the operational situation and relevant flight data. If, for example, a runway change occurs or a taxiway is closed, this is taken into account in the route planning.
A cleared route is authorised by a controller and is indicated by a solid green line as can be seen in Figure 3.5. This is the moment the aircraft starts following its planned route, which could have been changed manually

[2]Database containing information related to each aircraft or vehicle, like for example id, type, flight-plan, Secondary Surveillance Radar (SSR) code, stand and more [10].

by a controller. If manual changes by a controller impose a hazardous situation, an alert can be provided by the CMAC.
A pending route is visualized by a dashed green line as can be seen in Figure 3.5 just before the runway. It indicates a route, or part of a route, that has not been cleared yet.

Figure 3.4: Planned route presentation on HMI [10]

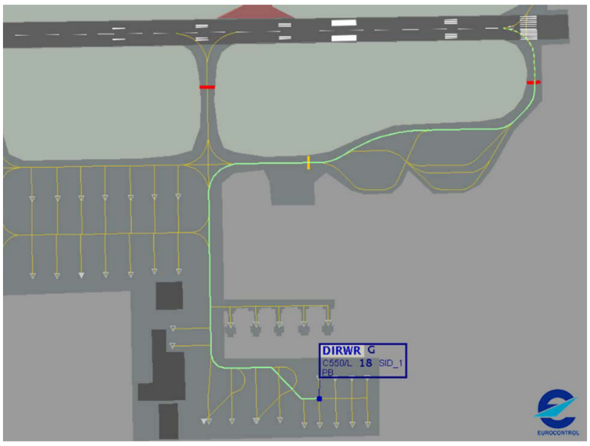


Figure 3.5: Cleared and pending route on HMI [10]

The routing service also calculates taxi times corresponding to a specific route. These taxi times, in minutes, are used by A-CDM (described in Section 2.1.3) for the determination of the TTOT for outbound, respectively EIBT for inbound flights.

Guidance Service

The guidance service consists of individual guidance for vehicles with a cleared taxi-route, by means of visual aids. A distinction is made between three functions, of which two have been described already:

- **Automated switching of Taxiway Centreline Lights (TCL) and stop-bars**, which denotes the FtG system previously described in Section 3.1.1.
- **Automated Activation of Advanced-Visual Docking Guidance Systems (A-VDGS)** provides automated docking guidance for the flight crew to park the aircraft in the correct position. It is situated such that the flight crew can easily read it off and indicates whether the aircraft is correctly lining up in front of the gate. An example is given in Figure 3.6, showing the aircraft type (A340) and its corresponding lateral guidance, indicated by the aircraft symbol and the centerline. Longitudinal guidance is also provided in the form of display text. The visual docking system completes the automated guidance from landing, via the taxiway centre lights, to the designated gate. At AAS the docking guidance system is manually operated by ground handling, which ensures safety and the correct aircraft type to be entered in the system.

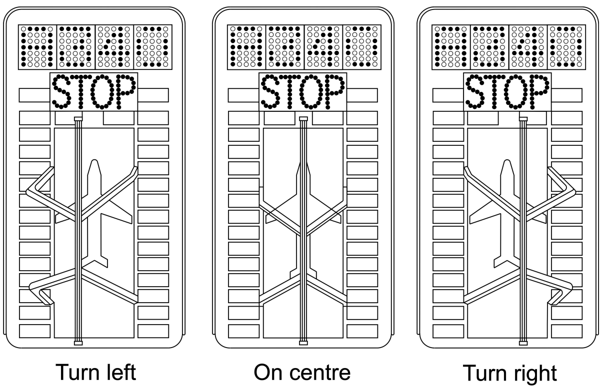


Figure 3.6: Visual docking system [2]

3.2. SESAR (Non-) Autonomous Engine-Off Taxiing

The SESAR programme focuses on the modernisation of the European ATM system by coordinating all ATM related research efforts in the European Union [43]. In the technical specification of its exploratory research call [3] a comprehensive description is given of all research topics, and their associated challenges, that are awarded by SESAR. It is based on the needs as identified in aviation-related planning documents, like the European ATM Master plan. A distinction between two areas is made:

- **ATM Excellent Science & Outreach** focuses on a conceptual level on investigating new research areas for ATM based on developments in non-ATM related sectors.
- **ATM Application-Oriented Research** aims at maturing concepts for ATM in order to further industrialise and validate these concepts. This is done by investigating and testing these technologies, and further developing them for use in practice.

With the objective of AAS to have an emission-free ground operation at all airports in the Netherlands [17] in 2050, a particularly interesting and relevant SESAR research topic is that of the implementation of (Non-) Autonomous Engine-Off Taxiing. It is a sub-part of the working area 'innovations in airport operations' and covers application-oriented research.

Application area 1 considers solutions to incorporate autonomous[4] and non-autonomous engine-off taxiing for surface operations at airports, to further increase safety and reduce fuel consumption and emissions [43]. The operation considers both taxi-out and taxi-in operations: from the gate to a holding point before line-up (taxi-out) and from the runway exit to the gate (taxi-in).
This can be achieved in a non-autonomous manner by means of external equipment, like for example towing trucks or a taxibot. In case of autonomous operation, this may be realised by means of electric motors on the Main Landing Gear (MLG) or Nose Landing Gear (NLG) and power supply from the Auxiliary Power Unit (APU). Further clarification of the current state-of-the-art systems is given in Chapter 4. In the case of autonomous operations, the pilot remains the operator with central control from the cockpit.

The goal for SESAR is to provide an operational concept description and associated operational procedures for (non-) autonomous engine-off taxiing. Important aspects that need to be covered are respectively [43]:

- **For non-autonomous engine-off taxiing**: the development of an operational concept which should include at least: (procedural) management of the towing vehicles, designated parking locations and necessary communications between all actors involved.
- **For autonomous engine-off taxiing**: procedural changes should be considered regarding engine start-up / shut-down, as these are supposed to change due to the implementation of taxi systems. Also, performance requirements need to be developed including speed and acceleration of the taxi systems, as well as the impact of new systems on airport operations and ATC.

Besides the operational challenges that need to be covered, there is also a clear scope for this research call:

- The main focus of research should not be to provide quantitative benefits due to engine-off taxiing, like reducing fuel burn/emissions, improved safety and similar benefits, as these have been determined already.
- Neither should research focus on technical aspects of systems to perform engine-off taxiing.

It can be concluded that significant research and validation have been done already on the topic of FtG, following from the fact that it is already being implemented at airports. On the other hand, the concept of (non-) autonomous engine-off taxiing is still in the conceptual phase and requires further research into all aspects related to, and affected by, the implementation of such taxiing operations. The ambition of AAS to further decrease emissions indicates that airports are searching for innovative ways to cut emissions. Autonomous engine-off taxiing is one of these opportunities. Therefore, the next chapter will focus on the state-of-the-art engine-off taxiing systems, as well as research done in this field. This should provide an overview of what has been done already, possibly indicating a research gap that can be filled by this study.

[3] Innovative or unconventional ideas, concepts, methods and technologies to increase the performance of the future ATM system [43]
[4] Autonomous taxiing refers to aircraft taxiing individually using on-board systems, while non-autonomous taxiing refers to taxiing operations by means of external machines [44]

4
State-of-the-art in Engine-Off Taxiing

Engines are designed to be as fuel-efficient as possible in cruising conditions, consequently meaning that aircraft engines are less efficient in idle conditions. During taxiing operations, aircraft engines are causing significant pollution: in 2002 it was found that 56% of the NOx generation for London Heathrow was caused by aircraft taxiing [11]. As described in Chapter 3, a promising innovation for airports and the aviation industry to further cut emissions is the concept of aircraft engine-off taxiing. A clear distinction has been made by SESAR between internal and external systems for engine-off taxiing. This chapter will describe the current state-of-the-art in engine-off taxiing systems: Section 4.1 will describe internal systems and Section 4.2 discusses external systems. Thereafter, in Section 4.3, research will be reviewed regarding engine-off taxiing to give a concise overview of the current state. This Section considers implementations of engine-off taxiing and their corresponding results.

4.1. Onboard Systems

Onboard systems are usually installed in either the NLG or the MLG. A major benefit of the installation of an onboard system in the NLG is that it is easier compared to the MLG, due to the structure being simpler because of the absence of brakes [45]. One of the first practical concepts of an onboard electrical taxiing system was the WheelTug in 2005. As can be seen from Figure 4.1, the WheelTug is installed on the NLG. It is powered by the APU, meaning it is not entirely emission-free. However, it is proven that emissions are significantly lower compared to engine taxiing [11]. The major benefits of the WheelTug are the fact that the Turn-Around Time (TAT) can be significantly reduced due to the automation of push-back maneuvers. Due to increased mobility, aircraft could be parked parallel to the terminal to be able to connect two passenger bridges at the same time [46]. On the other hand, the installation of a WheelTug system increases the total weight of the aircraft causing extra fuel-burn during flight. The extra weight also causes a change in aircraft architecture and possibly a mandatory new certification of (parts of) the aircraft.

Figure 4.1: WheelTug [11]

Figure 4.2: DLR Motor [12]

Another onboard electrical system located at the NLG is the DLR motor as shown in Figure 4.2. A Permanent-Magnet Synchronous Motor (PMSM) is installed in the wheel rim, having the capability of reaching a speed of 25 km/h when installing on an Airbus A320. It is based on fuel cell technology, which is still immature: hydrogen storage and safety are not yet at the level that the aviation industry needs [11]. Therefore, it currently cannot be regarded as available technology.

An onboard system located at the MLG has been designed by Safran in collaboration with Honeywell. Airbus also designed an identical MLG onboard electrical system called the eTaxi, which is intended purely for the A320 [47]. Both have been successfully developed and tested in accordance with the following requirements [11]: reach 20 kts in 90 seconds, reach 10 kts in 20 seconds for active runway crossing and a breakaway torque at 1.5% slope and Maximum Take-Off Weight (MTOW). Similar to the Wheeltug, both are powered by the APU. Difficulties arise when implementing systems in the MLG, due to the thermal influence of the brakes as well as forces due to landing and take-off. Although Safran decided to terminate the development of the Electric Green Taxiing System (EGTS), it remained involved in further research regarding EGTS on the MLG. To this date, only Airbus presented an operational copy of its eTaxi.

Studies have shown that onboard systems can significantly reduce fuel burn when looking at the entire operation of an aircraft [45]. Having a 500 kg Advanced Ground Propulsion System (AGPS) on board a mid-size aircraft can reduce fuel burn with around 2.5%. As mentioned by Okuniek et al. [44], a drawback of such an onboard system is that it is less beneficial for long-haul flights due to the extra weight. Therefore, it is to be determined whether onboard systems are useful for large hub airports. Other drawbacks mentioned are increased maintenance costs due to the new system and additional tyre wear due to different maneuvers [48]. These factors need to be carefully considered when implementing an AGPS. To conclude, the fact that Safran terminated its research into onboard electrical systems in 2016 is not promising for the future.

4.2. External Systems

External systems are AGPS that can be connected to the aircraft and tow it over the airport surface [45]. A way of increasing engine-off taxi time is the concept of operation towing: human-controlled tugs that tow the aircraft from A to B [49]. An example of an external system that is used in practice is the towing truck with an external tow-bar, as shown in Figure 4.3. Implementation of operational towing has the advantage that these kinds of trucks already exist in airports. Also, it reduces flight crew workload and increases redundancy in taxiing due to extra personnel. Major drawbacks are the additional complexity in airport ground movements and extra personnel needed for carrying out the towing operations. Besides, the tow-bar increases fatigue loads shortening the NLG life cycle [45, 50].

Figure 4.3: Towing operation using external tow-bar [13]

Figure 4.4: Towing operation using TaxiBot [14]

Another external AGPS that will be considered is the TaxiBot (Figure 4.4), as developed in 2012 by Israel Aerospace Industries [50] and certified for dispatch towing for both the A320 and B737 [11]. It is operational and has been deployed in 2018 in India. It looks very similar to current, clamped to the NLG, towing vehicles in operation at AAS and many other airports and allows for taxiing up to 23 knots. The aircraft NLG is lifted off the ground onto a platform and the TaxiBot wheels rotate with the NLG wheel, while ensuring the NLG force limits are not exceeded.

The push-back operation has to be carried out by a TaxiBot driver, whereas the remainder of the taxi operation is to be controlled by the pilot. The pilot is in control of the TaxiBot via steering commands from aircraft

tiller and brake pedals, as in regular taxiing operation. The major benefits of the system are that an aircraft can taxi with its engines off, leading to a lowering of emissions as well as noise mitigation. Besides that, the TaxiBot provides better grip in case of snow/ice, immediate taxi after push-back and no engine blast in the vicinity of gates [50].
There are also drawbacks to the TaxiBot system, besides the investment costs needed to purchase such equipment. As mentioned in Guo [45], airports may need additional roads and parking locations for TaxiBots when not towing an aircraft, returning from a towing operation or when waiting for an arriving aircraft. Secondly, the TaxiBots must be guided on the airport surface, increasing the complexity of airport surface operations. Even if dedicated roads were to be made, a responsible party for controlling this extra traffic must be appointed.

As mentioned by Postorino et al. [51], a system like the TaxiBot is preferred over a tow-bar system as it allows higher speeds, resulting in faster operations and also a safer operation for the aircraft itself due to the NLG being more secured by the clamping system. Operational challenges for external systems arise when considering the engine warm-up time needed before take-off: areas must be designated for engine warm-up or procedures must be designed in such a way that the engines can be started during towing. Besides that, the taxi operations for aircraft will change significantly, having effects for the parties involved in the ground operations. At last, the increased complexity of the airport operations should be re-evaluated to be able to handle all traffic in a safe manner [45]. Such examples of procedural changes and how this is currently tackled in research will be covered in the next section.

4.3. Relevant research on the implementation of engine-off taxiing systems

Although the aforementioned taxiing systems all have the main goal to reduce fuel consumption and engine emissions during taxiing, internal and external systems have different specifications and face different challenges. This consequently leads to a division in research regarding implementations of engine-off taxiing. Similar to the distinction in taxiing systems, research considers either an implementation of internal taxi systems, external taxi systems or compares the benefits and drawbacks of both on a more general basis. The major benefits and drawbacks of engine-off taxiing systems have been touched upon already in the previous sections. This section will focus on the current state of research regarding implementations of engine-off taxiing systems. Implementations of internal systems will be discussed in Section 4.3.1 and external systems in Section 4.3.2.

4.3.1. Research on internal taxi systems

Research on internal taxi systems considers the implementation of onboard taxi systems as have been described in Section 4.1. It was found that a clear distinction can be made between two types of research in literature:

- Research that considers the development of internal taxi systems. For example, in [52] the concept of regenerative braking is tested to recapture energy from the electric taxiing system. On the other hand, Schier et al. [12] developed and tested a nose wheel system based on permanent magnets. A significant amount of papers and patents can be found online that consider the technical aspects of onboard taxi systems. However, this type of research is more related to the technical development of such systems based on a set of requirements. As this study focuses more on the operational aspect of taxi systems, this branch of research shall not be considered further in this literature study.

- Research analysing the operational benefits associated with the implementation of onboard taxi systems. Such studies are carried out in [44, 45, 48, 53–55] and focus, for example, on the quantification of fuel savings or reduction of emissions due to onboard taxiing systems.

Regarding the second branch of research, the emphasis is mainly put on the quantification of fuel savings due to onboard electrical taxi systems. Some studies explicitly investigate a specific onboard system, like the EGTS [44, 48, 54] while others only quantify the benefits of electrical taxi systems without mentioning the specific taxi system [44, 45, 53]. It was mentioned in all papers that onboard taxi systems could significantly reduce both fuel consumption and emissions. However, these fuel savings are mainly achievable for short- to medium-haul flights. Because of the increased aircraft weight due to installing an onboard electrical taxi system, additional fuel is burnt in the flight phase. Aircraft that are primarily suitable for onboard systems are the A320 and B737 family, due to their high ratio of taxi time to flight time [53].

From a comparison of the papers, the following remarks can be made regarding research on internal taxi systems :

- As mentioned, the main objective in research is to quantify the fuel savings due to onboard systems for a specific aircraft [45, 53, 54]. Hospodka and Dzikus et al. [48, 53] analysed fuel savings for different combinations of flight times and taxi times, while in [54] only a single flight cycle is considered. Guo et al. [45] used emission/fuel factors per aircraft with the taxi-times and aircraft types at the top 10 U.S. airports. They included only a single fuel flow factor during taxiing, i.e. they did not include any start/stop movements.
- Only in [48, 53] a sensitivity analysis is done on the variables determining the fuel savings. Hospodka analysed the sensitivity of fuel savings when varying: taxi time, price of fuel, average maintenance cost or flight time. Dzikus et al. investigated the sensitivity of fuel savings on changing APU fuel flow, engine start-up time before take-off or weight of the onboard system. In both papers, it was found that the fuel savings are mainly dependent on the total taxi time while being less dependent on the flight time.
- Roling et al. [56] are the only ones to investigate the minimum speed capabilities of on-board electrical systems. They found that for operations at AAS, the system must be capable of reaching a taxi speed of 10 m/s.
- Dzikus et al. [53] are the only ones to compare their fuel savings for specific taxi time/flight times to actual taxi/flight time data. Using taxi-in and taxi-out times as well as flight distances from over 2 million flights from 10 US carriers, it was verified that onboard systems could be particularly suitable for A320/B737 type of aircraft.
- Several papers mention the consequences of autonomous taxiing using onboard systems for ATC [44, 48, 55]. These are related to aircraft needing time to warm up the engines for take-off mode (2-5 minutes), which changes procedures and clearances for the controllers. Okuniek and Beckmann [44] are the only ones to really investigate the possible impact of onboard taxi operations on ATC procedures. They provided timelines of both dual-engine taxiing as currently being used and (future) onboard electrical taxiing. It was concluded that workload is being transferred from apron control to ground control, due to the push-back being shortened and the need for aircraft to ignite their engines further on in the taxi process. They also pointed out the need for ATC to efficiently guide aircraft during mixed operations (dual engine, single-engine, and engine-off taxiing) on the airport to limit delay and keep the fuel savings due to engine-off taxiing. However, no modelling has been carried out to test these procedural changes.
- Okuniek and Beckmann [44] considered possible locations on the airport layout for engine warm-up. It is concluded that a designated area for engine warm-up is preferred, giving the ground controller the opportunity to focus on selected areas. These areas should be chosen in such a way that aircraft are able to return to the gate without blocking active taxiways. There should also be authorised personnel available to assist the flight-crew with engine ignition. The main drawback of this research is that no practical application has been considered. A schematic airport layout is used as an example, which barely resembles reality.
- Hospodka [48] is the only one to mention additional drawbacks of an onboard system like certification, changes in the Master Minimum Equipment List (MMEL) and issues under poor friction conditions. It can be questioned whether the implementation of onboard systems is reasonable if such drawbacks are still mentioned.

Although some studies conducted a (very limited) comparison of fuel savings for different aircraft types [54] or an entire set of flights [45], little attention is given to the mixed fleet of aircraft at an actual airport. Besides, neither did any of the papers take into account the operational consequences of the changes in taxi procedures when considering a full flight schedule. It could, for example, occur that taxiways or runways get congested due to delayed taxi procedures, as ATC must accurately take into account the engine warm-up location and time just before take-off. This could lead to more start-stop movements of aircraft and possibly

a reduction or even vanishing of the fuel savings. An operational analysis should be carried out regarding the choice of location for the engine warm-up areas.
On the other hand, it is mentioned in every paper that fuel savings are primarily feasible for A320/B737 type of aircraft. As shown in Chapter 2, around 20% of the traffic at AAS originates from intercontinental flights. Therefore for an application at AAS, other means to reduce fuel burn and emissions could be considered like for example external taxi systems.

4.3.2. Research on external taxi systems

It is found that the TaxiBot is the only external system currently being considered in practice for engine-off taxiing. Although the TaxiBot had been developed already in 2012, there is not a significant amount of research on the implementation of it. From the papers considered [45, 49, 51, 57–62] a branching between research goals can be made:

- Research that focuses on the benefits and/or consequences of the implementation of external taxi systems like the TaxiBot [45, 51, 58, 59, 61, 62]. These papers are mainly focused on the quantification of fuel savings when introducing external taxi-tugs [45, 51, 59] or optimal allocation of external taxi-tugs to minimise fuel usage [61]. A single paper investigated the consequences of external taxi systems for air traffic controllers [58]. Guillaume [62] is the only one to optimise for a combined minimisation of: cost of travelling an edge, waiting time at a node, cost of delay and cost of using a towing vehicle.

- Research that focuses on the automation of towing vehicles (like TaxiBot) [49, 57, 60]. These papers investigated the concepts of self-managing conflict resolution and self-driving of the towing vehicles.

- A single research focused on the technical requirements of a fully electric towing truck [61]. Although a quite thorough description is given of the technical specification and feasibility of the fully electric concept has been proven, this part of research is outside of the scope of this literature study.

The first branch of research mainly focuses on case-studies proving the value of implementation of external taxi systems. They assume that external taxi systems are fully developed and ready to be implemented. Regarding this branch of research, the following remarks can be made:

- Some papers verified fuel savings by means of a case-study at a specific airport [51, 59], while others quantified savings based on data from several airports [45]. All papers considered fuel factors and emission factors from the ICAO database instead of real fuel use/emission numbers. Only a distinct set of operations are considered (accelerate, decelerate, taxi at constant speed), while not taking into account deviating taxiing procedures or other interactions between taxiing aircraft at an airport.

- A few studies consider different types of taxi-tugs: for NB or WB aircraft [59, 62] or medium/heavy/super heavy towing trucks [61]. Only [62] made a distinction in maximum speeds for towing vehicles: NB trucks taxi at 14 m/s alone, or 12 m/s when towing an aircraft, while WB trucks always taxi at 10 m/s. Khammash et al. found that fuel/emission benefits relate linearly to the amount of NB taxi-tugs, while this effect is less evident for WB aircraft. Guillaume found that for both type of taxi-tugs, the fuel savings reduce per tug added to operations. It is concluded that fuel savings are dependent on both the amount of flights and the airport lay-out, and can thus differ per study. Also, it differs per study if taxi-bots are used for all taxi operations, or taxi-out operations only.

- The majority of the papers do not mention the changes in taxi procedures due to the implementation of the external taxi systems. In [51, 62] the procedural changes compared to the current situation are visualized and the need for engine warm-up before take-off is mentioned. The engine warm-up times vary significantly: 1 minute [51] or 3 minutes [59], while other papers assume 2-5 minutes [48, 62] or 5 minutes [61]. Postorino and Guillaume assume that all taxi operations are carried out perfectly, according to the proposed time-schedule.

- In a single study, a detachment area for the TaxiBot near the runway is missing [51] or it is not argued why a specific location has been chosen [59]. None of the studies analysed the consequences of these changes for a full flight schedule, and the possible congestion on different parts of the airport surface due to changing (engine start-up) procedures.

- Chua et al. [58] are the only ones to study the effect of different taxi procedures on ATC. A real-time simulation has been carried out in which ATCO's were responsible for the route assignment of the taxi-tugs. It was found that the autonomous tugs increase workload and did not significantly improve performance. An ATCo remained responsible for contact with the taxi-tugs, logically increasing workload for increasing number of vehicles. This is in contrast with the concept to create more automated and autonomous movements.
- All studies consider a single, or limited amount of, taxi-path(s) for an aircraft from the gate to the runway or vice-versa. Deviations from these routes are not considered, which in a single study meant that aircraft are held at the gate if their path is not conflict-free [59].
- Only in [61, 62] a case-study has been conducted for AAS. In [62] the parking locations from which towing tugs are operated is located at the root of the B-Pier, similar to the current NB tug parking location of KLM as described in Section 2.1.4. In [61] a parking location is chosen above the E-pier, which currently is not an official parking location for tugs at AAS. Neither of the studies made a distinction in parking locations for different types of taxi-tugs.
- In [61, 62] taxi-tugs are also allowed to use service-roads. This has the consequence that the rise of traffic density, due to the extra movements on the taxiway system, can be altered. However, priority rules for towing tugs, as mentioned in [57], are not included as conflicts are solved from a centralised point of view. More on the modelling part of these papers can be found in Section 5.1.1.
- All of the studies assumed that the speed and accelerations of taxi operations by means of the taxibot are similar to current conventional taxi operations. Only in [61] an extensive analysis has been done on the speed and acceleration profiles when using fully electric towing trucks.
- Both [62] and [61] assumed that taxiing is both possible with and without taxi-tug, while searching for an optimum distribution of operational towing and conventional taxiing. In [62] the amount of taxi-tugs was determined beforehand, while in [61] the model was forced to use all 10 trucks. Conflict resolution is not included for towing trucks, as it is assumed that they can drive close together and can cross eachother.

The second branch of research focuses on the autonomous part of the taxi-tugs. These papers look more into the feasibility of techniques to further automate towing tugs and make these towing trucks self-driving [49, 60] or self-managing [57]. From the papers considering the automation of towing vehicles, the following remarks can be made:

- Chua et al. [57] investigated the best decision cue strategy for prioritising between two autonomous taxi tugs. They found that priority should be given to the tug that is closest to the intersection. If they are equidistant to the intersection, the tug with more trailing aircraft should have priority. If there is still no discrimination, priority should be given to the tug that is closest to its final destination. There is no sign of implementation or actual testing of such reasoning technique.
- Both Morris et al. [49] and Sirigu et al. [60] discretized the tug operation in a set of phases:

  1. Tug sits at a depot.
  2. Tug is provided with the time, route and gate it should travel to.
  3. Tug travels to the gate and follows the assigned route.
  4. Tug detaches and provides a signal to the crew that is successfully detached.
  5. Tug navigates back to the depot.

  In Morris et al. route planning is still performed by an ATCo, but the taxi-tug autonomously maintains separation and corrects its speed. The ATCo is provided with an A-SMGCS-like system that suggests routes for the tugs and tug/aircraft combinations. In Sirigu et al., a tug/aircraft combination is assigned by an ATCo while route scheduling is done autonomously by the tug.
- The consequences of self-driving taxi-tugs on an airports taxi operation has not been considered by both studies. Sirigu et al. consider a single tug mission for either a departing or arriving aircraft, while Morris et al. mention that they have carried out fast time simulations to test the concept. There are however no signs of any of the results from these simulations to verify this.

Similar to the research conducted on internal taxi-systems, no real analysis has been done on the consequences of full implementation of external taxi systems. All studies consider a limited implementation of operational towing movements instead of a full automated external towing operation. A designated area for the tugs is needed, however, the majority of the studies do not explicitly discuss such a location. A concept for the priority cues at intersections is given in [57], but there is no test of such a concept on an actual airport. The performance of these priority cues should be tested in a model.

It has been proven that taxi-tugs can carry out a specific mission to taxi an aircraft autonomously [60], but it is yet to be investigated how external taxi systems would perform in a complex taxiway system while interacting with other vehicles. A major drawback of the models in [61, 62] is that routes are optimised from a central perspective; interactions and uncertainty is not taken into account. The next chapter will focus on the techniques available to model an airport ground surface operation. More on the respective modelling technique from [61, 62] can be found in Section 5.1.1.

5

Relevant Research in Airport Ground Surface Modelling

As described at the end of Chapter 4, suitable modelling techniques for airport ground surface movements will be reviewed in this chapter. This gives the ability to choose the most suitable modelling technique for this specific study. Besides aircraft taxiing, ground surface operations commonly refer to gate scheduling, runway scheduling and similar related operations. As this research mainly focuses on aircraft taxi operations, research is gathered that mainly focuses on this same topic. It is, however, noticed that the topics are heavily related. The most common modelling techniques used in airport taxiing are described in Section 5.1. Thereafter, in Section 5.2, a concise overview will be given of the pros and cons of the aforementioned techniques. The available data that can be used for this study will briefly be discussed in Section 5.3. To conclude, a research gap will be presented in Section 5.4, following from the findings from Chapters 4 and 5.

5.1. Modelling of an Airport Taxiing Operation

It is found that the following modelling techniques are most frequently used in research regarding airport taxiing:

- Linear Programming models;
- Search algorithms (e.g. (meta-) heuristics);
- Shortest path algorithms;
- Modelling based on historical data;
- Agent-Based Modelling;
- Markov Chains.

Each of the aforementioned modelling techniques will be discussed below. A general description will be given of the respective technique together with the current state-of-the-art applications in research. At last, the pros and cons of the technique will be given, as well as some significant results obtained in the considered papers.

5.1.1. Linear programming models

Linear Programming (LP) models generally consist of three parts: an objective function, a set of variables and a set of restrictions. The values of the decision variables have to be chosen in such a way to optimise the objective function (for example a maximisation of the overall performance or a minimisation of costs, depending on the specific problem) while meeting the stated restrictions. These restrictions are called the constraints and pose limitations on the values of the decision variables [63]. In linear programming models, both the objective function and the constraints are a linear combination of the decision variables.

Due to restrictions on the decision variables, variations on LP models have emerged. If decision variables are restricted to integer values (which could be the case when the decision variables represent for example products), the problem is said to be an Integer Programming (IP) problem. It is also possible to restrict some decision variables to discrete values, while others can remain continuous. In this case, the problem constitutes a Mixed Integer Programming (MIP), or in the linear case a Mixed Integer Linear Programming (MILP) model. Whenever the decision variables are restricted to being binary values (0 or 1), a problem is said to be a Binary Integer Programming (BIP) model.

LP models have been widely applied in operations research. One of the major pros of using MILP is the fact that it is an exact method and an optimal solution is guaranteed [64]. On the other hand, the solutions are often restricted to discrete values due to binary or integer decision variables. This causes a significant increase in computational time, possibly making the method impractical for large problems [65]. This has to be taken into account by limiting the complexity of the model to keep computational times within bounds. A solving technique that is often used in practice is the Branch & Bound (B&B) technique. The concept for this technique is to divide the complete set of solutions in smaller and smaller sub-solutions. Thereafter, it is iteratively checked whether the subsets can give an optimal solution by bounding how good the solution of a subset can be, and checking whether it can contain an optimal solution of the entire problem [63].

One of the application areas of MILP models over the past decades is the taxiing operation at airports. This area of research considers the optimisation of taxi routes and the scheduling thereof. It has been found in literature that research is mainly focused on minimising one, or a combination, of the following:

- total aircraft taxi time [62, 64, 66–75];
- number of controller interventions [67];
- taxi or departure delay [62, 64, 67, 69, 71, 74, 76];
- total taxi distance [68];
- longest taxi time among all aircraft [68];
- aircraft emissions [64] or fuel consumption [61].

Some of the aforementioned papers consider the optimisation of an entire schedule at once [66, 67, 70, 73, 76]. As mentioned by Marin [66], the amount of variables and constraints, thus indirectly the problem size, depends on the amount of nodes, links, number of aircraft and time steps. Consequently, the problem grows exponentially when increasing the number of flights or time-steps in a specific model, significantly increasing computational times. Also, due to a single optimisation moment, it is assumed that routes are carried out perfectly and no form of uncertainty is included. Therefore, some papers adopt a rolling horizon approach [64, 68, 69, 71, 74, 75]. This boils down to the construction of several smaller time-frames covering a small portion of the flights instead of optimising an entire schedule at once. Aircraft that are ready to taxi are covered in a single time-frame, while overlapping routes of aircraft from previous time-windows are used as constraints. Several approaches to rolling horizons are available, differing in the way how (time-window) overlapping routes are handled with. Aside from the computational advantage, implementation of a rolling horizon approach has the benefit that feedback can be introduced in the model and uncertainties can be taken into account [64, 69].

When looking at the aforementioned papers, the following remarks can be made:

- Only in [61, 62] the implementation of external towing trucks in an airports taxiing operation is considered. The operational aspects of both studies has been covered in Section 4.3.2.
- Several papers considered an artificial airport lay-out in their routing optimisation [67, 76], while the majority of the papers either considered a simplified lay-out of an actual airport [61, 62, 64, 66, 69, 71, 73, 75] or a specific part of an airport [70, 72]. In [70, 72, 74, 76] an artificial flight schedule has been used, while Lee et al. are the only ones to verify their taxi-route results in a fast time simulation using SIMMOD [74].
- In the majority of the papers, aircraft taxi speeds are assumed to be either a fixed value [67, 73, 76] or dependent on the specific part of the airport [71, 72, 74, 75]. Only in [61, 62, 64, 68–70, 72] aircraft taxi speeds are varying in between a minimum and maximum speed. It is mostly assumed that aircraft are either taxiing with a constant speed on a segment, or standing still. Acceleration and deceleration are only considered in [61], by dividing each segment in an acceleration, constant speed and deceleration part.

- In all papers, aircraft are assumed to be at a specific link or node at a specific point in time. Time is discretized in all papers, usually varying in size between 10 to 30 seconds. This means that during a time-step, aircraft are either travelling a link with a specific taxi speed, or standing still. The time discretization directly affects the time an aircraft is stationary.
- Some studies included the possibility of gate-holding[1], usually limited to a maximum amount of time [68, 72, 75]. While some papers mention it, only in [75] an engine warm-up time of 3 minutes is explicitly used in the model.
- Only a single paper considered multiple runway exits and variations in the runway exit taken by aircraft [75].
- Most papers considered a single route [69, 70, 72, 74] or a limited amount of taxi routes per aircraft or tow-truck [61, 64, 71, 76]. This reduces flexibility, as routes have to be scheduled conflict-free. Only a few papers considered full routing flexibility [62, 66–68, 73, 75], although this comes with high computational cost. Increasing the possible amount of paths significantly increases the number of variables and constraints, as mentioned in [71].
- The majority of the papers considered the optimisation of complete taxi routes at once [66, 67, 70, 71, 75, 76]. They determine the time at which an aircraft needs to be at a specific node to optimise for the global optimisation. Only in [64, 68] the routes are subdivided into pieces, which increases flexibility of the model.
- Although a single study included random push-back times in their simulations [70], most of the studies used a data-set that defined the gate - runway pair for each aircraft. Disturbances like runway configurations are not considered in any of the studies. In [75] it is even mentioned that an hour of operations during the test-case has been excluded, due to adverse weather conditions during this time period. This indicates the difficulty to include any uncertainty in MILP models.

One of the major drawbacks of LP models is the dependence of the amount of variables, constraints and indirectly the computational time on the number of nodes, links, aircraft, time-steps and the paths considered for routing [71]. A rolling horizon approach, as described above, could be applied to reduce computational time while modelling an entire day of operations. It has been shown that such a rolling horizon gives the possibility to take into account uncertainty in flight operations [64].
MILP models optimise from a central point of view, meaning that all vehicle routes are optimised for simultaneously to achieve the best global performance. It is assumed that these routes are carried out accordingly, not taking into account disturbances or interactions along the route. Conflicts are solved by only allowing a single vehicle per edge or node.
On the other hand, it has been found that aircraft taxi speeds are either discretized or flexible within an upper and lower bound. Acceleration and deceleration of aircraft, and the associated times related to these operations, are often not taken into account in MILP formulations. As this MSc study considers the implementation of AGPS, it can be assumed that accelerating and decelerating procedures, as well as interactions between vehicles, are important to consider. It is therefore questionable whether LP models provide sufficient accuracy to model an automated taxi operation for a complex hub airport.

5.1.2. Search algorithms

Aforementioned LP models are an exact method that guarantee optimal solutions. However, it has been noted that more complex problems are not (always) solvable within a limited amount of time. Search algorithms like heuristics and meta-heuristics have been introduced to solve for this. Heuristics are usually problem-specific, while meta-heuristics provide a high-level framework, independent of the problem, to search for near-optimal solutions. Both branches of algorithms are based on the concepts of exploitation[2] and exploration[3] [15]. They are approximate algorithms in the sense that they can provide near-optimal solutions within satisfactory computational times. This has the consequence that optimality is not guaranteed.
In general, (meta-) heuristics can be divided into single-solution methods (searching for an individual solution) and population-based methods. There exists a variety of algorithms in both directions, as depicted

[1]holding aircraft at the gate until their taxi-path and take-off runway is conflict-free instead of incurring delay at the runway
[2]The ability to search around a promising solution to reach the optimal solution
[3]The ability of terminating searching only in local areas, to escape from local optimal solutions

in Figure 5.1. As it will be too extensive to explain all algorithms individually, this section focuses on some commonly used search algorithms in airport taxiing problems.

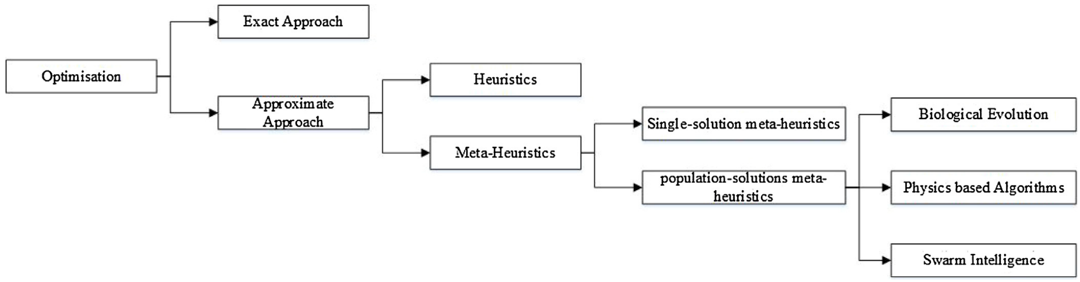


Figure 5.1: Classification of search algorithms [15]

One of the most commonly used population-based methods for optimisation of aircraft taxiing found in literature is the Genetic Algorithm (GA). GA is based on the idea of cumulative selection: it resembles biological natural selection and is based on the genetic operators of selection, crossover and mutation. These three definitions ensure new solutions from a population by exploration and exploitation of the search space: crossover produces new solutions from parent individuals in the population of solutions [77]. A fitness function is usually introduced to indicate the goodness of solutions and thus the direction of search.
Two papers considered applications of swarm intelligence algorithms: a particle swarm optimisation [78] and an ant colony optimisation [79]. Particle swarm optimisation shows similarity with genetic algorithms, differing in the sense that each potential solution (particle) is assigned a random velocity and particles are 'flown' through hyperspace [80]. The coordinates of each particle are tracked and are associated to the best solution. Changing the velocity and acceleration of the particles ensures exploration and exploitation of the solution space towards local and global optima. The ant colony algorithm can be compared to Dijkstra's algorithm (5.1.3), although here artificial ants traverse a map to search for the shortest path. Ants leave chemical pheromone on their path: more pheromone indicates moving towards a good route that other ants will follow, while less pheromone indicates moving towards a worse solution [81].

As mentioned, there is a large variety of algorithms used under the branch of search algorithms. A selection has been made of research using search algorithms, from which the following remarks can be made:

- Almost all papers considered both arrivals and departures. Only in [77] just departures are considered.
- The majority of the papers optimised for (a combination of) minimisation of taxi times [77–79, 82–84] and minimisation of runway queue delay or taxi waiting times [77, 82]. Weiszer et al. [83] are the only ones to consider a simultaneous minimisation of taxi time and aircraft taxi fuel burn, while Gerdes and Temme [85] included, besides the minimisation of taxi-time, a minimisation of the number of speed changes, holdings and distance to target time.
- The majority of the papers excluded any form of uncertainty in their optimisations. The obtained results provide a routing for every aircraft, which is assumed to be carried out accordingly. Aircraft interaction is not included and sudden changes in the taxiway network are by no means taken into account. Only in [82] uncertainty is included in the form of a speed uncertainty of 10% of the nominal taxi speed.
- Aircraft separation is in all cases taken into account beforehand, by timing and scheduling of the individual taxi routes from a central point of view. The proposed separation requirements differ per model: distance-based separation or time-based separation. Separation distances differ between 50 m [79, 84], 60 m [82, 86] to up to 200 m [77], while time-based separation differs between 12 sec [83] and 43 seconds [78]. Although it is noted that [85] uses a time-based separation in their model, the exact value is not clear from the paper. In [77, 82, 85] waiting nodes are introduced, either at the gate or on the taxiway, to resolve conflicts by waiting at a specific point. Gotteland et al. [82] used 30 seconds waiting time, even for arriving aircraft.
- Some papers consider a single taxi-route per aircraft [83], while others only use a limited amount of pre-determined routes for scheduling [84, 86].

Several studies highlighted the benefits of search algorithms over shortest path algorithms like Dijkstra. It is, however, noted that some studies applied their research to a relatively small and simple airport lay-out with, for example, a single runway [78, 79, 84], or use a limited amount of flights in their test schedule which barely resembles reality [77]. Verification of the obtained results with real taxiing operations is often neglected.
Two major drawbacks of search algorithms are the fact that uncertainty is often excluded and aircraft routing is optimised from a central point of view. It can be expected from taxiing operations at busy hub airports that circumstances are likely to change. Therefore, the possibility to include some sort of uncertainty and robustness will be necessary in order to accurately model an airports taxiing operation, even in case of runway configurations or taxiway closures. Another drawback is the fact that aircraft interaction is not taken into account, as is the case for the aforementioned LP models. This makes it hard to model interactions between different parties in an airport ground operation, making this modeling technique less suitable for applications of autonomous taxi operations.

5.1.3. Shortest path algorithms

Another commonly used method for optimisation of aircraft taxi routes are by means of shortest path algorithms. Shortest path algorithms can be applied to networks consisting of nodes and arcs (arcs being either bi-directional or unidirectional) with an associated distance or cost. The shortest path algorithm searches for the shortest path from an origin node to a destination node, by minimising the total distance or costs of the considered arcs [63].
Two of the most commonly used shortest path algorithms in literature are the Dijkstra and A* algorithms. Dijkstra's algorithm considers the distance of all nodes to the starting node: the distance of the starting node is labelled zero and all other nodes are labelled as infinite. Now, for each directly linked node the distance to this node and the sum of the starting node label (zero) are compared to the previously stated bounds (infinity). The smallest of the two is the new label for this node combination. After all nodes have been evaluated, the next iteration starts by considering all direct links to the node with the smallest distance to the initial node. If smaller distances are found than currently labelled, these are updated. This process is repeated until all nodes are covered [87].
Although the A* algorithm is labelled as a shortest path algorithms, it uses a heuristic to search in the direction of the destination node. It uses a valuation function which is to be minimised, usually consisting of the distance from the starting node (as in Dijkstra) and an initial guess of the distance to the destination. A* has the advantage that the number of nodes to calculate can be reduced and search speed can be increased [88, 89].

Aircraft taxi trajectories are routed one after another when optimising with shortest path algorithms. Although aircraft routes can be determined relatively quick, aircraft interaction is hardly taken into account. It is found in literature that some papers consider the principle of gate-holding, as discussed in Section 5.1.1, to ensure conflict-free routing of all aircraft [90, 91]. It is thus assumed that aircraft consequently carry out their assigned routing perfectly and no uncertain actions occur during taxiing.

The following remarks can be made regarding literature on shortest path algorithms [88–92]:

- Aircraft speeds are either adopted as a constant dependent on whether taxiing on a straight segment or in a turn [88–90] or on a specific taxiway segment [92]. This leads to discontinuous speed profiles, as showed in [89]. In [91] a maximum taxi speed has been proposed for straight segments and corners.
- Only in [90] different transitional phases are included in the speed profiles: acceleration, constant speed, deceleration and fast deceleration. However, the effects of these transitional phases on surrounding traffic is not considered.
- Some studies only considered a single flight [88] or a limited amount of flights [91]. Although it has been proven that the shortest path algorithms provide optimised individual taxi routes within limited computational time, such small flight schedules do not give an accurate representation for a busy and complex hub airport like AAS. Only in [89, 90, 92] the simulation results have been compared to real-world data, in order to verify the numbers.
- In [92] some sort of uncertainty is included. Flights are assigned a route with time-windows per node, allowing small taxi speed deviations and delays. More significant effects like runway rescheduling or adverse weather conditions are not mentioned.

- In all papers, conflicts are resolved beforehand in the planning process. Aircraft routes are not allowed to use the same edge at the same time. It is assumed that all aircraft carry out their assigned route perfectly and no unforeseen problems occur in the taxiway system. In [91] arrivals get priority over departing aircraft, whereas in [88] conflicts are not taken into account.
- In neither of the studies, a global optimum is found as aircraft routes are considered one after another instead of all at the same time [89].

Aforementioned fact that aircraft interaction is not taken into account by means of shortest-path algorithms is a major drawback of this modelling technique. Also, the fact that a global optimum is not ensured and aircraft speeds are discretized makes this type of modelling technique of less interest. A lot of papers use the principle of gate-holding to obtain an optimised conflict-free route for each aircraft [90–92]. It can be questioned whether this is feasible, as gate scheduling is quite complex and extra delay at the gate could incur additional taxi delay for incoming traffic elsewhere.

5.1.4. Analyses based on historical data

Historical data is often used in research to provide some kind of aircraft taxi time prediction, in order to increase airport efficiency or take-off time accuracy. It is found that there are roughly three (branches of) techniques applied in research that are based on historical data. A first branch of techniques commonly considered are (linear) regression models or statistics [93–96]. The papers considered used multiple linear regression [94, 95] or even up to quadratic relations in their regression functions [93]. A second branch of techniques used in research is machine learning [96, 97]. It was found that one of the papers considered a specific machine learning technique in the form of Support Vector Regression [97], while others focused on a comparison of different techniques [96]. The machine learning techniques considered by Lee et al. are: support vector machines, k-nearest neighbors, random forest and artificial neural network. As can be noticed already, machine learning covers a variety of techniques, each with its own benefits and drawbacks. An elaboration of all different techniques will not be carried out in this study, as this would be too extensive for this literature study. A third technique applied in research is by means of queuing models [98][4].

Considering the aforementioned papers, the following remarks can be made:

- Only in [93, 94] both arrivals and departures are considered in the taxi-time estimation. The majority of the papers [95–98] only considered taxi-out time estimation, i.e. departures.
- In case of, for example, Machine Learning techniques, the prediction accuracy is based on the variety and accuracy in the historical data used for training. Therefore, it is never completely obvious what is included in the taxi time prediction, as mentioned in [94]. This has the consequence that real-world measures like gate-holding, intersection take-offs or taxi detours are often excluded, as mentioned in [95, 96].
- The amount of operational data used in the papers differed significantly, being either a single/few day(s) [94], a week [97], a month [95, 98] or a year [96] of operational data. Only a single paper compared the obtained results with empirical data [94]
- Several papers investigated the influence of different variables on the taxi-time. In two papers, it is stated that the queue position / runway queue is the most determinant factor in accurate predictions of taxi-times [95, 98].

One of the major drawbacks of modelling techniques based on historical data is the fact that they are often limited to steady state operations. Circumstances like changing RMO, aircraft interactions on the taxiways or other unusual circumstances are often excluded in analyses [96]. Especially the lack of aircraft interactions is limiting for real-world applications at, for example, AAS, due to its complex taxiway system. Even in case of steady-state situations, taxi time prediction accuracy is limited to around 80-90%.
On the other hand, as mentioned in several papers [93, 96, 97], erroneous data needs to be removed or certain data is missing in a specific data-set. Filtering is needed before accurate analyses can be carried out. As mentioned, the dependency of accurate predictions lies solely with the training set used and therefore poses significant limitations to these types of modelling techniques.

[4]Arranging the service, for a set of customers (or aircraft) that are waiting in a queue, in the most effective way [63]

5.1.5. Agent-Based Modelling

Agent-Based Modelling (ABM) is a relatively new modeling and simulation technique in comparison to, for example, the aforementioned LP models and search algorithms. Agent-based systems are based on the concepts of (autonomous) agents and a multi-agent system and consists of a bottom-up approach of modelling: a model consists of interacting agents within a pre-defined (simulated) environment. Each agent possesses specific characteristics and a corresponding behaviour and can interact with other agents and the environment. Agents may refer to, for example, aircraft, cars, sensors, people etc, while the environment can be modelled with certain characteristics and specifications. Therefore, different types of agents with different behaviour can be implemented in a single model and interaction between agents is included. The pre-defined characteristics of agents may lead to behaviour in the simulation that could not have been foreseen, called *emergence*.

ABM provides a means for solving problems from a local perspective, instead of complex central solutions for which it is not feasible to include all necessary details and constraints. The model accuracy is, however, dependent on the level of detail in the agent specification. A benefit of ABM is the high flexibility of agents to cope with changes in the structure of the system. Agents may be able to adapt their behaviour to a changing environment by controlling their local relations and behaviour, which is highly relevant in transportation related problems. However, the development of a complete representation of agent characteristics and attributes may require significant time, consequently leading to large development times. Also, computational times may increase with increasing complexity of the model. Therefore, some level of abstraction may be necessary, depending on the goal of the model [99, 100].

When looking at applications of ABM in airport taxiing, it is found that the largest differences are found in the type and amount of agents modelled, as well as the complexity of the operation considered. It was found that the majority of the papers implemented an agent type for each specific part and actor on the airport [8, 23, 101–103]: aircraft, runway, intersection(s), apron and taxiway. Zhu et al. are the only ones to consider both airport operations and traffic in the TMA in their model, and modelled three different types of ATC agents: terminal airspace agent, tower controller agent and ground controller agent. Only in [8, 23, 102] the principle of distributed control was implemented: ATC is not modelled as a central controller. Conflict resolution is done locally by means of intersection agents that prioritise incoming aircraft.

Fines [23] investigated the use of cooperative coordination mechanisms for intersection agents to increase resilience during runway reconfigurations, and analysed the effect on taxi time and taxi distances. An extensive review has been done on applicable multi-agent path finding techniques, as well as coordination techniques before/ during and after route planning. A trade-off, based on several criteria of implementation, performance and connection ability of the algorithms considered, indicated the Conflict-Based Search (CBS) algorithm (with highways) to be most suitable. The trade-off and results of this paper could be useful for considering a path planning algorithm for autonomous towing trucks and could provide a good reference for this study in the future.

All ABM applications use a shortest-path algorithm (See Section 5.1.3) to determine initial aircraft routes from gate to the runway and vice-versa. It is noted that half of the papers considered are master theses [8, 23, 102], emphasising the rise of ABM applications in the airport taxiing domain.

Regarding the aforementioned papers, the following remarks can be made:

- Half of the papers modelled a simplified artificial airport [101, 102, 104], while the majority of the papers used an artificial flight schedule to test their model [101–104]. Only in [8, 23] a real flight schedule has been used, obtained from extensive Automatic Dependent Surveillance - Broadcast (ADS-B) data filtering.
- All considered papers modelled both departures and arrivals. Only [102] considered an arrival / departure ratio of 2:1, while the rest of the papers assumed a 1:1 ratio.
- In [8, 23, 101] interaction is limited to the surroundings of an agent. In [101], aircraft agents can retrieve information from agents up to 2 intersections ahead, while in [8, 23] only ATC intersection agents can retrieve information from up to 2 ATC intersection agents up- / down stream.
- Rafegas made a distinction between traditional airlines and low cost carrier airlines, where aircraft from traditional airlines get priority in the taxiway system. The other papers did not make a distinction and used a single type of aircraft agent in their model.

- For some papers, it is not clear how conflicts are solved [103, 104]. In [101] conflict resolution is done on an edge-basis: reduced distances due to short edge-length or corners are not taken into account. Rafegas solves for conflicts on a first-come first-serve basis.

Although ABM has not been applied a lot in airport taxiing problems, compared to, for example, LP models, it has proven to be a very useful modelling technique. This can be appointed to the possibility to model interactions between agents, as well as its dynamic ability to easily expand to larger systems as shown in [8, 23]. The fact that an agent's characteristics and specifications can be fully modelled makes ABM useful to implement autonomous behaviour of towing trucks. Accurate modelling of a large-size hub airport was proven to be possible [8, 23]. Coordination techniques for agent path planning have proven to be valuable and increase the systems resilience [23]. These benefits make ABM an interesting technique to consider.

5.1.6. Markov Chains

Markov chains are a specific type of stochastic process. In Markov chains, the evolving of processes only depends on the current state and is independent of events from the past [63]. This lack of memory property is defined as the *Markovian property*; a stochastic process is called a Markov chain if it has this specific 'lack-of-memory' property. A distinction can be made between continuous- and discrete time Markov Chains, dependent on the time-points it is observed to determine its state: A discrete-time Markov chain can, for example, be observed only at a specific point in time at the end of each day. It is possible to make decisions about which actions should be taken in a Markov chain. These actions affect the transition probabilities (probability of transition from the current state to a next state) and the costs/rewards associated to these actions. Choosing actions to be as optimal as possible is referred to as a Markov Decision Process [63].
It was found in literature that MDPs are often used in combination with Reinforcement Learning (RL) [105–108], due to the large number of different state/action combinations. The RL algorithm uses the system state as input and outputs a reward function, which in literature often is the difference between actual taxi-out time and predicted taxi-out time. The objective is to minimise this reward function, often referred to as the prediction error. Morris et al. [109] considered Markov-chains in combination with a multi-agent path finding model for both aircraft and autonomous taxi-tugs in their long-term research, as have been touched-upon already in Chapter 4 [49]. The goal of this paper is similar to the other papers: minimising taxi delays and avoiding congestion. A last study approached the airport ground movement problem from a different perspective by controlling the push-back rate at the gate at Boston airport to minimise taxi-out times [110].
Regarding the aforementioned papers, the following remarks can be noted:

- The majority of research focuses on the determination, or prediction, of taxi-out times [105–108, 110]. Morris et al. [49] are the only ones to consider all ground operations at the airport, including (future) autonomous taxi tug movements.

- Balakrishna et al. considered applications of RL and Markov Chains to predict taxi-out times at New York [105], Tampa [107], Washington and Detroit [106]. It was concluded that high prediction accuracy of up to 95%-100% of taxi-out times have been obtained for Tampa, Detroit and Washington. Prediction accuracy for New York was significantly lower (60 %), due to the high taxi-out times and high variance of taxi times at JFK airport [105].

- All of the papers used significant amounts of data to train the RL model, ranging from a full month of operational data [108], a few months of data [106, 107] and even over half a year of data [105, 110]. In [49] it is not clear what operational data has been used for the analysis.

- All studies considered a case-study at a specific airport. However, only in [110] the theoretic concept has been tested in practice. Besides, the concept has been compared to regular operations to verify the hypothesised benefits.

As mentioned before, the application of Markov chains with RL does not provide sufficiently accurate results in case of high (variations in) taxi-times. As taxi-times at AAS can vary significantly depending on the RMO, it can be assumed that Markov chains do not prove to be a suitable modelling technique for ground operations at AAS. Another major drawback of Markov chains is the lack-of-memory property, as in current ATC procedures the history of an aircraft's taxi trajectory is usually taken into account in routing. To conclude, the large amount of operational data necessary to provide accurate predictions with this technique pose limitations in its applicability.

5.2. Review of modelling techniques

As described above in Section 5.1, there is a wide variety of techniques available to model an airports ground surface operation. Each of these techniques has its benefits and drawbacks and are more or less frequently applied in the domain of airport operations. An overview of the six most frequently used modelling techniques in the airport taxiing domain can be found in Table 5.1. The table provides a high-level overview of the characteristics of each modelling technique, as have been described extensively in Section 5.1. The characteristics are indicated as a pro (+), a con (-) or just a modelling characteristic (•). Further elaboration is provided underneath the overview.

Modelling technique	*Characteristics*
Linear programming models	• Global optimisation, using an objective function, variables and constraints. + An exact method that guarantees an optimal solution. - Computational times increase significantly when increasing problem size. - Modelling accuracy is limited: uncertainty and aircraft interaction are often excluded. - Optimised routing schedule is assumed to be carried out perfectly over the considered time-domain.
Search algorithms	• Explore and exploit the solution space by means of heuristics. • Obtains near-optimal solutions within satisfactory computational times: optimality is not guaranteed. - Uncertainty is often excluded and it is assumed that the optimised plans are carried out perfectly. - Conflicts are solved during optimisation before the considered operations commence.
Shortest path algorithms	• Path by path, local, optimisation regarding costs or distance to travel from node to node in a network. + Low computational times due to sequential optimisation. - From a global perspective, solutions are often sub-optimal as a single path is considered at a time. - Conflicts are solved beforehand; aircraft interaction is hard to take into account.
Historical data	• Usage of historical data to predict some sort of process, for which several algorithms are available. - The model must be trained by means of a training data-set. Prediction accuracy is thus dependent on the accuracy of the training data-set. - A significant amount of training data is necessary to train the model. - Accurate in steady-state operations, with small variations in the data. Less accurate in transient operations.
Agent-Based Modelling	• Based on the concept of (autonomous) agents within a simulated environment. + Ability to model both agent characteristics and interactions between agents and the environment: a bottom-up approach. + Problem-solving from a local (agent) perspective, instead of complex central solutions. • Good understanding of the modelled operation is necessary to be able to create and interpret the model correctly. - Development of agent characteristics and specifications can take significant time. + High flexibility of the model to changes in system structure, due to the bottom-up approach.
Markov-Chains	• Stochastic process in which the transition from the current state to a next state is indicated by a probability. - Transition is only dependent on the current state and independent of events from the past. • Often combined with a reinforcement learning algorithm, to handle high amounts of state/action combinations. - As mentioned before, large amounts of data are necessary to accurately train a RL algorithm.

Table 5.1: Overview of modelling techniques and their characteristics

When looking at the modelling techniques as described above it can be concluded that some modelling techniques lack the ability to include essential aspects of an autonomous operation, as will be considered in this study. Global optimisation methods like LP and search algorithms provide a (near-) optimal solution, indicating for each aircraft a specific point in time and space to ensure conflict-free, optimal taxi-routes. It is assumed that aircraft carry out their route perfectly, having the drawback that both interaction and uncertainty are often excluded. Besides, computational times pose a limitation to the overall complexity of the model. Shortest-path algorithms provide a quick means to calculate optimal routes for individual aircraft, one after another. This consequently means that solutions are sub-optimal from a global perspective. Similar to applications of LP and search algorithms, shortest path algorithms do not, or very limited, take into account aircraft interaction. Therefore, these techniques provide limited options to model autonomous behaviour. Prediction techniques that either use historical data or are combined with Markov-chains, need significant amounts of data to train the model. Also, they are sensitive to large variations in the data, causing bad predictions in case of transient operations. Both are drawbacks, as it can be expected that transient states of operations will play a significant role in ground operations of a hub airport. It can be concluded that these techniques are not suitable for this study, also because this study will cover a futuristic concept for which no historical data is available yet.

Agent-based models provide a means to explicitly include agent interaction and behaviour. Although developing each individual agent and a corresponding environment sometimes requires significant development time, the models often resemble reality more accurate than the aforementioned techniques. They are flexible to changing operational circumstances and are modular, meaning that models can be expanded easily. They also provide a means to distribute control to a local level, instead of controlling all traffic from a central perspective. This gives the ability to further investigate the position of ATC, which changes from providing active guidance to a more supervisory role. Regarding the implementation of additional towing trucks on the airport surface, it is relevant to model acceleration and deceleration profiles for these vehicles instead of assuming constant speeds. ABM provides a means to model these speed profiles.

ABM has not been applied a lot yet in the airport taxiing domain and, therefore, this research can further expand on the knowledge on ABM of airport operations, as well as explore the possibilities of agent technology in aviation. Next section will elaborate more on the gap which this research aims to fill.

5.3. Availability of relevant data

In order to be able to model airport surface movements, relevant operational data will be necessary. Therefore, this section gives a brief description of the operational data available, as well as relevant knowledge and expertise within To70 that could be used for this research.

Operational data available for reference/use in this research:

- ADS-B data is available, as obtained by To70 and stored in a database. A tool has been made in a previous research [8], in which a data-set of 2 weeks of ground operations has been created for analysis by filtering and cleaning the raw ADS-B data. Although some entries had to be removed to create this set, a useful and representative data-set has been created that could be used for modelling. This data-set contains per flight: aircraft identification, origin, destination, start-time and a label indicating whether it is an arrival or departure flight. The type of aircraft is not provided in the filtered data-set, but is available in the raw ADS-B data and could thus be used.
- A-CDM data from AAS is available, from which accurate times (as described in Section 2.1.3) can be retrieved. These times could be useful both for taxiing operations, as well as scheduling times for the taxi-tugs. As the taxi-tugs are mainly dependent on the indicated TSAT, as mentioned in Section 2.1.4, relevant and accurate A-CDM data could be of importance for this research.
- As mentioned in Section 2.1, LVNL has a publicly accessible website from which the current RMO can be obtained. The current RMO is updated on this website every 5 minutes and can be used as a reference to compare the obtained results with.

As mentioned in [8] (ADS-B) data can be less accurate, or missing. Therefore, experiences from personnel could be useful for clarification of certain operations. At To70, both pilots and (ex-) air traffic controllers can be consulted. These persons can help in case of confusing data, or in case of verification of obtained results. Especially their procedural expertise can help in achieving a more realistic model.

5.4. Conclusions and Research Gap

Even though a distinction has been made in internal and external taxi systems in Chapter 4, it has been found that there are both similarities and differences in research. Research on internal systems is on average less practical oriented than for external systems. Although practical applications have been shown in the form of the WheelTug, internal systems are still in the development phase and do not provide as capable taxi capabilities as external systems. Another drawback of internal systems is the expected new certifications of aircraft due to the new system. As mentioned before, internal taxi-systems are mainly beneficial to NB aircraft due to their high taxi-time to flight-time ratio. As AAS services around 100.000 WB movements per year, it seems more obvious to consider the implementation of external taxi systems, like the TaxiBot, for the remainder of this study.

Regarding external taxi systems, some research has been done on concepts like self-driving, self-managing conflict resolution and automation. Although for both internal and external systems it has been proven that significant fuel savings can be achieved, considerations of operational implementation at hub airports are limited. Case-studies focus mainly on a single taxi operation and/or a small-sized airport, or optimise for the assignment of towing tugs to minimise fuel costs/taxi time. These studies do not take into account the interactions between aircraft on a complex high-density taxiway system and possible drawbacks of procedural changes due to new towing truck operations. In all applications, personnel is needed to either drive the towing truck back to its parking position, or remote parking spots are needed to park the vehicle after operation. Although the feasibility of autonomous operational towing has been proven for a single movement, an analysis of a full implementation on a complex hub airport is yet to be done. This MSc thesis aims to fulfil a role in this part by investigating the implementation of autonomous external taxi systems in the ground surface operations of AAS. It aims to expand to the knowledge regarding the feasibility of operational towing by means of external taxi systems, as well as to show the effects of implementation on current ground operations.

[61] and [62] are the only papers found that modelled the implementation of external towing trucks for an entire airport ground operation. They both considered a MILP application for route scheduling of both conventional and operational towing operations at AAS. However, as discussed in above sections, vehicle interactions are often neglected in MILP formulations, despite being of significant importance when considering autonomous behaviour. There are numerous other modelling techniques available to model an airport ground surface operation that can be used for this study, although these techniques have not been applied (a lot) yet to this specific domain. As mentioned in Section 5.2, agent-based modelling is an interesting paradigm to explicitly include vehicle interactions and behaviour. Although development times can be significant, model accuracy and flexibility can be higher compared to the other modelling techniques. Therefore, agent-based modelling will be used in this study for implementation of autonomous external taxi systems. As described in Section 5.3, several data sources are available for this study. These data-sources provide enough means to consider an accurate model of operations at AAS.
This research aims to expand to the knowledge regarding agent-based modelling applications in the airport ground surface operations domain. Although the present study will focus on AAS, other airports could use this knowledge to consider future studies in this domain.

6
Research proposal

The Chapters described before highlighted the current state in research regarding both engine-off taxiing and airport ground surface modelling. This Chapter will provide a description of the problem statement for this study, leading to a formal statement of the research objective. Thereafter, the main research question and corresponding sub-questions are formalised.

6.1. Problem statement

The aviation industry keeps growing annually regarding number of flights and passenger numbers. This consequently gives rise to the call to lower fuel burn and associated emissions, as these have a global impact. Engineering companies are further developing aircraft and their engines to be as fuel-efficient as possible during the cruising phase, trying to mitigate the environmental impact of aviation. However, the increasing number of flights also has consequences near airports. Airport infrastructures are utilised up to, or even over, their maximum capacity. Expansion of airport resources is often difficult as hub-airports are located in populated areas, leading to congestion on the airport surface and increased fuel burn during taxiing. The corresponding extra pollution and emissions affect the living quality of people near the airport and give rise to complaints. It forces airports to optimise their utilisation of current resources. The operational goals of an airport are often focused on accommodating more flights, while these developments forces airports to consider their environmental footprint.

In order to optimise airport taxiing operations, lower the associated emissions and fuel burn and further automate airport ground operations, the concept of Follow-the-Greens has been developed. It allows aircraft to more accurately follow their assigned taxi-path, while the role of ATC changes more to a supervisory role. This way, the number of aircraft that can be guided along the airport surface is not directly dependent on ATC workload. Although this type of distributed control lowers airport taxiing emissions to some extent, it does not solve the problem entirely as aircraft continue taxiing on their main engines. Therefore, current research focuses on the concept of aircraft engine-off taxiing as described in Chapter 4. Internal taxi systems, like the eTaxi, are attached to the aircraft structure. It has been proven in research that these types of systems are especially beneficial to aircraft with a high taxi-time to flight time ratio, like the A320 or B737 family. External taxi-systems like the TaxiBot could be used more flexibly and implementation at airports can be decided for seperately. These taxi systems could significantly lower aircraft emissions, especially if the systems were to be operated on electrical power. A major consequence of the implementation of external systems would be the increase in number of movements on the airport surface. As these movements currently have to be controlled by ATC this would possibly lead to more congestion, resulting in a disappearance of the obtained benefits.

Similar to the concept of Follow-the-Greens, external engine-off taxi systems could be considered by means of distributed control. This would mean that route guidance and conflict resolution is solved locally, instead of via radio communication with an ATCo. In this case, an increase in airport ground traffic can be covered from an operational point of view, while simultaneously lowering aircraft taxi emissions. This leads to the research objective for this study:

> *"To design and evaluate a novel concept of autonomous towing of outbound traffic at Amsterdam Schiphol Airport based on distributed coordination and planning."*

As have been presented in Section 5.4, current research is limited to a few studies that considered implementation of external towing systems in an airport ground surface operation. Although the concept of engine-off taxiing has been studied for a while, research into actual implementation is currently scarce. This study will explore and investigate the feasibility of external towing systems in a hub airport's ground surface operation. This will be a unique study in the sense that it will combine the concepts of engine-off taxiing and follow-the-greens. Previous studies excluded any form of interaction between agents while optimising ground operations from a central point of view. This study aims to fill a gap in engine-off taxiing, by proving its autonomous usability in a distributed control environment at a complex hub-airport and explicitly model the local interactions between the agents involved. The operational effects of implementing autonomous engine-off taxiing will be assessed by comparing the system parameters with current ground surface performance.

6.2. Research questions

In order to be able to fulfil the research objective as stated above, a set of research questions has been formed. These research questions have been divided into a single main research question and supporting sub-questions. The main research question for this study is:

> *"How does the implementation of a novel operational towing concept with autonomous Taxibots, by means of distributed coordination and planning, affect the ground surface operations as currently carried out at AAS?*

A set of supporting sub-questions has been formulated to be able to answer the aforementioned main research question. These sub-questions are:

1. How are the ground surface operations currently carried out at AAS?
    (a) Which parties are involved in an airport ground operations and how do they interact?
    (b) What procedures are in place to safely carry out airport ground operations, both for ATC and for other relevant parties involved?
    (c) What are the (operational) goals of the parties involved in airport ground operations?
2. What is the current state-of-the-art in aircraft engine-off taxiing?
    (a) What aircraft engine-off taxiing systems are available?
    (b) What are their specific characteristics and operational capabilities?
    (c) Which system is most suitable to consider for modelling?
3. What modelling technique (as described in Chapter 5) is most suitable for modelling autonomous engine-off taxiing in a hub airport ground surface operation?
    (a) What modelling techniques are available for airport ground surface modelling?
    (b) What are the characteristics of these modelling techniques?
4. To what extent can the concept of autonomous engine-off taxiing be implemented in a model of a hub airport ground surface operation?
    (a) Which agents, interactions and procedures need to be included in the model?
    (b) Which interactions or procedures need to be simplified in, or excluded from, the model?
    (c) What are the specifications and simplifications of the model environment?
    (d) How will multi-agent path planning and coordination be implemented and carried out for the towing trucks (both short and long-term)?
5. Which input and output variables will be used in the aforementioned model?
    (a) Which input data is available?
    (b) Which input parameters will be used for modelling, i.e. model parameters?
    (c) Which KPIs will be used to analyse the output of the model?
6. What are the operational effects of the implementation of a full autonomous operational towing concept into a hub airport ground surface operation?

(a) How does the outcome compare to current ground surface operations regarding the KPIs?
(b) What are non-measurable effects of the autonomous operational towing concept?

7. What conclusions can be drawn from the obtained results from the model of the autonomous engine-off taxiing concept?
8. What recommendations can be made regarding the feasibility of an autonomous operational towing concept in a hub airports ground surface operation?

III

Supporting appendices

A

Model Elaboration and Results

A.1. Model Elaboration

A.1.1. Methodological Steps

Figure A.1 provides a graphical overview of the methodological steps taken in this research to meet the research objective. Phase 1 resembles the literature study, which can be found in part II of this report. The results of the literature study have been combined with operational knowledge of AAS ground surface operations to formalise a novel concept of taxi operations. This concept has been used to create an Agent-Based Model representation and both are discussed in the MSc Paper in part I. Phase 3 resembles the translation of the conceptual- and Agent-Based Model into a working computer model. The computer model has been iteratively created, starting with a relatively simple airport layout to verify and validate correct agent behaviour in the creation phase. Iteratively, the ground surface movements were simulated in the model and a set of KPIs was analysed to assess model performance. After confirmation of its functioning, the model has been expanded to resemble the complete AAS infrastructural layout and consequently verified and validated again. Phase 4 indicates the last phase, during which several days of operational data has been used as input to the model and the model results have been analysed. A continuous check with the formalised research objective was carried out to check if the model met the stated objectives.

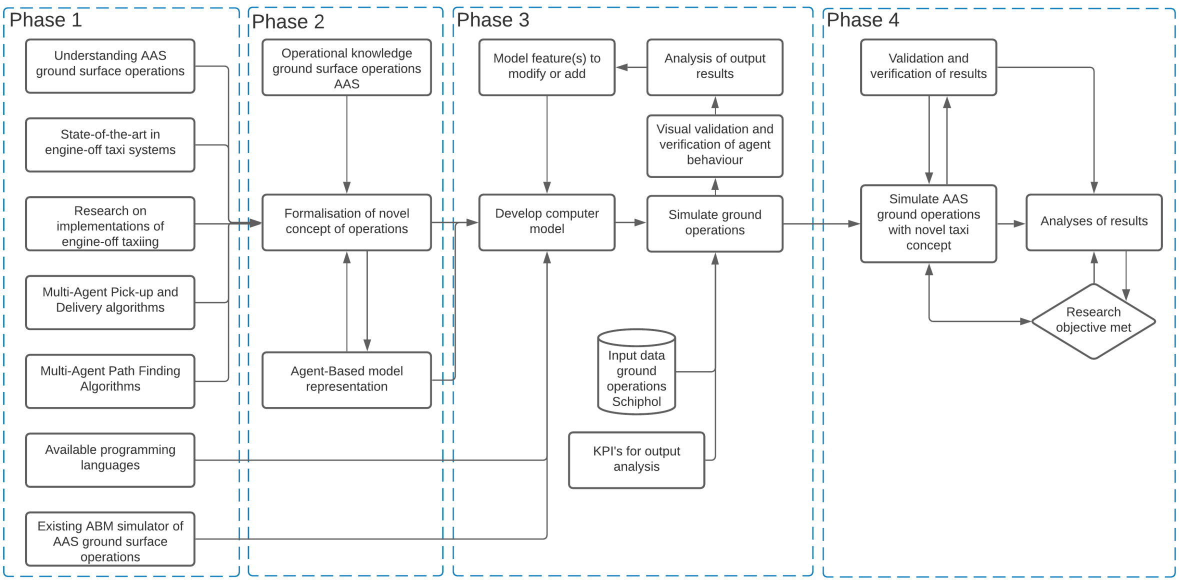


Figure A.1: Methodological steps Model creation and analysis

A.1.2. Agent-Based Model environment

The Agent-Based Model environment as depicted in the paper originates from Figure A.2. The Figure indicates the node locations as created in QGIS, a geographic information system program. The red dots indicate locations of either ATC agents (when located on the taxiway intersections, runway or gates) or intersection nodes on the service-roads. The service-roads, taxiways and runways can be clearly distinguished in the Figure. The two yellow dots in the middle of the figure indicate the parking locations for Taxibots, currently used as parking locations for push-back Taxibots. The node locations have been implemented and exported from QGIS, to allow for inclusion in the Python model.

Figure A.2: Visualisation of node locations on map

The most important model assumptions have been stated already in the MSc Thesis Paper. The following lists contains an elaboration on these assumptions, as well as some of the less striking modelling assumptions made:

- Aircraft movements are only modelled from their respective apron to the runway and vice-versa. Arrival aircraft are assumed to leave the runways at either maximum speed, via rapid-exit taxiways, or maximum turn-speed via regular runway exits. Aircraft depart the gate from a standstill when a Taxibot is attached.
- Three different separation values are assumed, dependent on the type of vehicle and the location on the airport's surface. The airport has been divided into landside and airside, i.e. service-roads and taxiways. On airside, individual Taxibots need to maintain a separation distance of 50 meters with other Taxibots or aircraft. The minimum separation between aircraft is set to 150 meters. A separation distance of 35 meters is assumed for Taxibots on the airport's landside [111]. The border between landside and airside is set at the aircraft gates, meaning that the service roads up to a gate are still part of the airport's landside, while all taxiways after the gate are part of airside and thus covered by ATC.
- Only Taxibots and aircraft ground surface movements are modelled. This means that other ground support vehicles are not taken into account. Also, only operational tow movements are modelled. This means that maintenance towing, aircraft towing to remote stands and similar movements are not included in the simulations.
- A flight schedule has been retrieved from real-world ADS-B data of ground tracks at AAS. This flight schedule consists of flight id's, flight type, origin, destination and start-time.
- A single aircraft type is modelled, namely the Airbus A320. This aircraft type has been chosen due to it being a narrow-body, which have significant taxiing times compared to their operating times [51]. For the Taxibot, the real-world TaxiBot has been used as a reference.

- It is assumed that both aircraft and Taxibots have a maximum acceleration and deceleration level, as well as a maximum speed on straight segments and turns. It is assumed that both vehicles, and also when travelling together, have a maximum level of acceleration and deceleration.
- It is assumed that Taxibots carry out their assigned tasks perfectly. Also, the flight crew carries out their instructions from ATC perfectly without any errors.
- Both Taxibots and aircraft try to travel at maximum speed over the airport's surface unless instructed otherwise by ATC.
- Runway occupancy times, as well as gate occupancy times for departures and arrivals, are assumed such that aircraft cannot take-off or arrive at the same gate simultaneously. This means that a departure or arrival aircraft toggles a runway occupancy window during which no other aircraft can use the runway.
- Service roads are always two-lane bidirectional roads. Also, there are sufficient parking spots to accommodate all Taxibots at the parking facility. On the other hand, taxiways are modelled as single-lane bidirectional roads. This means that only a single traffic direction is possible and is determined by ATC.
- It is assumed that Taxibot energy levels are sufficient to guarantee operations for an entire day. As obtained from interviews with personnel of Smart Airport Systems (see Section C) , it was found that in the current situation there is not enough information to predict energy consumption of full electric TaxiBots. The TaxiBot as being tested at this moment does not have energy consumption limitations as it is powered by diesel and allows up to 20 consecutive hours of operation.
- It is assumed that engine warm-up for departures, as well as engine cool-down for arrivals, can be done while a Taxibot is attached to the aircraft. This results in no added waiting time for an aircraft at the gate or runway.
- It is assumed that the aircraft is ready to depart the gate at the moment it is spawned into the network, i.e. the Taxibot does not need to couple to the aircraft. The ADS-B data denotes the start-time of taxi operations of the aircraft, where the push-back operation has been excluded from analysis. Therefore, the coupling time is set to zero seconds to allow the aircraft to leave at its scheduled time and be able to compare the results with the real-world scenario.

A.1.3. Relevant Agent properties

Algorithm 1 depicts the forward simulation algorithm as used in the Conflict-Based Search implementation for ATC agents. The algorithm determines for each active vehicle in the taxiway system the time of passing and distance for each node along a vehicle's route. Whenever two vehicles are expected to pass through the same node within 15 seconds of eachother, they are denoted as a potential conflict-pair and ATC intersection agents will try to prevent the conflict from happening through speed/route commands. Some additional properties that have been excluded from the Thesis paper are elaborated below.

Source/Sink agent properties:

Check Flight Schedule Property: This property consists of interactions between the source agent and the environment, i.e. the flight schedule. Every second, all source agents check the flight schedule whether an aircraft needs to be released from their location at the current time-point. If the spawn-time of an aircraft is equal to the current time-point, it is added to the release list and the Aircraft Spawn Property is triggered.

Aircraft Spawn Property: This property consists of interactions between the source agent and ATC agents. For inbound traffic, the source agent communicates with ATC whether the corresponding runway is available. If the runway-exit is free, the aircraft is spawned and begins its taxi operations. Regarding outbound traffic, the source agent contacts the ATC gate agent about other traffic in the vicinity of the gate. The ATC gate agent checks for the adjacent edges if there is any traffic expected to use the meta-gate. If not, and the gate-release is considered safe, the source agent is informed and spawns the aircraft into the network. The ATC gate agent is now responsible for the guidance of the aircraft. Whenever a gate is not free or the aircraft cannot be spawned due to other traffic, the aircraft release is delayed.

Algorithm 1: Forward simulation of agent route traversal time

$route$ ← vehicle route
$current\_node\_index$ ← index of current node in route
$remaining\_nodes$ ← total number of nodes - $current\_node\_idx$
t_i ← empty list for time estimations to each consecutive node along route
d_i ← empty list for distances to each consecutive node along route
$WasTurn$ ← None
t_{delay} ← 0
if v > 0 **then**
    $v_i \leftarrow v_{current}$
else
    if $vehicle_{type}$ is Taxibot **then**
        v_i ← 5.2 m/s
    else
        v_i ← 2.6 m/s
    end if
end if
d_i ← distance to first node
t_i ← distance to first node / v_i
distance ← distance to first node
for j from $current\_node\_index$ to $remaining\_nodes$-1 **do**
    distance ← distance + distance from node j to node j+1
    if WasTurn is True **then**
        $t_{delay} = t_{delay} + \frac{|v_{vehicle} - v_{max,vehicle}|}{acceleration_{vehicle}}$
    end if
    WasTurn ← False
    Try:
        Edge ← edge from node [j+1] to node [j+2]
        if Edge contains a turn **and** $v_{vehicle} > v_{max,turn}$ **then**
            $t_{delay} = t_{delay} + \left|\frac{v_{max,turn} - v}{deceleration_{vehicle}}\right|$ WasTurn ← True
        end if
    except *IndexError*:
        if node j+1 is an endpoint **then**
            Edge ← route[j+1] + _-1
        else
            Edge ← $Goal\_node_{vehicle}$
        end if
        if Node j to Edge contains a turn **and** $v_{vehicle} > v_{max,turn}$ **then**
            $t_{delay} = t_{delay} + \left|\frac{v_{max,turn} - v}{deceleration_{vehicle}}\right|$
            WasTurn ← True
        end if
    except *KeyError*:
        pass
    $Time \leftarrow t_{current}$ + (distance / v) + t_{delay}
    t_i ← Time
    d_i ← distance
end for

Aircraft Removal Property: This property consists of interactions between ATC agents and sink agents. Whenever an aircraft reaches its destination node, the corresponding ATC agent contacts the sink agent regarding the upcoming aircraft removal. The ATC agent hands over the aircraft agent to the sink agent, which on its turn removes the aircraft from the simulator. The sink agent toggles an occupancy time, either for the gate or the runway, if an aircraft is removed from the simulator. During the occupancy time, no other traffic can take-off from the considered runway or arrive/spawn at the gate.

Aircraft agents:

Communicate Readiness Property: This property considers interactions between 1: Aircraft agents and Taxibot agents and 2: Aircraft agents and ATC agents. Whenever a departing Aircraft agent is spawned at the gate, it communicates to the ATC gate agent and vehicles nearby that it is ready for departure. This message initiates the attachment procedure of the Taxibot if the Taxibot signals to be ready.

Taxibot agents:

Communicate Status Property: This property involves interactions between Taxibot agents and the Taxibot coordinator agent. Each time-point, a Taxibot agent sends out a message to the Taxibot coordinator agent about its current status. This status update includes the current location, status and speed of the Taxibot agent and is internally processed by the Taxibot coordinator agent to keep track of all active Taxibot agents.

Taxibot coordinator agent:

Obtain Available Taxibots Property: This property involves interactions between the Taxibot coordinator agent and the Taxibot agents. Whenever the *Assign Task property* is run and future tasks are identified, a list of available Taxibots for task allocation needs to be generated. From the Taxibot agent's *Communicate Status Property* the Taxibot coordinator maintains an internal model of all available Taxibots and their locations. This list is used in the *Assign Task Property*.

A.1.4. Relevant Steps in Model Expansion and Analysis

The AAS Agent-Based Model has been created iteratively due to several reasons. First of all, the concept of operational towing with autonomous Taxibots is novel. Therefore, the agent characteristics and behaviour had to be created from scratch. To accurately verify correct agent behaviour, the initial model development was carried out with a single runway and a reduced taxiway lay-out. This allowed for a better verification of agent behaviour via model animations and individual agent traces. After verification of this (reduced) model functioning, the model has been expanded to consider the infrastructural network of AAS more realistically with all runways. In expanding the model, some extra model additions and modifications were required due to the expanded simulator environment and its specifications. Some of the problems and how they are solved are elaborated below, as they provide a basis for some of the choices made and could be of use for any interested reader.

Missing return routes for Taxibots from 18R/36L runway

Problem

One of the first necessary additions to the model came to light when simulating Taxibot detachments at the 18R/36L runway. An example situation explaining this necessity is depicted in Figure A.3. It can be seen that, on the left hand side, a tow Taxibot is being detached from the aircraft it has been towing to the runway. A first remark can be made regarding this detachment; after the detachment is completed, the Taxibot has nowhere to go as it cannot travel back in opposite direction of the aircraft's travel direction. However, let's assume that the Taxibot is able to return to the taxiway segment between nodes 113 and 128, also referred to as taxiway Victor. It can be seen from the Figure that another aircraft is already travelling this segment, as it has the goal to take-off from node 134. This means that no route could be determined for the individual Taxibot back to its parking facility. This causes significant problems, as it greatly increases the risk for grid-locks on this specific taxiway sector. The problems concerning the one-way travel direction of taxiway Victor have been identified in former studies [23]. The inclusion of operational towing of aircraft further increases these problems.

Solution:

In order to ensure a route back for Taxibots without interfering with traffic on taxiway Victor, additional dual-lane service roads have been implemented in the model. As mentioned in [8], taxi operations to 18R/36L via the taxiways surrounding 18C/36C (northern and southern taxiways) allow only a single direction of travel. Therefore, if a vehicle is guided from node 128 (see Figure A.3) to a gate via the southern taxiway surrounding 18C/36C, the entire southern segment can only be used in this direction. Therefore, dual-lane service roads

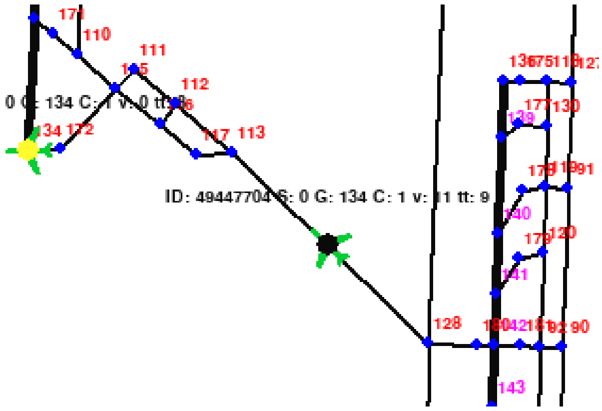


Figure A.3: Missing of return route options Taxibot after detachment 18R36L

have been adopted that surround these one-way taxiways, as is depicted in Figure A.4. It can be seen that there are two points where Taxibots have to be handled by an ATC agent to cross a taxiway; this is both at node 128 (start of taxiway Victor), as well as node 288 (tail of Quebec segment). At this locations, ATC checks whether the Taxibot(s) can safely cross the taxiway segment and handles the crossing. The inclusion of the service roads as depicted allows taxibots to travel to and from the 18L/36R and 18C/36C runways back to the gates.

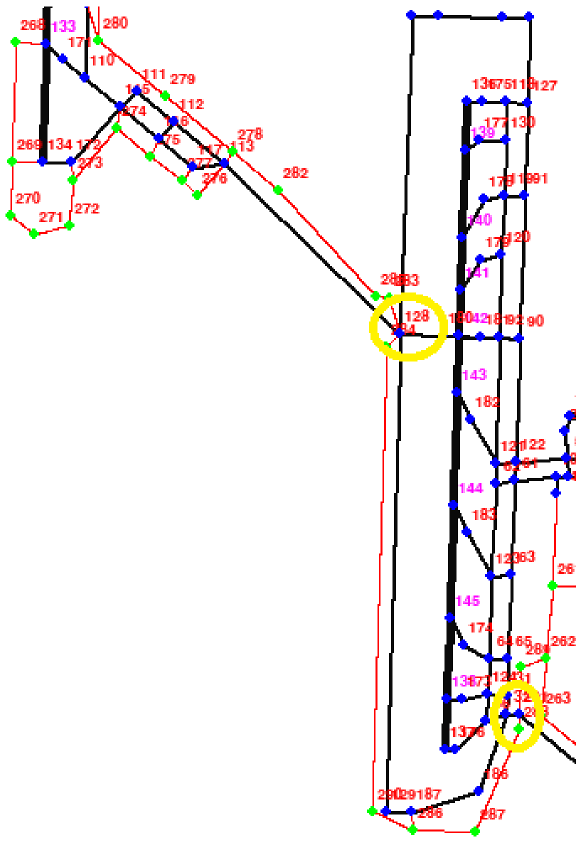


Figure A.4: Service roads 18L/36R and 18C/36C

Crossing and detachment issues departures 36C
Problem:
Another problem occurred when using the Taxibot for departures on runway 36C. These problems are two-folded; on one hand an issue was noticed with the service-road crossing near the 36C runway entries, while

on the other hand issues were encountered regarding the routing of Taxibots after detachment from departures using runway 36C. As both problems occurred in roughly the same airport section, they are discussed altogether in this Section. The first issue relates back to the inclusion of service roads originating from the 18R/36L runway. These service roads lie around the northern (Yankee) and southern taxiways (Zulu) of the 18C/36C runway. Therefore, at some point, a taxibot traversing these service roads has to cross a taxiway to return to the service road network at Schiphol center. Regarding the 36C runway, this point is indicated roughly at node 290 in Figure A.5. It can be seen that at this point in the infrastructural network, the service road crosses the taxiway. As this taxiway can be active and busy, a suitable point of crossing had to be determined. This will be elaborated upon in the next part.
Secondly, Taxibot return routes had to be implemented when considering Taxibot procedures for 36C departures. The original taxiway lay-out as implemented by previous studies [8, 23] and indicated by the red dots in the Figure, only provide a single taxiway lane up to the take-off point. This means that detachment must occur at the third to last node in the route, or even earlier. As the Alfa and Bravo taxiways (dual lane taxiway depicted in the right part of the Figure) are often busy with traffic, return routes of Taxibots via these taxiways is not desirable. Return routes of the Taxibots via either the Alfa or Bravo taxiways quickly causes grid-locks in the system, as other aircraft are using these taxiways in opposite direction to travel to their take-off runway point. Therefore, new roads had to be implemented in the model to ensure a safe return route for the Taxibots. This will be discussed next.

Figure A.5: Service road crossing and Taxibot detachment 18C/36C

Solution:
The resulting lay-out of service roads and taxiways near the 36C runway is depicted in Figure A.5. Firstly, a solution had to be found to connect the southern service road (below the Zulu taxiway) to the Yankee apron (aircraft parking location in the right part of the Figure). An extra ATC intersection node has been implemented, indicated by id 290, to ensure both a safe crossing of Taxibots from the Zulu taxiway to the Yankee apron and vice-versa, as well as a crossing of aircraft from the Quebec taxiway to the Alfa taxiway. This provides the Taxibots to return to Schiphol center both from the 18R/36L and 36C runway.
To allow Taxibot procedures for departures from the 36C runway, nodes have been added on the left hand side of the 18C/36C runway (see Figure). Central and decentral coupling locations for the Taxibot have been obtained from Schiphol experts for a specific set of runways currently considered in the Taxibot pilot. These decoupling points have been adopted for both the 18R/36L and 18C/36C runways. For the 36C runway, P4 (preferred detachment point, indicated by node **300**) and P5 (indicated by node **292**) will be used as decoupling locations. These additional taxiways are visualised in the Figure by means of the green diamonds with id's: 291, 292, 293, 294, 300 and 301. A first return direction for the Taxibot, if detachment is done at node 300, is via node 301 through node 187 back to the Zulu service-road. However, as traffic could build-up on this

taxiway crossing at node 187 may not be the preferred procedure. A second, more preferred return route is facilitated through node 292 and 297. This return route crosses the Zulu taxiway at node 298 and then enters the Zulu service-road again. These routes allow the Taxibots to safely return to their parking location, without having the need to travel any active taxiway.
By conducting interviews regarding these return routes for departures on 36C, it has been obtained that this new departure taxi-route causes other runway entries to be used less often. As can be seen in the Figure, the 36C runway can now be entered from both the left- and right hand side. From expert judgement, it has been confirmed that the right hand side runway entries (i.e. nodes 173 and 176) will not be used anymore if the left-hand side entries are to be opened up because of the Taxibot procedures. Therefore, nodes 173 and 176 are removed from the simulator.

Consecutive departures at same gate
Problem:
Another problem that was identified relates to consecutive aircraft departing from the same gate. An example of such a situation is given in Figure A.6 and the issue will be discussed based upon this figure. In the figure, an aircraft departing from gate 24 is shown. As indicated by the black dot, a Taxibot had just been attached to the aircraft and the aircraft has commenced taxi operations from its stand. In front of the aircraft, another individual Taxibot arrives at the gate (from the airside part of the network) to pick up the next aircraft departing from this gate. These two aircraft depart from the same gate with a short period of time in between and, because of the coupling time of the Taxibot, incur a problem. Because the arrival direction of a Taxibot at the gate is not hard-coded in a specific direction, Taxibots are allowed to travel to the gates either via airside or landside. In this situation, the arrival of the next Taxibot via airside causes a grid-lock in the system.

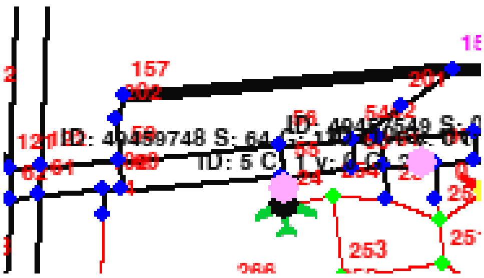


Figure A.6: Short interval between consecutive departures at the same gate

Solution:
In order to solve the aforementioned problem, it has been chosen to allow Taxibots to temporarily park at the gate. As the gates are modelled as meta-gates [8], they represent a cluster of gates at a specific pier. Therefore, the fact that aircraft can depart from the same gate within short intervals can be assumed valid. In the real world, these aircraft would depart from different gates, providing Taxibots with parking locations at the gate to wait before the aircraft signals to be ready. Therefore, it has been assumed that whenever an aircraft is not yet ready for attachment, a Taxibot is allowed to wait at the gate without it causing any hinder to other vehicles. This means that whenever the Taxibot is at standstill and parked at the gate, any other aircraft can be released from the gate and other Taxibots can travel from landside to airside or vice-versa. Logically, before an aircraft is released from the gate it is verified that no other traffic is occupying the first segment of its route.

Mixed flights at the same gate
Problem:
Slightly similar to the aforementioned problem with consecutive departures at the same gate, problems occurred due to mixed flights at the same gate. This problem concerns an arrival aircraft and departing aircraft travelling to, respectively from, the same gate. The origin of this problem is slightly more complex than the problem related to consecutive departures and therefore asks for additional measures, as will be elaborated upon here.
The problem is mainly related to aircraft arriving at the second to last segment of their route up-to their assigned gate. An ATC intersection agent will determine whether the next intended segment of this arriving aircraft is available and, if so, hands the aircraft over to the last segment in its route. It can, however, happen that another aircraft and Taxibot are coupling at the gate. Due to the fact that these vehicles are not yet

coupled, they are not handed over to the ATC intersection agent yet and the ATC agent thus does not know about their existence. This means that the arriving aircraft could be handed over to its last segment, while a departing aircraft is still busy coupling and will be handed over to its first segment in a few seconds. This situation causes a grid-lock at the gate, as both aircraft have nowhere to go.

Solution:
To solve the aforementioned problem, two additional features have been implemented. On one hand, the **ATC endpoint agent** (gate) checks whether the adjacent edges are free of other traffic intended to travel towards the ATC agent's corresponding gate. Additionally, for some small segments the ATC agent includes a limited amount of nodes further down the network to ensure a safe release of departing aircraft. This is due to segments being too short to ensure minimum separation distances between consecutive vehicles (< 150m). On the other hand, the ATC intersection agents have been extended with the option to communicate with the ATC endpoint agent an arriving vehicle is travelling towards. If this vehicle needs to be handed over by the ATC intersection agent to its last segment, it must determine whether this can be done safely. The ATC intersection agent determines: 1) whether the next segment is free of any traffic and/or 2) whether the gate is available. If both conditions are met, the vehicle is handed over to its last segment. In case the next segment or gate is unavailable, the respective vehicle is commanded to stop at separation distance from the intersection until the aforementioned conditions are met and it can safely attach to the gate.

Runway occupancy due to (active) runway crossing

Problem:
There are several locations on the airport lay-out at which a vehicle (aircraft or Taxibot) can or must cross a runway. In a very specific case, this runway crossing is forbidden (at W5) because it can cause inefficiencies near the corresponding runway, i.e. vehicles stopping too close to the runway because they have to give priority to another vehicle. Aside from this unique runway crossing, other runway crossings may be necessary whenever the runway is being active. Such a situation can be imagined if an aircraft departs from gate 2 to the 18R/36L runway and the 06/24 runway is being active. In such a case, the runway crossing of this aircraft must toggle a runway occupancy to prevent any other aircraft from landing or taking off from this same runway during this time-period, to ensure a safe minimum separation time in between. Another case relates, again, to the 18C/36C runway. From expert interviews it has been noted that the Yankee and Zulu taxiways may not be used freely in specific RMOs. If the 18C runway is being used for departures **or** the 36C runway is being used for landings, the Zulu taxiway directly underneath the runway cannot be used freely. In the opposite case, if runway 18C is being used for landings **or** runway 36C is being used for departures, the Yankee taxiway may not be used freely.

Solution:
To ensure safe minimum separation times between runway operations, stopbar procedures have been altered to toggle a runway occupancy and prevent aircraft from taking off / landing while an aircraft crosses such an active stopbar. This runway occupancy (30 seconds), provides the vehicle with a time-window in which the stopbar to stopbar crossing can be traversed. These crossings are active for all stopbar-stopbar taxiway segments whenever the corresponding runway is active **and** for the specific cases of the Yankee and Zulu taxiways as explained above. An example of such a crossing near the 18C/36C runway is depicted in Figure

A.5 for nodes 186-187 (both stop-bars).

A.1.5. Input data specifications

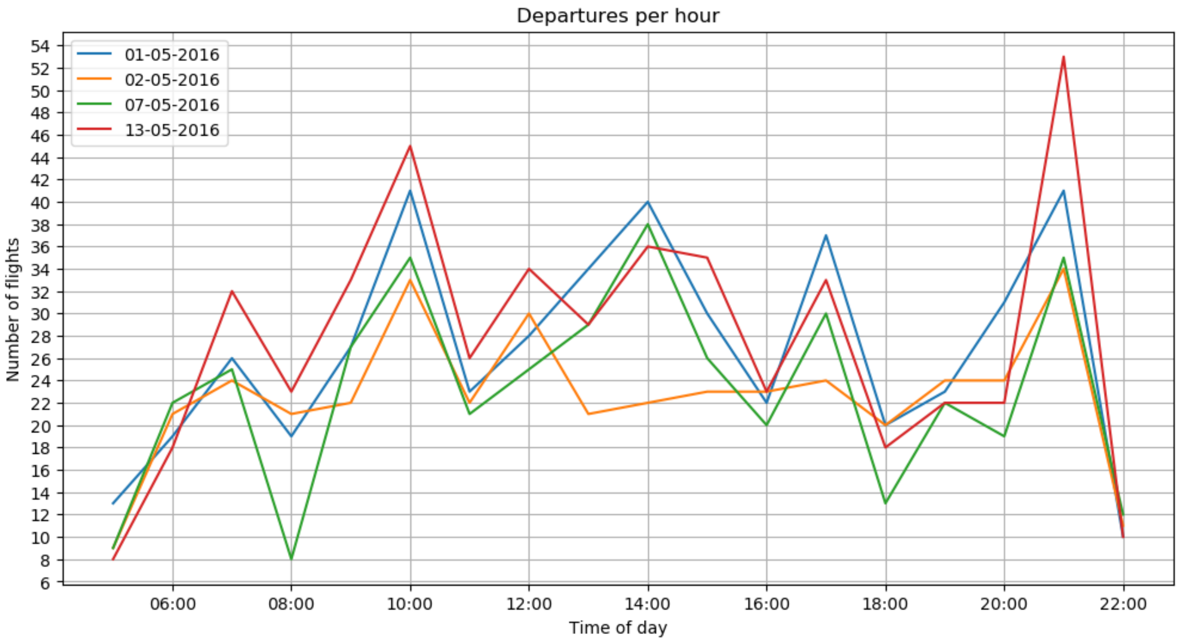


Figure A.7: Departures per hour for considered input data-sets

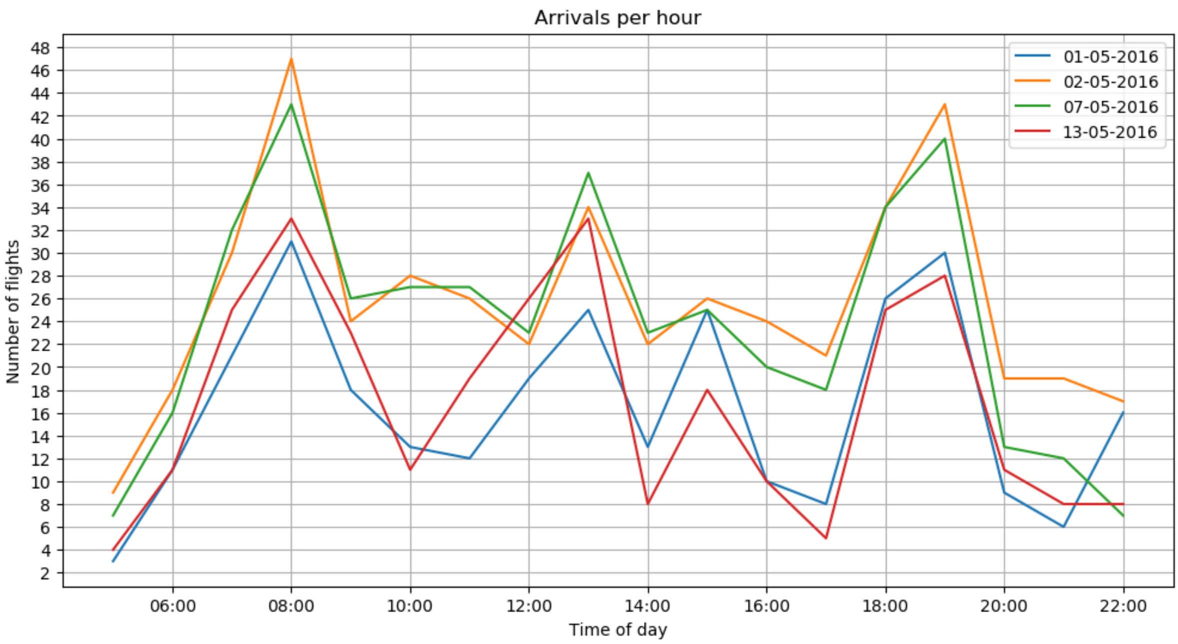


Figure A.8: Arrivals per hour for considered input data-sets

A.1.6. Simulator parameters

Symbol	Description	Value
v_{max}	Maximum taxi-speed	15.4 m/s
v_{turn}	Maximum taxi-speed in turns	5.144 m/s
$acc_{comfort}$	Comfort acceleration level	0.26 m/s^2
$dec_{comfort}$	Comfort deceleration level	0.77 m/s^2
dec_{max}	Maximum deceleration level	5.14 m/s^2
Radar range	Radar range within which other vehicles can be detected	250 m

Table A.1: Aircraft agent dynamics

Symbol	Description	Value
v_{max}	Maximum taxi-speed	11.8 m/s
v_{turn}	Maximum taxi-speed in turns	5.92 m/s
acc_{max}	Comfort acceleration level	0.41 m/s^2
dec_{max}	Maximum deceleration level	1.23 m/s^2
Radar range	Radar range within which other vehicles can be detected	80m
t_{couple}	Detachment time for couple to aircraft	60 s

Table A.2: Taxibot agent dynamics

Symbol	Description	Value
$sep^{airside}_{taxibot}$	Minimum separation distance on taxiways between Taxibots	50 m
$sep^{landside}_{taxibot}$	Minimum separation distance on service-roads between Taxibots	35 m
$sep^{airside}_{aircraft}$	Minimum separation distance between aircraft	150 m
$t^{departures}_{runway}$	Time between consecutive runway departures in seconds	60 s
$t^{arrivals}_{runway}$	Time between consecutive runway arrivals in seconds	60 s
occ_{gate}	Occupancy time after aircraft gate usage	30 s
$t^{crossing}_{runway}$	Occupancy time toggle for runway crossing	30 s

Table A.3: ATC related simulator parameters

Symbol	Description	Value
$\Delta t_{arrival}$	Arrival time-window of Taxibot before allocated task start-time	60 s
R_{max}	Maximum number of task reassignments by Taxibot Coordinator agent	1
Δt_{future}	Time-window of future upcoming tasks to consider for task allocation	10 min
TA_{rate}	Task allocation rate interval	10 s
$TA_{sorted,list}$	Task Allocation with sorted list on Taxibot distance	True
$TA_{active,TB}$	Task Allocation with active Taxibots considered	True

Table A.4: Taxibot coordinator parameters

Symbol	Description	Value
dt	Timestep in simulator	1 s
$deg_{no,turn}$	Maximum turn degree for which no braking is required	30 deg
$\Delta t_{window,CBS}$	Time-window CBS algorithm	15 s

Table A.5: General simulator parameters

A.2. Simulation Results

A.2.1. Q-Q Plots initial Aircraft results

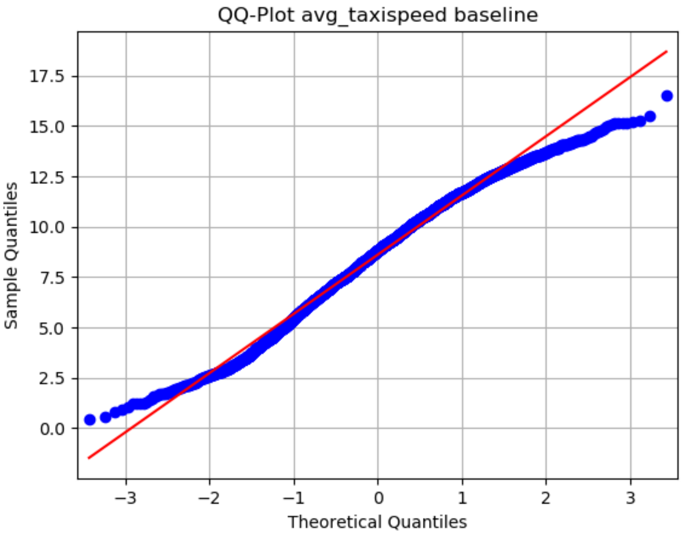


Figure A.9: QQ-plot aircraft average taxi-speed real-world

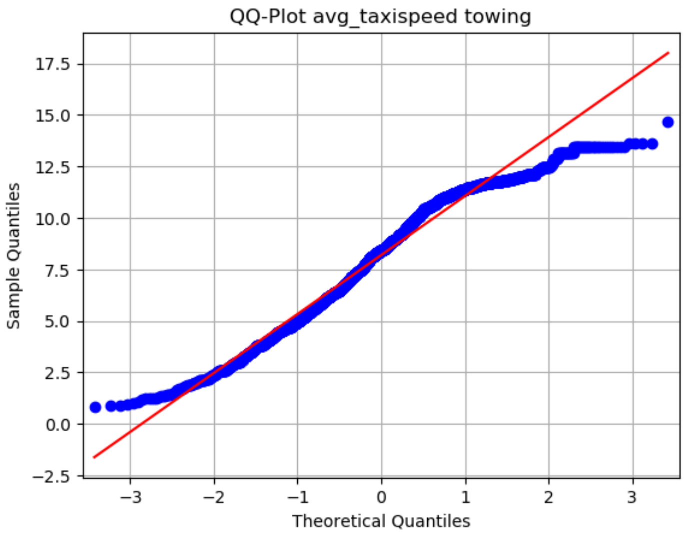


Figure A.10: QQ-plot aircraft average taxi-speed towing

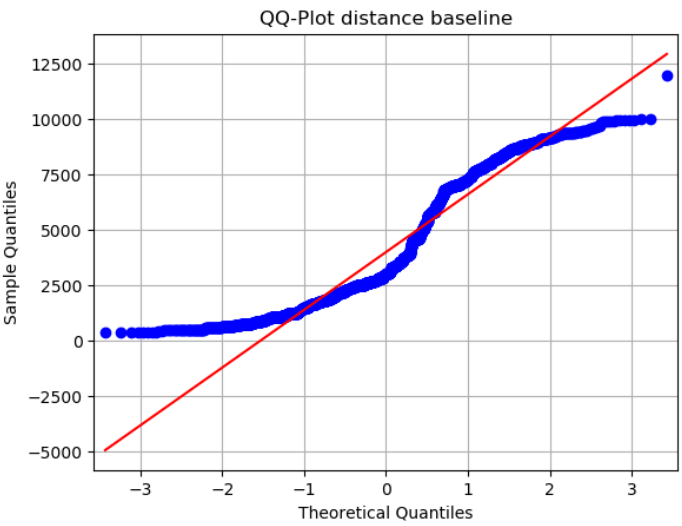


Figure A.11: QQ-plot aircraft taxi-distance real-world

Figure A.12: QQ-plot aircraft taxi-distance towing

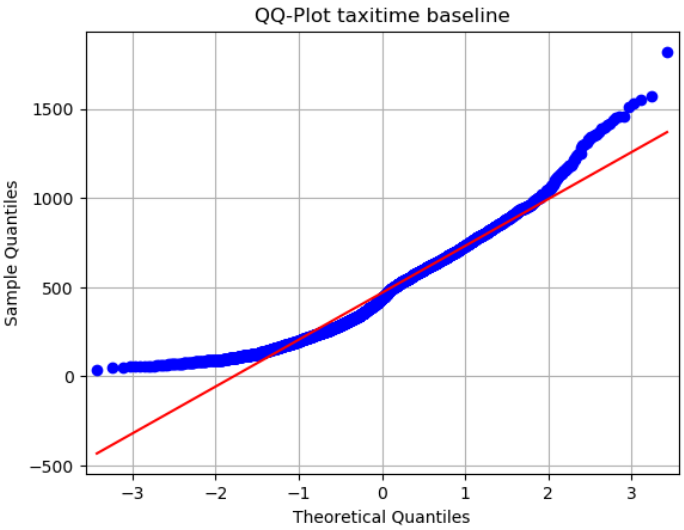


Figure A.13: QQ-plot aircraft taxi time real-world

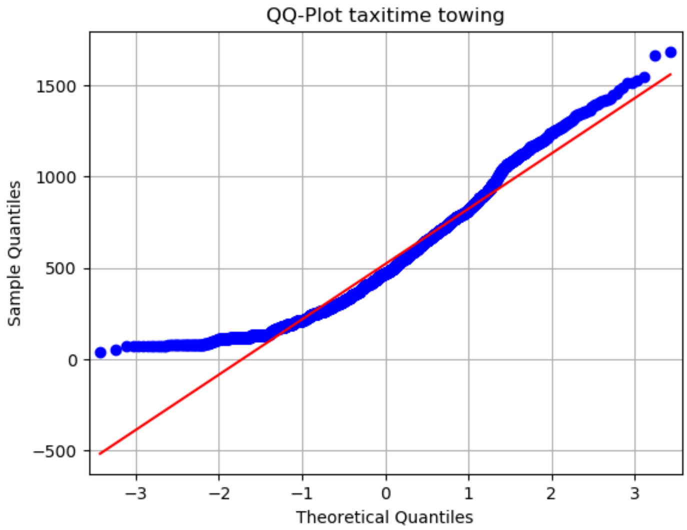
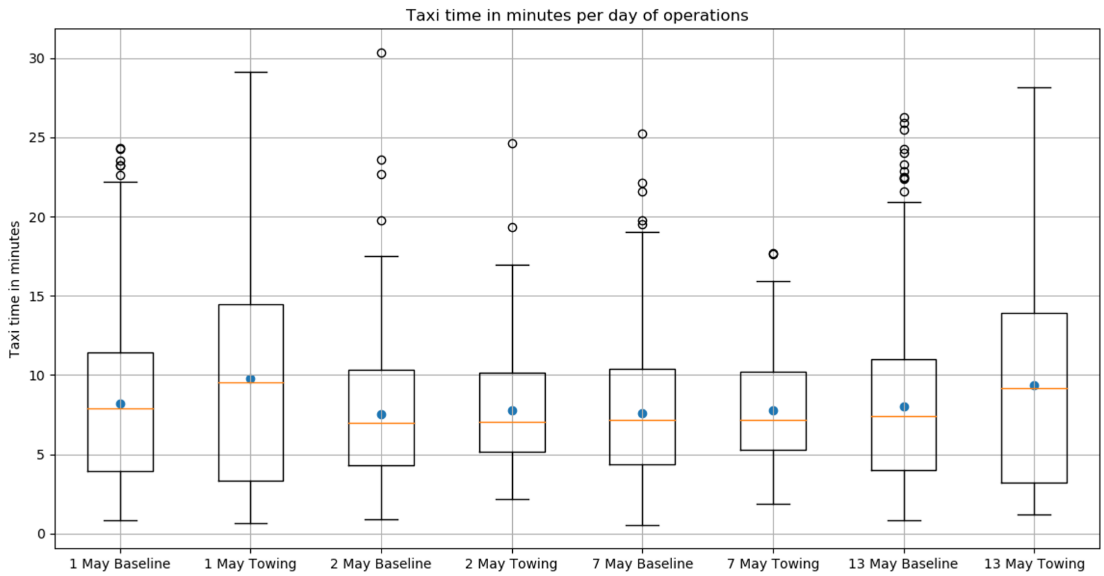


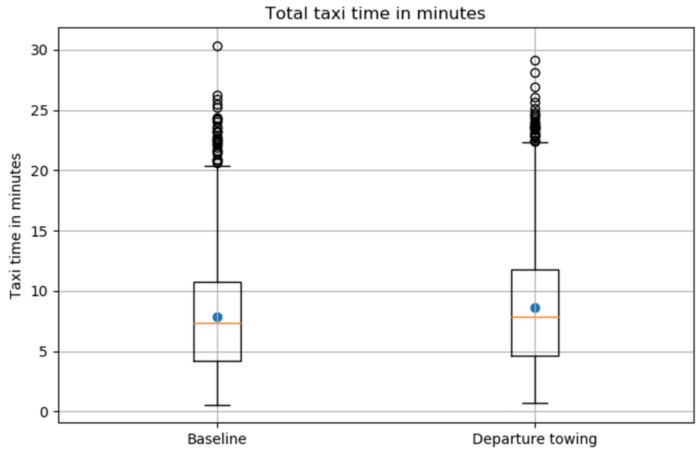
Figure A.14: QQ-plot aircraft taxi time towing

A.2.2. Simulation Results



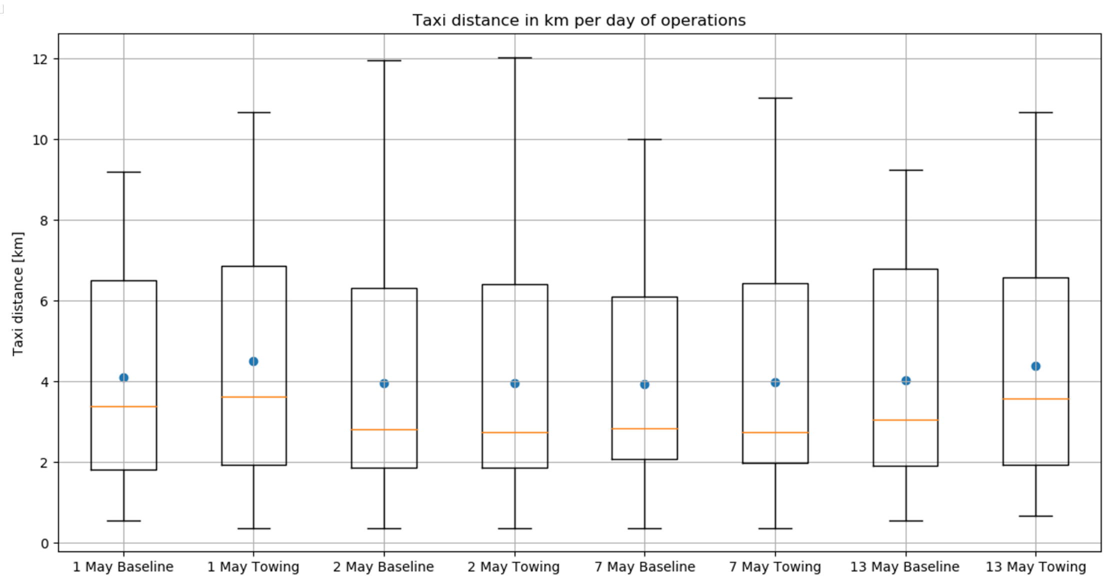
μ	8.20	9.78	7.50	7.77	7.58	7.75	8.03	9.38
σ	4.93	6.54	3.88	3.28	3.80	3.12	4.85	6.16
Min.	0.87	0.67	0.88	2.17	0.55	1.90	0.85	1.22
Max.	24.33	29.13	30.35	24.62	25.25	17.70	26.25	28.15

Figure A.15: Aircraft taxi time baseline scenario and departure towing only per day



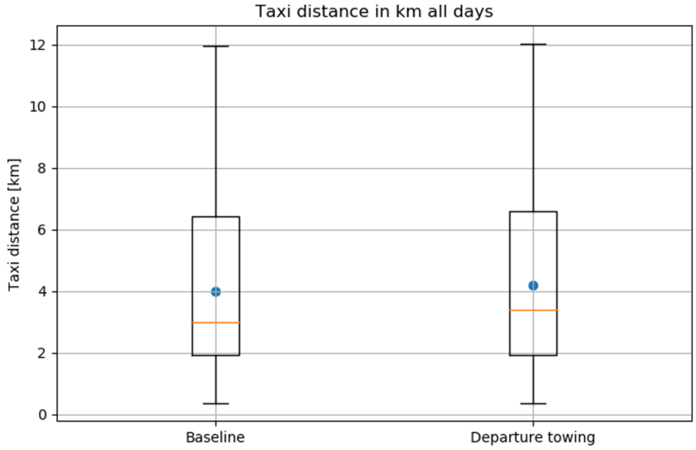
µ	7.81	8.63
σ	4.38	5.05
Min.	0.55	0.67
Max.	30.35	29.13

Figure A.16: Total aircraft taxi time baseline versus departure towing only for all days



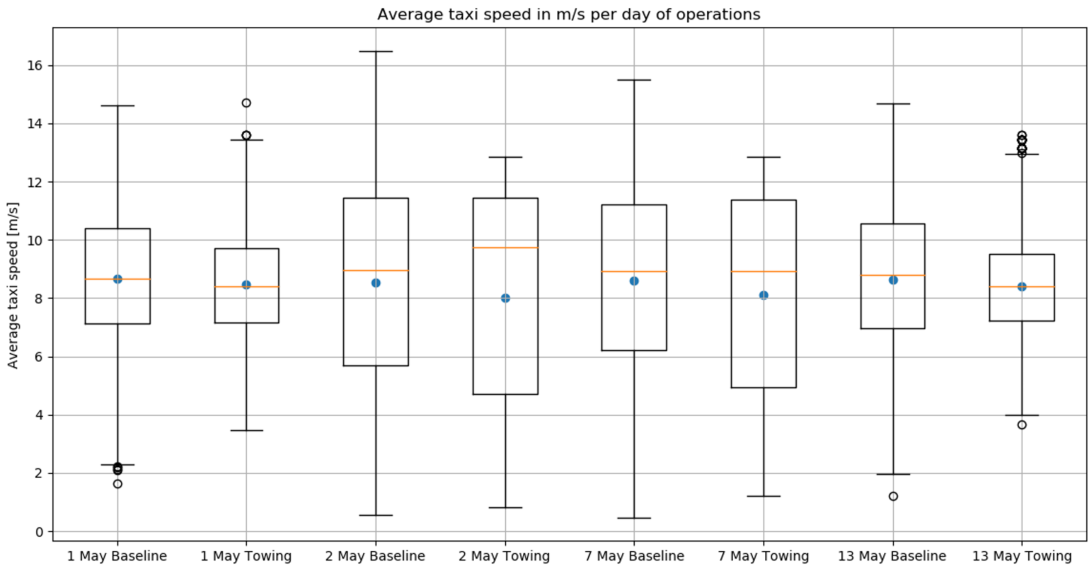
µ	4.09	4.50	3.95	3.96	3.92	3.97	4.02	4.39
σ	2.49	2.85	2.79	2.78	2.58	2.64	2.54	2.86
Min.	0.56	0.37	0.36	0.37	0.37	0.37	0.56	0.67
Max.	9.19	10.67	11.95	12.03	9.99	11.03	9.24	10.67

Figure A.17: Aircraft taxi distance baseline scenario and departure towing per day



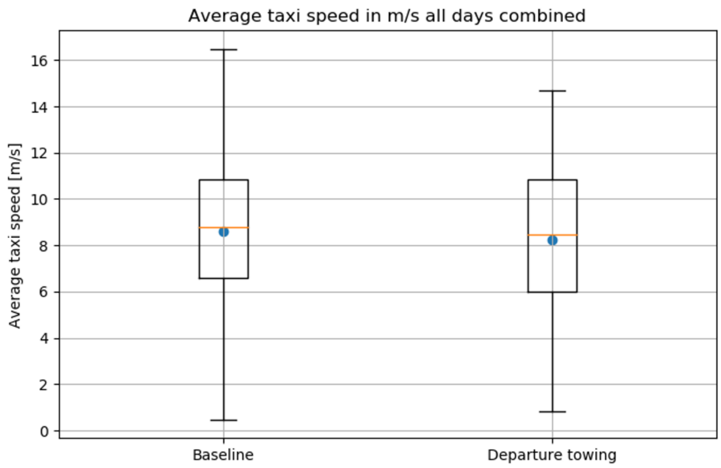
	Baseline	Departure towing
μ	3.99	4.19
σ	2.61	2.79
Min.	0.36	0.37
Max.	11.95	12.03

Figure A.18: Total aircraft taxi distance baseline versus departure towing for all days



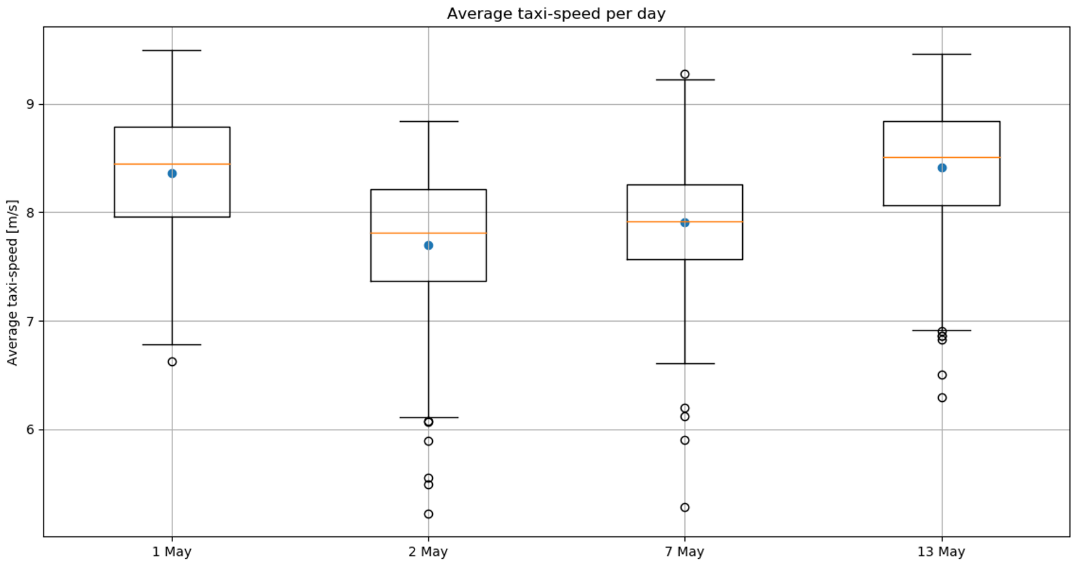
	1 May Baseline	1 May Towing	2 May Baseline	2 May Towing	7 May Baseline	7 May Towing	13 May Baseline	13 May Towing
μ	8.67	8.48	8.53	8.01	8.58	8.10	8.63	8.40
σ	2.38	2.11	3.48	3.62	3.22	3.33	2.45	1.90
Min.	1.65	3.48	0.55	0.83	0.47	1.23	1.22	3.65
Max.	14.60	14.70	16.48	12.86	15.50	12.86	14.68	13.60

Figure A.19: Aircraft average taxi speed baseline scenario and departure towing per day



μ	8.60	8.24
σ	2.94	2.87
Min.	0.47	0.83
Max.	16.48	14.70

Figure A.20: Aircraft average taxi speed baseline versus departure towing for all days



μ	8.36	7.70	7.90	8.41
σ	0.62	0.68	0.61	0.60
Min.	6.63	5.22	5.28	6.29
Max.	9.50	8.84	9.28	9.46

Figure A.21: Average speed Taxibots [m/s]

A.2.3. Heat-maps for consecutive days in vehicle movements

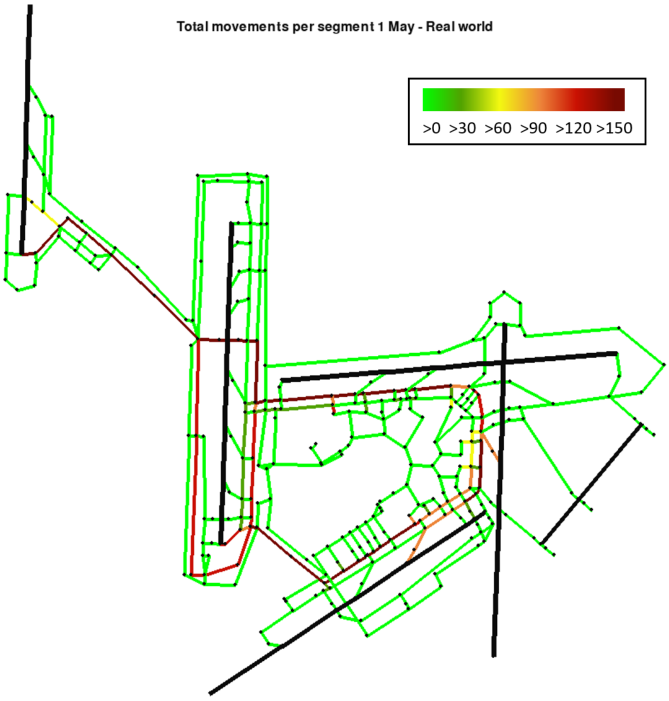


Figure A.22: Vehicle movements per segment, 1 May - Real world

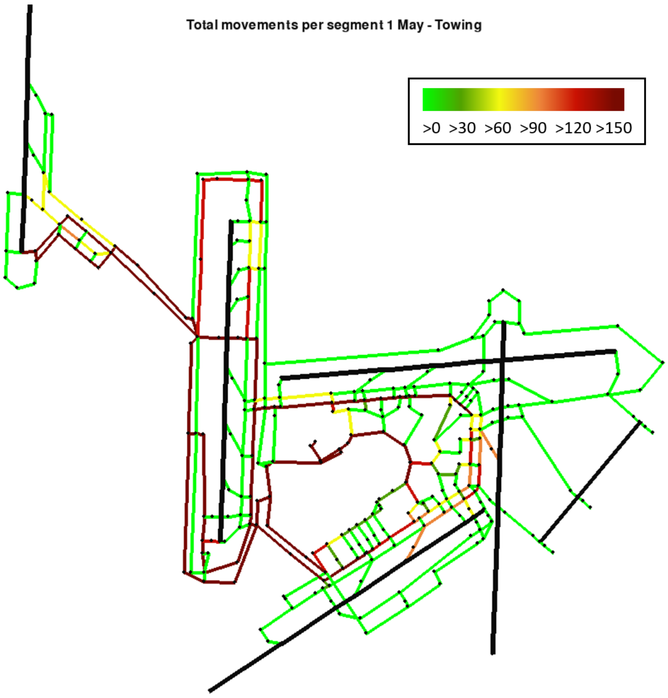


Figure A.23: Vehicle movements per segment, 1 May - towing

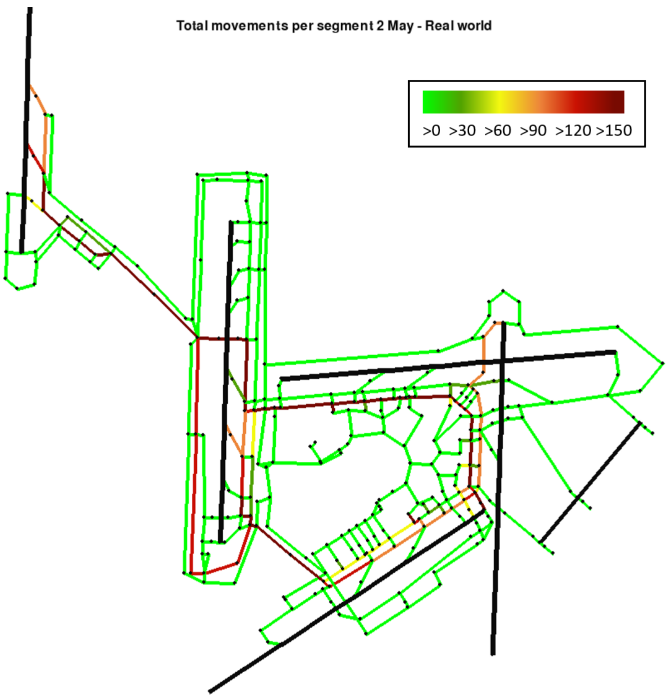


Figure A.24: Vehicle movements per segment, 2 May - Real world

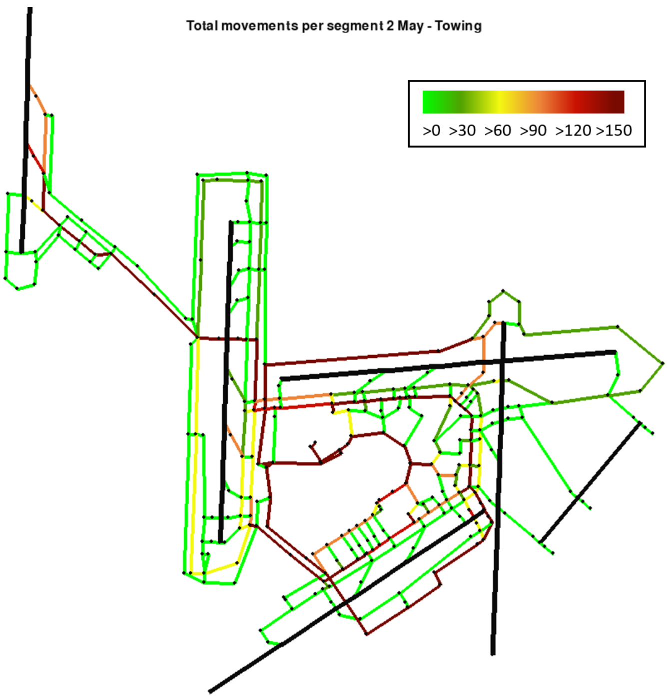


Figure A.25: Vehicle movements per segment, 2 May - towing

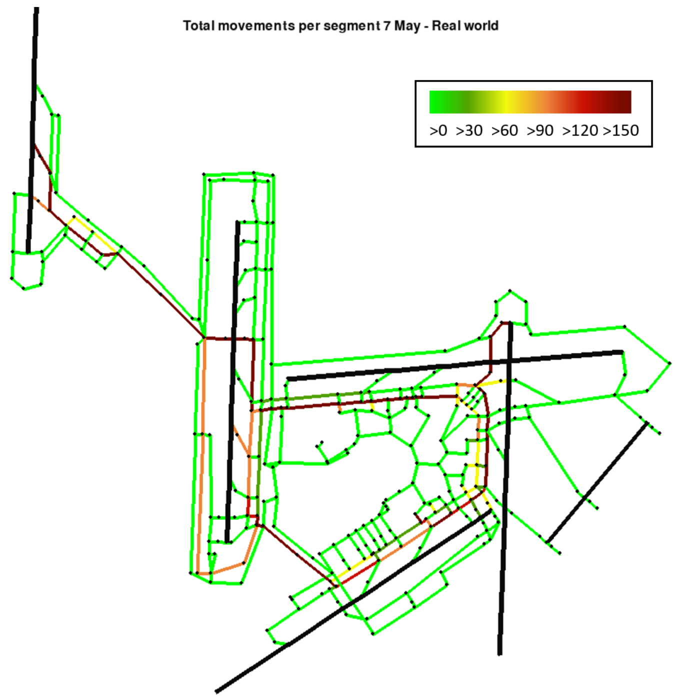


Figure A.26: Vehicle movements per segment, 7 May - Real world

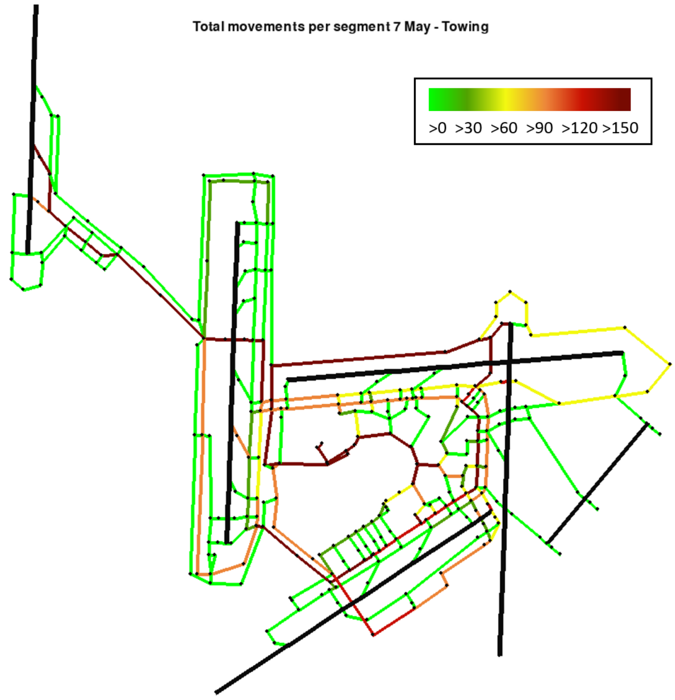


Figure A.27: Vehicle movements per segment, 7 May - towing

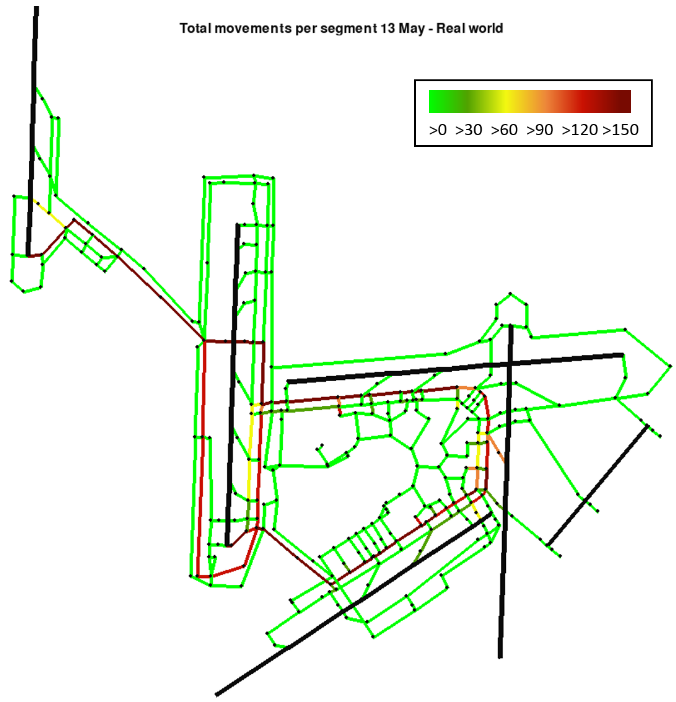


Figure A.28: Vehicle movements per segment, 13 May - Real world

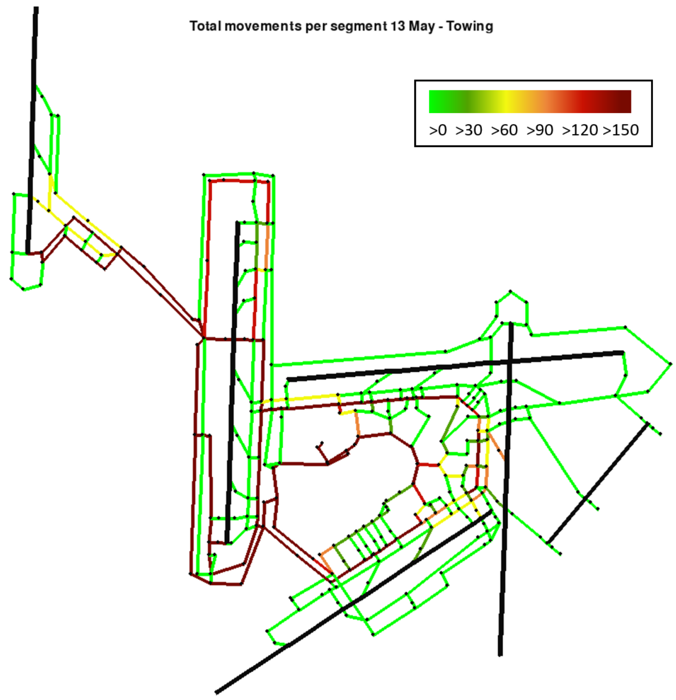


Figure A.29: Vehicle movements per segment, 13 May - towing

A.2.4. Heat-maps for consecutive days for average speed per segment

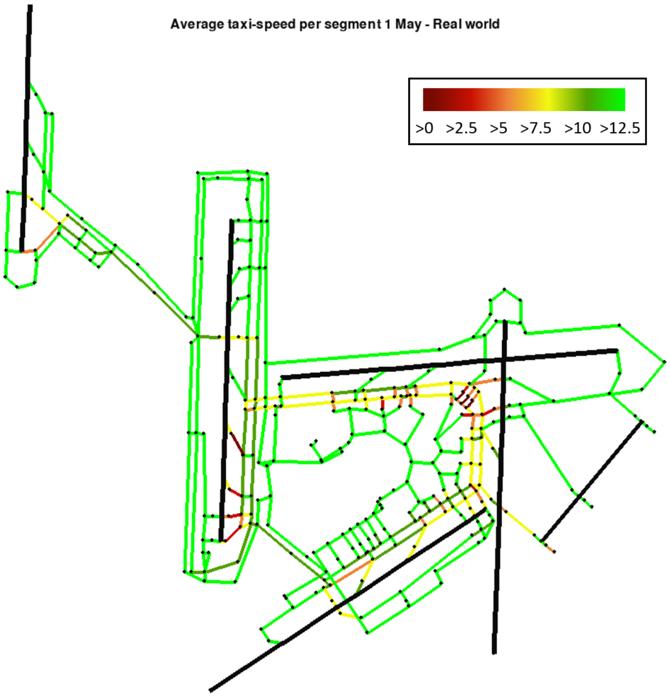


Figure A.30: Average taxi-speed per segment [m/s], 1 May - Real world

Figure A.31: Average taxi-speed per segment [m/s], 1 May - towing

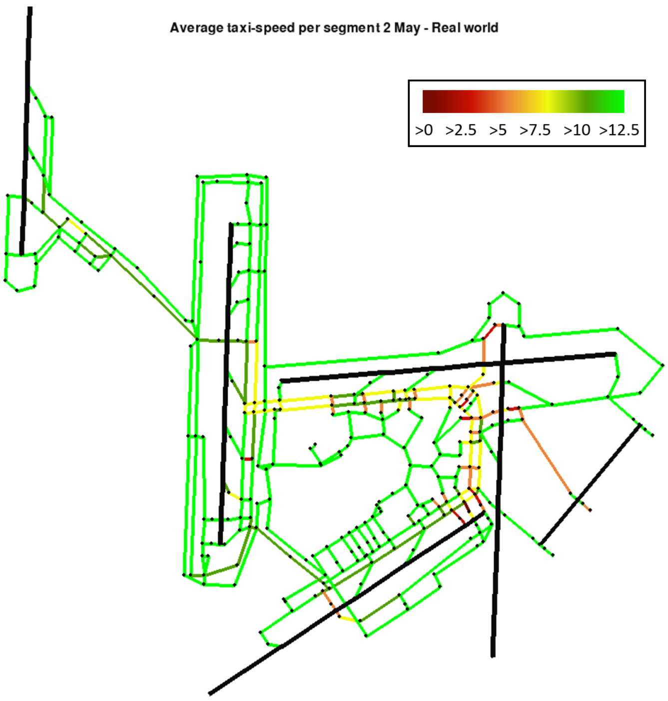


Figure A.32: Average taxi-speed per segment [m/s], 2 May - Real world

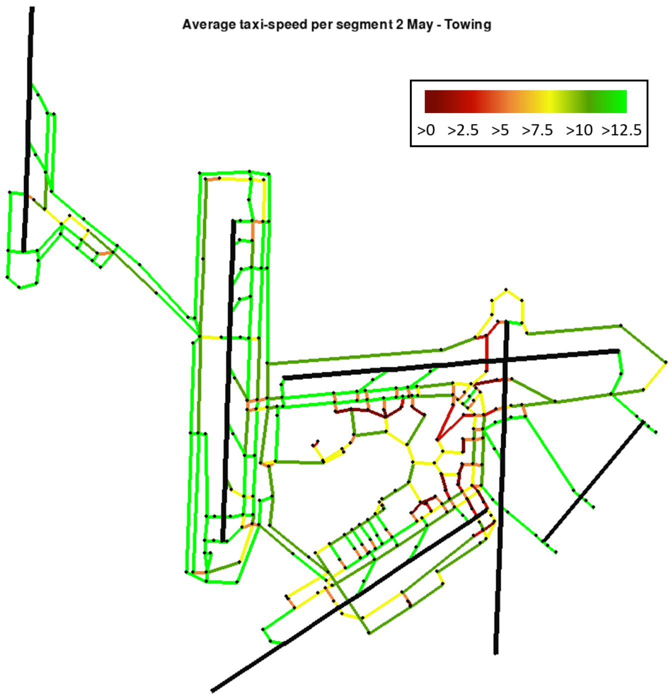


Figure A.33: Average taxi-speed per segment [m/s], 2 May - towing

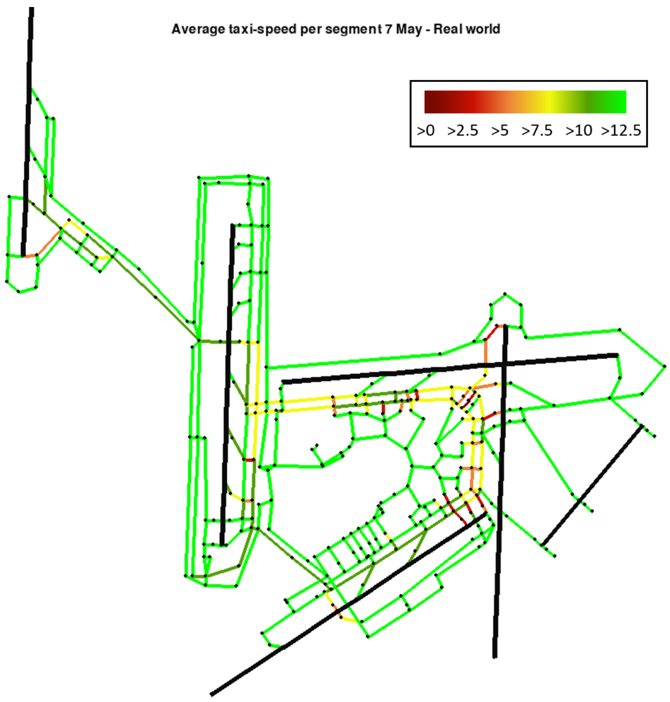


Figure A.34: Average taxi-speed per segment [m/s], 7 May - Real world

Figure A.35: Average taxi-speed per segment [m/s], 7 May - towing

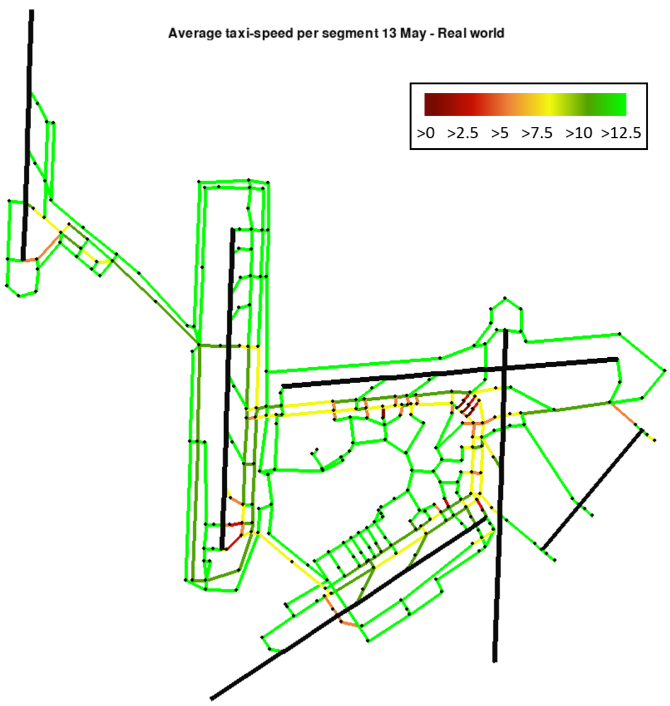


Figure A.36: Average taxi-speed per segment [m/s], 13 May - Real world

Figure A.37: Average taxi-speed per segment [m/s], 13 May - towing

A.2.5. Visual observations of model simulations

A few visual observations were made while carrying out the simulations for each of the four days of operations. These observations are elaborated below and provide an insight in the considerations that needed to be made in creation of the model.

Taxibot decoupling for R24 departures and R18L intersection starts

The Taxibot detachment locations as used in the simulation model have been obtained from experts of Schiphol. For both 36C and 36L departure traffic these locations have proven to be acceptable in the simulation model; it allows the Taxibots to return to their parking facilities without causing significant nuisance to taxiing aircraft. However, for two runway directions it was found that the preferred detachment points caused significant problems for the rest of the taxiing vehicles. These two issues will be discussed after another.

The first problem arises for intersection starts from runway 18L. As only Taxibot detachment locations for the runway head of 18L are provided by Schiphol, an initial location for detachment from 18L intersection starts had to be determined. In line with the detachment locations of runway 24, it has been chosen to detach the Taxibot at the Bravo taxiway at the nearest node from the designated runway entry. This situation is visualised on the left hand side of Figure A.38. A major problem this location causes is that no other traffic is able to travel towards the 18L runway head during the detachment of the Taxibot for an aircraft intersection start at 18L. Also, the Taxibot will have to move a significant distance over the taxiway, increasing the density on this part of the airport. It can be seen from the figure that, in this example, two aircraft have to wait unnecessarily long for the detachment to be completed and the taxiway to be cleared again before they can travel towards their Taxibot detachment location. As this is far from ideal, it has been chosen to adopt a different detachment location for intersection starts from 18L: the last node before the runway entry is adopted as Taxibot detachment node. One drawback from these locations is that the Taxibot has to cross the runway to travel back to its parking facility. However, it does not cause taxiway congestion as for the situation described before and the Taxibots runway crossing time is significantly smaller than the blockage time of an aircraft that cannot continue its taxiing operations.

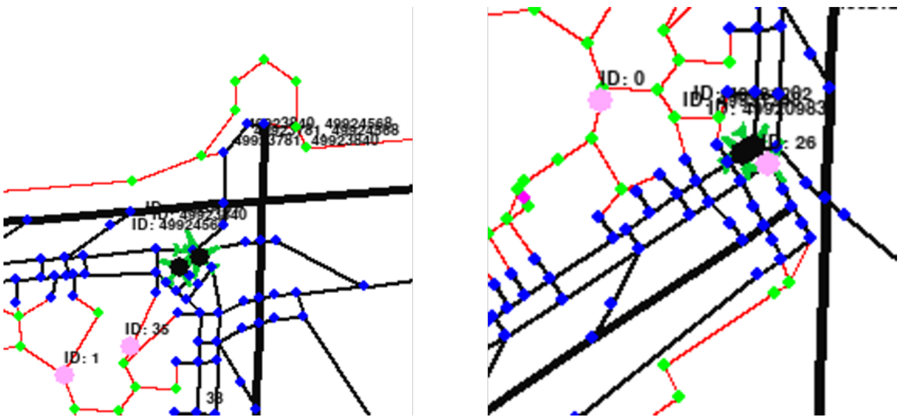


Figure A.38: Detachment issues runways 18L (left) and 24 (right)

The second detachment issue that came to light is related to detachment of the Taxibot for runway 24 departures. As detailed by AAS, decoupling for R24 will happen at the Alfa and/or Bravo taxiways from which the Taxibot will move towards the service road in between the two aforementioned taxiways. This service-road is not implemented in the model, which means that the Taxibots must travel via either Alfa or Bravo back to their parking facility. It can already be seen that this causes similar issues as described before, both blocking an active taxiway and increasing local traffic density on this part of the airport. A visualisation of such a detachment procedure can be found on the right of Figure A.38. Here, an aircraft is busy with the decoupling procedure while two other aircraft are waiting for their turn: one from the left hand part of Alfa and the other from the upper part of Alfa. This situation implies that no other inbound aircraft could travel through this part of the airport lay-out, or it has to wait for the aircraft to clear the taxiways again. In a similar reasoning as for intersection starts at 18L, it has been chosen to adopt the second to last node before the runway entry as Taxibot detachment point. Although the Taxibot has to cross an active runway, it provides way less nuisance to other taxiing aircraft and a better line-up of outbound traffic can be realised.

Crossing issues 06/24 for departures node 2

Gate node 2 is a special gate as it denotes the Sierra cargo apron at AAS. Some flights from the ADS-B data depart from this 'gate' and therefore a specific problem has to be discussed in this Section. This issue relates to the crossing of runway 06/24 of aircraft intended to take-off from a runway other than 36L, 36C or 24. Because of the implemented shortest path algorithm any aircraft departing from node 2 to, for example, runway 18L will be routed the yellow route as indicated in Figure A.39. This route crosses the 06/24 runway at the head of runway 24. After observing several simulations it was found that in some cases, a grid-lock occurred due to two aircraft being head to head on the S7 taxiway (upper yellow highlighted part above runway 06/24). Aircraft with the intention to take-off from runway 24 and arrive from either the Alfa or Bravo taxiways have no alternative options from the moment they move towards the runway entry. As the Taxibot still has to detach from this aircraft but is not yet included in the CBS algorithm, any aircraft from gate 2 does not receive rerouting commands yet. On the other hand, the aircraft departing from gate 2 has no alternative route options whenever it turns right on the S-taxiway. Therefore, the grid-lock can only be prevented if an aircraft is held at gate 2 until its route is clear again.

To solve the aforementioned problem, it has been chosen to force outbound aircraft from gate 2, that will not take-off from runway 24, to travel via the middle runway crossing of 06/24. This way, these outbound aircraft will not be of nuisance to other aircraft taking off from runway 24 and it prevents grid-locks from happening at the head of runway 24.

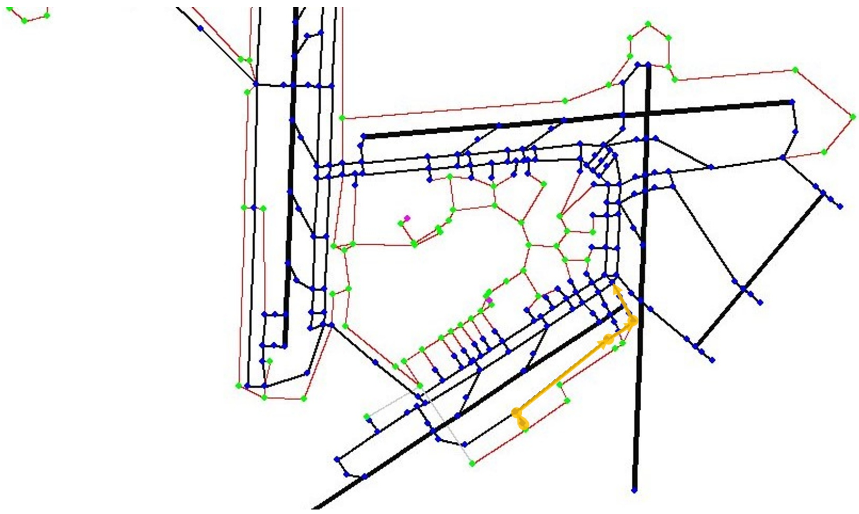


Figure A.39: Crossing issues departures gate 2

Line-up for Taxibot detachment

As touched upon already in Section A.2.5, some detachment locations for the Taxibot could create significant congestion and chaotic line-ups of aircraft for detachment. As departing aircraft arrive at the detachment location from different directions, often aircraft line-up in a relatively chaotic manner. Therefore, the aircraft line-up directions for Taxibot detachment must be carefully considered when implementing autonomous towing with the Taxibot. For outbound RMOs including runways 36L, 36C or the end of runway 18L, the line-ups do not cause any nuisance to other traffic. On the other hand, departures from runway 24 or intersection starts from 18L could cause the alfa and bravo taxiway to get congested relatively quickly, if no operational procedures are implemented for aircraft line-up. Such operational procedures could be agreed upon with ATC.

It must be noted that the input data sets consider only 800-900 flights, whereas AAS (before COVID-19) was averaging around 1500 flights per day. Besides, the simulation model only considers outbound traffic for Taxibot towing. For increasing traffic numbers, the aforementioned line-up problem could cause even more problems.

Difficulties in shortest path determination for speed restrictions

An issue that came to light during sensitivity analyses is related to the weight determination of taxiway and

service-road segments. As mentioned before, the model uses the A star algorithm to determine the shortest route between any two nodes in the network. For each vehicle type a corresponding weight is calculated for each segment in the graph. In the determination of this weight, the maximum speed of the corresponding vehicle is used.
Now, problems arise when implementing a maximum allowed speed on the landside network. A maximum allowed speed on landside segments causes these segments to be more heavily penalised in comparison to taxiways, as the weight increases due to the lower taxi-speeds. This resulted in the situation in which Taxibots were routed via a taxiway to their assigned gate, instead of just via the service-road network. As these routes cause significant issues in the taxiway network, measures had to be taken. These measures are somewhat similar to the route principles when Taxibots detach from their aircraft. In both cases, priority is given to route Taxibots via the service roads. This way, they should be able to reach their assigned gate via the service road network only and cause as less nuisance to taxiing traffic as possible. This prioritisation of service-roads has been achieved by significantly lowering the weights. After the route has been determined, a calculator for the actual path length is used to maintain accurate path lengths.

Infeasible decoupling locations

For a single take-off point, it was found that Taxibot detachment was infeasible to accommodate as this would lead to chaotic routes for the Taxibots and aircraft. This specific point relates to the W8 take-off point for runway 36C. Detachment of the Taxibot would require the aircraft to come at a standstill at either the alfa or Bravo taxiway, near the A25 crossing. However, it can be assumed that significant traffic taxis via these two taxiways, whenever 36C is used for outbound traffic. It was found in the simulation model that detachment at this point greatly reduces safety and caused grid-locks in the system, due to the Taxibot travel via the taxiways. Therefore, it has been chosen to omit take-offs from this specific intersection node for Taxibot towing. As the amount of flights from this specific intersection point is limited, the overall results remain valid.

Braking issues for very short service-road segments

An issue that was countered while running simulations for the sensitivity analyses relates to (very) short service-road segments. It was noticed that Taxibots were not capable of braking in time for the gate, leading to them surpassing the gate and the model failing to complete the simulations. This has to do with the limited information of a Taxibot, as the service-roads are not covered by ATC. It has been chosen to slightly adapt the towing Taxibot properties, such that they know which service-road segments are relatively short and require earlier braking. This has led to the expected behaviour in which Taxibots know when to brake earlier to realise standstill at their allocated gate.

A.3. Sensitivity Results

The results of the local and global sensitivity analyses are provided below. In each specific table, the parameter variation is indicated in the first column. The respective day of operations can be found in the title of the Table, together with the focus area of the sensitivity analysis: Taxibot coordinator or Taxibot dynamics parameters. The first row indicates the KPIs the A-test values are calculated for. A description of the respective symbols can be found in the list of Symbols. The subscripts A/D/All indicate the subset of flights included in the respective calculation: Arrival flights, Departure flights or All flights.

A.3.1. Sensitivity analysis results Taxibot Coordinator parameters

	AC_{tt}	AC_{td}	AC_{ts}	AC_{cd}	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$\Delta t_{future,6}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{future,8}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{future,12}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$R_{max,0}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$R_{max,2}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$R_{max,3}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$TA_{rate,20}$	0.50	0.50	0.50	0.50	0.53	0.53	0.43	0.50
$TA_{rate,30}$	0.50	0.50	0.50	0.50	0.54	0.53	0.40	0.50
$TA_{rate,40}$	0.50	0.50	0.50	0.50	0.55	0.52	0.37	0.50
$TA_{rate,45}$	0.50	0.50	0.50	0.50	0.55	0.49	0.36	0.50
$TA_{rate,50}$	0.50	0.50	0.50	0.50	0.56	0.47	0.34	0.50
$TA_{rate,55}$	0.50	0.50	0.50	0.50	0.57	0.48	0.33	0.50
$TA_{rate,60}$	0.50	0.50	0.50	0.50	0.58	0.49	0.33	0.50
$\Delta t_{arrival,30}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{arrival,45}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{arrival,75}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{arrival,90}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50

Table A.6: A-test values for Taxibot coordinator parameters 7 May

	AC_{tt}	AC_{td}	AC_{ts}	AC_{cd}	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$\Delta t_{future,6}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{future,8}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{future,12}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$R_{max,0}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$R_{max,2}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$R_{max,3}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$TA_{rate,20}$	0.50	0.50	0.50	0.50	0.51	0.52	0.48	0.50
$TA_{rate,30}$	0.50	0.50	0.50	0.50	0.51	0.49	0.46	0.50
$TA_{rate,40}$	0.50	0.50	0.50	0.50	0.53	0.52	0.43	0.50
$TA_{rate,45}$	0.50	0.50	0.50	0.50	0.54	0.49	0.39	0.50
$TA_{rate,50}$	0.50	0.50	0.50	0.50	0.54	0.51	0.39	0.50
$TA_{rate,55}$	0.50	0.50	0.50	0.50	0.54	0.53	0.39	0.50
$\Delta t_{arrival,30}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{arrival,45}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{arrival,75}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
$\Delta t_{arrival,90}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50

Table A.7: A-test values for Taxibot coordinator parameters 13 May

	AC_{tt}	AC_{td}	AC_{ts}	AC_{cd}	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$TA_{rate,30}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.53	0.02	0.54	0.96	0.60
$TA_{rate,40}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.53	0.03	0.52	0.96	0.60
$TA_{rate,45}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.54	0.03	0.53	0.96	0.60
$TA_{rate,50}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.54	0.02	0.50	0.97	0.60
$TA_{rate,55}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.53	0.03	0.53	0.97	0.60
$TA_{rate,60}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.53	0.03	0.56	0.97	0.60

Table A.8: A-test values for Taxibot coordinator varying TA_{rate} and $\Delta t_{arrival}$ 7 May

	AC_{tt}	AC_{td}	AC_{ts}	AC_{cd}	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$TA_{rate,30}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.55	0.05	0.51	0.92	0.57
$TA_{rate,40}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.55	0.05	0.51	0.92	0.57
$TA_{rate,45}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.55	0.05	0.53	0.92	0.57
$TA_{rate,50}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.55	0.05	0.52	0.92	0.57
$TA_{rate,55}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.55	0.05	0.52	0.92	0.57
$TA_{rate,60}$, no $\Delta t_{arrival}$	0.50	0.50	0.50	0.55	0.05	0.52	0.91	0.57

Table A.9: A-test values for Taxibot coordinator varying TA_{rate} and $\Delta t_{arrival}$ 13 May

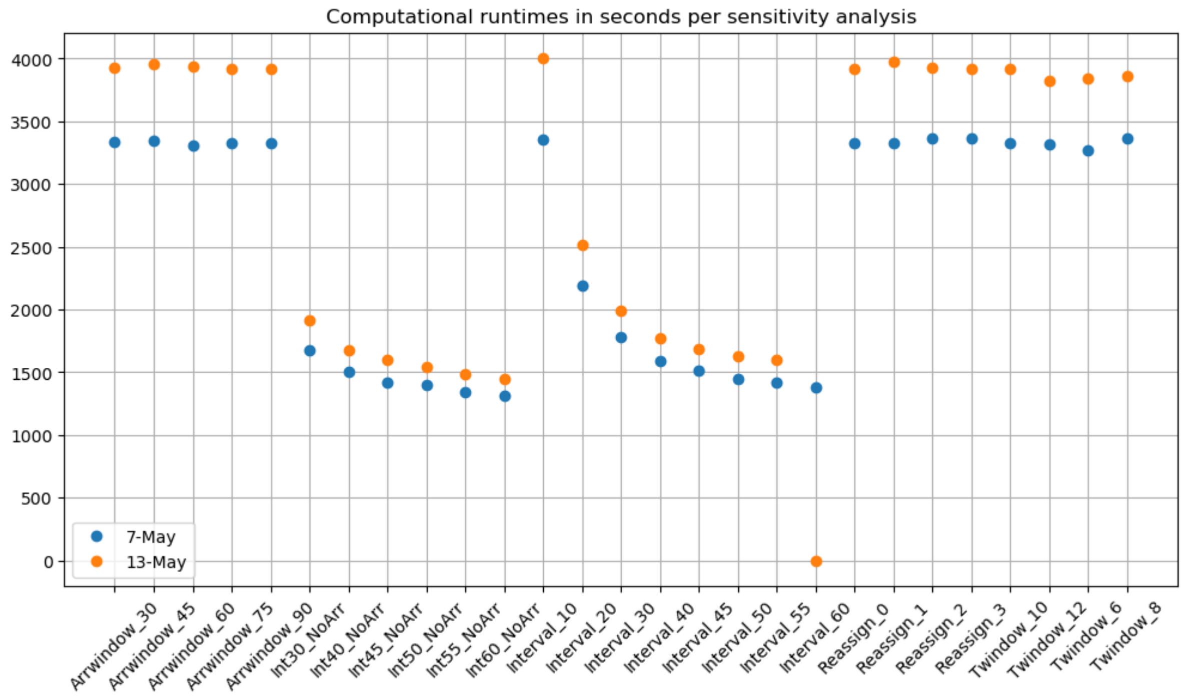


Figure A.40: Model runtimes comparison. Arrwindow indicates $\Delta t_{arrival}$, int and interval indicate TA_{rate}, Reassign indicates R_{max} and Twindow indicates Δt_{future}.

	AC_{tt}	AC_{td}	AC_{ts}	AC_{cd}	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$TA_{active,TB}$ = False	0.50	0.50	0.50	0.49	0.51	0.45	0.27	0.52
$\Delta t_{arrival,None}$	0.50	0.50	0.50	0.54	0.02	0.50	0.96	0.60
$TA_{sorted,list}$ = False	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50

Table A.10: A-test values for Taxibot coordinator model assumptions 7 May

	AC_{tt}	AC_{td}	AC_{ts}	AC_{cd}	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$TA_{active,TB}$ = False	0.50	0.50	0.50	0.47	0.55	0.46	0.22	0.52
$\Delta t_{arrival,None}$	0.50	0.50	0.50	0.55	0.03	0.49	0.93	0.57
$TA_{sorted,list}$ = False	0.50	0.50	0.50	0.50	0.48	0.48	0.49	0.50

Table A.11: A-test values for Taxibot coordinator model assumptions 13 May

A.3.2. Local sensitivity analysis results Taxibot dynamics

	$AC_{tt,A}$	$AC_{tt,D}$	$AC_{tt,all}$	$AC_{td,A}$	$AC_{td,D}$	$AC_{td,all}$	$AC_{ts,A}$	$AC_{ts,D}$	$AC_{ts,all}$	AC_{cd}
$TB_{acc,-20\%}$	0.50	0.50	0.50	0.50	0.50	0.50	0.49	0.50	0.50	0.59
$TB_{acc,+20\%}$	0.50	0.49	0.50	0.50	0.50	0.50	0.50	0.51	0.50	0.45
$TB_{dec,-20\%}$	0.50	0.50	0.50	0.50	0.50	0.50	0.49	0.50	0.50	0.56
$TB_{dec,+20\%}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.47
$TB_{v,max,-20\%}$	0.50	0.52	0.51	0.50	0.49	0.50	0.50	0.48	0.50	0.52
$TB_{v,max,+20\%}$	0.50	0.46	0.49	0.50	0.51	0.50	0.50	0.54	0.51	0.51
$TB_{v,turn,-20\%}$	0.50	0.50	0.50	0.50	0.50	0.50	0.49	0.50	0.50	0.49
$TB_{v,turn,+20\%}$	0.50	0.49	0.50	0.50	0.50	0.50	0.50	0.51	0.50	0.50
$t_{couple,-50\%}$	0.50	0.38	0.46	0.50	0.50	0.50	0.50	0.63	0.53	0.50
$t_{couple,+50\%}$	0.50	0.63	0.55	0.50	0.50	0.50	0.50	0.38	0.47	0.50

Table A.12: A-test values sensitivity Taxibot dynamics 7 May

	$AC_{tt,A}$	$AC_{tt,D}$	$AC_{tt,all}$	$AC_{td,A}$	$AC_{td,D}$	$AC_{td,all}$	$AC_{ts,A}$	$AC_{ts,D}$	$AC_{ts,all}$	AC_{cd}
$TB_{acc,-20\%}$	0.50	0.51	0.50	0.50	0.50	0.50	0.51	0.51	0.51	0.63
$TB_{acc,+20\%}$	0.50	0.49	0.50	0.50	0.50	0.50	0.50	0.51	0.51	0.42
$TB_{dec,-20\%}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.51	0.51	0.60
$TB_{dec,+20\%}$	0.50	0.49	0.50	0.50	0.50	0.50	0.50	0.52	0.51	0.45
$TB_{v,max,-20\%}$	0.50	0.60	0.54	0.50	0.49	0.50	0.49	0.32	0.42	0.56
$TB_{v,max,+20\%}$	0.50	0.46	0.49	0.50	0.51	0.50	0.50	0.60	0.55	0.53
$TB_{v,turn,-20\%}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.51	0.50	0.49
$TB_{v,turn,+20\%}$	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.51	0.50	0.49
$t_{couple,-50\%}$	0.50	0.41	0.47	0.50	0.50	0.50	0.50	0.64	0.56	0.50
$t_{couple,+50\%}$	0.50	0.61	0.54	0.50	0.50	0.50	0.50	0.39	0.46	0.49

Table A.13: A-test values sensitivity Taxibot dynamics 13 May

	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$TB_{acc,-20\%}$	0.42	0.55	0.45	0.50
$TB_{acc,+20\%}$	0.60	0.52	0.49	0.50
$TB_{dec,-20\%}$	0.52	0.54	0.47	0.50
$TB_{dec,+20\%}$	0.56	0.47	0.48	0.50
$TB_{v,max,-20\%}$	0.09	0.59	0.55	0.53
$TB_{v,max,+20\%}$	0.82	0.50	0.50	0.48
$TB_{v,turn,-20\%}$	0.33	0.59	0.51	0.51
$TB_{v,turn,+20\%}$	0.69	0.48	0.42	0.50
$t_{couple,-50\%}$	0.78	0.50	0.46	0.48
$t_{couple,+50\%}$	0.27	0.57	0.48	0.53

Table A.14: A-test values sensitivity Taxibot dynamics 7 May

	TB_{ts}	TB_{util}	$TB_{waiting,gate}$	$\rho_{airport}$
$TB_{acc,-20\%}$	0.44	0.53	0.50	0.50
$TB_{acc,+20\%}$	0.57	0.53	0.49	0.50
$TB_{dec,-20\%}$	0.52	0.52	0.45	0.50
$TB_{dec,+20\%}$	0.55	0.53	0.52	0.50
$TB_{v,max,-20\%}$	0.10	0.54	0.50	0.55
$TB_{v,max,+20\%}$	0.72	0.52	0.47	0.47
$TB_{v,turn,-20\%}$	0.40	0.52	0.53	0.50
$TB_{v,turn,+20\%}$	0.61	0.53	0.43	0.49
$t_{couple,-50\%}$	0.67	0.50	0.45	0.48
$t_{couple,+50\%}$	0.35	0.54	0.52	0.53

Table A.15: A-test values sensitivity Taxibot dynamics 13 May

A.3.3. Interaction plots Taxibot Dynamics

	Baseline value	**Parameter variations**
$TB_{v,max}$	11.83 m/s	8.3, 9.5, 10.7, 11.8, 13.0
TB_{acc}	0.41 m/s^2	0.33, 0.37, 0.41, 0.45, 0.49
Δt_{couple}	60 s	30, 90

Table A.16: Global sensitivity analysis set-up Taxibot dynamics

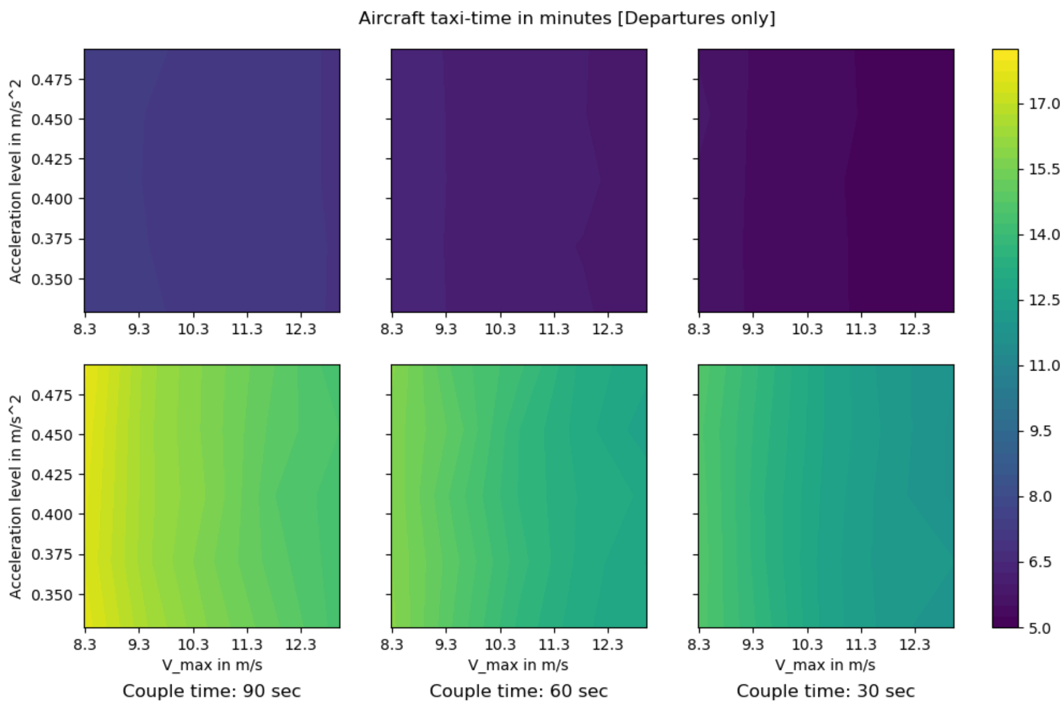


Figure A.41: Contour plot average taxi-time in minutes for 7 May (Top-row) and 13 May (Bottom-row)

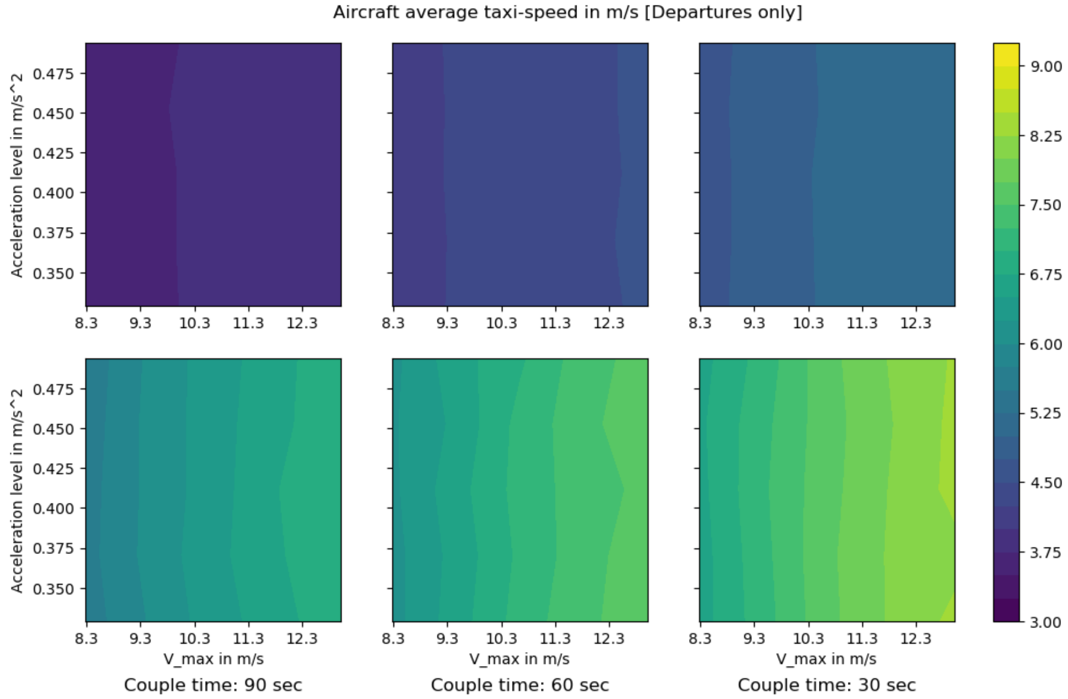


Figure A.42: Contour plot average taxi-speed in minutes for 7 May (Top-row) and 13 May (Bottom-row)

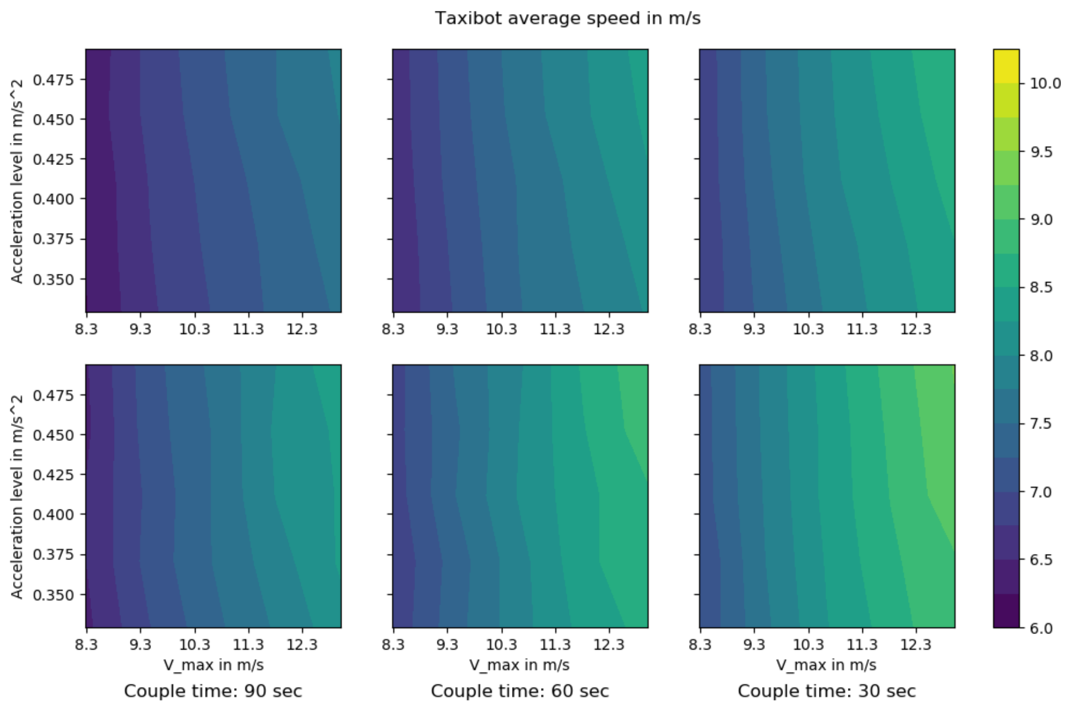


Figure A.43: Contour plot Taxibot average speed in for 7 May (Top-row) and 13 May (Bottom-row)

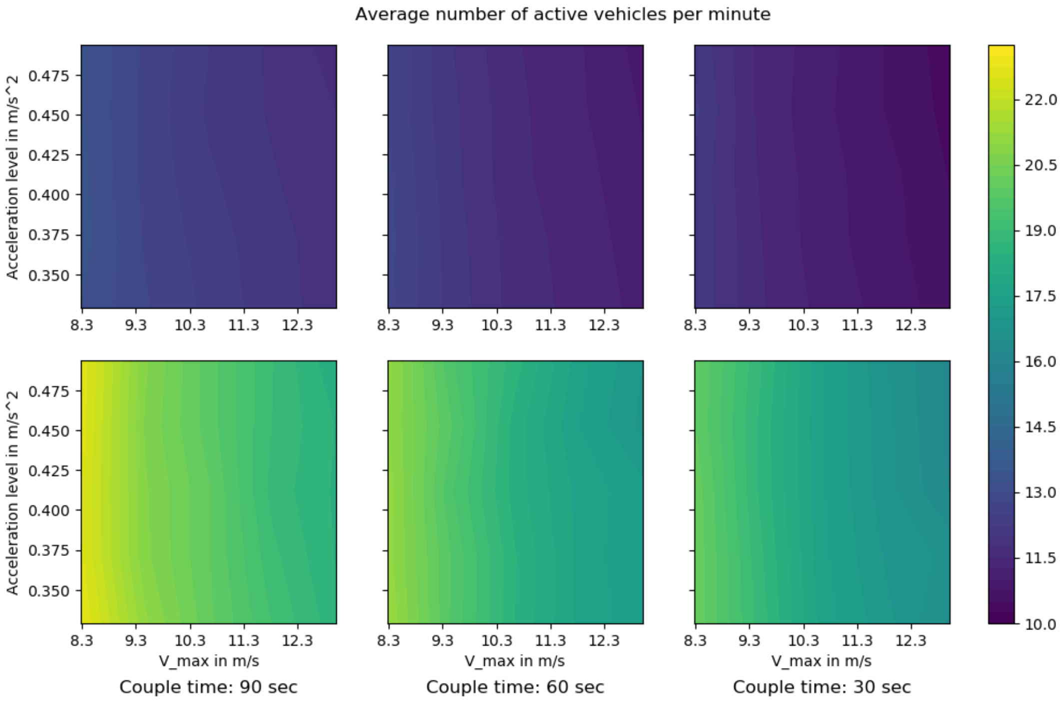


Figure A.44: Contour plot airport density for 7 May (Top-row) and 13 May (Bottom-row)

B
Amsterdam Airport Schiphol Layout

The figures below provide a visual means for any non-aviation related reader to explain the different runway notations at Schiphol Airport. Figure B.1 indicates the six runways with their corresponding Dutch name and runway direction indicated between brackets. Figure B.2 details the AIP Netherlands airport map of Schiphol with all specific details.

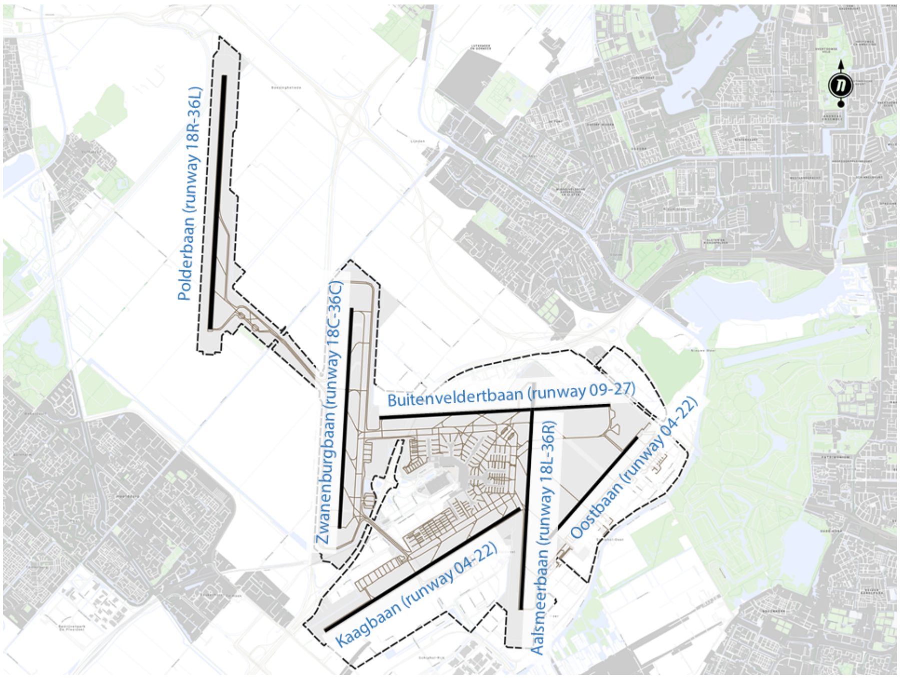


Figure B.1: Runway directions Schiphol Airport [5]

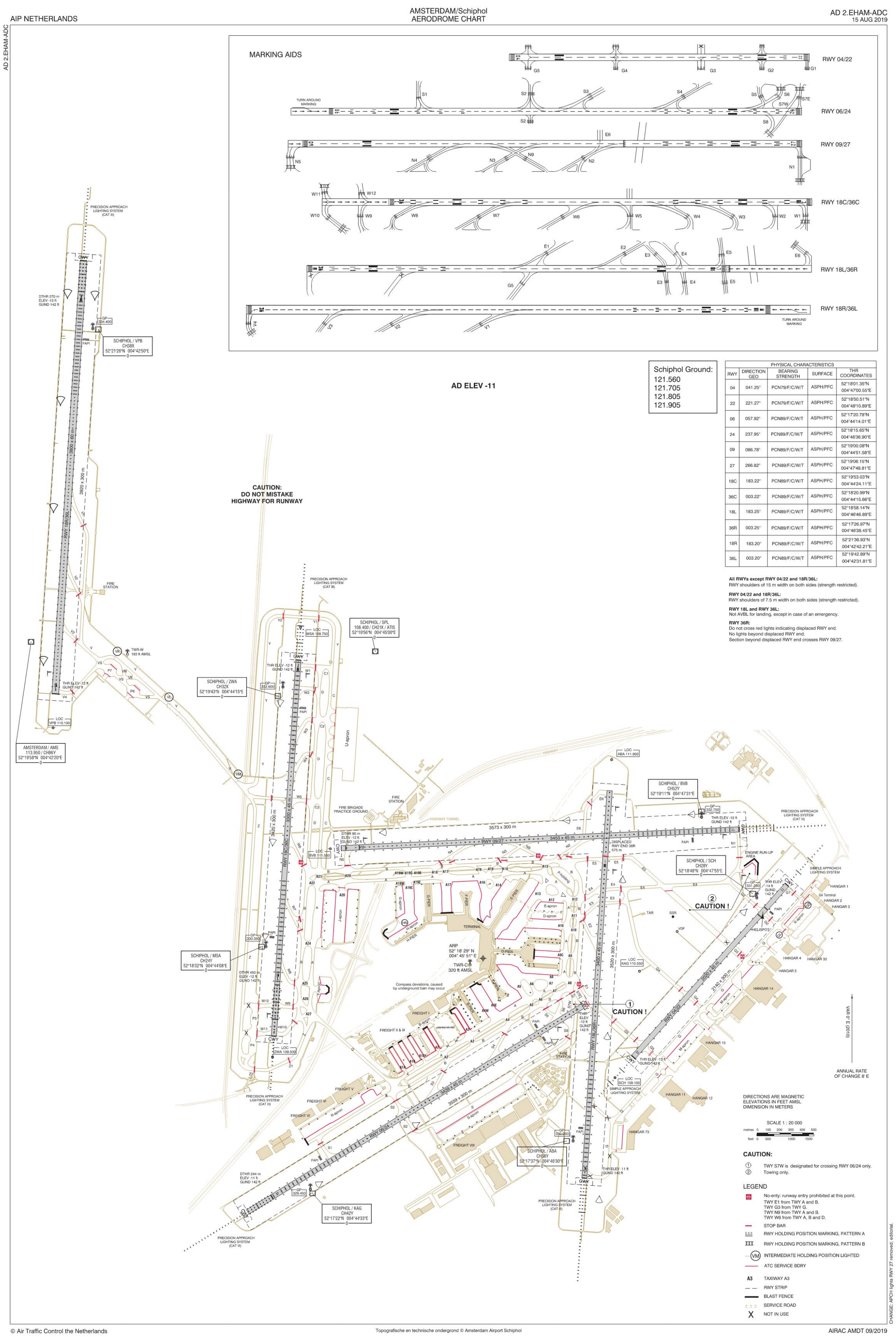


Figure B.2: AIP Netherlands map of infrastructural network Schiphol Airport [2]

C

Interview Frederic van Oost

Frederic van Oost is the Sales Director Europe of Smart Airport Systems, a company co-responsible for building the TaxiBot. A set of questions has been asked on April the 23th 2020, in order to get more insights into the TaxiBot capabilities, as well as verify some of the assumptions that have been made regarding the model. An elaboration of the questions and answers can be found below:

General Questions

- How long does attachment/detachment take for the TaxiBot?
  Attachment and detachment times are somewhat dependent upon the experience with the TaxiBot. However, we see that it takes between 30 seconds to a minute. Currently, the coupling time is around 45 seconds, based upon current experiences.

- For an individual TaxiBot; Are there any values regarding its acceleration/deceleration level?
  These values are not known precisely. When towing an aircraft, the aircraft acceleration/deceleration levels are similar as to conventional engine-on taxiing. As the aircraft-Taxibot combination uses the aircraft brakes for braking, the deceleration procedure is not changed compared to current taxiing operations. For individual TaxiBots it is not precisely known how fast it accelerates/decelerates.

- Are aircraft capable of taxiing through turns with a higher speed due to TaxiBot towing?
  No, the turn speed remains the same. The straight-taxi maximum speed of the TaxiBot (for NB aircraft) is limited to 42 km/h, meaning that aircraft taxi at a slower speed on straights.

Questions related to energy consumption

- Does the TaxiBot need recharging? And if yes, how long does recharging take?
  The electronics of the TaxiBot are powered by two Diesel generators. Everything is present in two-fold, to have redundancy in case of a system failure. Therefore, battery recharging is not needed and the TaxiBot only need to be filled with Diesel. Recharging is not necessary, the TaxiBot can operate without the need for Refill. The goal for Smart Airport Systems for the future is to move to a fully electric TaxiBot. However, the current state of battery capacity does not allow this yet, as current battery capacity would only guarantee a maximum of 1 hour of operation before recharging will be needed. The goal is to have a fully electric TaxiBot within the next few years (1-3 years). The full electric TaxiBot must allow a full day of operations without the need to recharge. Another thing that is being researched is the ability to fast-charge the TaxiBot, for quick recharging in between operations.

- How much energy/diesel does a TaxiBot consume when carrying out operations?
  At full operational load, the TaxiBot consumes around 0.5 liters of Diesel per minute. It has a tank capacity of 600 liters, meaning that a non-stop 20 hours of operations can be achieved. 1 kg of fuel emits 3.24 kg of CO2, however, it must be noted that the Aircraft APU is ON whenever the TaxiBot is attached. The APU, logically, also emits, which should be taken into account when comparing fuel/emission savings with conventional taxiing.

D
Model Architecture

This Section elaborates upon the created simulation model in Python. It gives a short description of the Python files used in the (basic) model environment and how they are structured within the model. Each file as depicted in the Figure below is briefly discussed next.

Run_me.py
The run_me file is the main file in the simulator and through this file, the simulation parameters can be set. The run_me file initialises the simulator.py file to start the simulation.

Simulator.py
The simulator.py file is initialised by the run_me file with a set of simulation parameters and constants. These parameters and constants are first structured within the program and decide for which input files have to be loaded in (i.e. flight- and runway schedules). The Simulator.py file connects all other py files and consists of the model iteration loop, through which all objects are updated.

data_import.py
The simulation parameters from the run_me file decide for which import data is to be used in the simulator.py file. The data_import.py file determines the excel input files used in the simulator. Other data-files, not included in this py file, determine the parameters for aircraft and Taxibot agents and are provided as separate .txt files.

ATC_class.py
The ATC_class file is both called within the simulator environment and is a grandfather file for the inheritance of all ATC agents. The ATC_class initialises all ATC Agents within the simulator environment.

Graph_structure.py
The Graph_structure file details the creation and all adjustments to the graph representation of the Agent-Based simulator environment. It initialises the graph environment and through each iteration performs the necessary adjustments and modifications as initiated by ATC agents.

Fleet.py
The Fleet.py file brings together all updates and actions for the ATC, Taxibot and aircraft agents. Through each iteration in the simulator, the internal states and actions for all underlying agent types are updated.

ATC_intersection_class.py
The ATC_intersection_class file details the ATC intersection agent description. It consists of all decision making and is updated each simulator iteration through the Fleet file.

ATC_stopbar_class.py
This file contains the agent description and specifications for ATC stopbar agents. These agents are located at the last node up to a runway crossing.

ATC_endpoint_class.py
This file contains the agent description and specifications for ATC endpoint agents. These agents are located at the gates and the runway points on the airport.

ATC_service_intersection.py
This file consists of a few relatively simple properties to keep track of vehicle movements on the service-roads (i.e. for Taxibots). The file is necessary to manage the autonomous Taxibot agents' locations.

parking_node_agent.py
Denotes the parking facilities for Taxibot agents and ensures a safe release and return of Taxibots near the parking facilities. These agents are commanded by the Taxibot coordinator agent.

towing_truck_class.py
This file contains the internal model and specifications for Taxibot agents.

aircraft_class.py
This file contains the internal model and specifications for Aircraft agents.

Map.py
This file details the Pygame visualisation of the model environment. If the model is to be run without visualisation, this file is not called.

Airport_ops_status.py
This file details the airport operational mode information system and can be consulted by ATC agents. This file determines the runway mode of operations for the next 15 minutes and keeps track of all past RMOs.

Sink/Source_agent.py
These files denote the sink and source, or entry and exit, agents within the model environment. They are responsible for the creation and removal of aircraft agents within the simulator.

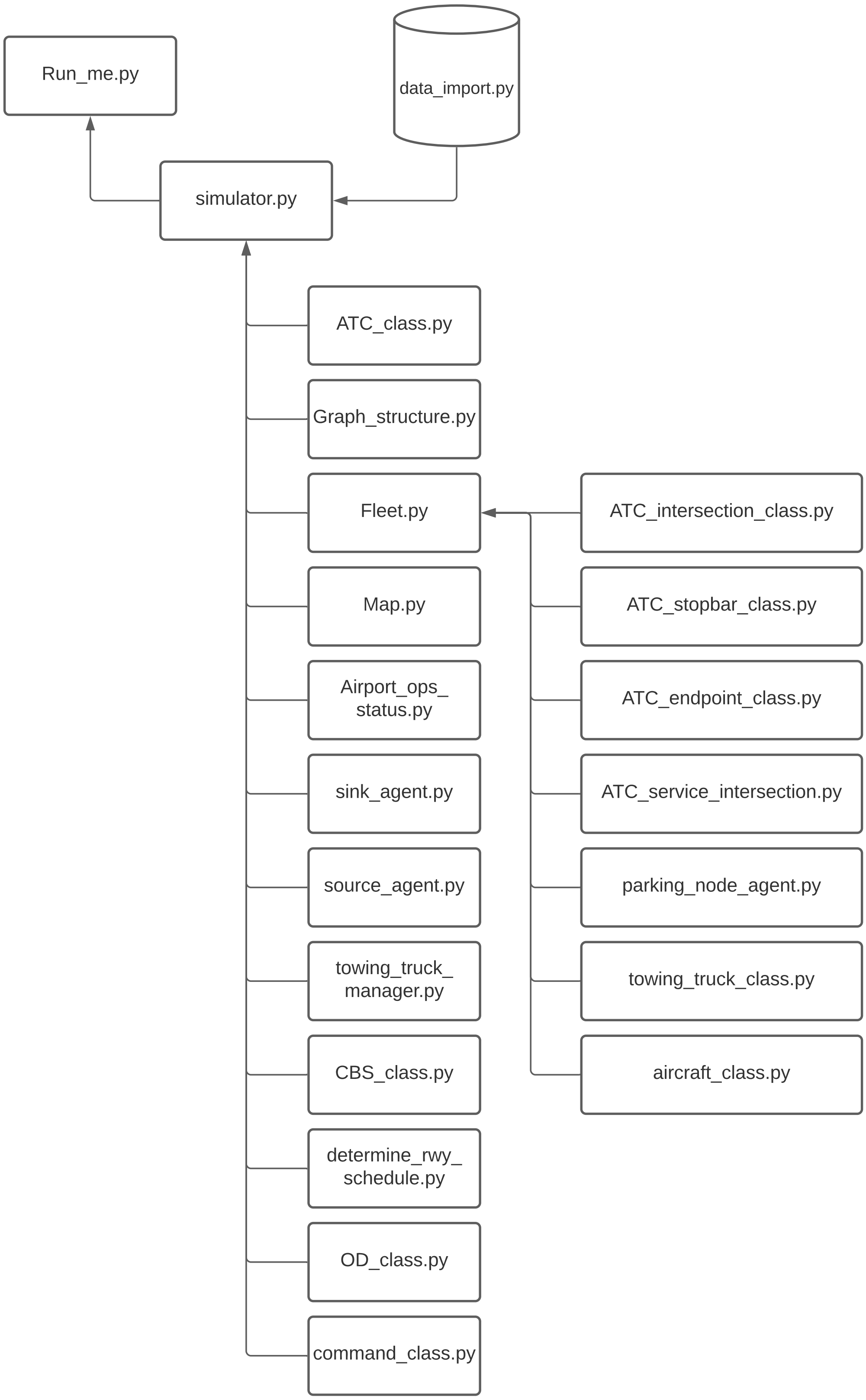
towing_truck_manager.py
This file details the internal model and all related specifications of the Taxibot coordinator agent.

CBS_class.py
This supportive file runs the CBS algorithm when called by ATC agents. It returns a list of information on all potential conflicts within the simulator environment and allows for conflict solving by ATC intersection agents.

determine_rwy_schedule.py
This file is supportive of the model and creates an excel file of the RMOs throughout the day of simulations. This speeds up the simulator, as the airport_ops_agent can directly obtain the active RMO per time-point.

OD_class.py
Each gate and runway point is also initialised by the OD class. The OD class contains a schedule of Aircraft releases and ensures an occupancy toggle when an Aircraft arrives or leaves the location of the OD agent.

command_class.py
This file contains a class description of commands that can be sent by ATC agents. Due to the Object-Oriented Programming nature of the model, it is created as a single file to clarify the set-up of the commands used by ATC agents.


Run_me.py
data_import.py
simulator.py
ATC_class.py
Graph_structure.py
Fleet.py
Map.py
Airport_ops_
status.py
sink_agent.py
source_agent.py
towing_truck_
manager.py
CBS_class.py
determine_rwy_
schedule.py
OD_class.py
command_class.py
ATC_intersection_class.py
ATC_stopbar_class.py
ATC_endpoint_class.py
ATC_service_intersection.py
parking_node_agent.py
towing_truck_class.py
aircraft_class.py

E
Recommendations for Future Work

The recommendations for future work on the topic of operational towing of aircraft within an airport ground surface operation are:

Inclusion of Apron Operations — The current model implementation excludes apron operations. Inclusion of the apron operations would allow for a better comparison of the novel taxi-concept timeline with current taxi-operations. Besides, the inclusion of apron operations would further increase the realism of the model compared to real-world operations.

Additional service-roads — The model could be expanded by implementing more service-roads to accommodate Taxibot movements. In that case, specifically, the implementation of the intermediate service-road in between the Alfa and Bravo taxiways should be considered. This way, the Taxibot detachment points for departures from runway 24 and intersection starts at 18L, as proposed by AAS experts, could be accurately simulated and tested. Also, expansion of the model could consider the inclusion of runway 09/27 in operational towing. Furthermore, different return routes for Taxibots could be tested when more service-roads are included.

Decentralisation or Expansion of Task Allocation Algorithm — The current Task Allocation algorithm follows a centralised approach and, therefore, task allocation is dependent upon a single piece of software. The algorithm should be further improved to be more robust, either by creating a sort of back-up algorithm or creating a decentralised solution.

Implementation of ATC agent coordination on short segments — The current model implementation ensures a conflict-free operation through the CBS algorithm. However, it was found that for very short segments the model has difficulties to ensure safe operations. Some sort of coordination algorithm could be considered in which ATC agents that are within a close distance to each other can coordinate future passings. This way, Aircraft- or Taxibot agents can be commanded far before they reach such a short segment, to maintain a safe ground surface operation.

Operational Towing of Inbound Flights — This research has only considered outbound flights for operational towing. However, it can be expected that significant benefits could be achieved by towing inbound flights. It is recommended to only consider inbound flights for operational towing that exceed a specific taxi-time, i.e. for which significant reductions in pollutants and fuel-burn can be achieved, to limit operational complexity. The combination with remote parking locations as elaborated below could provide an initial direction for consideration of operational towing of inbound flights.

Remote Parking Locations for Taxibots — The model does not take into account the possibility for Taxibot agents to wait for longer times at specific locations within the airport environment. However, research considers the implementation of waiting spots at the end of an agent's task, such that an agent waits at this specific location for its next assignment [112]. Such locations would possibly require infrastructural modifications to the airport layout and, therefore, the location of these points could be investigated. The use of remote parking locations could significantly lower the individual travel distances for Taxibots and amount of vehicle movements at critical crossings. Besides, it can ensure that a Taxibot is already parked near its next assignment to prevent the aircraft from waiting for the Taxibot. Remote parking spots are expected to be necessary for consideration of inbound towing.

Implementation of Runway Scheduling — The model assumes that an outbound aircraft is released from its gate as close to its spawn-time as possible. Runway scheduling must be realised through speed commands during an aircraft's taxi operation, as gate holding is excluded. Implementation of the apron operations could allow for gate-holding and consequently could mean that runway scheduling could be implemented. From this schedule, more optimal release times of aircraft could be determined that would potentially also benefit the airport taxi operations.

Improving the Forward Simulation Algorithm — The CBS implementation uses a relatively simple forward simulation algorithm to determine the time-passings at each future node of a vehicle's route. A more sophisticated algorithm could be implemented, as less accurate predictions cause unnecessary speed reductions and thus less efficient ground operations.

Different Aircraft and Taxibot types — could be implemented to investigate the feasibility of Taxibot towing for different aircraft types. It is expected that for lower maximum taxi speeds, such as for wide-body aircraft, the Taxibot implementation has less impact on conventional taxi procedures. The Agent-Based model can be used to investigate the environmental and financial impact of the implementation of the Taxibot, for which different vehicle types can be taken into account.

Different Traffic Scenarios and More Realistic Flight Schedules — The model uses four input days of operations with 800-900 flights per day. Compared to pre-COVID 19 flight numbers, this is only 60-70% of the average number of flights at AAS. Therefore, more flights and different traffic scenarios should be investigated to further test the feasibility of the towing concept.

Consideration of Other Airports — This thesis focused on the design and evaluation of a novel taxi-concept for AAS specific. However, due to the general modelling of agent types, other airport layouts could be considered to test the feasibility of operational towing. The focus area should be on airports with significant aircraft taxi movements and (relatively) long taxi-times.

IV

Reference list

Bibliography

[1] Schiphol. https://www.schiphol.nl/en/.

[2] LVNL. Aeronautical information publication netherlands, September 2019. Obtained on 14/10/2019 from https://www.lvnl.nl/eaip/website_EFF_10OCT2019/2019-09-12-AIRAC/html/index-en-GB.html.

[3] Airportspotting. Spotting at amsterdam schiphol - views inside the terminal, 2019. Obtained on 27/11/2019 from https://www.airportspotting.com/spotting-amsterdam-schiphol-terminal/.

[4] Schiphol airport Group. Gebruiksprognose 2019. -, 2019.

[5] Schiphol 24. Runways, 2019. Retrieved on 27/11/2019 from https://www.schiphol24.nl/wp-content/uploads/2016/04/banen.jpg.

[6] Zhihua Liao Yu Jiang and Honghai Zhang. A collaborative optimization model for ground taxi based on aircraft priority. 2013.

[7] AAS. Schiphol Airport CDM Operations Manual. Technical Report September, 2019.

[8] T.E.H. Noortman. "Agent-Based Modelling of an Airport's Ground Surface Movement Operation", 25-5-2018. Understanding the principles and mechanisms of decentralised control.

[9] SESAR. OFA04.02.01 (Integrated Surface Management) Final SPR. Technical report, 2016.

[10] Eurocontrol. *EUROCONTROL Specification for Advanced-Surface Movement Guidance and Control System (A-SMGCS) Services Specification for Movement Guidance and Control System.* 2018.

[11] M. Lukic, A. Hebala, P. Giangrande, C. Klumpner, S. Nuzzo, G. Chen, C. Gerada, C. Eastwick, and M. Galea. State of the Art of Electric Taxiing Systems. *2018 IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles and International Transportation Electrification Conference, ESARS-ITEC 2018*, (November), 2019.

[12] Michael Schier, Frank Rinderknecht, Andreas Brinner, and Heribert Hellstern. High Integrated Electric Machine for Aircraft Autonomous Taxiing. *International Conference on Electric Vehicles and Renewable Energies (EVER 11)*, (March 2011), 2011.

[13] Freightquip. Conventional aircraft tractors.

[14] Alain Clapaud. Air france to evaluate taxibot, the robotic aircraft tractor, November 2014.

[15] K. K.H. Ng, C. K.M. Lee, Felix T.S. Chan, and Yaqiong Lv. Review on meta-heuristics approaches for airside operation research. *Applied Soft Computing Journal*, 66:104–133, 2018.

[16] Eurocontrol. Performance Review Report (PRR) 2018 | EUROCONTROL. -(May), 2019.

[17] Actieplan Luchtvaart. Slim én duurzaam. Technical report, 2018.

[18] Frank Bussink, Max Mulder, and Marinus M Van Paassen. Evaluation of an Autonomous Taxi Solution for Airport Operations during Low Visibility Conditions. (September 2014), 2011.

[19] International Airport Review. The top 20 largest airports in the world by passenger number, April 2019. Retrieved on 14/10/2019 from https://www.internationalairportreview.com/article/32311/top-20-largest-airports-world-passenger-number/.

[20] Info Schiphol. Retrieved on 14/10/2019 from http://www.infoschiphol.nl/home/geschiedenis_schiphol.html.

[21] International Civil Aviation Organization. *Aerodrome Design Manual*, fourth edition, 2005. DOC 9157. Approved by the Secretary General and published under his authority.

[22] Schiphol Group. Kerngegevens 2018, Maart 2019. Retrieved on 14/10/2019 from `https://www.schiphol.nl/nl/schiphol-group/pagina/feiten-en-cijfers/`.

[23] K. Fines. "Decentralized Control for Resilient Airport Surface Movement Operations", 2019.

[24] Michael Schultz and Hartmut Fricke. Improving aircraft turnaround reliability. June 2008.

[25] Dutch Safety Board. Take-off from Taxiway Amsterdam Airport Schiphol. 2011.

[26] Schiphol group. Completing schiphol's dual taxiway system, 2019. Obtained on 15/10/2019 from `https://www.schiphol.nl/en/projects/page/completing-schiphols-dual-taxiway-system/`.

[27] Eurocontrol. Amsterdam schiphol airport adopts a-cdm, March 2018. Obtained on 15/10/2019 from `https://www.eurocontrol.int/news/amsterdam-schiphol-airport-adopts-cdm`.

[28] EUROCONTROL. Airport CDM Implementation THE MANUAL EUROCONTROL. (March):363, 2017.

[29] G. Baxter and I. Sommerville. Socio-technical systems: From design methods to systems engineering. *Interacting with Computers*, 23(1):4–17, 2011.

[30] Amadeo Odoni Peter Belobaba and Cynthia Barnhart. *The global airline industry*. John Wiley & Sons, Ltd., first edition, 2016.

[31] Aaron Flodin. Transponder and squawk codes, 2019. Obtained on 17/10/2019 from `https://www.vatsim.net/pilot-resource-centre/general-lessons/transponder-and-squawk-codes`.

[32] IVAO Nederland. Verkeersleiders, 2019. Obtained on 17/10/2019 from `https://nl.ivao.aero/training/atc/posities/del`.

[33] Live atc, 2019. `https://www.liveatc.net/search/?icao=eham`.

[34] Eurocontrol. Network operations, 2019. Obtained on 17/10/2019 from `https://www.eurocontrol.int/network-operations#about`.

[35] Eurocontrol ATM Lexicon. Flight progress strip, July 2016. Accessed on 18/10/2019 on `https://ext.eurocontrol.int/lexicon/index.php/Flight_progress_strip`.

[36] Peter Brooker. Sesar and nextgen: Investing in new paradigms. *Journal of Navigation*, 61(2):195–208, 2008.

[37] Karsten Straube, Marcus Roßbach, Björn D. Vieten, and Kerstin Hahn. Follow-the-greens: The controllers' point of view results from a sesar real time simulation with controllers. In Neville A. Stanton, Steven Landry, Giuseppe Di Bucchianico, and Andrea Vallicelli, editors, *Advances in Human Aspects of Transportation*, pages 837–849, Cham, 2017. Springer International Publishing.

[38] SESAR. Release 5 SESAR Solution #47 Guidance assistance through airfield ground lighting. Technical report, 2016.

[39] ATRiCS. Follow the greens, 2017. Accessed on 21/10/2019 on `http://followthegreens.com/`.

[40] SESAR European Airport Consortium. Follow-the-greens: The future of guidance is green, 2017. First SEAC Validation.

[41] Schiphol. Veiligheid in de nederlandse luchtvaartsector. Obtained on 23/10/2019 from `https://safetyroadmap.stage-env.com/nl/`.

[42] ICAO. Doc 9830 - Advanced Surface Movement Guidance and Control Systems (A-SMGCS) Manual. Technical report, 2004.

[43] Eurocontrol. Technical Specification of SESAR 2020 Exploratory Research Call. Technical report, 2019.

[44] Nikolai Okuniek and Dustin Beckmann. Towards higher level of A-SMGCS: Handshake of electric taxi and trajectory-based taxi operations. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2017-September, 2017.

[45] Rui Guo, Yu Zhang, and Qing Wang. Comparison of emerging ground propulsion systems for electrified aircraft taxi operations. *Transportation Research Part C: Emerging Technologies*, 44:98–109, 2014.

[46] WheelTug, 2019. Driving Aerospace.

[47] Yann Nicolas. etaxi - taxiing aircraft with engines stopped. Obtained on 28/10/2019 from `http://www.smartcockpit.com/docs/taxiing-aircraft-with-engines-stopped.pdf`.

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

[49] R.Archer et al. R. Morris, M. Chang. Self-Driving Aircraft Towing Vehicles : A Preliminary Report The Case for Autonomous Towing. *Artificial Intelligence for Transportation: Advice, Interactivity and Actor Modeling*, pages 41–48, 2015.

[50] Taxibot, 2013. The world's only certified and operational taxiing alternative.

[51] Maria Nadia Postorino, Luca Mantecchini, and Ettore Gualandi. Integration between aircraft and handling vehicles during taxiing procedures to improve airport sustainability. *International Journal of Transport Development and Integration*, 1(1):28–42, 2017.

[52] Maximilian T.E. Heinrich, Fabian Kelch, Pierre Magne, and Ali Emadi. Investigation of regenerative braking on the energy consumption of an electric taxiing system for a single aisle midsize aircraft. *IECON Proceedings (Industrial Electronics Conference)*, pages 3023–3029, 2014.

[53] A. Lau N. Dzikus, J. Fuchte and V. Gollnick. Potential for fuel reduction through electric taxiing. *11th AIAA Aviation Technology, Integration,and Operations (ATIO) Conference, including the AIAA Balloon Systems Conference and 19th AIAA Lighter-Than-Air Technology Conference*, (September):1–9, 2011.

[54] Iniobong Edem, Owunna Ikechukwu, Ikpe Aniekan E, and Ememobong E Ikpe. The Future of Conventional Aircraft Ground Propulsion Systems in Relation to Fuel Consumption and CO 2 Emission. 13(2):91–100, 2016.

[55] Fabrizio Re. Viability and State of the Art of Environmentally Friendly Aircraft Taxiing Systems. *2012 Electrical Systems for Aircraft, Railway and Ship Propulsion*, pages 1–6, 2012.

[56] Pjotr Sillekens Paul Roling and Richard Curran. The effects of electric Taxi systems on Airport Surface Congestion. (June):1–10, 2015.

[57] Zarrin K Chua. Self-Managing Conflict Resolution for Autonomous Taxiing Tugs : An Initial Survey. *SESARWPE*, (December):1–8, 2015.

[58] Zarrin Chua, Mathieu Cousy, Mickaël Causse, and François Lancelot. Initial assessment of the impact of Modern Taxiing techniques on airport ground control. *Proceedings of the International Conference on Human-Computer Interaction in Aerospace, HCI-Aero 2016*, 2016.

[59] Laura Khammash, Luca Mantecchini, and Vasco Reis. Micro-simulation of airport taxiing procedures to improve operation sustainability: Application of semi-robotic towing tractor. *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017 - Proceedings*, pages 616–621, 2017.

[60] Giuseppe Sirigu, Mario Cassaro, Manuela Battipede, and Piero Gili. A Route Selection Problem Applied to Auto-Piloted Aircraft Tugs. *Wseas Transactions on Electronics*, 8(-):27–40, 2017.

[61] E.V.M van Baaren. The feasibility of a fully electric aircraft towing system. *TU Delft repository*, 2019.

[62] N J F P Guillaume. Finding the viability of us-ing an automated guided vehicle taxiing system for aircraft. *TU Delft repository*, 2018.

[63] Gerald J. Lieberman Frederick S. Hillier. *Introduction to Operations Research*. McGraw-Hill Education, 10th edition, 2015.

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

[65] J. A. D. Atkin, E K Burke, and S Ravizza. The airport ground movement problem: Past and current research and future directions. *Research in Air Transportation*, pages 131–138, 2010.

[66] Ángel G. Marín. Airport management: Taxi planning. *Annuals of Operations Research*, 143(1):191–202, 2006.

[67] Angel Marín and Esteve Codina. Network design: Taxi planning. *Annuals of Operations Research*, 157(1):135–151, 2008.

[68] Gillian Clare and Arthur G. Richards. Optimization of taxiway routing and runway scheduling. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1000–1013, 2011.

[69] J. W. Smeltink and M. J. Soomer. An optimisation model for airport taxi scheduling, 2004.

[70] Sivakumar Rathinam, Justin Montoya, and Yoon Jung. An optimization model for reducing aircraft taxi times at the Dallas Fort Worth International Airport. *ICAS Secretariat - 26th Congress of International Council of the Aeronautical Sciences 2008, ICAS 2008*, 1:3470–3483, 2008.

[71] J. Guépet, O. Briant, J. P. Gayon, and R. Acuna-Agost. The aircraft ground routing problem: Analysis of industry punctuality indicators in a sustainable perspective. *European Journal of Operational Research*, 248(3):827–839, 2016.

[72] Hamsa Balakrishnan and Jung Yoon. A framework for coordinated surface operations planning at dallas-fort worth international airport. *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference 2007*, 3:2382–2400, 2007.

[73] Paul C. Roling. Airport surface traffic planning optimization: A case study of Amsterdam Airport Schiphol. *9th AIAA Aviation Technology, Integration and Operations (ATIO) Conference, Aircraft Noise and Emissions Reduction Symposium (ANERS)*, (September):1–7, 2009.

[74] Hanbong Lee and Hamsa Balakrishnan. A comparison of two optimization approaches for airport taxiway and runway scheduling. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, pages 4E1–1–4E1–15, 2012.

[75] X. Zou, P. Cheng, W. D. Liu, N. Cheng, and J. P. Zhang. A two-stage taxi scheduling strategy at airports with multiple independent runways. *Transportation Research Part C: Emerging Technologies*, 95(December 2017):165–184, 2018.

[76] H. G. Visser and P. C. Roling. Optimal airport surface traffic planning using mixed integer linear programming. *AIAA's 3rd Annual Aviation Technology, Integration, and Operations (ATIO) Forum*, 2008, 2008.

[77] Jesús García, Antonio Berlanga, José M. Molina, and José R. Casar. Optimization of airport ground operations integrating genetic and dynamic flow management algorithms. *AI Communications*, 18(2):143–164, 2005.

[78] Qing Liu, Chunling Li, and Xianfei Luo. Airport taxi scheduling strategy based on cooperate particle swarm optimization algorithm. *International Journal of Advancements in Computing Technology*, 3(9):165–172, 2011.

[79] Changyou Liu and Yongcheng Wang. Aircraft taxi path optimization based on ant colony algorithm. *Proceedings - 2011 4th International Symposium on Computational Intelligence and Design, ISCID 2011*, 2(2):226–229, 2011.

[80] New optimizer using particle swarm theory. *Proceedings of the International Symposium on Micro Machine and Human Science*, pages 39–43, 1995.

[81] Qing Liu, Tongshui Wu, and Xianfei Luo. A space-time network model based on improved genetic algorithm for airport taxiing scheduling problems. *Procedia Engineering*, 15:1082–1087, 2011.

[82] Jean Baptiste Gotteland and Nicolas Durand. Genetic algorithms applied to airport ground traffic optimization. *2003 Congress on Evolutionary Computation, CEC 2003 - Proceedings*, 1:544–551, 2003.

[83] Michal Weiszer, Jun Chen, and Giorgio Locatelli. An integrated optimisation approach to airport ground operations to foster sustainability in the aviation sector. *Applied Energy*, 157:567–582, 2015.

[84] Changyou Liu and Kaifeng Guo. Airport taxi scheduling optimization based on genetic algorithm. *Proceedings - 2010 International Conference on Computational Intelligence and Security, CIS 2010*, 300300(4):205–208, 2010.

[85] Ingrid Gerdes and Annette Temme. Taxi routing for aircraft: Creation and Controlling: Ground movements with time constraints. *SIDs 2012 - Proceedings of the SESAR Innovation Days*, (November), 2012.

[86] B. Pesic, N. Durand, and J.-M. Alliot. Real-World Applications Aircraft Ground Traffic Optimisation using a Genetic Algorithm. *GECCO conference*, pages 1–8, 2001.

[87] Joseph L. Balintfy, William J. Duffy, and Sinha Prabhakant. An Appraisal of Some Shortest-Path Algorithms. *Operations Research*, 22(4):711–727, 1974.

[88] Yassine Dabachine, Belaid Bouikhalene, and Abdessamad Balouki. Bidirectional Search Algorithm for Airport Ground Movement. *ACIT 2018 - 19th International Arab Conference on Information Technology*, pages 1–9, 2019.

[89] Charles Lesire. An iterative A* algorithm for planning of airport ground movements. *Frontiers in Artificial Intelligence and Applications*, 215(December):413–418, 2010.

[90] Stefan Ravizza, Jun Chen, Jason A.D. Atkin, Edmund K. Burke, and Paul Stewart. The trade-off between taxi time and fuel consumption in airport ground movement. *Public Transport*, 5(1-2):25–40, 2013.

[91] Nan Li, Qing Zhao, and Xiao Hao Xu. Research of taxing optimization for aircraft. *Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering, TMEE 2011*, pages 25–28, 2011.

[92] Stefan Ravizza, Jason A.D. Atkin, and Edmund K. Burke. A more realistic approach for airport ground movement optimisation with stand holding. *Journal of Scheduling*, 17(5):507–520, 2014.

[93] Richard Jordan, Mariya A. Ishutkina, and Tom G. Reynolds. A statistical learning approach to the modeling of aircraft taxi time. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, pages 1.B.1–1–1.B.1–10, 2010.

[94] S. Ravizza, J. A.D. Atkin, M. H. Maathuis, and E. K. Burke. A combined statistical approach and ground movement model for improving taxi time estimations at airports. *Journal of the Operational Research Society*, 64(9):1347–1360, 2013.

[95] Amal Srivastava. Improving departure taxi time predictions using ASDE-X surveillance data. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, pages 2B5–1–2B5–14, 2011.

[96] Hanbong Lee, Waqar Malik, and Yoon C. Jung. Taxi-out time prediction for departures at charlotte airport using machine learning techniques. *16th AIAA Aviation Technology, Integration, and Operations Conference*, (June 2016):1–11, 2016.

[97] Guan Lian, Yaping Zhang, Jitamitra Desai, Zhiwei Xing, and Xiao Luo. Predicting taxi-out time at congested airports with optimization-based support vector regression methods. *Mathematical Problems in Engineering*, 2018, 2018.

[98] Husni Idris, John-Paul Clarke, Rani Bhuva, and Laura Kang. Queuing Model for Taxi-Out Time Estimation. *Air Traffic Control Quarterly*, 10(1):1–22, 2002.

[99] Franziska Klügl Ana L. C. Bazzan. A review on agent-based technology for traffic and transportation. 29(May 2013):375–403, 2019.

[100] Alison J.J. Heppenstall, Andrew T. Crooks, Linda M. See, and Michael Batty. Introduction to Agent-Based Modelling. *Agent-Based Models of Geographical Systems*, (January):1–759, 2012.

[101] Yang Peng, Gao Wei, Sun Junqing, and Sun Bin. Evaluation of Airport Capacity through Agent Based Simulation. *International Journal of Grid and Distributed Computing*, 7(6):165–174, 2014.

[102] C. V. Rafegas. Taxiway Management Through Multi Agent System. pages 1–81, 2013.

[103] Xinping Zhu, Xulin Duan, Xinmin Tang, and Songchen Han. Agent-oriented simulation and evaluation of aerodrome airside. *2009 IEEE International Conference on Grey Systems and Intelligent Services (GSIS 2009)*, pages 1715–1719, 2009.

[104] Fu Chen. Aircraft taxiing route planning based on multi-agent system. *Proceedings of 2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference, IMCEC 2016*, pages 1421–1425, 2017.

[105] Poornima Balakrishna, Rajesh Ganesan, Lance Sherry, and Benjamin S. Levy. Estimating Taxi-out times with a Reinforcement Learning algorithm. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, pages 1–12, 2008.

[106] Poomima Balakrishna, Rajesh Ganesan, and Lance Sherry. Airport taxi-out prediction using approximate dynamic programming: Intelligence-based paradigm. *Transportation Research Record*, (2052):54–61, 2008.

[107] Rajesh Ganesan Poornima Balakrishna and Lance Sherry. Accuracy of reinforcement learning algorithms for predicting aircraft taxi-out times: A case-study of Tampa Bay departures. *Transportation Research Part C: Emerging Technologies*, 18(6):950–962, 2010.

[108] Elizabeth George and Shamsuddin S. Khan. Reinforcement learning for taxi-out time prediction: An improved Q-learning approach. *2015 International Conference on Computing and Network Communications, CoCoNet 2015*, pages 757–764, 2016.

[109] Robert Morris, S P Corina, Kasper Luckow, Waqar Malik, Hang Ma, T K Satish Kumar, and Sven Koenig. Planning , Scheduling and Monitoring for Airport Surface Operations. pages 608–614, 2016.

[110] Ioannis Simaiakis, Melanie Sandberg, and Hamsa Balakrishnan. Dynamic control of airport departures: Algorithm development and field evaluation. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):285–295, 2014.

[111] Sebastian Frank, Per Martin Schachtebeck, and Peter Hecker. Sensor concept for highly-automated airport tugs for reduced emisson taxi operations. *30th Congress of the International Council of the Aeronautical Sciences, ICAS 2016*, pages 1–9, 2016.

[112] Ji Ma, Daniel Delahaye, Mohammed Sbihi, Paolo Scala, and M Mujica Mota. A Study of Tradeoffs in Airport Coordinated Surface Operations. *EIWAC 2017, 5th ENRI international workshop on ATM/CNS*, 2017.