Agent-Based Distributed Planning and Coordination for Resilient Airport Surface Movement Operations

Master of Science Thesis

K.Fines

Agent-Based Distributed Planning and Coordination for Resilient Airport Surface Movement Operations

Master of Science Thesis

by

K. Fines

For obtaining the degree of Master of Science in Aerospace Engineering at
Delft University of Technology

To be publicly defended on November 29th, 2019

Student number:	4223934	
Thesis committee:	Prof. Dr. R. Curran	TU Delft, Chairman
	Dr. O.A. Sharpanskykh	TU Delft, Supervisor
	M. P. J. Vert	TU Delft, Supervisor
	Dr. Ir. J. Ellerbroek	TU Delft, External Examiner

Cover Image: Schiphol Airport, Amsterdam, Netherlands.
`https://i.redd.it/mdfxspq60yd01.jpg`

Preface

This thesis concludes my life's chapter at Delft University of Technology, where I was able to expand my horizons and become a Master of Science in Aerospace Engineering. I was fortunate enough to have encountered a huge variety of professional and academic experiences which have set the foundations for my future life and career. I would like to thank everyone who I have had contact with and who has helped me along my journey over these years.

Firstly, I would like to express my gratitude to my supervisors Dr. Alexei Sharpanskykh and Matthieu Vert whose continuous interest, support and guidance in my work was crucial throughout this project. Not only did they introduce me to the concepts of Agent-Based Modelling and resilience, but their willingness to transfer their deep academic and scientific knowledge proved to be invaluable throughout this research.

Secondly, thank you to all my friends and colleagues, which I have been able to share my experiences with and who have supported me along every step of my journey. Thanks Bryan, Michael, Thomas and everyone else for all the fun times and great moments that we've had together. A special thanks, truthfully, to my loving girlfriend Renée for holding my hand every step of the way. Without her unlimited support, warm-heartedness, and kindness, my time here would have been very different.

Finally, I would like to thank my parents, Panos and Katerina, who are the most inspirational figures of my life. Their unconditional love, support and guidance not only enabled me to complete my long studies at Delft University of Technology, but have made me who I am today.

K. Fines
Delft, November 2019

Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Acronym / Abbreviation	**Description**
A-CDM	Airport Collaborative Decision Making
A-SMGCS	Advanced Surface Movement Guidance and Control System
A-VDGS	Advanced Visual Guidance Docking Systems
ABM	Agent-Based Model
ACARS	Aircraft Communications and Reporting System
ACC	Area Control Centre
ADS-B	Automatic Dependent Surveillance Broadcast
AGHT	Actual Ground Handling Time
AGL	Airfield Ground Lighting
AI	Artificial Intelligence
AIBT	Actual In Block Time
AIP	Aeronautical Information Package
ALDT	Actual Landing Time
ANSP	Air Navigation Service Provider
AOBT	Actual Off Block Time
ARDT	Actual Ready Time
ASAT	Actual Startup Approval Time
ASRT	Actual Startup Request Time
A-SMGCS	Advanced-Surface Movement Guidance and Control System
ATC	Air Traffic Control
ATCO	Air Traffic Control Officer
ATM	Air Traffic Management
ATIS	Automated Terminal Information Service
ATOT	Actual Takeoff Time
CATC	Conflicting ATC Clearances
CBS	Conflict-Based Search
CBS+CB HWYs	Conflict-Based Search and Conflict-Based Highways
CBS+PM HWYs	Conflict-Based Search and Point-Merge Highways
CPDLC	Controller Pilot Data Link Communications
CPDSP	Collaborative Pre Departure Sequencing Planning
CMAC	Conformance Monitoring Alerts for Controllers
CT	Constraint Tree
CTOT	Calculated Take Off Time
DAG	Directed Acyclic Graph
EHAM	ICAO designator for Schiphol
EXIT	Estimated Taxi In Time
EXOT	Estimated Taxi Time along a predefined route
HMI	Human Machine Interface
ICAO	International Civil Aviation Organization
KNMI	Koninklijk Nederlands Meteorologisch Instituut
KPI	Key Performance Indicator
LVNL	Luchtverkeersleiding Nederland
LVP	Low Visibility Procedures
MAPF	Multi-Agent Path Finding
MAS	Multi-Agent System

Acronym / Abbreviation	**Description**
METAR	Meteorlogical Aerodrome Report
MLAT	Multilateration
QNH	Barometric pressure setting
RMCA	Runway Monitoring and Conflict Alerting
RNAV	Area Navigation
SAPF	Single-Agent Path Finding
SES	Single European Sky
SESAR	Single European Sky ATM Research
SID	Standard Instrument Departure
SSR	Secondary Surveillance Radar
TAF	Terminal Area Forecast
TCL	Taxiway Centreline Lights
TOBT	Target Off Block Time
TSAT	Target Startup Approval Time
TTOT	Target Takeoff Time
TTOT’	Earliest Target Takeoff Time
VHF	Very High Frequency
VOR	VHF Omnidirectional Radiobeacon

I

Master of Science Thesis Paper

Agent-Based Distributed Planning and Coordination for Resilient Airport Surface Movement Operations

K. Fines, *MSc Student, Delft University of Technology*
O.A. Sharpanskykh, *Assistant Professor, Delft University of Technology*
M.P.J. Vert, *PhD Candidate, Delft University of Technology*

Abstract*—Airport surface movement operations are complex systems that experience many types of adverse events which require resilient, safe and efficient responses. One regularly occurring adverse event is that of runway reconfigurations. Agent-based distributed planning and coordination has shown promising results in controlling and maintaining operations in complex systems, especially during disturbances. In contrast to the centralised approaches currently used by air traffic controllers, distributed planning is performed by several agents, which coordinate plans with each other. This research evaluates the contribution of agent-based distributed planning and coordination to the resilience of airport surface movement operations when runway reconfigurations occur. A Multi-Agent System (MAS) model was created based on the layout and airport surface movement operations of Schiphol Airport in the Netherlands. Within the MAS model, three distributed planning and coordination mechanisms were incorporated, based on the Conflict-Based Search (CBS) Multi-Agent Path Finding (MAPF) algorithm and adaptive highways. MAS simulations were run based on eight days of real-world operational data from Schiphol Airport. The MAS results show that the distributed planning and coordination mechanisms were effective in contributing to the resilient behaviour of the airport surface movement operations, closely following the real-world behaviour, and sometimes even surpassing it. In particular, the mechanisms were found to contribute to more resilient behaviour than the real-world when considering the taxi time after runway reconfiguration events. Finally, the highway included distributed planning and coordination mechanisms contributed to the most resilient behaviour of the airport surface movement operations.

***Index Terms*—Resilience, Airport Operations, Air Traffic Management, Multi-Agent Path Finding, Conflict-Based Search, Multi-Agent Systems**

I. INTRODUCTION

AIRPORT surface movement operations form complex systems, which are constantly exposed to different types of adverse events. One regularly occurring adverse event that the airport surface movement system must be able to cope with is that of runway reconfigurations. These occur due to changing weather, traffic demand, or time of day and often result in an altered airport surface situation, as the airport surface movement operations must adapt in order to accommodate these adverse operational changes. For example, different aircraft taxi routings may be needed, or new traffic streams may suddenly emerge, which require modifications in the handling of the altered traffic flows. These adaptations of the airport surface movement operations should be done in a resilient, safe and efficient manner, in order to achieve the fundamental goals of air traffic services [1]. In this study, a system is considered resilient if it has the intrinsic ability to adapt its functioning prior to, during, or after such adverse events so that it can sustain the required operations [2].

Currently, the airport surface movement operations are controlled by Air Traffic Control Officers (ATCOs) at airports. The ATCOs closely resemble a centralised system where one ATCO is responsible for controlling and coordinating aircraft in their sector in order to ensure the safe and efficient flow of traffic. Within this centralised system, the performance of operations is significantly influenced by the amount of traffic under an ATCO's control, their experience, and adverse events that affect the operational environment [3], [4]. This raises questions with respect to whether such a centralised approach is the most effective for dealing with the complex traffic situations caused by runway reconfigurations.

Agent-based distributed planning and coordination is an alternative approach which has shown promising results with respect to maintaining safe and efficient levels of operations in complex systems, especially in the case of disturbances. For example, this has been achieved in automated, large power grid systems [5], controlling the plan executions of robots [6], and its benefits have been highlighted in certain aspects of air traffic control [7]. Furthermore, decentralisation in airport surface movement systems [8], [9] has been shown to be effective in controlling aircraft on the airport's surface. In particular, decentralised bidding coordination mechanisms [9], [10] resulted in the airport surface movement system performing more efficiently than the real-world. These studies concluded that such approaches were able to safely and efficiently handle complex and chaotic taxi operations with only a limited amount of information exchange being required. This suggests that a distributed planning and coordination approach may enable, potentially, a better organization of the system, especially in the case of disturbed situations due to adverse events. However, no research was done on the application of such methods for studying the resilience of airport surface movement operations. For these reasons, the goal of this research is formulated as follows:

"To evaluate the contribution of agent-based distributed planning and coordination to the resilience of airport surface movement operations when runway reconfigurations occur."

Multiple distributed planning and coordination techniques exist such as Generalized Partial Global Planning [11] or delegate multi-agent systems [12]. Although these mechanisms have been shown to be effective in small and static systems, their effectiveness decreases significantly with scalability and their success in coordinating dynamic systems is questionable. Furthermore, the airport surface movement system is a complex, large and dynamic system [13]. Instead, to address the limitations of these approaches, a new, state-of-the-art family of planning and coordination mechanisms is called Multi-Agent Path Finding (MAPF) algorithms. These algorithms focus on planning and coordinating the paths of numerous agents within complex, dynamic systems. Multiple MAPF planning and coordination mechanisms exist such as Push and Rotate [14], Multi-Agent Path Planning [15] or with specific kinematic constraints [16]. However, these have long computational times, and require specific geometric arrangements of the environment [13]. Instead, the Conflict-Based Search (CBS) MAPF algorithm [17] which coordinates agents by resolving anticipated conflicts was found to be more suitable for the airport surface movement system. No geometric arrangements are required by this algorithm, and it was demonstrated to scale adequately with the number of agents. For these reasons, it was chosen to be used within this study.

Furthermore, plan mergining through highways is also desirable in order to deal with adverse situations in complex systems. Highways encourage agents to follow specific paths, resulting in the coordination of traffic flows. This is used in the resilience of city evacuations [18], and in current ground procedures at Schiphol Airport [19] where they promote platooning [20]. Specifically, highway integrations have been made with the CBS MAPF algorithm such as in warehouses [21]. However, in all these cases, the highways were fixed and pre-defined, limiting their effectiveness during the potential adaptations of airport surface movement operations. For this reason, two types of adaptive highway mechanisms are used within this research. These are integrated in the model proposed in this study, in addition to the CBS MAPF algorithm.

In order to achieve the research goal, a Multi-Agent System (MAS) model was created based on the layout and airport surface movement operations of Schiphol Airport in the Netherlands. This airport was chosen as it contains a complex taxiway and runway layout, exhibits 14 runway reconfigurations on average per day [9] and has large traffic volumes. Within the MAS model, a distributed implementation of the CBS algorithm and two additional adaptive highway mechanisms were incorporated and separately experimented with. Eight days of real-world data from Schiphol Airport was used as the input to the MAS model. Real-world operational situations surrounding runway reconfigurations were simulated using the three distributed planning and coordination mechanisms. Within these simulations, the taxi time and taxi distance indicators were measured, as they are commonly used to characterize airport surface movement operations [22]–[24]. The deviations of these indicators in the transient phase during runway reconfiguration events and in the new rest phase after runway reconfigurations were measured, with respect to the nominal level prior to the runway reconfiguration events. These were compared to the real-world deviations and thus the contribution of the three distributed planning and coordination mechanisms to the resilience of airport surface movement operations was evaluated in a retrospective manner.

This paper is structured as follows. First, the planning and coordination mechanisms are presented in Section II. Then, the Schiphol Airport surface movement operations system is presented in Section III. After this, the MAS model is presented in Section IV. Then, the verification and validation of the MAS model is presented in Section V. Afterwards, the specific techniques used to evaluate the resilience of the airport surface movement operations are presented in Section VI, followed by the results and analysis of the MAS simulations in Section VII. A discussion is then presented in Section VIII. Finally, conclusions are presented in Section IX.

II. THE PLANNING AND COORDINATION MECHANISMS

Three planning and coordination mechanisms were chosen and implemented in a distributed manner within the MAS. These mechanisms will now be further introduced and elaborated upon.

A. *CBS MAPF Algorithm*

The first mechanism is the CBS MAPF algorithm. The baseline CBS mechanism [17] is chosen to be used within this study. This mechanism works by determining and resolving anticipated conflicts between the plans of agents. This is achieved by predicting the time at which agents pass segments of their routes. If two agents are predicted to pass the same segment at the same time, or within a time-window to each other, then one of the two agents is delayed or re-routed, thus resolving the conflict. This mechanism is defined as CBS from now on.

B. *Point-Merge Highways*

The second mechanism includes the addition of point-merge highways to the CBS mechanism. This adaptive highway mechanism is based upon the point-merge technique [25], where traffic flows are directed towards specific points and are merged for subsequent paths. However, within this research, the merge point is dynamically created, based on common route segments of aircraft paths which follow the same traffic flow. The point at which common route segments of same flow aircraft begins is the starting point of the highway. The point at which the common route segments end is the ending point of the highway. Same flow aircraft are therefore encouraged to follow the specific highway streams after the starting point until the highway end point. Furthermore, the highways are also gradually removed if they are not being used. As the paths of aircraft vary, due to new runways being used, the highway starting and ending points change over time. Fig. 1 presents an example of these highways. The red aircraft are encouraged to follow the red highway edges and make one traffic flow. The green aircraft are encouraged to follow the the green highway edges and make a different traffic flow. This mechanism is defined as CBS+PM HWYs from now on.

C. *Conflict-Based Highways*

The third mechanism includes the addition of conflict-based highways, instead of point-merge highways, to the

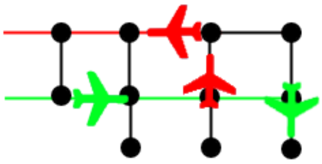

Fig. 1. Point-merge highways example. Red highways and red aircraft are one flow. Green highways and green aircraft are another flow.

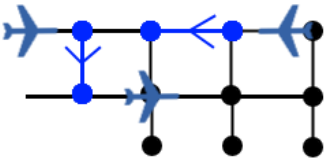

Fig. 2. Conflict-based highways example. Blue edges indicate the conflict-based highway and its directionality. The blue nodes are the areas with a high number anticipated conflicts.

CBS mechanism. This adaptive highway mechanism is based upon the amount of anticipated conflicts in localized areas. Highways are created in regions with a high number of anticipated conflicts, as determined by the CBS algorithm. These highways are composed of ”one-way” taxiway segments in the most travelled direction of aircraft. This results in short taxiway segment highways, creating more localized directional flows in areas with a high number of anticipated conflicts. Furthermore, these highways are removed if they are not being used. Fig. 2 presents an example of these highways. The blue nodes are nodes with a high number of anticipated conflicts, and the blue edges are the conflict-based highways. Their directionality is also indicated by the blue arrow. This mechanism is defined as CBS+CB HWYs from now on.

III. SCHIPHOL AIRPORT SURFACE MOVEMENT OPERATIONS SYSTEM

The Schiphol Airport surface movement operations system is a large, dynamic and complex system [13]. For this reason, certain abstractions were made and applied within the MAS model. This section presents five abstractions in the considered Schiphol Airport surface movement system.

First, only the airside operations were considered, where the sole users of the airport surface were aircraft, controlled by flight crew. Their goal is to reach their destination upon the airport's surface in a safe and efficient manner.

Second, only the taxiing movement operations of aircraft along the airport's surface were considered, as these operations fundamentally form the largest element of the airport surface movement operations system.

Third, arrival aircraft were considered from the moment that they vacate the runway. Furthermore, they were deemed as not being able to immediately alter their speed, as they are turning off from the runway at a higher-than-taxi speed [13], making it difficult for them to slow down or stop. Departure aircraft were considered from the moment that they have completed push back and are ready to taxi. It is possible for them, however, to wait before receiving a taxi command as they have not yet begun moving. Furthermore, all aircraft taxi at their maximum taxi speed, slowing down for turns, other aircraft, and adhere to any mechanism commands. If they reduce their speed, for any of these reasons, then they attempt to accelerate to their maximum taxi speed again.

Fourth, runway occupancy time was included, where departure aircraft may not take off within a specific time-window of each other. Instead, they must wait by forming queues upon the taxiway network near the runway holding points in order to wait for their turn to take-off.

Finally, historical real-world operational data of Schiphol Airport was used as an input to the MAS model in order to perform simulations based on real-world scenarios, using derived runway and flight schedules. The historical real-world operational data was acquired from the Delft University of Technology archives [9]. The data contains the real-world taxi times, taxi distances and origin-destination pairs of aircraft upon Schiphol Airport's surface. This information was derived from extensive Automatic Dependent Surveillance-Broadcast (ADS-B) surveillance data analysis undertaken within a previous study [9], [10].

IV. THE MULTI-AGENT SYSTEM MODEL

This section presents the MAS model and its specifications. The baseline MAS architecture and code is taken from previous studies at Delft University of Technology [8], [9] which resulted in an Agent-Based Modelling (ABM) style simulator of Schiphol Airport movement operations. This research continues this development by adapting the last version of the simulator for the needs and requirements of this research.

A. *Environment Specifications*

The environment is modelled by an abstraction of Schiphol airport's surface as a graph. This abstraction is performed by placing graph nodes at the taxiway intersections, pier entry/exit points and runway entry/exit points. Furthermore, graph edges are placed along the taxiways, thus linking the nodes with each other. The pier entry/exits are simplified and locations chosen based on previous studies [9]. Fig. 3 presents a representation of the graph.

The graph edges can either be bi-directional or uni-directional. Also, the graph edges can be removed. Furthermore, each edge on the graph has a corresponding weight associated with it. The weight, directionality, and removal of graph edges is altered by the ATC Agents to which it is connected to. The graph is a static environmental object and is accessible by all agents.

Additionally, the graph edges can be declared as being part of a point-merge highway (PM HWY) or a conflict-based highway (CB HWY). PM HWY and CB HWY information can also be stored on the edges.

The PM HWY information consists of the flight type of Aircraft Agents which the highway is made for, and the time point at which the edge was declared a PM HWY.

The CB HWY information consists of the time point at which the edge was declared a CB HWY.

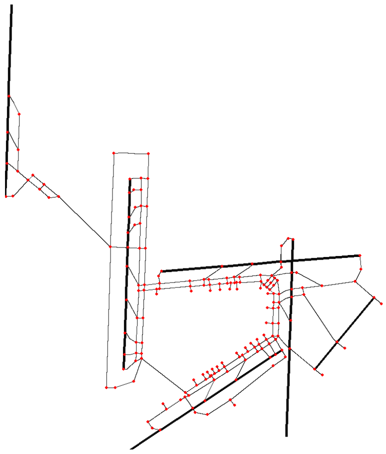

Fig. 3. The graph representation of the taxiways and runways of Schiphol Airport.

The edge weight is initialized by the computed time that it takes to taxi along the edge, at the maximum taxi speed of the Aircraft Agents [9]. The edge directionality is also initialized as bi-directional and all edges are initialized as not being a PM HWY or CB HWY, with no associated information.

F_n represents the flight schedule being used at any time point, where n represents the real-world day which it corresponds to. This is a static environmental object, accessible only by the Entry/Exit Agents and the Airport Operation Status Agent. The flight schedule is represented as a matrix with structure and information as follows.

$$F_n = \begin{bmatrix} FlightID_1 & Origin_1 & Dest_1 & ScheduledTime_1 \\ \vdots & \vdots & \vdots & \vdots \\ FlightID_i & Origin_i & Dest_i & ScheduledTime_i \\ \vdots & \vdots & \vdots & \vdots \\ FlightID_j & Origin_j & Dest_j & ScheduledTime_j \end{bmatrix}$$

Where i represents the information of a flight, and j represents the index of the last considered flight within F_n

B. Agent Specifications

This section presents the specifications related to the agents that were developed and included in the MAS. There are four types of agents: Entry/Exit Agents, Aircraft Agents, ATC Agents and the Airport Operation Status Agent.

1) ***Entry/Exit Agents:*** The Entry/Exit Agents are responsible for creating initial routes for the Aircraft Agents, safely releasing them into that taxiway network, as well as removing Aircraft Agents from the taxiway network. They are located at all pier entry/exits and runway holding points within the environment. Also, the agents know which type of coordination mechanism is being used and the runway occupancy time. The Entry/Exit agents have three properties.

Check Flight Schedule Property: This property involves interactions between the Entry/Exit Agents and the environment. At each time point, the agent observes the flight schedule and checks whether there is an Aircraft Agent which needs to be created according to the flight schedule. Additionally, the agent knows the origin and destination of the Aircraft Agent from this flight schedule.

If the agent determines that there is an Aircraft Agent which needs to be created, then the agent executes the *Route Generation Property*.

Route Generation Property: This property involves interactions between: 1. Entry/Exit Agents and the environment, 2. Entry/Exit Agents and Aircraft Agents.

This property is executed if the agent determines that an Aircraft Agent must be created, at the same time point as the *Check Flight Schedule Property*. Whenever a route is generated, the Dijkstra algorithm [26] is used.

If the CBS or CBS+CB HWYs mechanism is being used, then the agent observes the graph and generates a route for the Aircraft Agent from its origin to destination.

If the CBS+PM HWYs mechanism is being used, then PM HWYs must be included. The agent determines the flight type of the Aircraft Agent and observes the graph. Then, PM HWY edges in the graph which are equal to the flight type of the Aircraft Agent are temporarily set to a very low edge weight, and those which are not equal are temporarily set to a very high edge weight. Afterwards, the agent generates a route for the Aircraft Agent from its origin to destination.

Then, the agent communicates the route to the Aircraft Agent and executes the *Release Mechanism Property* [13] in order to enable it to start taxiing.

Remove Aircraft Agents Property: This property involves interactions between: 1. Entry/Exit Agents and ATC Agents, 2. Entry/Exit Agents and the environment.

At each time point, the agent checks if it has received a handover of an Aircraft Agent from an ATC Agent. If it has, then the agent observes where it is located in the graph.

If the Entry/Exit Agent is located at a pier entry/exit, then the agent removes the Aircraft Agent from the simulation.

If the Entry/Exit Agent is located at a runway holding point, then the agent checks if the runway is not occupied, and then removes the Aircraft Agent from the simulation. Whilst doing so, it triggers a runway occupancy time which prevents subsequent Aircraft Agents from being immediately removed from the simulation. Instead, the subsequent Aircraft Agents must wait until the runway is not occupied. In this way, the Aircraft Agents form queues and are removed on a first come first served basis.

2) ***Aircraft Agents:*** The Aircraft Agents follow the commands of the ATC Agents and taxi along their assigned routes in a safe and efficient way. They maintain a safe distance from other Aircraft Agents, trying to accelerate to their maximum taxi speed whenever possible, and slow down for turns. Also, the agents measure their taxi distance and taxi time. Additionally, the agents have a radar which is used to observe other Aircraft Agents in their vicinity. Furthermore, the agents know their scheduled time from the flight schedule.

Finally, each agent has a flight type. Flight type is defined as the arrival or departure property of the agent, and the agent's destination. In total, there are 14 possible flight types, which can be found in [13]. The Aircraft Agents have four properties.

Motion Properties: This property involves interactions between: 1. Aircraft Agents and the environment, 2. Aircraft Agents and Aircraft Agents.

At each time point, the agent follows its acceleration, heading and decision making motion protocols as described in [9] in order to taxi along its route. The important aspects of these protocols, which are executed at each time point, will now be presented.

First, the agent checks if it is taxiing at its maximum taxi speed V_{max}, otherwise it attempts to accelerate to it using its acceleration a_{accel}.

Second, if the agent is approaching a turn, and if the turn angle, θ_{turn} is greater than a specific turn angle $\theta_{maxturn}$, then the agent decelerates at a_{decel} to its maximum turn speed V_{turn} in order to execute the turn.

Third, the agent observes if there any other Aircraft Agent on its radar. If there are, then the agent determines whether it is following the other agent. If the agent determines that it is following the other agent, then the agent slows to match the speed of the other Aircraft Agent and maintains visual separation from it.

, the agent additionally executes the *Speed Control Command Property* described below.

Speed Control Command Property: This property involves interactions between ATC Agents and Aircraft Agents.

At each time point, the agent checks whether it has received a "slow down command" from an ATC Agent.

If it has, then the agent decelerates at a_{decel} for the subsequent time points until it receives a "cancel slow down command" from an ATC Agent. Once it receives this, the agent stops its deceleration.

Compute Distance Travelled Property: At each time point, the agent computes its distance travelled at that time point, t using Eq. 1.

$$d_t = v_t \cdot \Delta t \tag{1}$$

Δt represents the time step and v_t represents the speed of the agent at the current time point. The distance travelled to the current time point is computed using Eq. 2, where t_0 is the time point that the agent was created in the simulation.

$$d = \sum_{t_0}^{t} d_t \tag{2}$$

Compute Taxi Time Property: At the time point when the agent is removed from the simulation by the *Removing Aircraft Agents Property* of the Entry/Exit Agent, the agent computes its total taxi time. This is computed using Eq. 3, where $ScheduleTime_i$ corresponds to the Aircraft Agent's scheduled time from the flight schedule.

$$T_{taxi,i} = t - ScheduleTime_i \tag{3}$$

*3) **ATC Agents:*** The ATC Agents are responsible for detecting and resolving anticipated conflicts, creating and removing highways, and guiding the Aircraft Agents along their routes. These agents primarily incorporate and make up the distributed planning and coordination mechanisms. They are located at all nodes within the taxiway network that are not Entry/Exit Agents. Also, the agents know which type of coordination mechanism is being used and the associated mechanism variable values. The ATC Agents have twelve properties.

Anticipated Conflict Detection Property: This property involves interactions between the ATC Agents and Aircraft Agents.

At each time point, the agent determines the Aircraft Agents that will pass by it. Then, the agent computes a predicted time point at which these Aircraft Agents are anticipated to pass by it. This is computed by the Forward Simulation presented in Algorithm 1. Afterwards, the agent analyses the passing times of the Aircraft Agents. If two Aircraft Agents are anticipated to pass by the agent within a time window, T_{window}, then the agent is declared as a conflict node and the Aircraft Agents are declared as the conflict pair C_{pair}. Then, the agent executes the *Determine Anticipated Conflict Type Property*.

Determine Anticipated Conflict Type Property: This property involves interactions between the ATC Agents and Aircraft Agents.

This property is executed if the agent has determined a conflict pair C_{pair}, at the same time point as the *Anticipated Conflict Detection Property*. The detected anticipated conflict must be further assessed in order to determine what type of conflict it forms and how it can be resolved.

As the agent is the conflict node, it is common to the routes of both Aircraft Agents in the conflict pair C_{pair}. The agent determines the next node which the Aircraft Agents will head to after passing the agent, and the previous node which the Aircraft Agents come from prior to passing the agent.

If the agent determines that the next node of one of the Aircraft Agents is the same as the previous node of the other Aircraft Agent, then the Aircraft Agents form an anticipated head-on conflict and the agent then executes the *Anticipated Head-on Conflict Resolution Property*. Fig. 4 presents an example of such a head-on conflict case, where node B is the conflict node, node A is the next node of Aircraft Agent $AC2$ and node A is the previous node of Aircraft Agent $AC1$.

Algorithm 1 Forward Simulation of the Aircraft Agent's route

```
Input: Localized information of the Aircraft Agent
Output: Predicted time point, $t_{pass}$, of unimpeded passing the $ATC$ Agent
$V$ → velocity of Aircraft Agent at the current time point
$ATC_{init}$ → ATC Agent which the Aircraft Agent is approaching
$ATC$ → ATC Agent which is performing the Forward Simulation
$D$ → initialized as distance to $ATC_{app}$
$t$ → current time point
$wasturn$ → boolean variable. True if a turn was just forward simulated, otherwise false
$t_n \leftarrow t + D/V$
if $ATC_{app} = ATC$ then
    $t_{pass} \leftarrow t_n$
else
    for each $node$ in $route$ until $node = ATC$ do
        if $wasturn = True$ then
            $t_{delay} \leftarrow t_{delay} + |V - V_{turn}|/a_{accel}$
            $wasturn \leftarrow False$
        end if
        if $\theta_{turn} \geq \theta_{maxturn}$ and $V \geq V_{turn}$ then
            $t_{delay} \leftarrow t_{delay} + |V_{turn} - V|/a_{decel}$
            $wasturn \leftarrow True$
        end if
        $D \leftarrow D + D_{node}$
        $t_n \leftarrow t + D/V + t_{delay}$
    end for
    $t_{pass} \leftarrow t_n$
end if
return $t_{pass}$
```

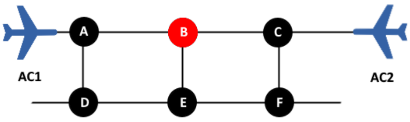

Fig. 4. Anticipated head-on conflict example. Node B is the conflict node.

If the agent determines that the previous nodes of the Aircraft Agents are not the same, then the Aircraft Agents do not form an anticipated head-on conflict, but form an anticipated crossing conflict. In this case, the agent then executes the *Anticipated Crossing Conflict Resolution Property*. Fig. 5 presents such an anticipated crossing conflict, where node B is the conflict node, node A is the previous node of $AC1$, and node E is the previous node of $AC2$.

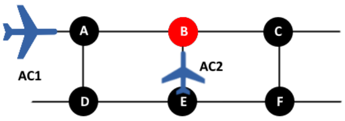

Fig. 5. Anticipated crossing conflict example. Node B is the conflict node.

Finally, if the agent determines that the next and previous nodes of both Aircraft Agents are the same, then the Aircraft Agents are following each other. For this reason, they will maintain visual separation to each other. In this case, the agent takes no further action for this anticipated conflict. This is shown in Fig. 6, where the previous node of both $AC1$ and $AC2$ is node A, and the next node of $AC1$ and $AC2$ is node C.

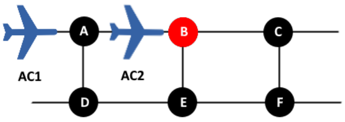

Fig. 6. Aircraft Agents following each other. Node B is the conflict node.

Anticipated Head-on Conflict Resolution Property: This property involves interactions between: 1. ATC Agents and Aircraft Agents, 2. ATC Agents and the environment.

This property is executed if the agent has determined that the Aircraft Agents in the conflict pair C_{pair} form an anticipated head-on conflict, at the same time point as the *Determine Anticipated Conflict Type Property*.

Head-on conflicts cannot be resolved by speed commands alone, but can be resolved by re-routing one of the two Aircraft Agents to avoid the head-on route segment.

The previously introduced Fig. 4 shows such a scenario and will be used to aid the explanation of this property. In this example, Aircraft Agent $AC1$ has the route which traverses along nodes A-B-C, Aircraft Agent $AC2$ has the route C-B-A, the agent at node B has been declared as the conflict node and $AC2$ is the furthest away aircraft from the conflict node B.

The agent observes the graph. Then, the agent determines a new route for the furthest away Aircraft Agent, in this case $AC2$, re-routing it along a new route of C-F-E-D-A, whilst the route of $AC1$ remains unchanged. The new route bypasses the common edges to both Aircraft Agents, namely edges A-B and B-C. Then, the agent communicates the new route to the Aircraft Agent $AC2$, thus resolving the anticipated head-on conflict. If the CBS+PM HWYs mechanism is being used, then the agent includes the PM HWY effects as described in the *Route Generation Property* of the Entry/Exit Agents whilst determining the new route.

Anticipated Crossing Conflict Resolution Property: This property involves interactions between: 1. ATC Agents and Aircraft Agents, 2. ATC Agents and ATC Agents.

This property is executed if the agent has determined that the Aircraft Agents in the conflict pair C_{pair} form an anticipated crossing conflict, at the same time point as the *Determine Anticipated Conflict Type Property*.

The agent determines which of the two Aircraft Agents of the conflict pair is furthest away from it, with respect to the remaining taxi distance of its route. The furthest away Aircraft Agent, AC_s, is declared as the Aircraft Agent which will receive the speed control command.

Then, the agent computes the required speed that AC_s must slow to using Eqs. 4 & 5:

$$T_{req} = t_s + T_{window} - (t_s - t_o) - t \tag{4}$$

$$V_{req} = \frac{d_s}{T_{req}} \tag{5}$$

where t_s is the estimated passing time of AC_s, T_{window} is the time window within which an anticipated conflict is declared, t is the current time point, t_o is the estimated passing time of the other aircraft in the conflict pair and d_s is the taxi distance of AC_s.

Afterwards, the agent communicates V_{req} to the ATC Agent which AC_s is currently approaching. Then, the agent increases its internal counter, $n_{resolutions}$, which keeps track of the number of resolved anticipated crossing conflicts.

Issue Speed Control Command Property: This property involves interactions between: 1. ATC Agents and ATC Agents, 2. ATC Agents and Aircraft Agents.

At the time point when the agent has received V_{req} from an ATC Agent, the agent communicates a "slow down command" to the Aircraft Agent that is heading towards it, AC_s. For the time points after this communication, the agent monitors the velocity of AC_s. If the agent observes that the velocity of AC_s is equal to V_{req}, then the agent communicates a "cancel slow down command".

At any time point whilst the agent is monitoring the speed of AC_s, if AC_s is handed over to the next ATC Agent, then the monitoring of the Aircraft Agent AC_s is also passed over to the next ATC Agent.

Create Point-Merge Highways Property: This property involves interactions between: 1. ATC Agents and Aircraft Agents, 2. ATC Agents and the environment.

This property is only executed if the CBS+PM HWYs mechanism is being used.

At each time point, the agent determines the number of Aircraft Agents per flight type that will pass by its connected edges. Then, the agent determines the edge for which the number of flight type specific Aircraft Agents is greater than or equal to the highway generation threshold. Afterwards, the agent attempts to declare the edge as a PM HWY for that flight type. This is done by the agent observing the graph edge PM HWY information in the environment. If the graph edge is not a PM HWY, then the agent declares it as a PM HWY and stores the flight type information and the current time point in the graph edge. If the graph edge is already a PM HWY for the same flight type, then the agent only stores the current time point in the graph edge.

Otherwise, if the graph edge already contains PM HWY information for a different flight type, or if the number of Aircraft Agents per flight type is less than highway generation threshold for any of its connected edges, then the agent executes the *Remove Point-Merge Highways Property*.

Remove Point-Merge Highways Property: This property involves interactions between the ATC Agents and the environment.

This property is executed if the agent was not able to declare an edge as a PM HWY or if the number of Aircraft Agents per flight type is less than the highway generation threshold for any of its connected edges. This property is executed at the same time point as the *Create Point-Merge Highways Property*.

The agent observes if any of its connected edges contain PM HWY information. If none of them do, then the agent takes no further action.

Otherwise, if any of its connected edges contain PM HWY information, then the agent determines the amount of time that the edge was a PM HWY. If the amount of time that the edge was a PM HWY is greater than or equal to the amount of time a PM HWY should persist for, then the agent removes all of the PM HWY information from the edge and it stops being a PM HWY. If the amount of time that an edge was a PM HWY is smaller than the amount of time a PM HWY should persist for, then the agent takes no further action.

Create Conflict-Based Highways Property: This property involves interactions between: 1. ATC Agents and Aircraft Agents, 2. ATC Agents and ATC Agents, 3. ATC Agents and the environment.

This property is only executed if the CBS+CB HWYs mechanism is being used.

At each time point, the agent checks if its anticipated crossing conflict resolution counter $n_{resolutions}$ is greater than or equal to the highway generation threshold. If it is, then the agent determines the neighbouring ATC Agents for which this is also the case. Then, the agent considers the graph edges which link the agent and these neighbouring ATC Agents. For these edges, the agent determines the most travelled direction of resolved anticipated conflicts. This can either be from the neighbouring ATC Agent to the agent, or vice versa. This determined direction is the uni-direction which the edges should be made in order to be CB HWYs.

Then, the agent attempts to declare these edges as CB HWYs. The agent checks whether each edge has already been made part of a CB HWY. If it has not, then the agent makes the edge uni-directional in the determined direction. Then, the agent stores the current time point information on the edge. If the uni-directionality of the edge was already in the same determined direction, then the agent only stores the current time point information on the edge.

Otherwise, if the edge was already made part of a CB HWY in the opposite direction, or if the agent's (or its neighbouring ATC Agents) $n_{resolutions}$ is less than the highway generation threshold, then the agent executes the *Remove Conflict-Based Highways Property*.

Remove Conflict-Based Highways Property: This property involves interactions between the ATC Agents and the environment.

This property is executed if the agent was not able to declare an edge as a CB HWY, or if the agent's (or its neighbouring ATC Agents) $n_{resolutions}$ is less than the highway generation threshold, at the same time point as the *Create Conflict-Based Highways Property*.

The agent determines if any of its edges were made uni-directional as part of a CB HWY. If none were, then the agent takes no further action.

Otherwise, if any of its edges were made uni-directional as part of a CB HWY, the agent determines the amount of time that the edge was a CB HWY. If the amount of time that the edge was a CB HWY is greater than or equal to the amount of time a CB HWY should persist for, then the agent makes the edge bi-directional again and removes all of its CB HWY information. If the amount of time that the edge was a CB HWY is smaller than the amount of time a CB HWY should persist for, then the agent takes no further action.

Remove Aircraft Agent Edge Property: This property involves interactions between: 1. ATC Agents and Aircraft Agents, 2. ATC Agents and the environment.

At each time point, if an Aircraft Agent is under the control of the agent, then the agent determines the edge and direction which the Aircraft Agent is travelling. Then, the agent removes the edge in the opposite direction of the Aircraft Agent's movement in the graph. At the time point that the Aircraft Agent finishes travelling over the edge, the agent makes the edge bi-directional again, unless the edge is part of a CB HWY.

Change Runway Crossing Edge Property: This property involves interactions between the ATC Agents and the Airport Operation Status Agent.

At each time point, the agent checks whether it has received information from the Airport Operation Status Agent about an edge which should be added or removed. If the agent receives information about an edge, then the agent observes whether the edge exists in the graph. If the edge exists, then the agent removes the edge from the graph. Otherwise, if the edge does not exist, then the agent adds the edge to the graph.

Handover Property: This property involves interactions between: 1. ATC Agents and ATC Agents, 2. ATC Agents and Entry/Exit Agents, 3. ATC Agents and Aircraft Agents, 4. ATC Agents and the environment.

At each time point, the agent checks if an Aircraft Agent is under its control and is about to pass it. If it is, then the agent checks if the edge which the Aircraft Agent is about to taxi upon exists in the graph. If it does, then the agent hands over the control responsibility of the Aircraft Agent to the next ATC or Entry/Exit Agent.

If the edge does not exist, then the agent determines a new path for the Aircraft Agent using the Dijkstra algorithm and communicates it to the Aircraft Agent. Afterwards, the agent hands over the control responsibility of the Aircraft Agent to the next ATC Agent in its new route. It is important to note that if the CBS+PM HWYs mechanism is being used, then the agent also includes the effects of the PM HWY as described in the *Route Generation Property* of the Entry/Exit Agents during the new route generation.

4) ***Airport Operation Status Agent****:* The Airport operation Status Agent is responsible for determining which runways are active and communicating, if required, to the ATC Agents which should add or remove certain edges depending on the runway use. In this way, the agent prevents Aircraft Agents from crossing active runways. The Airport Operation Status Agent has two properties.

Determine Runway Use Property: This property involves interactions between the Airport Operation Agent and the environment.

At the very first time point, the agent observes the flight schedule and determines a runway schedule. This is done by sorting the flights in the flight schedule based on their scheduled time and observing their origins and destinations to determine the arrival and departure runways throughout the day.

Remove Graph Edges Property: This property involves interactions between the Airport Operation Status Agent and ATC Agents.

At each time point, the agent determines which runways are active for arrivals and departures, based on the current time point and the runway schedule. Then, the agent determines which edges must be removed or added and communicates this information to the ATC Agents which are connected to them.

V. VERIFICATION & VALIDATION

Verification was performed using computerized model verification techniques as well as with plausibility consideration approaches [27]. Code verification was done by performing unit testing [28] and by resolving any compiler errors. Additionally, conceptual verification was performed by small scale sensitivity analyses and observing whether the emergent behaviour matches the expected behaviour of the elements. For example, the runway occupancy time was verified by observing the emergence of queues at runways. Additionally, fixed scenarios were simulated where it was known how the included mechanisms should behave. Finally, calculation verifications were also undergone in order to check whether the computation mechanisms matched the manually computed results. For example, this was done by manually calculating the elements of the CBS algorithm and comparing them to the computed elements within the MAS implementation.

Validation was performed by exploring the model behaviour through sensitivity analyses of model parameters using a range of operational scenarios as inputs. The face validity [29] of model outputs in these cases was assessed. Furthermore, real-world scenarios were simulated using the real-world data, and the MAS model outputs were compared to the historical real-world operational data of the same scenarios. Finally, a graphical display was used to observe the animated motion of the aircraft and of the mechanisms. This was compared to real-world animations.

VI. EVALUATING RESILIENT BEHAVIOUR

The main focus of the analysis of the results is to evaluate the contribution of the three distributed planning and coordination mechanisms to the resilience of the airport surface movement operations, using the MAS model outputs. This section presents the analysis methodology used to evaluate the resilience.

There are many diverse definitions for resilience and associated behaviours in systems. Within this study, resilience can be considered as the intrinsic ability of a system to adjust its functioning prior to, during, or after changes and disturbances, such that it can sustain required operations [2]. Applying this definition to this research, the mechanisms within the MAS contribute to the resilient behaviour of the airport surface movement operations if they re-organise the airport surface movement system such that it maintains at most the same deviations of taxi time and taxi distance as in the real-world, during and after undesirable runway reconfigurations events.

Furthermore, it was established that runway reconfiguration events do not always behave as adverse events. For example, a runway reconfiguration from a geographically far away runway to a geographically closer runway, with respect to the gate location, resulted in a decrease in taxi distance and taxi time after the runway reconfiguration event in the real-world. An adverse event can be considered as anything that impacts, or may impact, the functioning of a system undesirably. For this reason, within this research, the considered adverse events are the runway reconfigurations that decrease the performance of the system in the real-world. The MAS results of these events are used to evaluate the distributed planning and coordination mechanism contribution to the resilient behaviour of the airport surface movement system.

For these adverse runway reconfiguration events, the taxi time and taxi distance of each flight within the MAS model is measured and forms the output of the MAS simulation. This is done for all three distributed planning and coordination mechanisms. Then, these outputs are compared to each other and to the taxi time and taxi distance of the flights in the real-world. The deviations and behaviours are then evaluated before, during, and after the time at which the adverse runway reconfiguration occurred. In this way, the evolution of system behaviour caused by the three mechanisms can be analysed and compared to each other and to the real-world.

In order to aid and quantify the comparison between these outputs, the average taxi time and average taxi distance indicators are computed for the set of flights, F_s, which occur before, during and after the runway reconfiguration time, t_0. A time window limit of t_w is used to characterize the indicator levels before and after the runway reconfiguration, by selecting flights within the time window limits.

The "before runway reconfiguration" indicator level is defined as the nominal airport surface movement system level prior to a runway reconfiguration event. For this nominal level, the average taxi time and taxi distance performance indicators, $T_{nominal}$ and $D_{nominal}$, are computed by averaging the taxi time and taxi distance for all flights which arrived at their destination at $t_{arrival}$ prior to the runway reconfiguration event. For this case, the set of flights, F_s, is made up of the flights that occurred within the time window $T = \{t_0 - t_w \leq t_{arrival} < t_0\}$.

The "during runway reconfiguration" indicator level is defined as the transient airport surface movement system level during a runway reconfiguration. In this case the average taxi time and taxi distance performance indicators, $T_{transient}$ and $D_{transient}$, are computed by averaging the taxi time and taxi distance for all flights which were taxiing whilst the runway reconfiguration occurred. For this case, the set of flights, F_s, is made up of the flights that occurred within the time window $T = \{t_0 - t_w \leq t_{ScheduledTime} \leq t_0\} \cup \{t_0 < t_{arrival} \leq t_0 + t_w\}$.

The "after runway reconfiguration" indicator level is defined as the new airport surface movement system level, after the transient phase has come to rest following a runway reconfiguration. In this case, the average taxi time and taxi distance performance indicators, T_{rest} and D_{rest}, are computed for all flights which begin taxiing after the runway reconfiguration event. For this case, the set of flights, F_s, is made up of the flights that occurred within the time window $T = \{t_0 < t_{ScheduledTime} \leq t_0 + t_w\}$.

Then, the differences $\Delta T_{transient} = T_{transient} - T_{nominal}$ and $\Delta D_{transient} = D_{transient} - D_{nominal}$ are computed to evaluate the contribution of the mechanisms to the initial resilient behaviour of the system, during the runway reconfiguration. Afterwards, $\Delta T_{rest} = T_{rest} - T_{nominal}$ and $\Delta D_{rest} = D_{rest} - D_{nominal}$ are computed to evaluate the resilient behaviour of the system after the runway reconfiguration occurred. The resulting deviations of the MAS simulation values are then compared to the deviation values of the real-world and between the distributed planning and coordination mechanisms.

In order to not include the double counting of flights due to the overlapping of $t_0 \pm t_w$ when multiple runway reconfiguration occurred in a short period of time, runway reconfigurations that violate $t_{0,prev} + t_w \geq t_{0,next}$ are not considered, where $prev$ represents the previous runway reconfiguration and $next$ represents the next runway reconfiguration.

VII. RESULTS & ANALYSIS

The MAS model was developed in Python 2.7, and simulations were run using a Windows 10 machine with 16GB RAM and a hexacore Intel Core i7-8750H processor. The historical real-world data, as presented in Section III, was used as the input to the MAS model and was used to create the flight schedule, F_n. The simulations were run for all three distributed planning and coordination mechanisms using eight days of real-world data and took approximately 21 hours to complete. The specific MAS configuration can be found in [13].

The eight days of simulated operations equated to 6852 flights being simulated. All of these flights successfully reached their destination and were completely de-conflicted, with no collisions occurring between aircraft.

A. Taxi Time & Taxi Distance Behaviour

A typical and representative day with a high amount of runway reconfigurations is chosen to demonstrate the behaviour

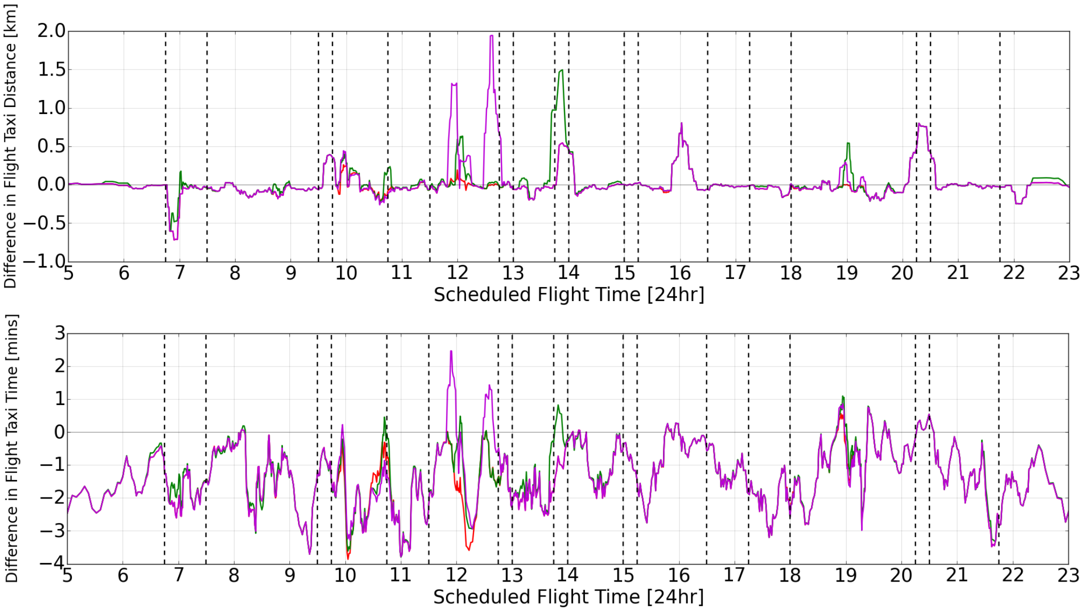

Fig. 7. Moving average differences with respect to the real-world data, grouped per 10 flights for the MAS simulation of 02-05-16. Red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance. Dotted vertical lines represent runway reconfiguration events.

of the mechanisms, namely the results of 02-05-2016. The results of all other simulated days can be found in [13]. Fig. 7 presents the taxi time and taxi distance moving average differences (computed for each 10 flights) of the MAS results with respect to the real-world.

The results in Fig. 7 show that all three mechanisms follow similar trends throughout the day, often overlapping with one another in periods where there are no runway reconfigurations. The average difference in flight taxi time with respect to the real-world of the CBS mechanism is -1.35 min/flight, -1.27 min/flight for the CBS+PM HWYs, and -1.28 min/flight for the CBS+CB HWYs mechanisms. This indicates that all mechanisms result, on average, in a shorter taxi time with respect to the real-world operations, with the CBS exhibiting the greatest average saving per flight. The average difference in taxi distance of the CBS mechanism is -0.02 km/flight, and 0.02 km/flight for the CBS+PM HWYs and CBS+CB HWYs mechanisms. This indicates that there is no significant difference in taxi distance, on average.

All mechanisms follow similar trends for the differences in taxi time and differences in taxi distances, with some notable deviations. Specifically, two clear peaks can be observed at 12:00 and 12:30 for the CBS+CB HWY mechanism, where the difference in taxi time performance significantly increases above the real-world performance and that of all other mechanisms. This can be explained by observing the difference in flight taxi distance graph for this same time periods. The deviation in taxi distance significantly increases, indicating that aircraft travelled longer routes, deviating from the shorter routes which they used in the real-world, or with the other mechanisms. Similarly, the CBS+PM HWYs mechanism also produces a taxi time difference peak at approximately 14:00, accompanied by an increase in the difference in taxi distance at the same time. Again, this shows that the longer taxi routes resulted in longer taxi times. For the CBS mechanism, no such unexpected peaks are observed. By further analysis, it was observed that such a behaviour could be explained by the fact that highways did not emerge in the most effective ways. Instead, long and convoluted highways were generated that were not beneficial to the flows of the inbound and outbound traffic, often having to cross each other multiple times. However, three remarks are made upon these findings. First, highways are able to successfully emerge from the included highway generation mechanisms. Second, the point-merge and conflict-based highways are able to influence the behaviour of the airport surface movement operations at different moments to each other. Third, further work is still required on the highway generation mechanisms such that ineffective, or disruptive, highways are avoided.

The average simulation results of the other days are presented in Table I. The results indicate that all three mechanisms result in a decreased average taxi time per flight, with the CBS mechanism contributing to the greatest saving of 1.07 min/flight with respect to the real-world taxi time. The Vargha-Delaney A-test [30] statistical value of all mechanisms is, on average, 0.40. This shows that there is a difference between the real-world and coordination mechanism-based taxi indicators for the airport surface movement results, where there is only an approximately 40% probability that a randomly selected flight had a taxi time larger than that of the real-world, for all three mechanisms. Hence, it is more likely that a randomly selected flight from the MAS simulation resulted in a smaller taxi time than the real-world. Furthermore, all three mechanisms have a standard deviation which is almost 7 times smaller than that of the real-world. This indicates that there were significantly more uncertainties or operational delays for the

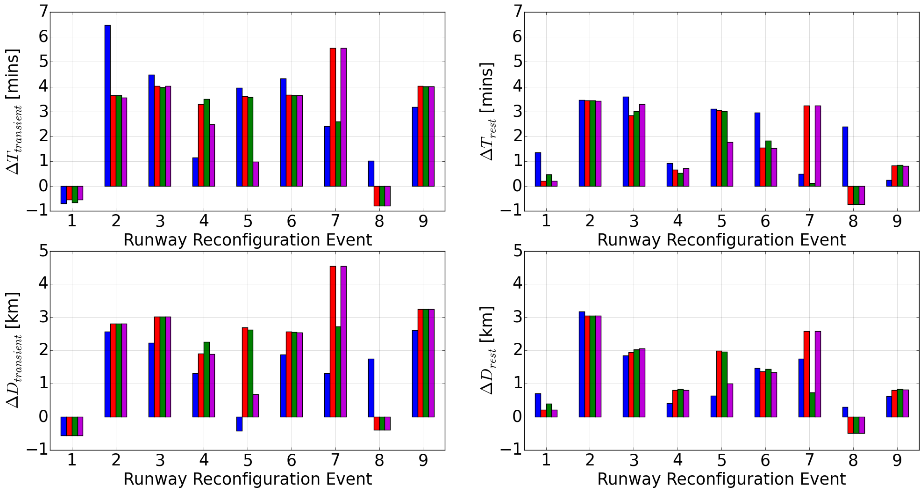

Fig. 8. Deviations in the transient and rest phases for the considered runway reconfigurations Blue bar: real-world performance, red bar: CBS mechanism performance, green bar: CBS+PM HWYs mechanism performance, purple bar: CBS + CB HWYs mechanism performance.

TABLE I
MAS SIMULATION AVERAGE TAXI TIME AND TAXI DISTANCE OF FLIGHTS ON ALL DAYS

Planning and Coordination Mechanism	Taxi Time [min/flight]	Taxi Time A-test Value	Taxi Distance [km/flight]	Taxi Distance A-test Value
Real-world	$\mu = 7.08$ $\sigma = 25.11$	-	$\mu = 3.82$ $\sigma = 2.47$	-
CBS	$\mu = 6.01$ $\sigma = 3.52$	0.40	$\mu = 3.81$ $\sigma = 2.54$	0.49
CBS+PM HWYs	$\mu = 6.04$ $\sigma = 3.53$	0.41	$\mu = 3.83$ $\sigma = 2.55$	0.50
CBS+CB HWYs	$\mu = 6.04$ $\sigma = 3.53$	0.40	$\mu = 3.83$ $\sigma = 2.55$	0.50

flights in the real-world, which was not the case for the flights in the MAS model. However, the results also show that there are almost no differences with respect to the average taxi distance per day. This is also further confirmed by the A-test value of, on average, 0.50 for all mechanisms. Overall, these results show that applying these types of distributed planning and coordination mechanisms may be beneficial to improving the performance of average taxi time of the airport surface movement operations.

B. Evaluating Resilience

Next, the changes between taxi time and taxi distance indicators prior to, during and after runway reconfigurations occurred are analysed. Fig. 8 presents the deviations in performance indicators between the nominal and transient levels, $\Delta T_{transient}$ and $\Delta D_{transient}$, and nominal and rest levels, ΔT_{rest} and ΔD_{rest}, as defined in Section VI. The runway reconfiguration events are numbered from 1-9 for convenience, and are considered to evaluate resilience since they are adverse events as described in Section VI. The times at which these runway reconfigurations occurred can be found in [13]. It is important to note that, for the results of 02-05-16 as well as for the average of all eight days in this section, $t_w = 15$ min was chosen as a suitable and representative time window in order to select runway reconfigurations and compare the deviations. However, the results when considering $t_w = 10, 20, 25, 30$ min can be found in [13].

The results show that, with respect to the evolution of $\Delta T_{transient}$ during runway reconfigurations, all mechanisms are able to contribute to greater resilient behaviour of the airport surface movement operations than the real-world for runway reconfiguration events 2, 3, 5, and 6. This can be seen by all mechanisms having smaller deviation values than the real-world. Runway reconfiguration event 8 indicates that all three mechanisms contributed to an improvement in taxi time and taxi distance during and after the runway reconfiguration. No immediate differences could be determined from the behaviour of the MAS to explain the differences with the real-world. However, after performing data analysis on the real-world data, it was found that aircraft in the real-world took, in some cases, up to three times as long to taxi from the same origin and destination as in the MAS. There may be multiple operational explanations as to why this occurred in the real-world. The aircraft may have taxied at significantly lower speeds due to their aircraft type or weather conditions, or had to wait for an abnormally long amount of time at the runway holding point. In this runway reconfiguration, runway 24 and runway 18L were used for departures, which require, in some cases, coordinated time-intervals for departures flying the same direction. It could be the case that some aircraft had to hold short of the runway for these reasons, thus creating the differences with the MAS model. Further research is required into whether or not these are the true reasons for this deviation, and, if so, how these effects can be implemented in the MAS model.

Furthermore, for runway reconfiguration 5, the CBS+CB HWYs mechanism contributed to the most resilient behaviour of the airport surface movement operations, indicated by the smallest deviation value. This is also seen in the small taxi distance deviations of $\Delta D_{transient}$, which indicates that aircraft did not have to taxi a long large routes. This effect was also propagated into the rest phase, where the ΔT_{rest} value for this mechanism and runway reconfiguration is significantly lower than of the other mechanisms and the real-world. This highlights that this mechanism was particularly effective at reorganizing the operations in this runway reconfiguration event. The highways had formed effectively prior to the runway reconfiguration, and their persistence into the rest phase was desirable. This shows that effectively formed highways are beneficial in contributing to the resilient behaviour of the airport surface movement operations.

For runway reconfigurations 4, 7 and 9, the mechanisms contributed to less resilient behaviour than the real-world of the airport surface movement operations during the runway reconfiguration. This is seen by the higher $\Delta T_{transient}$. Runway reconfiguration 7 indicates significantly worse resilient behaviour contribution of the CBS and CBS+PM HWYs mechanisms, indicated by their large values. After further analysis of the emergent behaviour, it was determined that the reason for this is due to the coordination aspects of the Airport Operation Status Agent. In the real-world, the crossing of runway 18C at taxiway W5 [19] was used for taxiing earlier than in the MAS simulation. Therefore, aircraft could taxi along the shorter route by crossing the runway, which was not the case in the MAS simulation. Instead, more aircraft than in the real-world had to taxi along longer routes to pass behind the runway 18C threshold as they were not able to cross the runway. The same issue is the case for the CBS+PM HWYs mechanism. However, it has a lower deviation value as the average taxi time and taxi distance nominal values before the runway reconfiguration are higher than that of the CBS and CBS+CB HWYs. This shows that the lack of anticipation to the runway reconfigurations is undesirable, as these situations could be improved. This should be further investigated such that the airport surface movement system can already reorganize in advance to the runway reconfiguration.

The same analysis is applied to the $\Delta D_{transient}$ performance indicator, which shows that the mechanisms only contribute to resilient behaviour in the airport surface movement operations for runway reconfiguration 1 and 8. This highlights that, overall, there are no improvements in the resilience of the deviations in taxi distance. The deviations are higher than that of the real-world, thus indicating that the mechanisms contribute to less resilient behaviour during runway reconfigurations than the real-world, overall.

In the rest phase, when observing ΔT_{rest}, the mechanisms contribute to more resilient behaviour of the airport surface movement operations than the real-world for runway reconfiguration events 1, 6 and 8. This is indicated by the smaller deviation values. For runway reconfigurations 2, 3 and 4, all mechanisms contribute to approximately equal levels of resilient behaviour than the real-world, as the deviation values are almost the same. When considering ΔD_{rest}, the mechanisms contribute to more resilient behaviour for only the first runway reconfiguration event.

The overall, average, values for these deviations are shown in Table II, which is used to evaluate the overall resilient behaviour of the mechanisms for this day.

TABLE II
MAS SIMULATION AVERAGE RESILIENT RESPONSES FOR THE RUNWAY RECONFIGURATION EVENTS ON 02-05-16

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	2.93	1.41	2.06	1.21
CBS	2.95	2.20	1.68	1.36
CBS+PM HWYs	2.62	2.03	1.39	1.20
CBS+CB HWYs	2.55	1.98	1.59	1.27

The averaged deviation values indicate that all mechanisms contribute to almost the same or more resilient behaviour of the airport surface movement operations than the real-world, with respect to taxi time deviation in the transient phase. The CBS mechanism has a marginally higher average $\Delta T_{transient}$ than the real-world indicating that it contributes to marginally less resilient behaviour than the real-world. However, as the value is almost equal, it can be argued that the CBS mechanism contributes to the same level of resilient behaviour as the real-world. The CBS+CB HWYs exhibits the smallest $\Delta T_{transient}$ suggesting that this mechanism contributes to the greatest degree of resilient behaviour in the airport surface movement operations. The CBS+PM HWYs mechanism exhibits less $\Delta T_{transient}$ than the real-world, indicating that it contributes to more resilient behaviour of the airport surface movement operations, but not as much as the CBS+CB HWYs mechanism which contributes to the most resilient behaviour of the airport surface operations. When considering the $\Delta D_{transient}$ performance indicator, all mechanisms contribute to less resilient behaviour of the airport surface movement operations than the real-world. This is indicated by all values being higher than that of the real-world. However, out of the three mechanisms, the CBS+CB mechanism contributes to the smallest deviation.

All mechanisms contribute to more resilient behaviour of the airport surface movement operations when considering the average taxi time deviation after a runway reconfiguration event. This is displayed by the smaller ΔT_{rest} performance indicator than that of the real-world. Furthermore, it is important to note that the CBS+PM HWYs mechanism contributes to the most resilient behaviour, having the smallest deviation value. Out of the three mechanisms, the CBS contributes to the least resilient behaviour of the airport surface movement operations. For the final indicator, ΔD_{rest} performance indicator, the CBS+PM HWYs contributes to the same resilient behaviour of the airport surface movement operations as the real-world, due to having almost the same value. The other two mechanisms both contribute to less resilient behaviour of the airport surface movement operations, due to the higher deviation values than the real-world.

There are some notable remarks to be made here. Overall, it can be determined that, for this day alone, the distributed planning and coordination mechanisms contributed to improvements in the resilient behaviour of the airport surface movement operations with respect to $\Delta T_{transient}$ and ΔT_{rest} when compared to the real-world. However, the mechanisms contributed to significantly less resilient behaviour of the airport surface movement operations when considering the taxi distance of aircraft during the transient phase. After further analysis, this can be explained by the fact that the mechanisms re-organise the airport surface system by keeping the aircraft moving along longer taxi routes, for all the mechanisms, instead of making them stop and wait. Furthermore, for the CBS+PM HWY mechanism, it is observed that the MAS is able to adapt to utilize the disturbed, transient state of the airport surface movement operations and propagate the highway effects to the rest phase more successfully, on average, than the CBS+CB HWYs, after the runway reconfiguration event has occurred. This highlights a strength of the point-merge highway generation mechanism, where generated highways are sustained beyond the transient phase to the rest phase in a more successful way, on average. However, as was highlighted, more work is required in order to ensure that only effective highways are generated as it was observed that this mechanism also contributes to ineffective highway generation in certain cases.

Although resilient behaviour has been demonstrated using the day of 02-05-16, the same analysis of the resilient behaviour was performed for all simulation days, and is shown in Table III. In total, 58 runway reconfiguration adverse events were considered for this analysis, in order to evaluate the contribution of the mechanisms to the resilient behaviour of the airport surface movement operations using a wider range of runway reconfiguration events.

TABLE III
MAS Simulation Average Resilient Responses for the Considered Runway Reconfiguration Events of All Days

Planning and Coordination Mechanism	$\Delta T_{transient}$ [min/event]	$\Delta D_{transient}$ [km/event]	ΔT_{rest} [min/event]	ΔD_{rest} [km/event]
Real-World	$\mu = 3.89$ $\sigma = 2.15$	$\mu = 2.15$ $\sigma = 1.32$	$\mu = 2.12$ $\sigma = 1.39$	$\mu = 1.19$ $\sigma = 0.76$
CBS	$\mu = 3.71$ $\sigma = 2.39$	$\mu = 2.71$ $\sigma = 1.70$	$\mu = 1.59$ $\sigma = 1.17$	$\mu = 1.13$ $\sigma = 0.84$
CBS+PM HWYs	$\mu = 3.65$ $\sigma = 2.38$	$\mu = 2.70$ $\sigma = 1.69$	$\mu = 1.50$ $\sigma = 1.17$	$\mu = 1.09$ $\sigma = 0.83$
CBS+CB HWYs	$\mu = 3.66$ $\sigma = 2.43$	$\mu = 2.71$ $\sigma = 1.74$	$\mu = 1.55$ $\sigma = 1.15$	$\mu = 1.12$ $\sigma = 0.83$

When considering these results, all mechanisms exhibited a slightly smaller taxi time deviation in the transient phase. This shows they contribute to slightly more resilient behaviour of the airport surface movement operations during a runway reconfiguration event than the real-world. The CBS+PM HWYs mechanism actually contributes to the greatest degree of resilient behaviour of the airport surface movement operations during the runway reconfiguration, which is almost the same as the CBS+CB HWYs mechanism. However, all mechanisms contribute to less resilient behaviour of the airport surface movement operations than the real-world in terms of the deviations in taxi distance. This is indicated by all values being larger than that of the real-world. This further supports the previously given argument that the MAS adapts by making aircraft moving along longer routes rather than making them follow shorter routes, but making them stop or taxi at slower speeds in the taxiway network.

When considering the taxi time and taxi distance performance deviations after the runway reconfiguration event, all mechanisms contribute to significantly more resilient behaviour of the airport surface movements operations than the real-world. Out of all the mechanisms, the CBS+PM HWYs contributes to the greatest degree of resilient behaviour, having the smallest deviation value. Finally, when comparing ΔD_{rest}, all mechanisms marginally contribute to a greater degree of resilient behaviour than the real-world, as they display only slightly better improvements, with the CBS+PM HWYs mechanism again yielding the most resilient behaviour in the airport surface movement operations.

Finally, Table IV presents the associated A-test values for these average results.

TABLE IV
MAS Simulation A-test Values of the Average Resilient Responses for the Considered Runway Reconfiguration Events of All Days

Planning and Coordination Mechanism	A-test value of $\Delta T_{transient}$	A-test value of $\Delta D_{transient}$	A-test value of ΔT_{rest}	A-test value of ΔD_{rest}
CBS	0.51	0.63	0.41	0.50
CBS+PM HWYs	0.50	0.63	0.38	0.49
CBS+CB HWYs	0.50	0.62	0.39	0.48

The A-test results further confirm the observations from Table III. For the $\Delta T_{transient}$, the A-test results show that there is negligible difference between the resulting resilient behaviour of all mechanisms. For the $\Delta D_{transient}$, the A-test results show that there is a difference in the resilient behaviour of the mechanisms and of the real-world. The value indicates that there is an approximately 63% probability that a randomly selected runway reconfiguration $\Delta D_{transient}$ result from any of the mechanisms is larger than a randomly selected runway reconfiguration $\Delta D_{transient}$ from the real-world. However, when considering the ΔT_{rest} indicator, all mechanisms result in A-test values of approximately 0.40. This shows that there is a difference between the resulting resilient behaviour of the airport surface movement operations created the mechanisms and of the real-world. In this case, there is a 60% probability that a randomly selected runway reconfiguration ΔT_{rest} of any of the mechanisms is smaller than a randomly selected runway reconfiguration ΔT_{rest} of the real-world. For the ΔD_{rest}, there is negligible difference between the resulting resilient behaviour of any of the mechanisms and the real-world.

VIII. Discussion

This study demonstrates that the implemented agent-based distributed planning and coordination mechanisms are effective in controlling the airport surface movement operations by only communicating routes and when aircraft must "slow down". This is within the capabilities of the upcoming Single European Sky for ATM Research (SESAR) [31] Follow-the-Greens concept [32]–[34] which uses Airfield Ground Lighting (AGL) to control aircraft. Specifically, routes could be illuminated by the AGL, and the "slow down" command could be executed by making the illuminated segment flash in front of aircraft, for example. Once the flight crew observes this, they apply brakes to decrease their speed. If the green segment stops flashing, then the flight crew stops decreasing their speed and can continue taxiing at their desired, unimpeded taxi speed, accelerating if required. Integrations with such a system may, therefore, be interesting to explore.

Furthermore, although this study was based upon Schiphol Airport, it is suspected that larger airport environments with more alternative routes for aircraft could be advantageous to the re-organization of the airport surface movement system by the mechanisms. In the previous section, it was determined that the mechanisms use longer taxi distances to re-organize the airport surface movement system. Therefore, larger environments with more route possibilities could further enhance this effect.

The MAS simulations were run for fixed kinematic aircraft profiles, as all Aircraft Agents had the same maximum taxi speed and accelerations. This is a limitation which could be further improved by using varying aircraft performance. In this way, the mechanisms' contribution to more variable airport surface movement operations could be investigated.

Finally, the Forward Simulation aspect used within the CBS MAS model is an approximation, as it does not account for the accelerations or slowing due to other Aircraft Agents nearby. Furthermore, it also does account for any other control commands from other ATC Agents. For these reasons, better prediction methods such as using machine learning [35] may result in improved CBS behaviour and better conflict-based highway generation.

IX. CONCLUSIONS

This study has taken the first steps into evaluating the contribution of agent-based distributed planning and coordination to the resilience of airport surface movement operations when runway reconfigurations occur.

A MAS model was created which incorporated a distributed implementation of the CBS MAPF algorithm and two types of adaptive highway mechanisms. Flights based upon eight days of real-world historical data from Schiphol Airport were simulated and the taxi time and taxi distance indicators were measured using the MAS model. The deviations in the taxi time and taxi distance indicators during and after runway reconfigurations were determined, with respect to the nominal levels prior to runway reconfigurations. These deviations were used to evaluate the contribution of the distributed planning and coordination mechanisms to the resulting behaviour of the airport surface movement operations within the MAS model. Furthermore, the deviations of each distributed planning and coordination mechanism were compared to the real-world behaviour and to each other.

In terms of overall performance, it was found that the distributed planning and coordination mechanisms resulted in an average saving of 1.07 min/flight taxi time improvement with respect to the real-world historical data. This suggests that such distributed planning and coordination approaches could result in more time efficient airport surface movement operations. No improvements were found with respect to the taxi distance, however.

In terms of resilience, it was found that the distributed planning and coordination mechanisms were all effective in contributing to the resilient behaviour of airport surface movement operations. Their contribution was found to be similar to that of the real-world, resulting from the centralized real-world system which uses highly experienced human ATCOs. When considering the average taxi time deviations during all considered runway reconfiguration events, the three mechanisms did not contribute to any significant differences of the resilient behaviour of the airport surface movement operations when compared to that of the real-world. When considering the average taxi distance deviations during the considered runway reconfigurations, all three distributed planning and coordination mechanisms resulted in significantly greater deviations than the real-world, thus showing that they all contributed to less resilient behaviour of the airport surface movement operations. However, in terms of the average taxi time deviations after the runway reconfiguration had occurred, all three distributed planning and coordination mechanisms contributed to significantly more resilient behaviour, indicated by smaller deteriorations in the nominal performance than that of the real-world. Although all mechanisms had similar results, the CBS+PM HWYs and CBS+CB HWYs mechanisms contributed to the most resilient behaviour of the airport surface movement operations. When considering the average taxi distance deviations after the runway reconfigurations, the three mechanisms did not contribute to any significant differences with that of the real-world.

Finally, it can be concluded that the distributed planning and coordination mechanisms incorporating highways are more beneficial in contributing to the resilient behaviour after runway reconfigurations, with the CBS+PM HWYs mechanism resulting in the most overall contribution to the resilient behaviour of the airport surface movement operations.

REFERENCES

[1] ICAO, "Annex 11: Air traffic services," International Civil Aviation Organization, Tech. Rep., 2001.

[2] E. Hollnagel, *Resilience engineering in practice: a guidebook.* Farnham: Ashgate, 2011.

[3] A. Majumdar and W. Ochieng, "Factors affecting air traffic controller workload: Multivariate analysis based on simulation modeling of controller workload," *Transportation Research Record*, vol. 1788, pp. 58–69, 2002.

[4] C. Wickens, A. Mavor, and J. Mcgee, *Flight to the Future: Human Factors in Air Traffic Control.* Washington: The National Academic Press, 1997.

[5] C. xia Dou, D. wei Hao, B. Jin, W. qian Wang, and N. An, "Multi-agent-system-based decentralized coordinated control for large power systems," *International Journal of Electrical Power & Energy Systems*, vol. 58, pp. 130–139, 2014.

[6] J. King, R. Pretty, and R. Gosine, "Coordinated execution of tasks in a multiagent environment," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 33, pp. 615–619, 2003.

[7] A. Camara, D. C. Silva, E. Oliveira, and P. H. Abreu, “Comparing centralized and decentralized multi-agent approaches to air traffic control,” *Proceedings of the 28th European Simulation and Modelling Conference*, pp. 189–193, 2014.

[8] H. Udluft, “Decentralization in air transportation,” Ph.D. dissertation, Delft University of Technology, Delft, the Netherlands, 2017.

[9] T. Noortman, “Agent-based modelling of an airport’s ground surface movement operation,” Master’s thesis, Delft University of Technology, 2018.

[10] T. E. H. Noortman, “Agent-based modelling of airport’s ground surface movement: Literature study,” Delft University of Technology, Delft, the Netherlands, 2017.

[11] B. Bold, “Generalized partial global planning (gpgp),” Tech. Rep., 2005.

[12] R. Claes, T. Holvoet, and D. Weyns, “A decentralized approach for anticipatory vehicle routing using delegate multiagent systems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, pp. 364–373, 2011.

[13] K. Fines, “Agent-based distributed planning and coordination for resilient airport surface movement operations,” Master’s thesis, Delft University of Technology, 2019.

[14] B. de Wilde, A. W. ter Mors, and C. Witteveen, “Push and rotate: a complete multi-agent pathfinding algorithm,” *Journal of Artificial Intelligence Research*, vol. 51, pp. 443–492, 2014.

[15] K.-H. C. Wang and A. Botea, “MAPP: a scalable multi-agent path planning algorithm with tractability and completeness guarantees,” *Journal of Artificial Intelligence Research*, vol. 42, pp. 55–90, 2011.

[16] W. Honig, T. K. S. Kumar, L. Cohen, H. Ma, H. Xu, N. Ayanian, and S. Koenig, “Multi-agent path finding with kinematic constraints,” *Proceedings of the Twenty-Sixth International Conference on Automated Planning and Scheduling (ICAPS 2016)*, pp. 477–485, 2016.

[17] 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.

[18] G. Pumpuni-Lenss, T. Blackburn, and A. Garstenauer, “Resilience in complex systems: An agent-based approach,” *Systems Engineering*, vol. 20, pp. 158–172, 2017.

[19] “Air traffic control the netherlands,” http://www.ais-netherlands.nl/aim/2018-10-25-AIRAC/html/index-en-GB.html, accessed: 25-10-2018.

[20] C. Bergenhem, S. Shladover, E. Coelingh, C. Englund, and S. Tsugawa, “Overview of platooning systems,” in *Proceedings of the 19th ITS World Congress*, 2012.

[21] L. Cohen, T. Uras, and S. Koenig, “Feasibility study: Using highways for bounded-suboptimal multi-agent path finding,” *Proceedings of the Eighth Annual Symposium on Combinatorial Search*, 2015.

[22] Civil Air Navigation Services Organisation, “Recommended key performance indicators for measuring ansp operational performance,” Civil Air Navigation Services Organisation, Tech. Rep., 2015.

[23] M. Ithnan, T. Selderbeek, W. B. van Blokland, and G. Lodewijks, “Aircraft taxiing strategy optimization,” Department Maritime & Transport Technology, Delft University of Technology, Tech. Rep., 2015.

[24] A. Kharina and D. Rutherford, “Fuel efficiency trends for new commercial jet aircraft: 1960 to 2014,” *The International Council on Clean Transportation*, 2015.

[25] Eurocontrol, “Point merge,” https://www.eurocontrol.int/concept/point-merge, accessed on 16-07-2019.

[26] T. H. Cormen, C. E. Lieserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Boston: The MIT Press, 2009.

[27] D. Helbing and S. Balietti, “How to do agent-based simulations in the future: From modeling social mechanisms to emergent phenomena and interactive systems design,” *Technical Report 11-06-024*, 2015.

[28] L. Morell, *Unit Testing and Analysis*, Hoboken, New Jersey, 1989.

[29] J. Carson, “Model verification and validation,” vol. 1, 01 2002, pp. 52–58.

[30] A. Vargha and H. 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, pp. 101–132, 2000.

[31] SESARJU, “Sesar joint undertaking,” https://www.sesarju.eu/, accessed on 07-10-2018.

[32] EUROCONTROL, “Eurocontrol specification for advanced-surface movement guidance and control system (A-SMGCS) services,” 2018.

[33] ATRiCS Advanced Traffic Solutions GmbH, “Follow the greens,” http://followthegreens.com/, accessed on 07-10-2018.

[34] K. Straube, M. Roßbach, B. D. Vieten, and K. Hahn, “Follow-the-greens: The controllers’ point of view results from a SESAR real time simulation with controllers,” *Advances in Intelligent Systems and Computing*, vol. 484, no. X, pp. 837–849, 2016.

[35] J. Yin, Y. Hu, Y. Ma, Y. Xu, K. Han, and D. Chen, “Machine learning techniques for taxi-out time prediction with a macroscopic network topology,” in *Proceedings of the AIAA/IEEE 37th Digital Avionics Systems Conference*, 2018.

II

Literature Study

This literature study has already been graded

1

Introduction

It is well established that the number of European flight movements is growing, with the total number of flights reaching a record 10.6 million in 2017 and forecasted to grow by an average annual growth rate of 2.3% to a total of 12.4 million flights in 2024, or an increase of 17% [16]. This results in increased congestion at airports which are are not able to fully accommodate the 17% increase in flight movements by 2024 [16]. A direct effect from this is the increased work load for Air Traffic Control (ATC) and stress on the Air Traffic Management (ATM) network, which faces the challenge of absorbing this growth without hindering the safe and efficient flow of flights.

Airports are constantly exposed to different types of disruptions, which require a resilient response to maintain the safe and efficient flow of operations. In the first quarter of 2018, a record 43.1% of departure flights had a delay greater than 5 minutes, and a record 39.3% of arrival flights experienced a delay in Europe [17]. These delays are caused by adverse events in the airport environment. A system is considered resilient if it has the intrinsic ability to adapt its functioning prior to, during, or after such adverse events so that it can sustain the required operations.

Currently, Air Traffic Control Officers (ATCOs) at airports closely resemble a centralized system where one ATCO controls and commands aircraft in their sector in order to ensure the safe and efficient flow of traffic. Naturally, workloads and performance of operations vary and possibly deteriorate with: the amount of traffic and number of movements under a controller's control, the ATCO's human performance and adverse events that affect the operational environment. One change in system architecture in order to improve these factors is that of decentralized control. This is where decision making is shifted from the tower control to a lower, local level by placing multiple interacting virtual agent controllers on the taxiway system which operate and issue commands on local information in order to accomplish global goals. This may result in higher autonomy and allows, potentially, a better organization of the system, especially in the case of disturbances. One such system that is actively being developed at an airport level in combination with the Single European Sky ATM Research (SESAR) programme is a decentralized control system called Follow-the-Greens. This will form a case study of a decentralized control system for airport surface movement operations.

This research, therefore, focuses on further elaborating and evaluating the mechanisms required to enable resilient behaviour of airport surface movement operations utilizing the decentralized control concept. Schiphol airport and the Follow-the-Greens concept will be used as case studies. One adverse event that is of particular importance for airport surface movement operations is that of runway reconfigurations that occur, on average, approximately 14 times a day at Schiphol airport [18]. This adverse event relates to the changing of departure and landing runway configurations based on multiple factors in order to allow the safe and efficient flow of operations from the ground to departure, and from approach to the ground. Runway reconfigurations pose challenges with respect to conflicting flows along the airport's surface. From the steady state of an already established runway configuration, a transient state is undergone where traffic already taxiing to an active runway must be deviated to the new runway reconfiguration, in addition to new flights which will request taxi to this new runway reconfiguration. At the same time, arrival traffic which is taxiing from the "old" active runway must also be dealt with, with respect to the new taxi routes caused by the new runway reconfiguration. This adverse event, therefore, causes significant disturbances in the airport surface movement operations with respect to the management and use of the taxiway network. For this reason, this adverse event is of particular importance and will be further studied as the decentralized control system must be able to respond in a resilient manner to minimize the disruptions on the airport surface movement operations.

The aim of this literature study is to familiarize with the state-of-the-art areas that are associated with this research and culminate in a research objective, questions and methodology which will be used throughout the following phases of the MSc thesis. Firstly, chapter 2 presents and elaborates upon the airport surface movement operations at Schiphol airport, as well as on the adverse event of runway reconfigurations. Secondly, the chosen case study for decentralized control of Follow-the-Greens is presented and elaborated upon in chapter 3. After this, literature which is related to resilience and resilient system behaviour is reviewed in chapter 4. Then, cooperative coordination mechanisms are identified as a key mechanism for the behaviour of the decentralized control system and are presented in chapter 5. Finally, after all the relevant literature has been presented and reviewed, the research proposal and research methodology for this MSc thesis is presented in chapter 6 and chapter 7 respectively.

2

Airport Surface Movement Operations

As this study focuses on airport surface movement operations, it is clear that this is an important area to study in order to understand the types of interdependencies, movements and procedures that are typically followed on airport surfaces. As mentioned in the introduction, Schiphol airport is used as a case study as it is a typical large scale international airport. This chapter is structured as follows: section 2.1 introduces Schiphol airport and discusses the types of airport surface movements that occur, section 2.2 presents a socio-technical system representation of the airport surface movement operations, section 2.3 discusses the adverse event of runway reconfigurations as they occur at Schiphol airport, and finally section 2.4 presents operational data sources that can be used in order to quantify the surface operations at Schiphol.

2.1. Schiphol Airport

This section presents a description of the ground operational environment at Schiphol in subsection 2.1.1 and an overview of typical airport surface movement operations in subsection 2.1.2.

2.1.1. Description

Schiphol Airport began on a piece of land in Haarlemmermeer in 1916 as a military airport [19]. Today it operates with over 63.5 million passengers, handles approximately 106 peak arrival movements and 110 peak departure movements per hour and has 6 runways, 90 gates and 7 piers [20]. In 2016, 478864 air transport movements were recorded with that number continuously increasing every year.

Schiphol is one of the first airports in Europe to install the Airport-Collaborative Decision Making (A-CDM) system [21]. This allows the sharing and connection of different elements of airport surface movement operations with Eurocontrol, as well as other stakeholders which utilize the airport's surface. The benefit of this is the reduction of delays and the facilitation as well as coordination of concurrent aircraft movements on both a European level through the Eurocontrol Network Manager, as well as locally to Schiphol operations. The reason as to why this is interesting to elaborate upon is because it gives insight into the types of data exchanges which occur during every day airport surface movement operations, as well as the scheduling of them. Figure 2.1 presents the milestones (flow) of CDM activities of flights, as established by Eurocontrol [1].

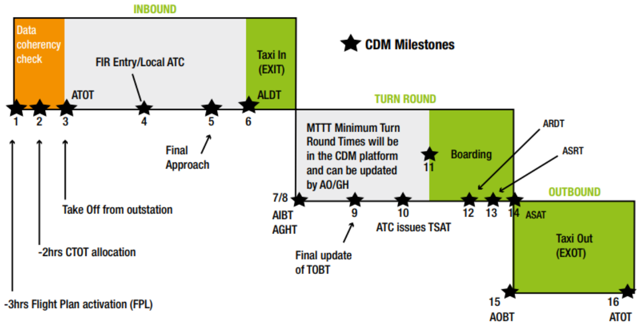

Figure 2.1: A-CDM Milestones [1]

The specific A-CDM milestones that are of interest for the airport surface movement operations (and this MSc study) are from milestone 5 (final approach) until milestone 16 which is the Actual Take-Off Time ATOT). At milestone 6, Actual Landing Time (ALDT), an Estimated Taxi-In Time (EXIT) is computed which is based on the routing from runway to stand. After this, the aircraft begins to taxi along the designated route until milestone 7/8. Milestone 7 is the Actual In Block Time (AIBT) after which Actual Ground Handling Time (AGHT) begins which relates to the turning-around of the flight (passengers de-planing, unloading of cargo, loading of new cargo, etc.). This happens until milestone 9, where the ground handler updates the Target Off Block Time (TOBT) of the flight based on the progress of ground handling processes which is used by a sequencing system based on the active runway configurations and other traffic TOBT states, in order to determine a Target Startup Approval Time (TSAT), which is a sequenced time when a flight is able to start the engines and begin the flight. The benefit of this TSAT is to prevent queues from forming at the runways and blocking the taxiway network. At milestone 11, the passengers begin to board. At milestone 12, the Actual Ready Time (ARDT) is reached, followed by mileston 13 where the Actual Startup Request Time (ASRT) is initiated by the flight crew (and is within the TSAT window). Once the clearance is issued to the flight from ATC, milestone 14 is reached which is the Actual Start Up Approval Time (ASAT). At the same time point (or very near this point if there are no unforeseen push-back problems), the Actual Off Block Time (AOBT) is recorded when the aircraft begins push back. After this moment, an estimated taxi time along a predefined routed is computed (EXOT), until milestone 16 when the aircraft takes off at the Actual Take Off Time (ATOT). Elements of this system that are of particular interest for this study will further be elaborated upon in subsection 2.1.2.

2.1.2. Overview of Typical Airport Surface Movement Operations

As with the ground operations at other airports, Schiphol Airport aims to provide the safe and efficient flow of traffic on the airport's surface, from stand to runway for departing and arrival traffic. This information is primarily derived from a very thorough description contained in a previous MSc thesis [4], as well as from conversations with current and retired ATCOs at Schiphol airport. A typical work flow of airport ground operations is depicted in Figure 2.2. This will be used to support the overview of typical airport surface operations.

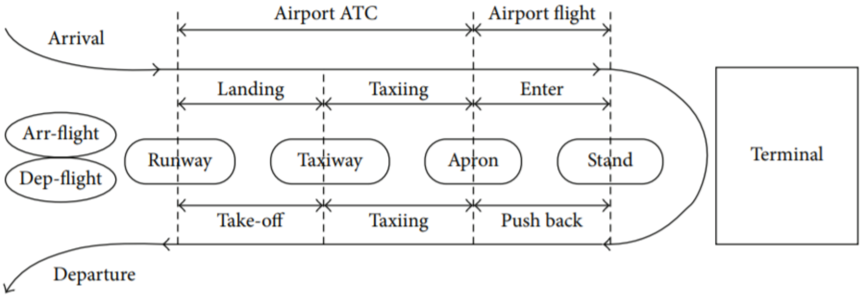

Figure 2.2: Work-flow of Airport Ground Operations [2]

Arrival aircraft receive landing clearances whilst on final approach by the tower controller. After their touchdown, ALDT is recorded in the CDM, and the runway must be vacated in order to allow sequenced traffic to take off or land as soon as is safely and operationally possible. This is aided with the use of rapid exit taxiways which are used to turn off from the runway. This allows arriving traffic to maintain their landing roll (and high speed) onto the taxiway network. Figure 2.3 presents the runway turn-off taxiways. All taxiways that are 30° [3] to the runway are rapid exit taxiways. The maximum speeds with which aircraft are able to vacate the runways via the rapid turn offs are 35kts [22], although speeds of up to 50kts are permitted depending on the rapid exit radius of turn-off curve. Taxiways V1 and V2 from runway 18R, and W4 and W3 on 36C are stated to have this higher turn-off radius and thus allow the higher maximum speed of 50kts to be utilized.

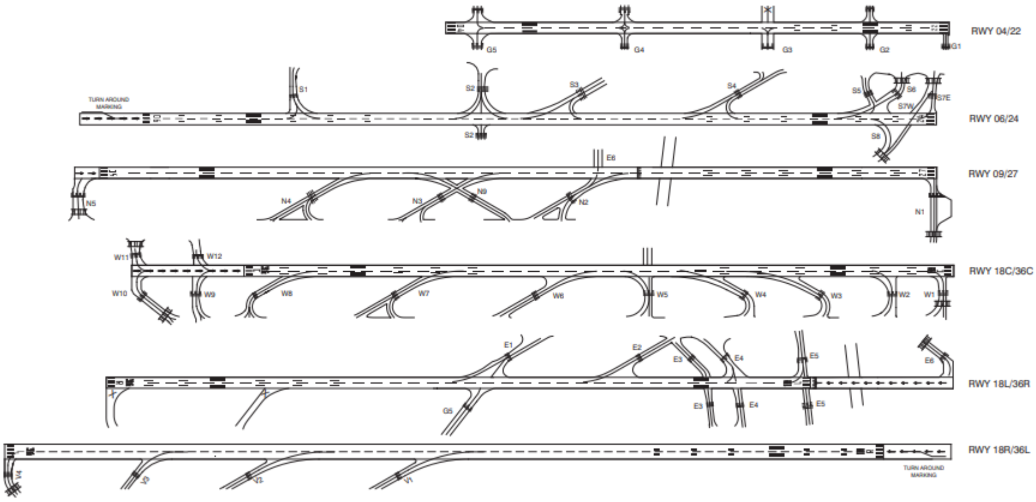

Figure 2.3: Runway Turn Off Taxiways [3]

Once the runway is vacated, the aircraft is handed over to the ground controller. Once the aircraft contacts the ground controller, it receives taxi instructions from the runway to a designated airline operator and aircraft specific stand. This route is used to compute the EXIT in the CDM system. The reason as to why stands are aircraft and operator specific is such that turn around processes on both the landside and airside can be performed as efficiently as possible, with vehicles and passengers not needing to transfer from large airport distances, for example. Figure 2.4 presents the standard taxi directions which are used at Schiphol. The inner ring (taxiway Alpha, depicted in green) follows a clockwise direction, where as the outer ring (taxiway Bravo, depicted in red) follows an anti-clockwise direction. Taxiway Quebec (depicted in orange) does not have a specific direction, and thus is flexible with respect to the direction which traffic can taxi over it. However, it is important to note that the ground controller can issue different directions of routing, as they deem necessary, depending on traffic levels and taxiway usage levels.

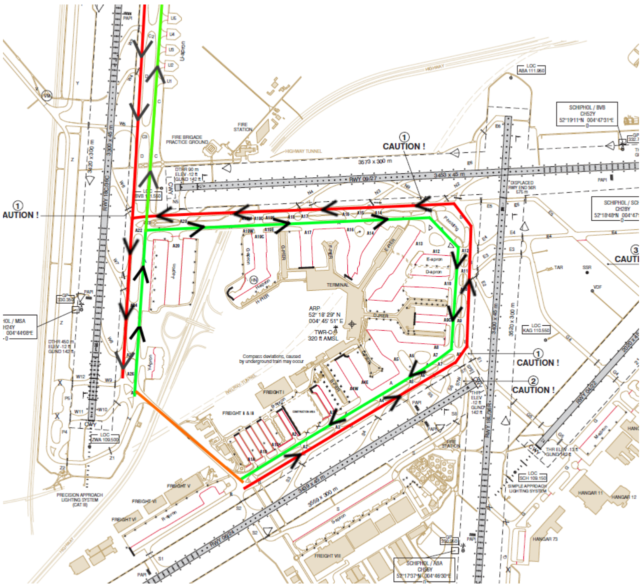

Figure 2.4: Taxi Flow Directions [3]

It may be the case that the stand which the arrival aircraft must taxi to is occupied by a departing aircraft, which may have experienced some delays during its turn around phase or the arrival aircraft arrived earlier than expected. If there is no alternative stand that the arrival aircraft can taxi to, it can be given instructions to taxi and wait just outside, or inside the apron where its stand is located, and must hold until its stand is cleared. Furthermore, if a longer waiting time is anticipated or waiting outside/inside of the apron creates too much of a blockage for other traffic, then the aircraft is given instructions to taxi to a remote holding area called the P (Papa) holding area, shown by the red circle in Figure 2.5.

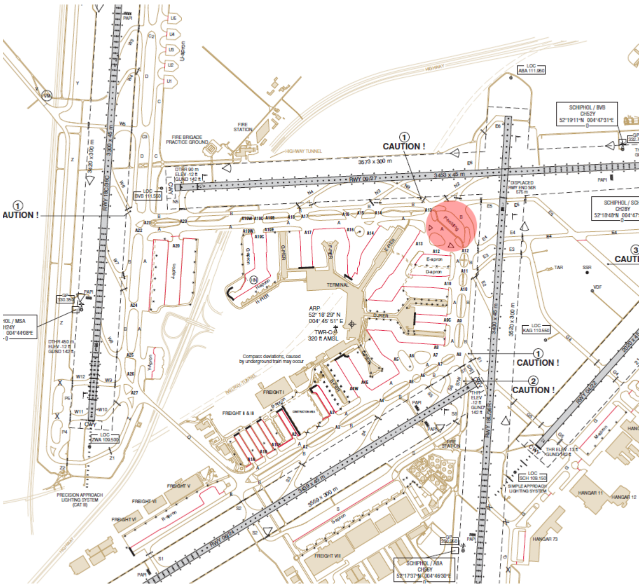

Figure 2.5: P-holding [3]

Once the arrival aircraft reaches the apron which contains its stand, it proceeds to its stand where it shuts down its main engines and systems, and turn around processes commence, such as refuelling, cargo loading/unloading, passenger boarding/de-boarding, maintenance checks etc. This is when the AIBT and AGHT times are recorded in the CDM system.

For departing traffic, the process is similar but is performed in a reverse order. Once all pre-flight turn around phases have been performed (and the final updated of TOBT has been issued), the flight crew request clearance at time ASRT. After clearance has been received by the delivery controller (at time point ASAT in the CDM system), the flight crew then contact the ground controller to request push-back and start-up. The controller assesses if the aircraft is within its allocated start-up window, if there is traffic taxiing behind it (or near to where it will push-back to) and gives the respective clearance (assuming there are no reasons to hold it at the stand) to commence push-back and start-up, as per the standard push-back procedures which vary from apron to apron and from stand to stand.

Once the aircraft commences push back, the AOBT is recorded in the CDM system. Once the push back is complete, the tow bar has been disconnected and the flight crew is ready to taxi, they notify the ground controller. Following from this, the ground controller gives the flight crew instructions to taxi to a holding point of the active take-off runway, which the flight crew is expecting based on their clearance from the delivery controller. EXOT in the CDM system is computed based on this routing. Upon reaching the designated holding point, the ground controller hands the aircraft over to the tower controller which is responsible for sequencing and giving it take-off instructions, leading to the take-off the flight at time ATOT in the CDM system.

The types of instructions that both arrival and departing traffic receive are a sequence (route) of taxiways which must be followed, runway crossings, hold short of other taxiways as well as give way to other traffic instructions. These instructions are given in a pre-determined ATC phraseology [23]. Naturally, this must be done for multiple aircraft types, operators, and from different spatial locations on the airport's surface, all of which are dynamically changing with respect to time and are influenced by environmental factors such as weather. According to air law, the pilot in command of an aircraft is directly responsible for, and has the final authority of the operations of that aircraft [24]. It is the sole responsibility of the pilot-in-command to take the actions necessary to avoid collisions or harm to the aircraft. The aim of ATC instructions and coordination, therefore, is to further enhance the rules which pilots-in-command must follow, by safely and efficiently guiding traffic from the stand to the runway and vice versa. More specifically, instructions should be given to avoid head on scenarios where two aircraft are facing each other (and thus are unable to continue taxi) or where aircraft may end up at the same place at the same time and thus raise the risk of a collision. The fact that the ATC has a much wider situational awareness of the airport surface as a whole, gives the ability to optimize and efficiently organist plans in order to avoid conflicts and minimize taxi distances (this will be further elaborated upon in chapter 3. Ultimately, however, the flight crew is responsible for following the commands issued by the ground controller, whilst maintaining visual separation from other traffic. Furthermore, they are responsible for adhering to other rules of the air. In the case of airport surface movements, these are as follows, taken directly from the International Civil Aviation Organization (ICAO) Rules of the Air [24]:

1. An aircraft shall not be operated in such proximity to other aircraft as to create a collision hazard.
2. An aircraft that has the right-of-way shall maintain its heading and speed.
3. An aircraft in flight, or operating on the ground or water, shall give way to aircraft landing or in the final stages of an approach to land.
4. An aircraft that is aware that another is compelled to land (emergency landing) shall give way to that aircraft.
5. An aircraft taxiing on the manoeuvring area of an aerodrome shall give way to aircraft taking off or about to take off.
6. In case of danger of collision between two aircraft taxiing on the movement area of an aerodrome the following shall apply:
    (a) When two aircraft are approaching head on, or approximately so, each shall stop or where practicable alter its course to the right so as to keep well clear.
    (b) When two aircraft are on a converging course, the one which has the other on its right shall give way.
    (c) An aircraft which is being overtaken by another aircraft shall have the right-of-way and the overtaking aircraft shall keep well clear of the other aircraft.
7. An aircraft taxiing on the manoeuvring area shall stop and hold at all runway-holding positions unless otherwise authorized by the aerodrome control tower.
8. An aircraft taxiing on the manoeuvring area shall stop and hold at all lighted stop bars and may proceed further when the lights are switched off.

2.2. Socio-Technical System Representation

Now that the airport surface operations have been elaborated upon, a socio-technical system representation of airport surface movement operations at Schiphol airport will be presented in order to provide a more specific set of elements and interactions that take place. This will be done using the overview presented in subsection 2.1.2 as well as using the literature of a a previous MSc thesis [4]. Figure 2.6 presents an overview of the elements and interactions involved in airport surface movement operations, taken from [4], and will be elaborated upon. subsection 2.2.1 describes the elements presented in Figure 2.6 and subsection 2.2.2 presents a description of the labelled interactions.

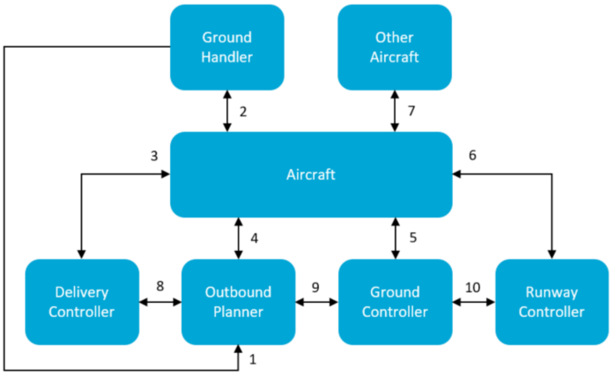

Figure 2.6: Socio-Technical Representation of Airport Surface Movement Operations [4]

2.2.1. Description of Elements

This section describes the elements that are included in the representation presented in Figure 2.6.

Delivery Controller

The delivery controller is responsible for issuing clearances to aircraft with respect to their flight plans. The flight plan is checked that it is activated, and verified to contain consistent information (i.e. route and correct destination) with respect to the intentions of the aircraft. Furthermore, a Standard Instrument Departure (SID)/departure instructions are issued with respect to the active departing runway, and a squawk code (transponder code) is issued to verify the surveillance information of the aircraft. Additionally, the slot time based on the Calculated Take Off Time (CTOT presented in the A-CDM milestone 2 in Figure 2.1) is checked to determine whether the flight is requesting clearance within the CTOT window. The goal of the delivery controller is to check the flight plan, provide each aircraft with departure instructions, squawk code, and to ensure that the flight crew has acknowledged this information.

Outbound Planner

Once an aircraft has acknowledged their clearance, and the flight crew has reported that they are ready for start-up, then the outbound planner is responsible for determining whether or not the aircraft can begin start-up. This is done by assessing the situation with respect to how many other aircraft have received clearances and are also "starting up". A sequence of aircraft startup times (TSAT) is determined with the aid of a Collaborative Pre-Departure Sequence Planning (CPDSP) tool to determine the times at which aircraft can start-up such that queues are avoided at the runways. This tool uses information from the A-CDM system to compute a TSAT sequence for all flights that will be departing from the same runway. For each flight, the earliest Target Take Off Time (TTOT’) is calculated by $TTOT' = TOBT + EXOT$. If there are many aircraft with the same or very similar TTOT’ which will be using the same departure runway, then a queue will form at the runway. For this reason, a sequencing delay (based on CTOT, SID, wake turbulence category, runway capacity) is added to sequence the flights in an algorithmic way. This then makes the TTOT of each flight different: $TTOT = TSAT + EXOT$, where $TSAT = TOBT + T_{Delay}$ as each flight has a different TSAT. It is important to note that whilst listening to live ATC operations at Schiphol, it was determined that the outbound planner and delivery controller were are often combined into the same position. Furthermore, "TSAT Windows" were heard to be used, thus suggesting that TSAT is not necessarily a fixed time point, but a time window. The goal of the outbound planner is to adhere to the TSAT sequencing and grant aircraft permission to start-up.

Ground Controller

The ground controller is responsible for all movement operations upon the airport’s surface. This involves giving push back and taxi instructions to both arriving and departing aircraft, as well as ground vehicles. As mentioned in subsection 2.1.2, the ground controller is responsible for giving taxi routings which enhance the safe and efficient flow of traffic on the airport surface, and avoid head on collisions or instructions that may create other collision scenarios between aircraft and/or vehicle. Furthermore, instructions should result in aircraft arriving at the runway as close as possible to the departure sequence as defined by the CPDSP system. The goal of the ground controller is to safely and efficiently guide departing traffic from stand to runway, arriving traffic from runway to stand, and ground vehicles from their origin to destination, upon the airport’s surface.

Runway Controller

The runway controller is responsible for all operations acting on or surrounding the active landing or departing runways. More specifically, this involves granting take-off and landing clearances to aircraft based on the departure sequence, as well ensuring timing separation between departures/arrivals based on wake turbulence criteria between different aircraft. Furthermore, when aircraft must cross active runways, they are handed to the runway controller from the ground controller, which sequences them to cross the active runway amongst the taking off and landing aircraft. Additionally, the runway controller is responsible for permitting and coordinating ground vehicles to perform inspection operations on the runways. The goal of the runway controller is to ensure the safe and efficient take-off, landing and runway related operations, by adhering to wake turbulence separation criteria as well as departure/landing sequences.

Aircraft

Aircraft are controlled by flight crews which control how the aircraft moves. Furthermore, the flight crews interact with the controllers, as well as with other aircraft. Additionally, the aircraft's flight crews are responsible for adhereing the rules of the air as presented in subsection 2.1.2. For departure aircraft, the goal is to travel from their stand (origin) to their destination (runway) along the airport's surface in a safe and efficient (with respect to taxi time/fuel burn) manner. For arrival aircraft, their goal is to travel from the arrival runway to a stand in a safe and efficient manner.

Ground Handler

The ground handler is responsible for performing and overseeing all pre-flight and post-flight operations of aircraft whilst it is at the stand. This primarily includes the oversight of turnaround operations such as refuelling, cleaning, catering, cargo and passenger loading/unloading. Further more, it is up to the ground handler to issue the final TOBT time in the A-CDM system which will be used to compute the TSAT of the aircraft. It is the goal of the ground handler to minimize the turnaround time of an aircraft and ensure the prompt and on time (or earlier) completion of turnaround processes with respect to the TOBT.

2.2.2. Description of Interactions

This section discusses the interactions as labelled in Figure 2.6.

1: Ground Handler to Outbound Planner

The ground handler updates and submits the TOBT time of the aircraft based on the progress of the turnaround (stand) processes to the outbound planner through the CDM system.

2: Ground Handler to Aircraft

The ground handler interacts with the aircraft's flight crew in order to update the flight crew about aspects relating to the turnaround processes, and discuss any other matters that are related to this.

2: Aircraft to Ground Handler

The aircraft's flight crew interacts with the ground handler to request further services or alterations to the turnaround processes that happen at the stand. Furthermore, they notify the ground handler of any issues related to the TOBT/TSAT which they may receive when communicating with ATC.

3: Aircraft to Delivery Controller

The aircraft's flight crew requests clearance as per their flight plan by contacting the delivery controller either through Very High Frequency (VHF) radio, Aircraft Communications and Reporting System (ACARS) or Controller–Pilot Data Link Communications (CPDLC). Furthermore, the aircraft's flight crew reads back/acknowledges any clearances or instructions which have been received from the delivery controller, and discusses any specific issues which may exist.

3: Delivery Controller to Aircraft

The delivery controller issues delivery related clearances to aircraft. This includes squawk (transponder) code, departure instructions and any further flight plan route clearances to the aircraft. Furthermore, the delivery controller listens to read-back instructions from aircraft in order to verify that all information has been properly received by the aircraft's flight crew. The interaction is either through VHF radio, ACARS or CPDLC. Additionally, the delivery controller notifies the flight crew to contact the outbound planner.

4: Aircraft to Outbound Planner

Once the flight crew of an aircraft have received and acknowledged their clearance from the delivery controller, they then contact the outbound planner (usually the same ATCO as the delivery controller) to report that they are fully ready, indicating that all stand related turnaround processes have been completed and the aircraft is ready to begin its flight.

4: Outbound Planner to Aircraft

The outbound planner grants permission to aircraft that have requested start-up whilst adhering to a TSAT sequence as mentioned in section 2.2. Information such as Automatic Terminal Information Service (ATIS) and barometric pressure level (QNH) is also passed on to the aircraft. Furthermore, the planner tells the aircraft that they should contact the ground controller.

5: Aircraft to Ground Controller
The aircraft's flight crew contact the ground controller once they have received start-up clearance from the outbound planner and are ready to push-back. They state the stand that they are on whilst requesting their push-back. Furthermore, once push-back has been completed, they communicate with the ground controller to notify them that they are ready to taxi.

5: Ground Controller to Aircraft
The ground controller issues push-back clearance once an aircraft has requested push-back (and it is safe to do so), as well as taxi instructions after the aircraft's push-back is complete and the flight crew have stated that they are ready to taxi. Furthermore, once the aircraft is approach their departure runway, they tell the aircraft to contact the runway controller. Once the aircraft is taxiing, the ground controller also observes that the aircraft is travelling along the cleared route.

6: Aircraft to Runway Controller
The aircraft's flight crew contacts the runway controller by stating that they are fully ready for departure at the runway holding point. The flight crew also reads back any clearances (such as hold short, take-off etc.).

6: Runway Controller to Aircraft
The runway controller communicates hold short, line up, take-off, landing and crossing instructions to aircraft. The controller also notifies arrival aircraft/crossing aircraft to contact the ground controller.

7: Aircraft to Other Aircraft
The aircraft's flight crew observe other aircraft and maintain a visual separation to them, by altering their movements as necessary. Furthermore, the flight crew's should adhere to the rules of the air by yielding to traffic that has right of way, as mentioned in section 2.2.

8: Delivery Controller to Outbound Planner
The delivery controller hands over flight responsibility to the outbound planner once a clearance has been given and acknowledged by an aircraft. This is done by transferring the (electronic/paper) flight strip associated with the flight.

8: Outbound Planner to Delivery Controller
The outbound planner hands over flight responsibility to the delivery controller if changes such to the clearance are required (such as a change in departure instructions). This is done by transferring the (electronic/paper) flight strip associated with the flight.

9: Outbound Planner to Ground Controller
The responsibility of the aircraft is handed over to the ground controller after the start-up clearance has been given to the flight.

9: Ground Controller to Outbound Planner
If the aircraft suddenly is not ready to push-back or commence their flight (due to an unforeseen circumstance, for example), the ground controller can hand the responsibility of the aircraft back to the outbound planner. This is done by transferring the (electronic/paper) flight strip associated with the flight.

10: Ground Controller to Runway Controller
If an aircraft is approach its departure runway, or requires crossing of an active runway, then the responsibility of the aircraft is transferred to the runway controller. This is done by transferring the (electronic/paper) flight strip associated with the flight.

11: Runway Controller to Ground Controller
After an arrival aircraft has vacated the runway, or a crossing aircraft has vacated the runway, then the responsibility of the aircraft is handed over to the ground controller. This is done by transferring the (electronic/paper) flight strip associated with the flight.

2.3. Runway Configurations

Airports are dynamic environments which are exposed to multiple adverse events. One such expected and regularly occurring event is that of runway reconfigurations, which is the changing of the active landing and departing runway. Runway configurations are the set of active runways that are used for either landing or taking off traffic. Each runway has a binary state which dictates if they are or are not available for landing/take-off traffic. At Schiphol airport, the runways are, on average, reconfigured 14 times a day [4]. This significantly dictates the types of the airport surface movement operations as the origins for arrival aircraft and destinations of departing aircraft are derived from which the active landing/departing runways are. For this reason, a ground control system must be able to cope with the varying runway configurations in order to avoid disrupting the taxiing flows of traffic, and maintain the safe and efficient flows of traffic. It should not be the case that the goals of the aircraft (of reaching their destinations on the airport's surface) cannot be achieved due to a runway configuration. This section elaborates upon the types of runway configurations that are possible, presented in subsection 2.3.1, contributing factors to runway reconfigurations in subsection 2.3.2, planned runway reconfigurations in subsection 2.3.3, and unplanned runway reconfigurations in subsection 2.3.4.

2.3.1. Possible Runway Configurations

Whilst studying the Aernautical Information Package (AIP) documents for Schiphol (EHAM) [3], it was observed that not all runways are available for both arrival and departure traffic. Table 2.1 presents the runways at Schiphol as well as their availability to be used for arrival and/or departures.

Table 2.1: Available runways at Schiphol for arrival and departure traffic

Runway	Arrivals	Departures
36R	yes	-
36C	yes	yes
36L	-	yes
18L	-	yes
18C	yes	yes
18R	yes	-
9	yes	yes
27	yes	yes
22	yes	yes
4	yes	yes
24	yes	yes
6	yes	yes

In addition to this, the types of runway combinations/configurations that occur at Schiphol are also of interest as these will provide indications into the types of configurations that are most commonly used, as well as which combinations are actually possible. Little publicly available literature was found in order to determine this information. However, Luchtverkeersleiding Nederland (LVNL) provides a real-time, publicly available website [25] which states which runways are active (for every 5 minute interval throughout the day) for landing and departing traffic. This website also has this information from 1 February 2018 until the present. This data was accessed for an arbitrary 1 month period from 9 November 2018 until 7 December 2018. Based on this, the runway changes presented on this LVNL website were grouped per frequency. Figure 2.7 presents the usage time of certain runway configurations throughout this period. The x axis presents the runway configuration in the format of X + Y, where X are the arrival runways and Y are the departing runways. If there are multiple arrival and/or departing runways, then the runways are separated with a comma (,) character.

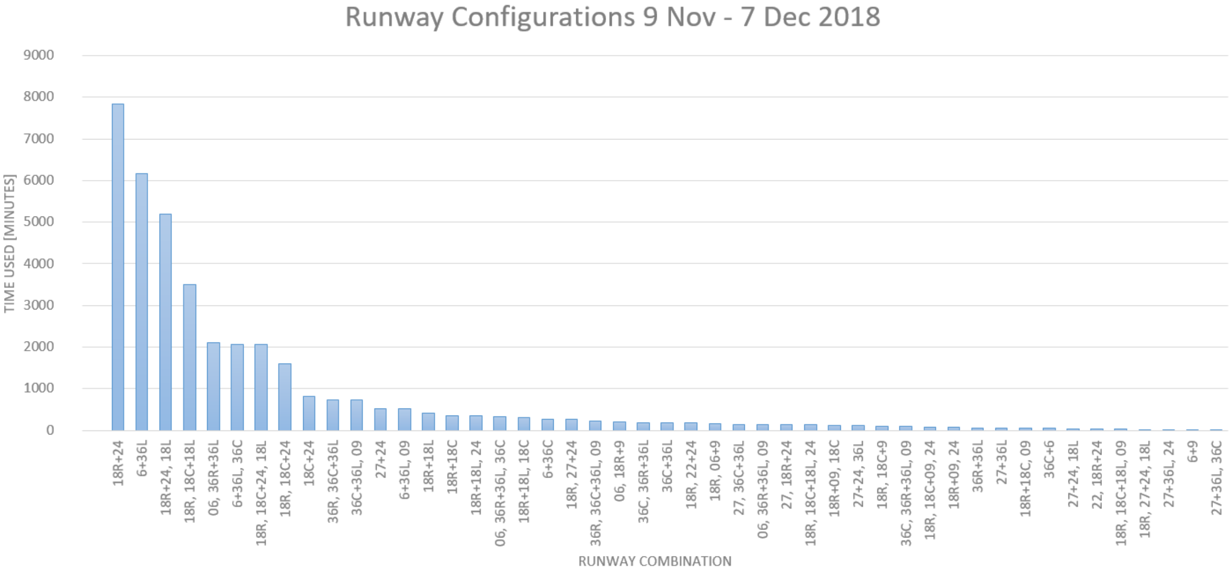

Figure 2.7: Runway configuration time usage at Schiphol between 9 November - 7 December 2018

In total, 48 different runway combinations with a combined total of 38880 minutes (derived from $27 days \cdot 24 hours \cdot 60 minutes$) were observed from the LVNL website data, and no violation of the arrival and departure runway availability of Table 2.1 is observed, further confirming its validity. Furthermore, Figure 2.8 presents the cumulative percentage of the time used of each runway reconfiguration, where 100% is the total time of 38880 minutes.

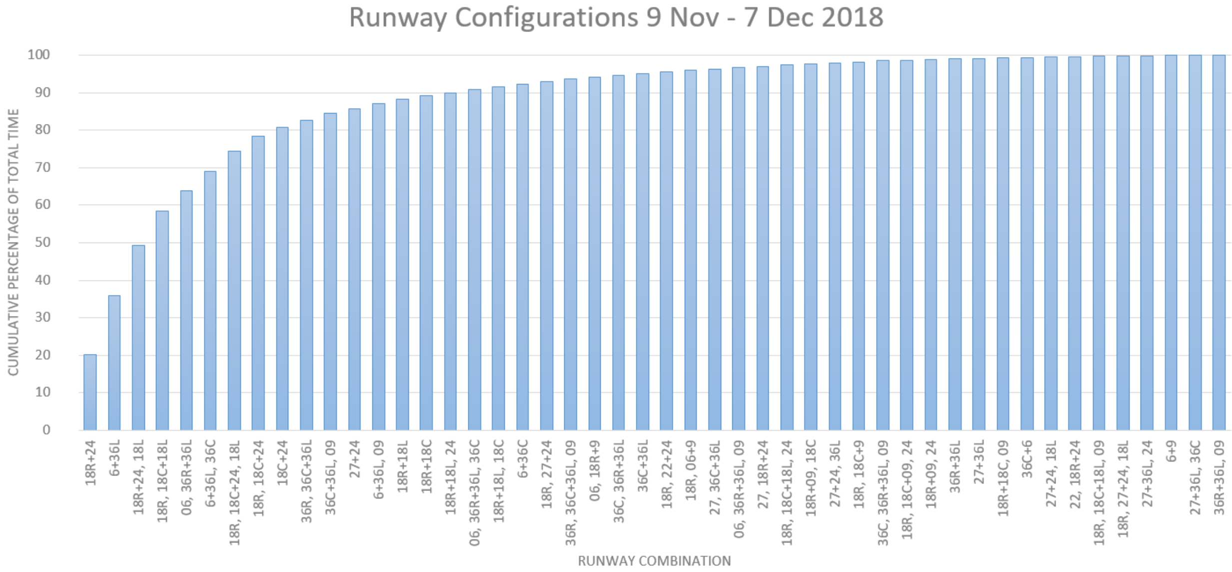

Figure 2.8: Cumulative percentage of total time of runway configurations at Schiphol between 9 November - 7 December 2018

The data shows that the time usage of the first 7 runway combinations makes up approximately 75% of the total time usage for the given time period being considered. In other words, 14.6% of the observed, possible runway combinations are used for 75% of the time period being considered. These are presented in Table 2.2.

Table 2.2: Most commonly used runway combinations at Schiphol

Runway Combination	Time Used [mins]	Percentage of Total Time [%]
18R+24	7835	20.2
6+36L	6170	15.9
18R+24, 18L	5190	13.3
18R, 18C+18L	3505	9.0
06, 36R+36L	2110	5.4
6+36L, 36C	2070	5.3
18R, 18C+24, 18L	2060	5.3

This preliminary investigation shows that it is possible to encompass and potentially model the adverse event of runway configurations at Schiphol utilizing only a certain number of runway combinations, instead of creating seemingly random combinations. This is recommended to further be investigated in this MSc study, as mentioned in the research proposal in chapter 7.

2.3.2. Contributing Factors to Runway Reconfigurations

Now that the types of runway combinations that are possible at Schiphol have been analysed, the subsequent sections will discuss why runway reconfigurations between such combinations occur. This section presents contributing factors to runway reconfigurations.

Weather

Wind direction and strength (wind speed) is an important contributor to determining which runways are active for departures and arrivals. If crosswind or tailwind components are higher than the certified crosswind limitations of aircraft as per the aircraft manufacturer, or higher than the aircraft's operators maximum components, then the aircraft is not able to make a takeoff or landing. As with aircraft, airports also have tailwind and crosswind limitations for active runways. The tailwind component is normally not greater than 10kts, and the crosswind limitations are between 15-25kts [26]. At Schiphol airport, crosswind limitations are set at 20kts and tailwind limits are set on 7kts [5, 18, 27]. The way these components are computed is using basic trigonometry as Figure 2.9 shows.

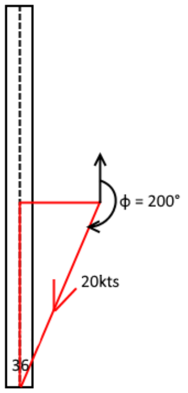

Figure 2.9: Crosswind and head/tail wind determination

The crosswind and tail/headwing component can be computed [26] by Equation 2.1 and Equation 2.2:

$$X_{wind} = V_{wind} \cdot sin(\phi_{wind} - \phi_{rwy}) \tag{2.1}$$

$$Y_{wind} = V_{wind} \cdot cos(\phi_{wind} - \phi_{rwy}) \tag{2.2}$$

A negative value of Equation 2.1 is towards the left adjacent direction of the runway's heading ($\phi_{rwy} - 90^\circ$), and a positive is to the right ($\phi_{rwy} + 90^\circ$). A negative value of is a tailwind, whereas a positive value is a headwind. In the above example in Figure 2.9,the crosswind component is therefore: $X_{wind} = 20 \cdot sin(200 - 0) = -6.8kts$ indicating that the cross wind is 18.8kts towards the 270° heading. Similarly, the headwind component is: $Y_{wind} = 20 \cdot cos(200 - 0) = -18.8kts$ indicating that there is a headwind of 18.8kts. Furthermore, runway availability wind roses can be created in order to incorporate the crosswind and tailwind limitations. Figure 2.10 presents such a diagram, taken directly from the Airport Operations course at TU Delft [5]. This diagram incorporates both the 7kts tailwind and 20kts crosswind limitations at Schiphol, as well as the permissible runway configurations.

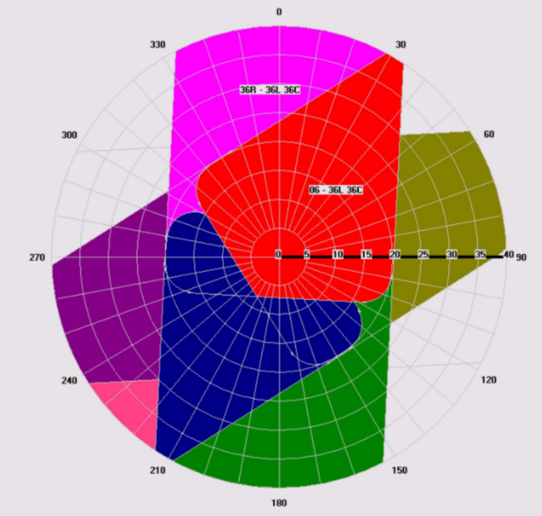

Figure 2.10: Runway Availability Wind Rose at Schiphol [5]

The different colours indicate the different permissible runway configurations. The wind rose diagram is used to assess which runway configurations are able to be used depending on the current wind. The wind rose is plotted for 0 – 359° wind directions and from 0-40kts strength (speed). For example, imagine that on a given day the wind is reported to be 030° with 25kts strength. One would read the wind rose to find the permissible runway configurations that can be used in this condition, by reading the 5th ring from the origin and the radial marked 30. By reading this, it can be seen that the 06/36L runway can be used for departures and 36C for arrivals. However, if the wind is 120° with 30kts strength, then all runways exceed the crosswind and tailwind limitation. This is also confirmed by applying Equation 2.1 and Equation 2.2. These results are presented in Table 2.3.

Table 2.3: Crosswind and Tailwinds for Winds from 120° and 30kts.

Runway	X_wind [kts]	Y_wind [kts]
04	29.5	5.2
06	26.0	15.0
09	15.0	26.0
18L/R/C	-26.0	15.0
22	-29.5	-5.2
24	-26.0	-15.0
27	-15.0	-26.0
36L/R/C	26.0	-15.0

The only runway that is able to be used (within the crosswind and tailwind limitations) is runway 09. However, this would result in single runway operations, and although runway 09 is a runway which is often used for departures (as it also has its own dedicated SID), it is not a common runway for arrivals, and has significantly less landing aids as it only has a VHF Omnidirectional Radiobeacon (VOR) approach [3]. Nevertheless, runways are built based on the most common wind/strength combinations, amongst other factors, and for this reason it can be assumed that such wind conditions rarely occur at Schiphol.

Although wind is a major weather related factor, visibility and cloud ceiling also dictate the types of runways configurations that can be used. If the visibility is between 1500m and 5km and/or the cloud ceiling is between 300 and 1000ft, then visual approaches and special procedures apply to dependent/parallel runways [27]. If the visibility further deteriorates to <1500m, then Low Visibility Procedures (LVP) are enforced which may further influence the special procedures applied to dependent/parallel runways as it visual contact is not able to be achieved. Furthermore, if there are thunderstorm conditions on the approach/departure of certain runways then the operations may temporarily stop in order to prevent the risk of lightning strikes to aircraft.

Noise

Runway use directly affects the noise contours produces on the ground. When certain runways are active, more noise is created on certain approach/departure paths than on others, which is also propagated by the wind direction and speed. For this reason, certain runways and configuration of runways are not permissible in certain time periods [27]; this will now further elaborated upon. Night time is considered between 2230-0600 (Schiphol time) and daytime is between 0600-2230. In the daytime, 2+1 (landing, take-off) runways or 1+2 runways are used. In the night-time, 1+1 is used.

During the night time, runways 06-24 and 18R-36L are the most preferred (weather allowing) as these cause the least noise pollution. If it is not possible to utilize these, then the second alternative is to runways 18C-36C or 09-27. In principle, runway 18L-36R is not used in the night time and runway 04-22 is closed in the night.

Runway Surface Conditions

The runway surface condition is directly linked to the braking action and operational safety of using the runway for take-off or landing. These changes in surface conditions are mainly caused due to weather effects such as downpours, snow, hail and icing. However, hydraulic fluid from aircraft or runway contamination may also deteriorate the runway surface. These are observed by pilot reports of such conditions but also from runway maintenance checks from ground vehicles. If a runway's surface becomes significantly deteriorated, then it may be temporarily deactivated in order to clean it or repair the damage to the surface.

Runway Systems

Runways contain more complex and necessary systems apart from just the tarmac surface. These include the runway lighting system, landing aids, and stop bars. The landing aids dictate the category of approach that is permissible. This, in combination with weather visibility conditions, may result in an approach not being able to take place on the runway and therefore making it unavailable for landing. However, take-off will still be permitted. Furthermore, if any of these systems fails, then it is possible that this technical failure affects the safety of operations and thus the runway is temporarily closed for traffic. Potentially, if a runway is closed and it is possible to do so, another runway may be used temporarily to accommodate the impacted traffic.

Demand

Schiphol, as with other major airport, operates large number of flight movements in wave-like schedules due to hub & spoke models of airlines, as well as short turnaround times. Schiphol operates 2 runways for landing and 1 for departures (2+1) during inbound peaks and 2 runways for departure and 1 for landing in outbound peaks (1+2) [27]. If the inbound and outbound peaks overlap, then a "2+2" configuration is possible with 2 runways open for departures and 2 runways open for landing during the transition from inbound to outbound peaks.

Emergencies

Emergencies, and aircraft accidents, are rare events that may require certain runways to be closed or others to be opened in an unplanned way, or even all operations to stop at the airport and flights to be diverted. For this reason these are the most unpredictable events which have significant impact on the runway configurations.

2.3.3. Planned Runway Reconfigurations

Planned runway reconfigurations are runway reconfigurations that can be prepared and scheduled for in advanced of the time when conditions of the factors previously mentioned dictate that a runway is not able to be used/should be used. This is how the vast majority of runway reconfigurations are executed by air traffic control. The operations of planned runway reconfigurations at Schiphol airport will now be described based on expert and ATC discussions as well as a previous MSc thesis [18]. In general, the planning of runway reconfigurations can be characterized in 2 stages of (time) dependent planning: tactical and operational planning.

Tactical Planning

At the evening of each day, tactical planning is performed through a briefing by ATC in order to determine the runway schedule for the following day until about mid day. The flight schedule for that time period is assessed and runway combinations based on the inbound, outbound and off- peaks are chosen whilst also utilizing weather forecasts of the same time period. More specifically, the flight schedule is assessed to determine when an inbound/outbound peak will occur, and therefore when it is best to use a 2+1, 1+2 as well as associated temporary 2+2 runway combination. The weather forecasts are used to determine the runway direction/which combination of runways can be scheduled. Furthermore, noise restrictions due to the time of day are also taken into account. This briefing and scheduling is then performed at 2 other time points during the day, with associated time periods. The first is performed in the morning of the following day which covers the runway schedules time period from midday to evening and the second is performed at midday which covers the time period of the rest of the evening. In this way, a runway schedule is determined for each day at 3 different time points in order to account for the dynamic and variable nature of both the flight schedule and meteorological conditions.

Operational Planning

About 1 to 0.5 hours prior to an inbound, outbound or off-peak runway combination transition, the ATC supervisors perform operational planning by discussing with the active controllers about the scheduled runway configurations. The approach and tower supervisors of an airport discuss, verify and conclude on the final runway combination that will be used. This is done by assessing the planned runway schedule, current as well as short-term meteorological conditions, outbound/inbound peak times, time of day, as well as the operational situation such as if there are many aircraft waiting to depart/already taxiing to a certain runway. If the runways are changed from landing runway 18R to take-off 36L, for example, and aircraft are already being sequences for 36L, then they must wait short of taxiway Victor (or at the stand) until the last arrival lands and vacates 18R, as this will need to taxi along taxiway Victor first (see Figure 2.11).

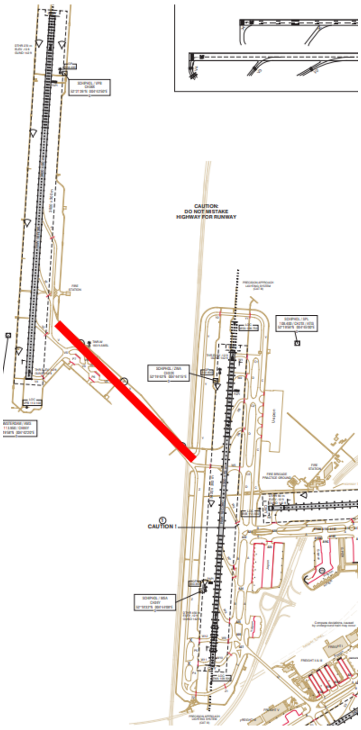

Figure 2.11: Taxiway Victor bottleneck [3].

This creates long delays for departing traffic [18]. For this reason, the supervisor discussions may conclude to temporarily open a different runway for take-off to prevent such unnecessarily long waiting times.

This overall operational planning provides a more real-time alteration/validation of the planned runway schedule. Furthermore, a time of change is decided upon in order to take into account the sequencing of inbound and outbound traffic flows. Once all of this has been determined, the runway combination as well as the activation time (time that new new runway reconfiguration will be active) is entered in the ATC and CDM systems which can be used by controllers in order to plan and schedule their individual goals as required. It is important to note that the activation time is not necessarily a very strict time point, but is more of a target indication, therefore a time window may be more appropriate to be used. This is recommended to be further investigated and modelled in the MSc study, and will be discussed in chapter 7.

2.3.4. Unplanned Runway Reconfigurations

Unplanned runway reconfigurations are runway reconfigurations that occur without the tactical planning of a runway schedule, but with operational planning (presented in subsection 2.3.3), usually taking place in a very short time period (minutes) after a triggering event that results in the state of a runway suddenly having to change (from open to closed or vice versa). In other words, the sudden deviation from a currently active runway combination is a form of unplanned runway reconfiguration. Such trigger events that result in unplanned runway reconfigurations are generally unforeseeable events such as failures or deteriorations below minimum operational levels of runway surface conditions or runway systems, resulting in runways suddenly having to close and, potentially, others to open in order to continue accommodating the demand of traffic, or emergency aircraft that may land on runways that were not active or may have to stop and temporarily close a runway.

2.4. Operational Data

In order to complete the literature review for the airport surface movement operations at Schiphol, data sources which could be used to quantify and characterize the operations are researched. This is useful in case the real world operations need to be used/consulted during the design for verification and validation reasons. Additionally, it may be interesting to consult this data in order to compare the final outcomes of this thesis to the real world operations at Schiphol.

As previously mentioned, LVNL has a publicly available website [25] that includes the active landing and departing runways from 1 February 2018 - present, taken at 5 minute intervals throughout the day. This could be used to further analyse the types of runway combinations that occurred at Schiphol.

The Koninklijk Nederlands Meteorologisch Instituut (KNMI) provides meteorological related data on a publicly available website on both an hourly [28] and daily [29] basis for a time period from the 2000s until the present. This contains a wide range of information from the Schiphol weather station about the wind, visibility, cloud, rain, pressure as well as on more meteorological factors. Furthermore, historical Meteorlogical Aerodrome Report (METAR) or Terminal Area Forecasts (TAFs) also contain the aviation published meteorological conditions, published almost every 20-30 minutes at Schiphol airport, with a public database found on a website [30].

As mentioned in section 2.1, Schiphol utilizes the A-CDM system in order to log all timestamps and states of ground movements. Although access to this type of data was not found to be publicly available, it may be possible to access this and utilize it in order to determine real world timings associated with flight movements, such as push back timings and estimated taxi timings, or this could be used to compare to the final results of this thesis simulation.

Finally, surveillance data which contains the trajectories of ground movements at Schiphol such as Automatic Dependent Surveillance Broadcast (ADS-B) or multilateration (MLAT) can be found on third party websites [31]. The already existing TU Delft ground simulator of Schiphol (which will be presented in chapter 7) also contains the detailed surveillance data of 2 weeks worth of flights at Schiphol between 01-05-2016 and 15-05-2016.

3

Decentralized Control: Follow the Greens

Decentralized control in a system is where lower level components operate and undertake decision making based on their own local information to accomplish global goals through the emergent behaviour of the system. This is different to that of centralized control where decision making and control is performed by a singular central controller element which instructs/controls lower level components. Figure 3.1 presents a graphical representation of these approaches.

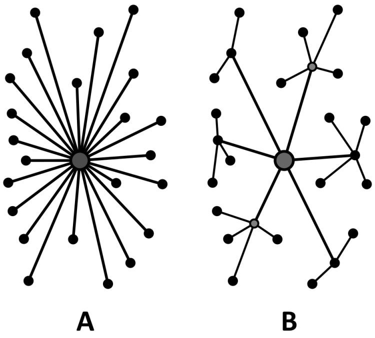

Figure 3.1: Centralized (labelled A) vs decentralized approaches (labelled B) [6]

Decentralization in complex systems such as large power (electricity) systems [32], coordinated plan executions of robots [33] as well as in air traffic control [34] all have promising results with respect to achieving the global goals of the systems. In particular, decentralization with respect to airport surface environments [4, 35] has also been shown to have promising effects with respect to controlling aircraft on the airport's surface. In these literatures, virtual agents were placed on the taxiway intersections that act on their local information to direct and guide aircraft along pre-defined routes. From this literature [4], it was concluded that such a decentralized control for airport surface movement operations at Schiphol airport resulted in:

- Only a limited amount of information being required to allow safe operations.
- Local information is not enough to accomplish good performance.
- The ability to handle a more complex and chaotic operation.
- Effect of coordination only becomes visible for highly congested taxiway networks.

For this reason, especially the first and third bullet point, such a decentralized control system is considered promising to be further expanded upon with respect to system disturbances such as runway reconfigurations and will be used for this MSc study.

The Single European Sky (SES) framework has been created by the European Commission, which aims to provide a legislative framework to meet future air transport safety, capacity and efficiency needs at a European level instead of at a national level [36]. As part of this framework, a technical pillar of the SES initiative was established called the Single European Sky for ATM Research (SESAR) which is a public-private partnership (Joint Undertaking) with the aim to harness the research, innovation expertise and resources of the entire ATM community [37]. One outcome of the SESAR programme is the concept of an Advanced-Surface

Movement Guidance and Control System (A-SMGCS) called Follow-the-Greens [38, 39]. This is a form of a decentralized control system for airport surface movement operations based on Airfield Ground Lighting (AGL) that will be used as a case study of decentralized control for airport surface movement operations. The global goal of any airport surface movement system is to provide services to aircraft and vehicles in order to maintain airport throughput under all local weather conditions, whilst maintaining the required level of safety [7]. This chapter is structured as follows: section 3.1 presents the general concept of the Follow-the-Greens, section 3.2 presents the SESAR validations of the Follow-the-Greens, section 3.3 presents the Eurocontrol A-SMGCS specifications and system outlines which are based on the Follow-the-Greens concept and section 3.5 presents a socio-technical representation of such a system.

3.1. Concept

Literature [39] provides a good source for understanding the general understanding of the Follow-the-Greens. The concept of this system is that taxi routes for all aircraft or vehicles on the airport's surface are computed by decentralized system elements based on origin-destinations. After the routes have been computed, the paths are communicated to aircraft/vehicles through the airport surface environment by using AGL visual navigation. Green taxiway center lights are illuminated along a defined taxiway segment in front of aircraft (or ground vehicles) to guide them along their cleared route from origin to destination, such as from the gate to the runway. All other taxiway center lights not included in the cleared route are deactivated. As the aircraft (or ground vehicle) taxiis along the route, the green lights over which it traverses are switched off and a further green light is illuminated at a point furthest away from the aircraft on the taxiway segment, giving the impression that the aircraft is "pushing" the lit green segment ahead of it. In this way, it is possible for the aircraft or ground vehicle to taxi along its route. If the aircraft is required to stop, for example due to traffic, red stop bars are illuminated, interrupting the taxiway route and making the flight crew wait for further green lights to follow. If no stop bars are present, then the green lights simply "stop" at an intersection indicating the aircraft/vehicles must await for further green lights.

The concept of using AGL by following green lights for a progressive taxi system is not a new concept, being used at airports such as London Heathrow and Munich Airport for the past 20 years in a manual or semi-automatic switching way [39], using a human operator. However, the new Follow-the-Greens concept from SESAR is based upon an automated decentralized control system which is able to assign taxi routes, manage taxiway illumination, as well as manage or alert ATCOs of potential conflicts. Eurocontrol has included such a concept in its latest specification for A-SMGCS services [7] which indicates its acceptance at an industry level, and will be discussed in section 3.3. Furthermore, this concept has also been validated by the SESAR program, which will be elaborated upon in section 3.2.

This overall concept is different to the currently established ATC procedures of an ATCO having to centrally compute paths for all aircraft or vehicles on the airport surface, as well as manage all the traffic, which is heavily based on the cognitive ability of the ATCO as well as traffic and workload factors. Furthermore, adverse events such as runway reconfigurations increase the complexity associated with carrying out the ATCO's goals as a ground controller due to the increased work load, traffic merging, de-merging and sequencing required. Additionally, these routes are communicated through radio communications where each specific taxiway segment in the path is transmitted from ATC to aircraft which in turn reads them back creating radio congestion. Other instructions such as hold short, or stopping instructions are additionally communicated to the aircraft through radio. This currently adopted ATC operation was described in more detail in chapter 2.

3.2. SESAR Validation

The Follow-the-Greens has been validated through a real-time simulation of Munich airport [40], developed by ATRiCS [41]. Two scenarios were conducted: one in clear weather conditions during the day time and the other in low visibility conditions. Both scenarios focussed on the apron segments of Munich airport, where flight crews were given instructions to taxi to a holding point at the exit of the aprons. During this validation of this concept, improvements in terms of capacity, environment, human performance, safety and predictability of airport surface movement operations through the use of this system were confirmed [40, 42].

However, during the SESAR validation, the scenarios that were used were restricted to the apron segment only which means that its feasibility and validity in other airport manoeuvring areas remains unknown. Furthermore, no regularly occurring adverse events that take place in the airport environment, such as runway reconfigurations, were taken into account, nor were any specific mechanisms with respect to how routes and de-conflicting services were computed. This therefore shows that this validation is limited and that this MSc research is able to contribute to this.

3.3. A-SMGCS Specifications

As previously mentioned, Eurocontrol has included this decentralized control concept to its latest A-SMGCS specifications manual [7]. It is included as the "Automated Switching of the Taxiway Centreline Lights (TCL)" system. This is a Follow-the-Greens system which provides individual guidance to any mobile (aircraft, ground vehicle or any other cooperative system) which has a cleared route, as per the A-SMGCS services. For this reason, it is interesting to further explore these.

There are, in total, four main contributors to the services which the Follow-the-Greens A-SMGCS provides. These are shown in Figure 3.2.

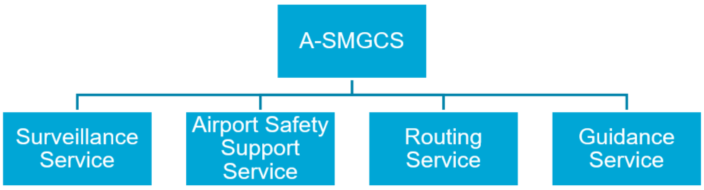

Figure 3.2: Contributors to an A-SMGCS system service.

ATCOs monitor all traffic and decisions made by the A-SMGCS through the the Human Machine Interface (HMI). Based on the information presented through this interface, clearance confirmations, alterations and other interactions are able to be selected and "shown" to flight crews/vehicles through the AGL. This is a clear shift of the ATCO's responsibility to a supervisory role, where the A-SMGCS plays a more dominant control role.

These A-SMGCS elements will be described in subsequent sections of this chapter based primarily on elaborations from the found literature of the latest Eurocontrol A-SMGCS Specifications [7] and ICAO Doc 9830 A-SMGCS Manual [43]. It is interesting to explore these as they form a basis of the state-of-the-art application of decentralized control for airport surface movement operations.

3.3.1. Surveillance Service

The surveillance service is the core A-SMGCS service, which, once established, allows the 3 subsequent services to be established. This service provides a representative situation on aerodrome traffic through identification, position and tracking of both aircraft and vehicles within the aerodrome surface and vicinity.

The transponders of aircraft provide the surveillance service with the location and identification of aircraft/vehicles on the airport surface. The flight crew is responsible for turning on the transponder when required, according to aerodrome procedures. This is, in the vast majority of cases, done when a clearance is issued from the controller to the flight crew, and the flight crew is ready for push-back and start up. A typical transponder scenario, as stated in the Eurocontrol manual, is shown in Figure 3.3.

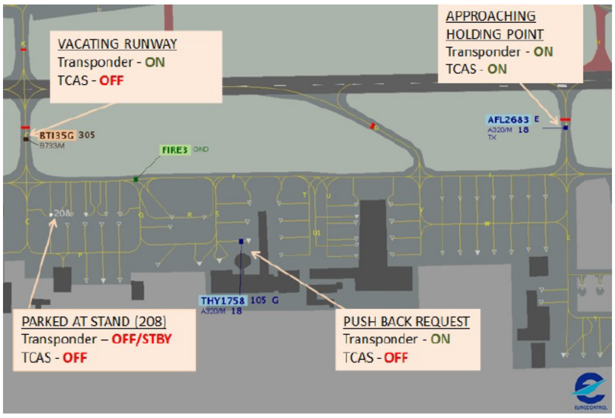

Figure 3.3: Transponder settings on the airport surface [7]

Furthermore, every vehicle operating on the airport surface manoeuvring area (which does not include the service roads) must have a vehicle transmitter ensuring detection and identification by the surveillance service. Callsigns and identification criteria are aerodrome specific and depend on the types of operations that they undertake on the airside.

3.3.2. Airport Safety Support Service

The airport safety support service enhances the safety of aerodrome operations by utilizing the features from the 3 other services (surveillance, routing and guidance). From this, the Eurocontrol specification states that it should be able to:

- Anticipate potential conflicts and hazards
- Detect conflicts and incursions
- Detect mobiles that are not following clearances
- Provide alerts

In order to achieve these abilities, this service is compromised by three specifically defined functions, as follows:

- Runway Monitoring and Conflict Alerting (RMCA): a short term conflict alerting tool which monitors moments near the runway and is able to detect conflicts.
- Conflicting ATC Clearances (CATC): provides an early prediction of a situation that may end up in a hazardous situation if not corrected.
- Conformance Monitoring Alerts for Controllers (CMAC): provides controllers with alerts when non-compliance to procedures or clearances of traffic on any of the airport surface.

These systems aim at helping the controller with assisting their working methods and following local procedures, as well as building confidence in the situational awareness. In addition, the conflict detection/infringements are implemented according to ICAO legistlations [22], including multiple runway line-ups (lining up in sequence or an aircraft lining up at an intersection further upwind from another aircraft), as well as intersection departures.

If alerts are triggered (which are both audible and visual), then the controller must verify the event and determine the appropriate action. Based on this, an instruction/clearance is issued to resolve the situation.

However, although descriptions of the specifically defined functions (RMCA, CATC, CMAC), the actual mechanisms and algorithms are not described or specified.

3.3.3. Routing Service

The routing service generates a route for each ground mobile based on the aerodrome infrastructure, parameters or by controller interaction. It is the prerequisite for the guidance service, as well as for updating aircraft times in the Airport-Collaborative Decision Making (A-CDM) platform. This is used for further aircraft calculations with respect to push back times, start-up times and take off times, as discussed in chapter 2.

Routes are airport surface routes are system generated based on the current operational aerodrome parameters from the Airport Operation Status (which will be discussed in subsection 3.5.1):

- Runways in use
- Taxiways in use

In addition to Mobile Information Database (which will be discussed in subsection 3.5.1) information which relates to each mobile. For aircraft the following information is stored and used for this:

- Identity
- Type
- Flight Plan
- SSR Code
- Stand
- Clearances
- Planned route
- Cleared route
- Assigned runway

- Timing information
- De-icing information
- Aircraft status

And for vehicles:

- Identity
- Type
- Type of movement
- Clearances
- Planned route
- Cleared route
- Timing information

Before a ground mobile starts moving on the airport surface, or before an arrival has landed, the system generates a planned route. When the controller authorises the planned route of the mobile to start moving, the route becomes a cleared route and the route status changes to cleared. This is up to a clearance limit, such as a holding point of a runway. Following the clearance, any part of the planned route that has not yet been cleared (such as if the route is only cleared to an intermediate holding point) is defined as a pending route which requires clearance to continue. Planned and cleared routes are shown to controllers through the HMI, as per the Eurocontrol A-SMGCS manual. An example of this is presented in Figure 3.4.

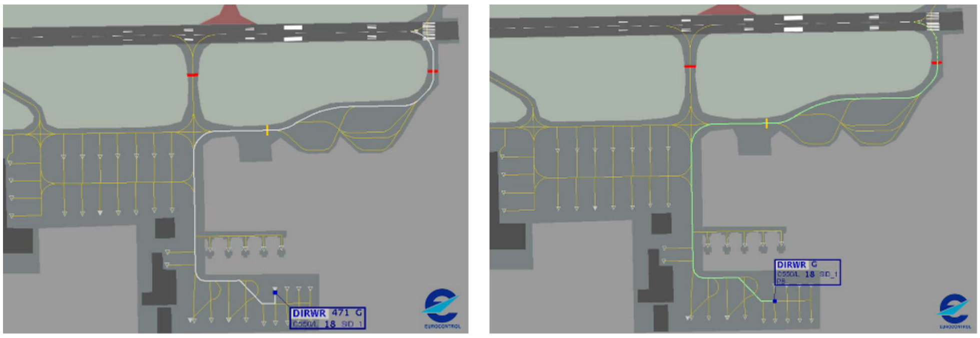

Figure 3.4: Planned (left) and cleared (right) route statuses [7]

The A-SMGCS specification also goes on to specify that planned routes are generated without controller interaction, when the start and end point of the trajectory is known. In most cases, this is the assigned runway and parking stand for an aircraft, or for two positions on the airport's surface, for vehicles. The route is generated based on:

- Known rules/procedures
- Shortest distance
- Standard taxi routes
- Category of taxiway and type of aircraft
- Actual & planned taxiway closures
- Restricted areas
- Assigned runway

Furthermore, the Eurocontrol specification states that a route should be responsive to operational changes such as runway changes or temporary hazards. This route planning occurs, typically, before the flight plan activation or when all necessary information is available. Furthermore, controllers are also able to modify or create new routes before or after the ground mobile has begun moving.

The Eurocontrol specification also states that controllers are responsible for monitoring and adjusting the generated routes, especially due to additional constraints that are not known to the routing service, such as other aircraft pushing back. When the initial clearance is issued to aircraft, it is the controller's responsibility to simultaneously enter the clearance input in the HMI, which then changes the route statues from planned to cleared. This is, also, the same workflow for route alterations.

However, no specific mechanisms, algorithms or computational methods are specifically mentioned as to how routes can be created, coordinated and combined with the individual routing plans of other airport surface users, although only important elements are mentioned. Furthermore, although it is stated that the routes should be able to respond to operational changes such as runway changes, no means or methods of how this should be done are presented. This is an additional research gap in this specification, which this MSc study will also help to contribute towards. This gap is an additional motivation as to why resilient behaviour and cooperative coordination mechanisms should be researched as potential mechanisms that could be integrated into this service.

3.3.4. Guidance Service

The guidance service is responsible for providing individual guidance information using visual aids to any mobile which has received a cleared taxi route. It is comprised by, in total, three functions:

- Automated switching of Taxiway Centreline Lights (TCL)
- Automated switching of stop bars
- Automated activation of Advanced-Visual Guidance Docking Systems (A-VDGS)

Automated Switching of Taxiway Centreline Lights (TCL)

This function provides individual guidance to any ground mobile that has received a clearance by illuminating taxiway lights to a specified distance ahead of the ground mobile, by switching the lights on and off. This can either be done one at a time, or in a group of lights in a short segmented. This is based on the infrastructure of the airport, and the length of the segmented that can be illuminated is dictated by the topographical influences of the taxiway lights, and aerodrome layout. This is synonymous to the "Follow the Greens" as the switching of taxiway lights gives the impression that aircraft are following green lights. The HMI projection to the controller, as per the Eurocontrol manual, is shown in Figure 3.5. It is not specified which algorithms and mechanisms are used in order to create this "Follow-the-Green" effect but it is assumed that the surveillance information from the surveillance service is used in additions to the ground infrastructure to determine which green lights to illuminate/turn off.

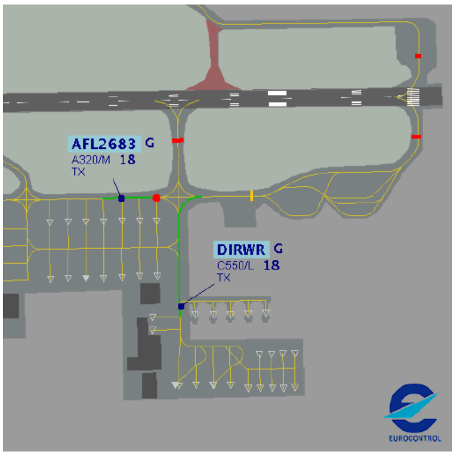

Figure 3.5: Follow-the-Green HMI [7]

Spacing between mobiles is the responsibility of the flight crew or vehicle drivers in good weather conditions. For this reason, the Automated TCL does not provide this spacing in good weather conditions. Instead, the guidance service takes other traffic into account for spacing in order to guide the ground mobile along its cleared route, and allocates priority between mobiles based on local operating rules such as:

- Runway exit vs parallel taxiways
- Aircraft vs vehicle
- Aircraft converging or crossing at intersections
- Taxiways passing close to push back routes
- Taxiways where insufficient wingtip separation exists

However, the types of mechanisms, communications and control used to achieve this are not specified. For this reason, when and how such routes (plans) must be shown on the AGL (i.e. executed) must be further investigated. This will be presented in the research proposal in chapter 7.

A typical recommended controller phraseology for this is: "[Callsign], Follow the Greens to Holding Point X, runway Y (or Stand Z)". This can be issued to any A-SMGCS cooperative ground mobile (such as aircraft and ground vehicles). Simultaneously to this, the controller confirms this route in the HMI which in turn triggers the appropriate TCL to illuminate. The controller then proceeds to monitor all TCL segments for traffic on the HMI.

After the flight crew (or vehicle drivers) read back these instructions, it is their responsibility to follow the green TCL segments, if they are illuminated in front of them. If there are no green lights illuminated in front of them, or if the TCL lights are red or a stop bar is illuminated, then the flight crew must stop the aircraft and hold position. Furthermore, if the TCL lights are yellow or they are flashing, the flight crew must exercise caution whilst moving.

Automated Switching of Stop Bars

This function provides the capability to switch stop bars on or off follow a clearance, as required. When a stop bar is illuminated it acts as a "red traffic light", signalling to flight crews that they must stop moving. The stop bars can either be at a holding position at a runway, or across a taxiway. If they are placed against a taxiway, they can also be used to enforce further separation along taxiway sections in LVP. According to ICAO regulations [22], the TCL segments must not be activated at least 90m after a lit stop bar, in order to not confuse flight crews.

Once a take-off/line-up/enter or cross runway clearance has been issued in the HMI, by a controller, this service uses the surveillance service position to switch runway holding position stop bars on or off, when the ground mobile is a set distance from them. If the aircraft is simply taxiing, and there are taxiway based stop bars, then this service, again, uses the distance from the stop bar to automatically control if it should be on or off. Furthermore, after the traffic has passed over the stop bar, it is automatically turned "on" again using either local sensors (on the taxiway infrastructure), or using the distance from it.

Automated Activation of Advanced-Visual Docking Guidance Systems

The A-VDGS is a docking guidance system that guides aircraft park in the correct parking position, depending on the aircraft type. In addition to this, the A-VDGS provides more information from the A-CDM system such as target off block times, or estimated start up times.

3.4. Research Gap

It is clear that the Eurocontrol specifications form a solid architectural basis for a state-of-the-art decentralized airport surface movement control system, and the role of the ground controller is now shifted to that of a supervisory role. However, it has also been determined that the intricate details about the specific types of mechanisms that could be used in this system are lacking. Out of the 4 pillars of the A-SMGCS system, the routing service and guidance service pose particularly interesting candidates to investigate the types of mechanisms that could be used to achieve their individual goals (which will be described in subsection 3.5.1). In order to shift the role of the ground controller to a supervisory role, these services have to be able to provide conflict free and coordinated routing plans for all aircraft on the airport's surface, in order to minimize the amount of control inputs that the ground controller is required to utilize. Furthermore, it is clear that these elements must be able to deal with the adverse events of planned and unplanned runway reconfigurations such that the airport surface movement operations remain as unaffected as possible. Additionally, the limitations of the SESAR validation described in section 3.2, further form this research gap with respect to how such a system performs on a larger airport surface manoeuvring area.

Therefore, these identified research gaps present an interesting area that this MSc study will additionally be able to contribute towards. For this reason, approaches that result in resilient behaviour of systems and cooperative coordination mechanisms will be elaborated upon in chapter 4 and chapter 5 respectively.

3.5. Socio-Technical System Representation

Now that the A-SMGCS and Follow-the-Green concepts have been elaborated upon, a high-level socio-technical represents of this system is created and presented in Figure 3.6. This is an augmentation of the the socio-technical system representation of airport surface movement operations, presented in section 2.2, thus showing how such a system fits in the overall airport surface movement operations. In other words, all of the decentralized elements of the Follow-the-Green A-SMGCS system are required in order to achieve the same goal of the ground controller as mentioned in section 2.2, and change the interaction of how the ground controller issues clearances to aircraft. Instead of the ground controller interacting directly with the aircraft by radio (interaction 5 in section 2.2), the controller now interacts specified cleared routes through the taxiway environment, which the aircraft observe and subsequently follow. This, therefore, significantly alleviates the work of the ATC who now has more of a supervisory role, and the ATCO and airport surface movement operations could be able to enjoy the benefits (improved capacity, safety, etc.) of such a decentralized control system presented by the SESAR validation in section 3.2.

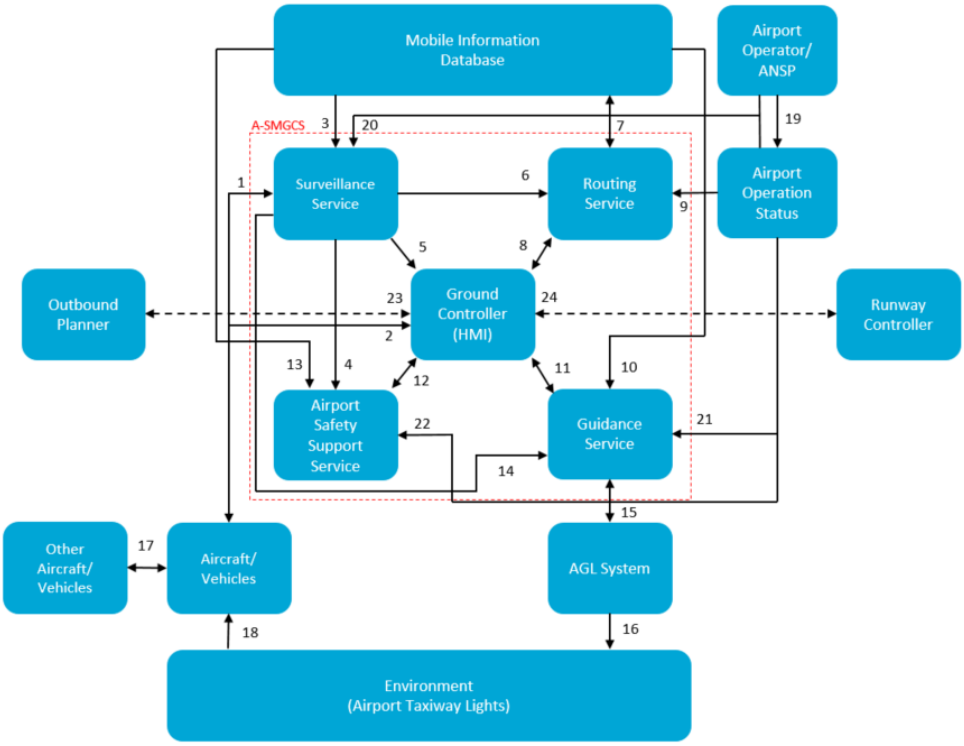

Figure 3.6: Socio-Technical Representation of the Follow the Greens A-SMGCS system

3.5.1. Description of Elements

This section describes the elements that are included in the socio-technical representation of the Follow the Greens A-SMGCS system, presented in Figure 3.6.

Ground Controller

The ground controller's role is responsible for supervising the four different services of the A-SMGCS, as well as monitoring all airport surface movement operations. This includes confirming system generated taxi route clearances of aircraft through the HMI, making any alterations as required, as well as listening for clearance requests from aircraft. Furthermore, they are still responsible for communicating and coordinating with the outbound planner and runway controller ATCOs, as specified in section 2.2. The ground controller is also responsible for giving pushback clearances and monitoring that all traffic is following their predefined routes and that no conflicts or deviations from any clearances occur. The goal of the controller is to provide each aircraft aircraft (or ground vehicle) a safe and efficient movement along the airport surface, from their origin to their destination (such as from the runway to the gate, or vice versa) by supervising the decisions of the A-SMGCS system.

Surveillance Service

The surveillance service is responsible for collecting and fusing flight and vehicle surveillance data from non-cooperative and cooperative (ADS-B, MLAT) systems [7]. The goal of the surveillance service is to provide the Routing Service, Ground Controller (HMI), Airport Safety Support Service and Guidance Service with comprehensive, up-to-date data about the position of objects detected within the airport's surface.

Routing Service
The routing service is responsible for generating routes for mobiles on the airport's surface, adhering to aerodrome taxi procedures and ensuring efficient, conflict free and coordinated routes. The goal of the routing service is to generate valid origin-destination routes for mobile units which have been identified as per the Surveillance Service and the Mobile Information Database. This is done using three internal states depending on the status of a ground mobile's route: planned (when a target with an origin-destination is received from the Surveillance Service and Mobile Information Database), cleared (when the Ground Controller issues the appropriate authorisation input through the HMI) and pending (for certain portions of the route that have not been given appropriate authorisation by the Ground Controller). This logic has been previously defined in subsection 3.3.3.

Guidance Service
The guidance service is responsible with determining how to operate the TCLs, and determining which segments and lights must be activated and when (including when green segments should stop to allow other traffic to pass), by operating the AGL. This should be done for all generated routes from the routing service. Furthermore stop bars and A-VDGS is also controlled by this service. The goal for the guidance service is to operate the AGL in such a way such as to illuminate the TCLs for cleared routes of ground mobiles generated by the Routing Service, which are stored with respective route status in the Mobile Information Database.

Airport Safety Support Service
The airport safety support service is responsible for detecting alert situations and triggers these alerts to notify the controller. It is responsible for detecting conflicts and incursions through an early prediction, as well as providing the RMCA, CATC and CMAC functions (see subsection 3.3.2). In order to achieve this, it uses the information received from the Surveillance Service. The goal of this service is to detect alert situations and trigger alerts to notify the ground controller to them. In turn, the Ground Controller resolves the conflicts.

Aircraft/Vehicles
Aircraft or vehicles are operated by flight crews/vehicle drivers which form the ground mobile users that are responsible with ensuring their safe movement from origin to destination, operated by either flight crew or vehicle drivers. They do this by communicating and following instructions from the ground controller, as well as observing the TCL to follow routes to their destinations. Furthermore, they observe other traffic (Other Aircraft/Vehicles block in Figure 3.6) in order to maintain separation and ensure their safety. The goal of aircraft/vehicles is to enable them to reach their destination in a safe and efficient way.

AGL System
This system is responsible for operating lights on the airfield in such a way as defined by the guidance service, in order to give the effect of Follow-the-Greens. The goal of this system is to execute light commands as required by the guidance service.

Mobile Information Database
The mobile information database is responsible for storing all information related to each ground mobile, as required by all services. A more detailed list of the types of information stored was presented in subsection 3.3.3. The goal of this database is to respond to requests for information from other services, and return the relevant information that they require.

Airport Operation Status
The airport operation status is responsible for providing a representative operational picture of the airport operations. It includes determining the active runways, active taxiways, as well as the operational aspects of the aerodrome layout (infrastructure layout, reference points, taxiway constraints due to wing tip distance and fixed obstacles). The goal for this element is to determine a representative picture of the airport operational environment and provide the Surveillance, Routing and Guidance Services with this information.

Airport Operator/ANSP

This element is a representation of the approach/departure segment, as well as other Area Control Centre (ACC) segments. It is responsible for interfacing certain information and environmental aspects to the A-SMGCS (ground) system. The goal of this system (within the context of the A-SMGCS socio-technical system representation) is to provide information about runway configurations and runway changes to the Airport Operation Status, as well as passing important approach surveillance information to the surveillance service.

3.5.2. Description of Interactions

Now that the elements involved in this socio-technical representation have been described, the types of interactions between them will be discussed. This will be done using the numbers presented in Figure 3.6.

1: Surveillance Service to Aircraft/Vehicle

The surveillance service sensors interrogates the aircraft/vehicle transponder which request a position report as well as an Secondary Surveillance Radar (SSR) code identifying the target.

1: Aircraft/Vehicle to Surveillance Service

After an interrogation is received, the transponder on board the aircraft/vehicle responds with a position report and SSR code identifying the target.

2: Aircraft/Vehicles to Ground Controller

After a clearance has been received by the delivery/clearance controller and all pre-flight preparations have been completed (such as boarding, re-fuelling, baggage loading, etc.), the flight crew of the departing aircraft establish that they are ready for push-back and start-up. Once this has been established, they contact the ground controller (through VHF radio communication) requesting this clearance. For arriving aircraft, the flight crew contacts the ground controller after they have vacated the runway and request taxi to the stand. Furthermore, throughout any time point whilst the aircraft is under the responsibility of the ground controller, the aircraft's flight crew are able to contact the ground controller with any issues or requests that they may require for their goal. Vehicles interact with the ground controller in the same way. Communication is made through ATC phraseology [23].

2: Ground Controller to Aircraft/Vehicles

The ground controller communicates with the aircraft/vehicles through VHF radio communications. The ground controller responds to any requests that aircraft/vehicle make, by providing clearances in ATC phraseology [23]. This includes push-back, start-up, as well as taxi clearances. Furthermore, any hold short or give-way to other traffic commands are also communicated through this. However, using the Follow the Greens procedure, it is expected that only a "Follow the Greens" clearance is issued to traffic, as the AGL will include the hold short or give-way commands. Additionally, the ground controller can issue any further commands to traffic for any time point whilst the traffic is under the responsibility of their control.

3: Mobile Information Database to Surveillance Service

The surveillance service interacts with the mobile information database based upon the SSR code which has been received by the data fusion of its sensors. It searches for the information associated with the SSR code. The information found is as per that presented in subsection 3.3.3, and is used to completely identify the aircraft/vehicle target.

4: Surveillance Service to Airport Safety Support Service

The airport safety support service receives the identity and current position of each mobile (at each time step) from the surveillance service. This is used in order to keep track of all mobiles and associated trajectories upon the airport's safety, and assess whether any alerts are required.

5: Surveillance Service to Ground Controller

The surveillance service presents the position and information of all identified aircraft and vehicles on the HMI (which in turn is what the ground controller observes).

6: Surveillance Service to Routing Service

The routing service receives the aircraft or vehicle identity and current position of each ground mobile from the surveillance service (for each time step).

7: Mobile Information Database to Routing Service

The routing service receives information depending if the ground mobile is an arrival, departure or vehicle. For departure aircraft, the following information is received:

- Additional flight information
- Aircraft type
- Stand
- Estimated Off Block Time (EOBT)
- Runway entry point
- If de-icing is needed (if yes, then subsequent de-icing information)

For arrival aircraft:

- Additional flight information
- Aircraft type
- Estimated Landing Time (ELDT)
- Runway exit
- Allocated stand

And for ground vehicles:

- Aircraft registration (if towing)
- Aircraft type (if towing)
- Identifier (if ground vehicle only)
- Origin position
- Destination position

Additionally, manual route inputs and clearances are also received.

7: Routing Service to Mobile Information Database

Routes, additional clearances and manual entries (as received from interaction 8 from the ground controller) are sent to the specific directory of the aircraft/vehicle in the mobile information database.

8: Routing Service to Ground Controller

The routing service presents information of planned, cleared and pending routes to the HMI of the ground controller, which in turn is observed by the ground controller.

8: Ground Controller to Routing Service

The ground controller is able to alter routes or enter manual clearances. These are entered through the HMI and then received by the routing service (which in turn is able to save this information in the mobile information database through interaction 7).

9: Airport Operation Status to Routing Service

The routing service receives the following information from the airport operation status:

- Aerodrome layout
- Runway and taxiway status
- Taxiway configuration
- Standard taxi routes
- Intermediate route points
- Active constraints
- ATC rules/procedures

10: Mobile Information Database to Guidance Service

The guidance service receives all information related to clearances and cleared routes of all aircraft/vehicles from the mobile information database.

11: Guidance Service to Ground Controller

The guidance service presents information as to which TCL are being illuminated/activated to the HMI of the ground controller, which is in turn observed by the ground controller.

11: Ground Controller to Guidance Service

The ground controller is able to select the current status of lights for all lights, or just those concerning a specific mobile (in order to de-clutter the light information).

12: Ground Controller to Airport Safety Support Service

The ground controller interacts with the types of alerts presented on the HMI from the airport safety support service. This includes dismissing the alerts when conflicts have been resolved.

12: Airport Safety Support Service to Ground Controller

The airport safety support service presents the RMCA, CATC and CMAC alerts to the ground controller through the HMI.

13: Mobile Information Database to Airport Safety Support Service

Clearance and cleared route information is received from the mobile information database, for all mobiles on the airport surface.

14: Surveillance Service to Guidance Service

The identity and current position of each mobile, for each time step, is received from the surveillance service.

15: Guidance Service to AGL System

The guidance service sends light commands to the AGL system, including TCLs and stop bars.

15: AGL System to Guidance Service

The guidance service receives the light status/state of the TCLs and stop bars (on/off, intensity/unservicable), used in order to determine which light commands must be sent to the AGL system.

15: Guidance Service to AGL System

The guidance service sends light commands to the AGL system, including TCLs and stop bars.

16: AGL System to Environment

The AGL system emits light colors and intensities to the TCL and stop bars in the environment.

17: Other Aircraft/Vehicles to Aircraft/Vehicles

All aircraft/vehicles observe each other in order to maintain their individual goals.

18: Environment to Aircraft/Vehicles

Aircraft/vehicles observe the emitted light colors and intensities from TCL and stop bars in the environment.

19: Airport Operator/ANSP to Airport Operation Status

The airport operation status receives active runway/future runway changes from the other airport and ANSP sources (such as from the airport approach segment).

20: Airport Operator/ANSP to Surveillance Service

The surveillance service receives positional and surveillance related data from other airport and ANSP sources (such as from the airport approach segment).

20: Airport Operation Status to Surveillance Service

The surveillance service receives fixed aerodrome configurations such as the aerodrome map, layout, fixed obstacles from the airport operation status.

21: Airport Operation Status to Guidance Service
The guidance service receives fixed aerodrome configurations such as the aerodrome map, layout, fixed obstacles from the airport operation status.

22: Airport Operation Status to Airport Safety Support Service
The airport operation status receives fixed aerodrome configurations such as the aerodrome map, layout, fixed obstacles from the airport operation status.

23: Outbound Planner to Ground Controller
Responsibility (and physical flight strip) of aircraft between the outbound planner to the ground controller is executed when an aircraft is ready for push-back and start-up. Furthermore, the outbound planner assess the workload (and traffic) which the ground controller is undergoing and decides when to handover further aircraft to them.

23: Ground Controller to Outbound Planner
If the aircraft is in fact not ready for push-back and taxi, then the ground controller hands the responsibility (and flight strip) of the aircraft back to the outbound planner.

24: Runway Controller to Ground Controller
The runway controller hands over the responsibility (and physical flight strip) of an arrival flight (or aircraft/vehicle that is crossing an active runway) to the ground controller, the moment the aircraft is vacating the runway.

24: Ground Controller to Runway Controller
The ground controller hands over the responsibility (and physical flight strip) of a departing flight (or aircraft/vehicle that must cross an active runway) to the tower controller, the moment the aircraft is reaching the runway threshold.

4

Achieving Resilient Behaviour in Systems

Resilience relates to the ability of systems to deal with disruptions or adverse events, in order to preserve their central system goals and nominal levels. In typical socio-technical systems such as in air transport, naturally, this is particularly of interest due to the complexity, risks and uncertainties involved with operations. In terms of this MSc study, this notion is of importance as a decentralized control system, as presented in chapter 3, should be able to preserve the performance of airport surface movement operations during runway reconfiguration adverse events. Therefore, it is of interest to investigate the current state-of-the-art approaches and techniques which are described in order to achieve resilient behaviour in systems.

The way in which a system can result in resilient behaviour is currently an active research field as there is no unified theory that can be applied in order to provide any system with specific mechanisms or conceptual frameworks in order to guarantee resilient behaviour. However, system adaptations caused by modifying system functionings as a response to adverse events is a fundamental concept that is common in resilience literature [44–50]. Figure 4.1 presents this in a diagram flow.

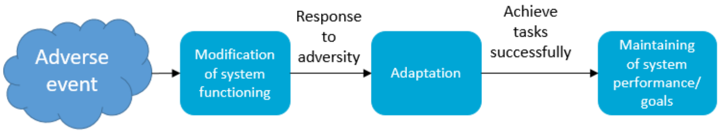

Figure 4.1: Steps to achieving resilient behaviour in a system

In complex systems, it is often the case that multiple modifications are possible in order to cause system adaptations. Therefore, performance based on a pre-determined set of performance indicators can be used in order to assess the resilient behaviour of an adaptation caused by a specific modification. In this way, it is possible to compare and determine which modifications result in the most resilient behaviours by determining the performance indicator value after the adaptation has been performed.

In general, multiple system properties are required to achieve this. Anticipation, monitoring, response and learning are common system properties that contribute to resilient system responses [49]. However, other sources state that avoidance, robustness, recovery and reconstitution are important system properties in order to deal with adverse events [51]. This section presents mechanisms and approaches that have been found to be used in order to achieve resilient behaviour when a system is exposed to an adverse event. It is important to note that these are not necessarily independent to one another, as there are overlaps and aspects of each contained within each other.

As presented in section 2.3, two main types of the adverse event of runway reconfigurations were determined: planned and unplanned runway reconfigurations. Planned runway reconfigurations are associated with respect to a time (window) at which they will be invoked, and unplanned runway reconfigurations relate to the event having just occurred. For this reason, the found resilient behaviour mechanisms and approaches/strategies are presented with respect to the time point at which they are executed in order to deal with disruptive events. This is, namely, prior to a disruption occurring (i.e. planned runway reconfigurations) and after a disruption has occurred (i.e. unplanned runway reconfiguration). The goal of this is to determine which mechanisms/approaches are appropriate for each types of this runway reconfiguration adverse event.

Firstly, section 4.1 presents modifications to system functioning prior to disruptions, and section 4.2 presents modifications after disruptions have occurred.

4.1. Modifying System Functioning Prior to Disruptions

Modifying system functioning prior to disruptions (i.e. for planned runway reconfigurations) relates to performing the modifications which result in adaptations prior to the disruptive event occurring. This section presents techniques that have been found in literature in order to encourage and achieve this in systems.

Anticipation

Anticipation is a form of adjusting system functions prior to an adverse event occuring. It is related to being able to know what to expect [49] or making a future-oriented action, decision or behaviour based on a prediction [52], where a prediction is a representation of a particular event. The predictions are usually made based on environmental stimuli or triggers, which an anticipatory actor senses. Literature [8] presents a state-of-the-art cognitive architecture for an anticipatory agent, which is applied at the security operations at an airport. This is presented in Figure 4.2. The reason as to why this is interesting is because this is a general agent architecture that can be applied to other areas.

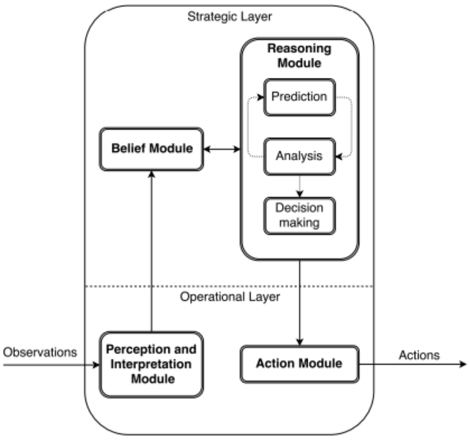

Figure 4.2: Anticipation agent as presented in [8]

This agent observes the environment and updates its beliefs, making it aware of the current state of the environment, potential changes and signals of malfunctioning that trigger anticipation. The belief module contains the knowledge of the agent, which is continuously being updated by input observations, therefore yielding situational awareness for the agent. An update of the belief module may result in an activation of the reasoning module. The belief module is also updated based on output from the reasoning module, including future predictions and outcomes of analysis/decision making. The reasoning module, which uses the output of the belief module, undertakes prediction, analysis and decision making. From this, simulated chains of interactions with the environment are generated and these are evaluated by action option valuation. This output is then the input to the decision making module in order to decide on which action to execute. Finally, this decision is sent to the action module which is responsible for the preparation and execution of the action.

Other works such as [48, 53] present anticipation in terms of imagination being directly linked to the ability to predict, based on the frequency of occurrence of the event. Regularly occurring historical events, such as machine failure or bad weather, are known and have been studied. For this reason it is possible to both imagine (predict) such events happening again in the future, as well as being possible to plan for them. Irregular events are historical events that are normally rare, such as earthquakes or fires. There is significantly less knowledge from these past events, or they occur so infrequently, that only limited planning/preparation is possible. However, as they have occurred in the past, it is still possible to imagine (predict) them but it is expected that they are not likely to occur in the short term. Unexampled events are events that are so rare that there is normally no preparation for them and are so unexpected that they could not be imagined for, such as the 9/11 terrorist attack. This link between plannability and imaginability as a direct outcome from whether or not the event has previously occurred is an interesting aspect. One could imagine that this link, therefore, also applies to the previously mentioned anticipatory agent [8], which also has the same issue: the decision making is only as valid/accurate as the types of predictions/imaginations that the reasoning module is able to make. Furthermore, this suggests that regular airport adverse events, such as runway changes or blocked taxiways, are more able to predict and thus anticipate. However, if a completely unexpected (or rare) adverse event such as a total electrical failure occurs at the airport, for example, then predictions are significantly limited and anticipation may not be possible.

In terms of this MSc study, only one specific and expected adverse event is being considered (runway reconfigurations) and therefore the unexpected event issues discussed in the previous paragraph are not foreseen to pose problems. Anticipation is a mechanism/approach that is recommended to be used for this MSc thesis (primarily for planned runway reconfigurations), and will be further discussed in section 4.3.

Minimizing Network Graph Topology

Literature [50] presents another approach for resilient behaviour prior to disruptions occurring. The paper discusses network resilience in order to combat spatial adverse events (such as bad weather, or volcanic eruptions) in air traffic networks (i.e. origin-destination traffic routes). This is done by comparing two different resilience mechanisms. The first mechanism which is presented is to modify the existing air traffic network (prior to any disturbance occurring) by minimizing the amount of air routes (edges in a graph) that a hub airport is connected to (when considering it as a node in a network graph). The reasoning behind this is that reducing the amount of edges results in the reduction of interdependencies between airports, minimizing the knock-on disruption throughout the entire network when bad weather forces these hub airport nodes to be unavailable for traffic. This is, fundamentally, a design approach (or a policy/procedure) when constructing a network graph that is expected to experience adverse events at its nodes. For this reason, it is questionable whether or not this is indeed a "resilience approach" (as presented in the paper), or if it is more related to the robustness of the network system. Furthermore, this concept, is not very applicable to this MSc study as the airport surface already has a fixed and pre-defined ground layout (i.e. a network graph) that cannot be further reduced (without causing an inefficient use of the taxiway resources) or expanded (as taxiways cannot simply be "built"). It is understandable how this could be applied to a much larger and more diverse network such as the air route network, but the aforementioned issues pose concerns if applying it to this MSc study. It is quite an unrealistic and inefficient approach as the ground surface already has fixed resources (taxiway infrastructure) that ideally should be utilized for efficiency purposes. This approach may also create bottlenecking and further disruptions due to traffic streams being restricted to not occupy the entire taxiway network.

4.2. Modifying System Functioning After Disruptions

Modifying the system functioning after the disruptions have occurred (i.e. unplanned runway reconfigurations) results in performing the modifications which result in system adaptations after the adverse event has occurred. This section presents the approaches that have been found in literature in order to achieve this.

Reacting to System Indicator Deviations

Monitoring relates to observing specific system indications/elements. It relates to knowing what to look for and being able take action based on the onset of an event [48, 49]. System indicators that can be used as resilience metrics may be split into three types of categories: attribute-focused metrics which consist of indices that rely on subjective assessments, data-based indicators which quantify the system attributes that contribute to resilience and performance-based methods which measure the consequences of system disruptions and impact. In the ATM socio-technical system, Key Performance Indicators (KPIs) are performance-based metrics that are used to monitor the current state of a system and detect the onsets of disturbances (such as the reduction of a specific KPI) [54]. ICAO defines such indicators as indicators for current/past performance and expected (estimate) future performance [55]. In essence, monitoring refers to continuously checking the state of a specific system element in order to detect possible cues that highlight the deviation from nominal functionality. The time at which cues are detected plays a key role in determining the response which the system is able to undertake. In general, the earlier the system is able to detect these deviations, the more flexible and faster the response to it can be [48]. Furthermore, core system goals (such as safety) should be monitored throughout any response in order to be able to relate the affect of the response [56].

Figure 4.3 shows an example of how monitoring a performance-based metric could result in a response.

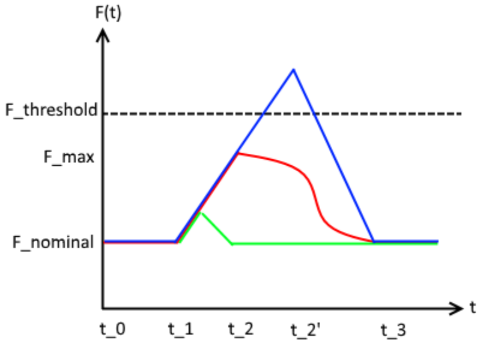

Figure 4.3: Concept of performance-based monitoring for resilience. Green line represents the performance indicator behaviour of an effective modification, the red line represents a less-effective modification, and the blue line represents an ineffective and non-resilient system adaptation.

A performance-based metric varies with time and is depicted by $F(t)$, and an effective modification with respect to resilient behaviour is (for this example) defined as how quickly the nominal $F_{nominal}$ value can be re-achieved[1]. For example, this could be the average pushback or taxi delay of flights. Consider the red line the graph. Between timepoints t_0 and t_1, the delay is at the nominal delay level of $F_{nominal}$. At time point t_1, a drift in nominal value is observed from $F_{nominal}$ to F_{max}, perhaps caused by an adverse event such as an unplanned runway reconfiguration. At time point t_2, the system initiates a modification in order to respond to the rising $F(t)$ value, which takes until timepoint t_3 to return to nominal levels, and is therefore effective in achieving resilient behaviour. Now looking at the blue line, a similar recovery behaviour is observed. $F_{threshold}$ denotes a pre-defined threshold value of $F(t)$ which should not be passed (for safety reasons, for example) for a modification and subsequent adaptation to be resilient. However, this recovery at timepoint t'_2 results in the metric value raising above the pre-defined $F_{threshold}$ and even though the the metric returns to nominal levels at timepoint t_3, it has not resulted in a resilient behaviour due to the violation of the $F_{threshold}$ value. Considering the green line, this results in a a recovery point at a time point between t_1 and t_2, which results in the fastest recover to nominal levels and is therefore more effective than the red line as it returned to the nominal levels faster (with respect to the definition of effectiveness in this example). This performance based monitoring can be used to assess different modifications/adaptations with respect to resilient behaviour, and it is important to define the effectiveness of modifications in a system. This approach is recommended to be used in this MSc thesis and is discussed in section 4.3 and the research methodology in chapter 7.

Modifying Task Flows

Source [54] formalizes a mechanism that represents task executions as task flows where each task has an associated contribution towards a specific resilience indicator (KPI). In the nominal system operating conditions, there is a nominal set and order of tasks, namely S_0, which must be carried out in order to maintain the nominal system level with respect to the corresponding (summation) of KPI contributions, $F(S_0)$ associated with S_0. However, when a disturbance occurs, an alternative set of tasks are required in order maintain the nominal operating system level. When such an alternative set of tasks are required during a system disturbance, the most set of tasks and order with the smallest deviation from the nominal summed KPI contribution, $F(S_0)$ must be found. In order to achieve this, a Directed Acyclic Graph (DAG) is constructed which contains all possible task flows involved from a triggering disturbance, V_{start} to the end of a scenario, V_{end}, such that the nominal system levels may be preserved through an alternative task flow. For example, V_{start} = airborne loss of separation alert and V_{end} = nominal airborne separation level. In between this start and end vertex, intermediate vertices are constructed which represent tasks T_{ij}, where i represents the actor and j represents the task, that could/are required to traverse from V_{start} to V_{end}. In order to distinguish and compare the different task flows which are valid paths, a flow distance function is created. For each valid V_{start} to V_{end} alternative task flow, namely S_i, the summed contribution to the KPI indicators $F(S_i)$ is computed. This is then compared to the nominal (non-disturbed) flow KPI contribution $F(S_0)$. The path that has the smallest difference to this has the smallest resilience loss, with respect to the nominal conditions, and is thus

---

[1] The "effective modification" can be defined in any other way (for any system) which results in different conclusions of the modifications.

the most resilient response. When the edges between the vertices in the DAG are constructed to have a distance as a detraction from the nominal $F(S_0)$, then the most resilient (least deviation from nominal operating $F(S_0)$) is then computed by the shortest path in the DAG. This approach is interesting as it effectively assesses and computes all possible tasks and associated modifications that could be carried out in order to return to nominal levels, and determines the order with which they must be carried out to result in the smallest deviation of nominal conditions. The issue with this is that all potential tasks between V_{start} and V_{end} must be established/known in advanced of the disruption in order to compute nominal and alternative (when a disturbance occurs in a system) task flows. Additionally, this response does not necessarily guarantee a return to nominal levels as there may not be an alternative path with the same nominal KPI level. Furthermore, the KPIs and associated contributions of each task to them must be established, which seems quite arbitrary as it is difficult to assign qualitative values to specific tasks in a normalised an non-biased way. For the scope of this study, which focusses on a specific adverse event, it may be possible to pre-determine all tasks associated with nominal and alternative flows of runway reconfigurations (especially those that are planned), although the large issue of determining the contributions to specific KPIs of certain tasks still remains. Additionally, specific KPIs must be decided upon which are useful indicators to assess the system performance during runway reconfigurations. In terms of expanding this approach to more adverse events, further analysis into the tasks and KPI contributions of other events must be performed in a scenario-per-scenario way, as this is quite a specialized and not a general mechanism for dealing with disruptive events.

Modifying Network Graph Topology

Literature [50], as discussed in the previous section 4.1, discusses network resilience by proposing resilience mechanisms in order to combat spatial adverse events (such as bad weather, or volcanic eruptions) in air traffic networks (i.e. origin-destination traffic routes). The second mechanism that is presented is more interesting than the first and will now be presented. When a spatial hazard (bad weather) disruptive event engulfs a hub airport, the edges to which it is connected to are removed, and reconnected to the closest operational airport, provided that there are sufficient resources (capacity) at the airports to which they are shifted to. This is done whilst the disruptive element is affecting the node, after which the nominal network connections are activated again. In this way, the connections between nodes can adapt and reorganize (by creating a new network bypassing the affected node) in dynamic ways to avoid the disruptive elements. In terms of the airport surface, this could be analogous to an unplanned runway reconfiguration where a runway becomes unavailable. This results in all edges associated to the holding points of this runway being removed and thus the runway bypassed. Therefore, not all runway hold short edges (from the main taxiway network) need to be activated at each time point, but only those which are associated with an active runway. In this way, no routes of aircraft will utilize any portion of the deactivated runway. However, although this is a feasible method, it prevents runway crossing from occurring, such as crossing runway 18C at holding point W5 which has consequences with respect to the efficiency of taxi routes. Furthermore, it also restricts runway taxiing operations, such as backtracking along a runway if required, as the runway can simply not be "reached" in the graph.

Agent Based Model Examples

An Agent Based Model (ABM) approach to a complex system resilience study [51] was found that presents restorative elements after a disruption (due to a natural event) occurs in a directed network graph of system where the nodes are the operations of the system, and the edges connecting them depict the flow of system operations in a similar way to literature [54]. When a disruption occurs, certain edges are disabled (i.e. removed) to de-connect certain nodes and thus model the disruption effects throughout the complex system. This method focuses on restorative mechanisms necessary to repair these broken edges and restore the system network graph. The ABM consists of restoration agents (repair team) that are responsible for repairing the network after a disruptive event, in a spatially fixed environment. Perturbation effects are also included in these restoration agents, increasing the difficulty of repairing edges. The restorative mechanisms are split into three strategies: a sequenced approach where all restorative agents repair the same edges one edge at a time, a random approach where restorative agents are randomly assigned to inoperative edges (and thus multiple edges can be repaired at the same time, and a shortest time to repair approach (where agents repair edges with the least effort to repair first, one by one). The type of repair modes which these agents can perform to repair the edges (once they reach them) are also varied as follows: constant effort (all agents have the same repair effort/ability), variable effort (all agents have variable repair effort/ability), event (perturbation events occur and edge repair difficulty requires more/less effort), and adaptation (where agents assess changes to edge repair difficulty and move to edges with lower repair difficulty if available). Twelve scenarios (repair strategy - repair mode combinations) were simulated. The study shows that ABM is a very feasible modelling approach in order to determine the effects of disruptions in complex systems, as well as for assessing the effectiveness of adaptations with respect to resilient behaviour. This, therefore, further suggests that ABM is a feasible modelling method for modelling disruptions on a graph (such as of the airport surface environment), which is based on spatial constraints (i.e. the existing taxiway network). Furthermore, the literature also shows that the most resilient results (based on a set of indicators which are used to assess the resilient behaviour of a system) differ depending on the type of disruption (i.e. if there are or are not perturbations in the repair mode in this example). This is an interesting conclusion that may suggest that there is not a single modification/adaptation that results in resilient behaviour that is able to deal with both planned and unplanned runway reconfigurations, but different approaches may have to be taken depending on each specific case.

A multi-agent system application to the resilient re-organizing of traffic networks due to natural disasters [9] literature was found that discusses how road networks could be changed based on a decentralized agent system. Each area of a road network is split into a zone which is controlled by a zone manager agent which analyses traffic information about the traffic flows at each intersection with the zone, defines road reversals, diverts traffic to evacuation routes and informs vehicles about changed network topologies. Within each zone, intersection controller agents are placed to each intersection which execute the decisions of the zone manager. At the high level, an overall traffic manager agent identifies global traffic management strategies between the zones and communicates with zone managers for information about their zones, as well as altering their zone boundaries if needed. The vehicles that operate on the road network are able to communicate with other vehicles as well as with intersection controllers and with the vehicle's driver. Initially, zones are created by the zone managers based on evacuation exit points (each zone must contain at least one), and then expands the road network based on evenly distribute traffic across the zones in proportion to the zones' capacities. After each traffic zone is created, a road reversal plan (which changes the flows along the roads from normal operation to directing the traffic towards the exit which is contained inside the zone) is defined in order to maximize the traffic flow towards the exit. However, certain roads may be blocked and therefore, the traffic zone manager coordinates with its intersection agents in order to by pass these road blockages, or send traffic to another zone's exit point. In essence, each zone learns a directed map and routes (based on shortest paths) that can be used if an evacuation is required, that can by dynamically adjusted (during the evacuation procedure) to avoid road blockages. In this way, a system modification can also be pre-defined prior to an adverse event occurring that is also able to further adapt after it has occurred. Furthermore, the zone managers interact with each other by adjusting zone boundaries depending on if too many roads are blocked in a certain zone and if it is better to send traffic to an exit point of another zone. This form of dynamically modifying boundaries based on the blockages results in adaptations based on the specific type of blockages. Figure 4.4 presents an outcome of this adaptively determined evacuation routes.

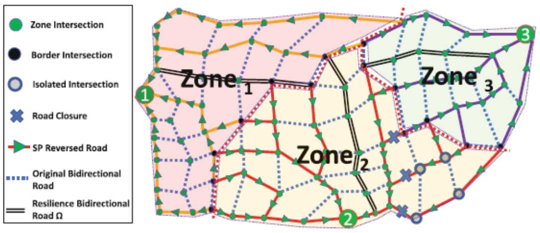

Figure 4.4: Zones with evacuation routes [9]

The results of this paper show that this dynamic adaptation of zones and avoiding road blockages resulted in significantly (almost 175%) more vehicle agents being evacuated than if the zones and roads were fixed. This suggests that modifying pre-defined taxiway routes (or creating them "on-the-fly") may yield higher successful origin-destination completions. This literature, in addition to further confirming the feasibility of ABM for resilience studies, indicates the decentralization has a positive influence with respect to resilient behaviour in system. Additionally, the structure presented in this literature is also of a similar form of decentralization as is presented in the Follow-the-Green concept in chapter 3, which suggests that such a decentralization may indeed result in resilient behaviour. Furthermore, this strategy is particularly interesting as it can be applied to the airport environment. For example, the airport surface can be split up into zones and similar style of agents as is presented in the literature. If a runway reconfiguration occurs, traffic that is currently taxiing and is affected (e.g. their departure runway suddenly closes) can be directed along a created highway (which accounts for any other traffic or blockages, for example) to another zone which contains the new departure runway which they will utilize. This concept with adapting routes and highways that may be most applicable due to an adverse event will be further discussed in chapter 5, and is recommended to be investigated further in this MSc study.

4.3. Concluding Remarks

The previous sections present approaches for modifications resulting in adaptations, and some conclusions can be made of approaches that are feasible to be used for this MSc study. Furthermore, these are included in the research methodology, presented in chapter 7.

Indicators which will be used to assess resilient behaviour should be determined, based on performance based ICAO KPIs [55], as these vary with time (and therefore can be used to encompass the performance prior to and after runway reconfigurations). This will be used in a manner similar to that described in section 4.2, and therefore the effectiveness of modifications with respect to these should be defined.

In terms of planned runway configurations, modifying system functioning prior to the runway reconfiguration taking place, and in particular anticipation, is an approach that is recommended to be used. The reason for this is that it is possible to anticipate such a runway reconfiguration, and adapt the airport surface movement operations prior to the reconfiguration taking place. This makes the anticipation mechanisms, and in particular the anticipation agent structure presented in section 4.1, of particular interest to be used.

In the scope of this MSc study, an anticipatory agent (or the internal processes) could be included in the the Routing Service (see section 3.5), for example, and can be used for planned runway reconfiguration adverse events. This agent can observe if a planned runway reconfiguration has been input through a communication interaction with the airport operation status element. The agent could observe if and what runway reconfiguration with associated activation time, as discussed in chapter 2, has been input.

The agent interprets this observation, updating the belief of the agent (that a planned runway reconfiguration will occur), which activates the reasoning module. The reasoning module predicts what the impact is on the aircraft that have planned routes (i.e. have not yet received a clearance from the Ground Controller). As previously mentioned in chapter 2, the time at which runways configurations are activated are more of a time window than a specific moment in time. This, therefore, is crucial for anticipation as the airport surface movement operations should be configured at a time point which is before the actual activation time (within this window), whilst not being configured too early in advance which may result in aircraft having to wait unecessarily at the runway until they are activated. This will need to be further developed in the later stages of this MSc study and will be discussed in chapter 7. However, the anticipation agent/processes could make 3 predictions as follows: the first option is about the impact of delaying the aircraft at the stand to wait until the runway reconfiguration is activated, the second prediction concerns the option where the aircraft's planned route is changed such that it arrives at the new runway reconfiguration at a time prior to the activation time, and the third prediction concerns using the initially planned route (and incurring the impact that the aircraft may have to receive a runway change whilst taxiing). The prediction outcomes are analysed, and a decision is made based on how to change the planned route. Based on this decision, the appropriate actions are taken in the action module. Of course this is a loose example of how it could be applied, although there may be more prediction scenarios possible or better implementation methods. Furthermore traffic that is already taxiing to a runway will have to be checked to see if they are affected by the runway reconfiguration and if they will need a route alteration.

In terms of unplanned runway configurations, mechanisms for creating modifications could be included in the Routing Service. Monitoring of if an unplanned runway configuration has been communicated/interacted from the airport operation status agent could be used in order to invoke modifications to the airport surface movement operations and adapt to this unplanned runway reconfiguration. This would include altering the routes of currently taxiing aircraft, as well as altering the routes that of aircraft which have received a route at the stand, and newly generated routes must also utilize the new runway reconfiguration.

For both types of runway reconfigurations, the possible modifications to the routes, or other operational aspects, must be further investigated in order to first determine which modifications are possible, and then determine, based on a valuation, which modification will result in the most efficient and resilient behaviour. Furthermore, the last literature which presents the learning and adapting of highway routes is recommended to be further investigated in combination with the chosen cooperative coordination mechanism in chapter 5.

5

Cooperative Coordination in Multi-Agent Systems

The global goal of any airport surface movement system is to provide services to aircraft and vehicles in order to maintain airport throughput under all local weather conditions, whilst maintaining the required level of safety [7]. However, airports have fixed resources, namely the taxiways, runways and gates, and have many users (aircraft, ground vehicles, people etc.). In addition to this global resource constraint, the users of the airport have interdependencies[1], and there is no unique individual that has sufficient competence, resources or information to be able to make the system reach the global goal as a whole. All users of the airport's surface must share these resources and achieve their individual goals. In order to achieve this, coordination is of particular importance which relates to the management of interdependencies between tasks and activities [60]. Additionally, the sharing of the airport's surface (from the ATC side) is typical of cooperative coordination where each ATCO works with other ATCO/sectors in order to fulfil the global goal of the safe and efficient flow of operations. Furthermore, no (regularlay occuring) "greedy" amount of resources are allocated to any particular aircraft (such as continuously assigning certain airlines a faster taxi route than others) as ATC tries to give the same level of service to each aircraft within their control. Furthermore, as identified in the research gap in chapter 3, it is clear that coordination of individual plans to provide conflict free and efficient routes along the airport's surface is of particular importance. For these reasons, a cooperative, coordinated and combined effort by multiple agents is desirable, especially if a decentralized approach is being considered [61]. First, multi-agent planning is introduced in section 5.1. After this, Multi-Agent Path Finding Algorithms (MAPF) are introduced in section 5.2 which form the main literature investigation for coordination techniques. Then, approaches for coordination before, during and after planning are presented in section 5.3, section 5.4 and section 5.5 respectively. Finally, a trade-off between the various coordination algorithms is performed in section 5.6 which culminates in a selection of an algorithm that is proposed to be used as the basis of the cooperative coordination mechanism in this MSc study.

5.1. Multi-Agent Planning

The issues described in the introduction are comparable to the formulation of multi-agent planning problems which are defined as: given a description of the initial world state, a set of goals, a set of agents, and a set of capabilities and private goals for each agent, find a plan for each agent that achieves its private goals, such that these plans are coordinated together to meet the global goals [62]. Multi-agent planning can be summarised by the combination of coordinating, planning and scheduling.

Coordination is the managing of interdependencies between tasks or activities of agents [60], and is important for preventing chaos in decentralized systems [61]. Without coordination, collisions and unsafe environments can occur at airports.

Planning relates to the determination of a sequence of actions that describe how an agent can reach their desired goal state from the current state of the world [57]. In terms of airport surface movement operations, a sequence of actions must be determined in order to traverse aircraft or vehicles from their origin to destina-

---

[1] Interdependencies occur when local tasks undertaken by one agent have an impact on another agent, or group of agents. For example, the local task execution T_1 of one agent a_1 influences performance-related characteristics associated with another agent a_2 [57–59].

tion locations, within the airport ground infrastructure, whilst considering the interdependencies with other users.

Scheduling is the ability to temporally restrict tasks or actions by assigning resources and times to them, whilst adhering to the capacity limitations shared resources. In the case of airport surface ground movements, the resources are the taxiways, runways, gates, and the times can be considered as the time points at which traffic traverses them. These must be shared amongst all agents utilizing this system.

If multi-agent planning is not taken into account, then merely reactive agent behavioural properties such as stopping or turning upon observation of other traffic may not be sufficient in order to enable the safe and efficient flow of traffic, especially in large, multi-mode traffic airports such as Schiphol. If ground mobile agents do not take route action dependencies into account within their plans, then conflicts during their individual plan executions are likely to occur. In general, coordination and planning can be split into three types: coordination before planning, coordination after planning and coordination during planning.

5.1.1. Phases of Multi-Agent Planning

This section presents an overview [57] of possible sequenced phases that can be used to achieve multi-agent planning.

1. Global goal refinement: the global goals or tasks are refined to result in subtasks that can be assigned to individual agents in order to achieve the global goal.
2. Task allocation: allocate the set of subtasks to agents.
3. Coordination before planning: define rules/constraints for individual agents to prevent them from producing conflicting plans (discussed in section 5.3).
4. Coordination during planning: individual planning of each agent to reach its individual goals (discussed in section 5.4).
5. Coordination after planning: coordinate the individual plans of the agents (discussed in section 5.5).
6. Plan execution: execute the plans.

5.2. Multi-Agent Path Finding Algorithms

In problems that require pathfinding, determining routes between two vertices in a graph (or points in space) with respect to a cost function and path constrains is a fundamental and important aspect. Well known centralized Single-Agent Path Finding (SAPF) search algorithms such as the A* algorithm are widely applied in artificial intelligence through a range of applications from robot path planning [63] to video games [64] as they are able to efficiently find the shortest path for an agent in an spatial environment. However, application of such algorithms in a spatially fixed environment with large number of mobile agent scenarios results in scalability and large computational time issues as the state space size is exponential to the number of agents [10, 11, 13]. Additionally, coordination of individual route planning is not dealt with, and thus issues with respect to conflicts, task scheduling and optimality occur. Often, sacrifices such as allowing conflicts to occur between agents in a video game [64] must be made, or other collision avoidance mechanisms [65] must be specifically designed. These are major issues in this MSc study research: conflicts must be avoided and systems/mechanisms external to aircraft should be used (otherwise new systems must be integrated with aircraft which is not the aim of the Follow-the-Green concept). This was also determined in the A-SMGCS specifications in chapter 3. In artificial intelligence, the state-of-the-art approach to solving such complex systems is a branch of path finding theories called Multi-Agent Path Finding (MAPF). These aim at coordinating the individual plans of agents within their environment.

In this MSc study, the airport environment can be represented as a graph $G(V,E)$ with V taxiway intersections, runway entries and gate vertices and E taxiway edges. Additionally, k number of aircraft or vehicles must operate upon the airport's surface whilst not colliding and in an efficient way. This representation directly coincides with formal definitions for Multi-Agent Path Finding (MAPF) problems [13, 66–69], thus yielding this as a feasible and interesting research area to review.

MAPF problems consist of finding paths for all $k > 1$ agents along the edges of a graph with a unique intial start and goal vertex whilst avoiding inter-agent collsions, often with an additional optimality constraint such as minimizing the makespan [13, 67, 68]. The MAPF algorithms will be presented in subsequent coordination sections.

5.2.1. DEC-A*: Decentralized A*

DEC-A* is a decentralized MAPF algorithm based on the A* search algorithm, presented in [10]. Although no coordination is associated with this algorithm, it is presented now as it forms an application of decentralization to such other MAPF algorithms.

The main idea behind this algorithm is to sub-divide the state space into smaller graphs which can be used to determine the shortest path from a start to an end state through a sequence of paths within the graphs.

The overall state space is defined as (grid-)search domain D which is sub divided into a set of smaller graphs $[G_1,...,G_n]$, which are linked between a set of inter-graph links $[l_{ij},...,l_{kl}]$. Each graph is defined by a finite number of states s_i, where a link is defined as $l_{ij} = s_i \rightarrow s_j$, where $i \neq j$. An agent A_i is associated to each graph G_i, where an initial state $init$ state is contained in an agent A_{init} and a goal state $goal$ is contained in agent A_{goal}. This is shown in Figure 5.1, where the complete state space is subdivided into 4 sub graphs with agents $A1$ - $A4$ associated with them, where$A_{init} = A1$ and $A_{goal} = A3$

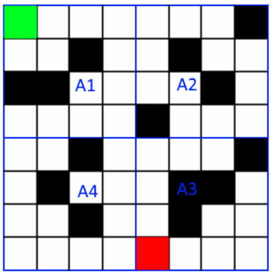

Figure 5.1: DEC-A* subdivision of state space [10].

The problem is defined as finding the shortest path between the $init$ and $goal$ states. In the domain D, a global heuristic computation algorithm is applied to determine which graphs (and associated links) are included in the shortest path from G_{init} to the graph of G_{goal}. The agents from $[A_{init},...,A_{goal}]$ then determine the optimal route base on a search tree algorithm by applying a local A* algorithm within the identified graphs of agents A_i to find the path from neighbouring entry link to the neighbouring exit link. The solution path is defined as $\Pi = [\pi_1, l_{1,2}, \pi_2, ..., l_{n-1,n}, \pi_n]$, where π_i is the local path within graph G_i and $l_{i,j}$ is the link between graphs G_i and G_j through states $s_i \rightarrow s_j$. Figure 5.2 presents a general sequence of steps which are undertaken in this algorithm.

Figure 5.2: Flow of DEC-A* algorithm

The benefit of this algorithm is that, through the splitting up of the complete state space into smaller state spaces, the time and space complexity is greatly reduced whilst still preserving the determining the shortest path.

An experimental evaluation is presented in the literature. Four grid cell sizes with a varying number of blocked grid cells is used. The runtimes of the A* and DEC-A* algorithm are compared. The results show that the A* algorithm is faster than the DEC-A* in the small grid cell sizes of 30 x 30 and 50 x 50. However, for larger grid cells size of 180 x 180, the DEC-A* outperforms the A*, arriving at a solution between 0.022 - 0.418s for all 3 probability blocking cases, in comparison to 2.2 - 27.4s for the A*. This trend continues into the largest grid cell size of 500 x 500, where the DEC-A* takes between 0.062 - 3.05s to arrive at a solution, in comparison to 1.5 - 33min (and even not finding a solution in one case) for the A*. These results show that decentralization is beneficial with respect to scaling, especially when applying this algorithm to large spatial environments such as the airport's surface. Furthermore, the splitting up of the airport surface into zones is very similar to the resilient literature presented in section 4.2. The blockages included in this algorithm could also be considered as a static representation of the other aircraft/vehicle in which use the airport layout and thus "block" certain parts of taxiways, for example.

However, this algorithm has one fundamental and major issue. It does not take any type of coordination into account, thus collision avoidance is not considered. It only applies the A* algorithm without any logic or sequencing validation amongst agents. Nevertheless, it does highlight potentials for decentralization, especially with respect to scaling, which is why it is included in this literature study. For this reason, this algorithm could be viewed as a general decentralized MAPF framework, due to its aforementioned benefits. Naturally, other coordination mechanisms must be involved in order to prevent collisions preventing algorithms. Figure 5.3 presents an arbitrary splitting into sub graphs of Schiphol airport.

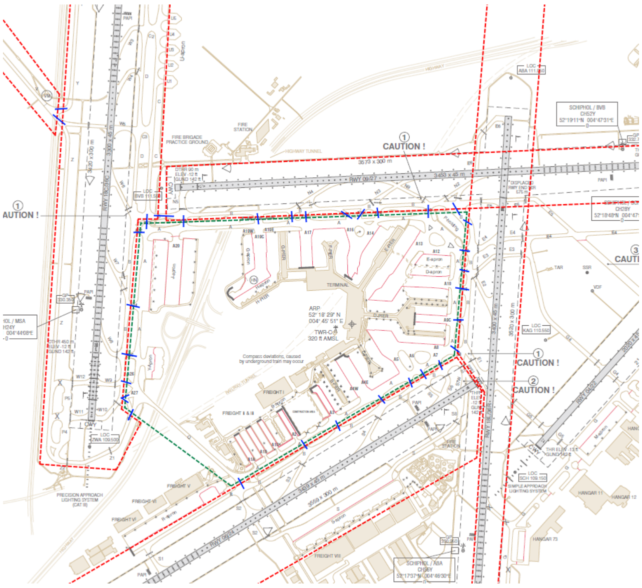

Figure 5.3: Arbitrary splitting of Schiphol into smaller graphs.

This highlights that it is possible to separate Schiphol into smaller sub-graphs in order to enjoy the benefits of decentralization of the graph of Schiphol. Red graphs include runways, and green includes the major piers at the terminals. The blue lines indicate the links between the sub graphs. Splitting arbitrarily has the advantage that the graph size (and therefore search space) is reduced depending on the runway configurations. For example, if runway 18L/36R is not active, then the whole section of that graph can be "deactivated", thus reducing the number of nodes and edges in the overall graph. Furthermore, if runway 18C/36C is not active, this portion can also be removed, and perhaps the central (green) graph expanded to include the runway turn off points of this runway. Internally to these graphs, different coordination mechanisms in order to avoid collisions can be applied, as will be discussed in the subsequent sections, for example. These mechanisms may also benefit with the reduced graph complexity. However, the handover (blue links) regions must be investigated as conflict free handovers are required in these regions, and agents must not be allowed to cross the same link at the same time in order to avoid conflicts.

The green region could also be subdivided into smaller graphs, such as North, East, South, West. For example, if an agent requires to taxi from a North pier, to runway 24, then a high level A* algorithm (for example) is possible to determine which grids should be "activated" and the graph constructed to include the North, East, South and runway 24 sub graphs. Then, based on this graph, local coordination could be performed (by junction agents) to provide a conflict free path to the runway.

5.3. Coordination Before Planning

Coordination before planning refers to the removing of interdependencies among agents' tasks by defining rules or constraints for individual agents to prevent them from producing conflicting plans. For example, implicit coordination mechanisms such as social laws and rules that constrain the actions of agents can be used to reduce communication, planning and coordination time such as in road traffic rules where cars drive on one side of the road and obey traffic signs. At Schiphol airport this form of coordination is already in use through the design of taxiway procedures such as the inner and outer ring directions presented in chapter 2, as well as the ICAO published rules of the air [24]. This is already a form of coordination before planning as these rules mean that head on collision scenarios can be prevented as it is not possible for aircraft to taxi in both directions on both the inner and outer taxiway rings. Furthermore, this form of coordination is also present in the runway reconfiguration selection as conflicting runways are not able to be chosen (landing and departing on opposite sides of the runway). It is assumed that a significant amount of effort and design of such procedures has been undertaken by ATC in order to train ATCOs and publish charts which utilize procedures and ground routing based on a pre-determined set of rules. For this reason, it is recommended that such published procedures be incorporated in any of the chosen algorithms which will be derived in the trade-off at the end of this chapter in section 5.6.

Although strict laws can result in coordination and significantly reduced communications/resources, it can be the case that agents are not able to find solutions from their origin to destination. Utilizing flexible laws [70], agents prefer to obey the laws but are also able to relax them in case a no solution outcome is determined. However, such a relaxation directly affects the optimality of a solution and yields sub-optimal solutions, but the main benefit is that a higher number of agents are able to find feasible solutions. This is actually how ATC performs their actions as it was observed using trajectory data that the inner and outer taxiway law is not always adhered to, especially if there are low levels of traffic in a certain airport surface area. Interestingly, this resulted in a more optimal solution (as the taxi time was significantly reduced), therefore indicating that the optimality is not always reduced.

Literature [71] presents a general method for determining social laws for concurrent actions of agents. This is based on 2 (extreme) approaches in a MAS, namely that either all information about the actions of all agents is supplied or that no information is supplied. The types of laws significantly depend on this type of information knowledge, and the main benefit of this form of coordination, and not planning, is that the design of such laws need only be computed once as opposed to frequently over time. For example, explicit coordination mechanisms such as agents communicating explicitly who does what can result in a distributed planning as opposed to simply following rules.

5.4. Coordination During Planning

Coordination during planning refers to taking into account the interdependencies/plans of other individuals whilst making the plans of other individuals. This section presents found algorithms and mechanisms which are based on coordination during planning.

5.4.1. MAPP: Multi-Agent Path Planning

Multi-Agent Path Planning (MAPP) is a MAPF algorithm, presented in [11] and is based on a concept of a sliding grid with one (or multiple) spaces inside it. This is a centralized planning method with decentralized plans. The problem is defined as a graph representation of a map with mobile unit agents U, each with a unique start and goal pair (s_u, t_u). The aim is to navigate each agent from s_u to t_u whilst avoid fixed and mobile obstacles. The MAPP algorithm is demonstrated to be complete (guarantees to find a solution) for $slidable$ problems. In order for a problem to be $slidable$, three conditions must hold as follows:

1. There must be an alternate path Ω for an agent for 3 node sequences along a shortest path $\pi(u)$ at the previous, current and the next nodes (not including the start and end nodes). This is shown in Figure 5.4.
2. The first node after the starting node in the first agent's path must be empty.
3. No destination location may interfere (intersect) with any of the paths and alternative paths of any agent.

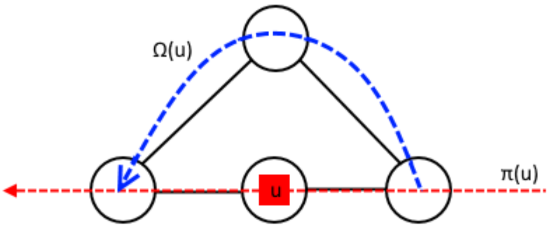

Figure 5.4: Alternate path for 3 node segments based on [11].

The algorithm first finds the shortest path $\pi(u)$ and alternative path for each 3 node segments along

$\pi(u)(\Omega(u))$ for all agents, determines which agents are $slidable$, and then follows a hierarchical progression to advance all agents to their goals. If the agents are not $slidable$, then they are prioritized at the bottom of the list as they will be dealt with when there is more "space" from $slidable$ agents reaching their goals first (although there is no guarantee that they will be able to reach their goal without a collision). In essence, when there is a crossing of paths, agents are pushed out of their $\pi(u)$ to their $\Omega(u)$ for a certain portion of their path, thus sliding them out of another agent's path and avoiding a collision. Furthermore, the paper then introduces a concept of "private zones" which are additional spacing factors behind agents to create better separation margins, as well as additional variations on the slidable problem and initial MAPP algorithm which relax the point 1 and point 3 of the slidable conditions. These are the Alternate Connectivity (AC) and Target Isolation (TI) MAPP variants. Figure 5.5 presents a flow chart of the basic MAPP algorithm (without any of these variants), which guarantees to solve slidable agents only.

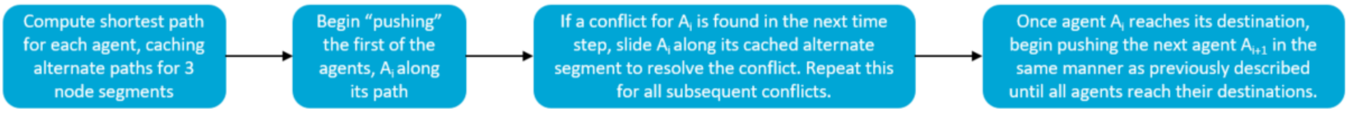

Figure 5.5: Flow of the MAPP algorithm [11].

An empirical evaluation is then performed. The MAPP algorithm is implemented and tested on 10 of the largest maps from a game[2]. These maps range from 13765 - 51586 traversable tiles [11], and offer a range of different geometrical features such as obstacles, open rooms and narrow lanes. Each map is tested with 100 - 2000 agents, and a time-out limit of 10 minutes is set (therefore if no solution is found within this time limit, it is counted as a failure).

For almost all the game maps, the best TI+AC MAPP (previously mentioned relaxed variant) solves between 92 - 99.7% (dependant on which of the 10 maps is being used), which is more accurate than the other algorithms which it is being compared to, which only solve between 25-90% with 2000 agents. However, the MAPP solutions are on average 7% longer than the other algorithms.

This is an interesting algorithm as there is no need for the continuous replanning of agents, which are able to "slide" onto another path temporarily to reach their destination. It is clear that this, therefore, does reduce the run-time of the algorithm. However this will have to be endured in the initialization of the algorithm as all alternative paths for each agents Ω must still be calculated, increasing the memory usage as these routes must be hashed.

It is unknown as to how available and practical slidable actions are on the Schiphol airport layout, as the infrastructure is fixed and dynamic constraints such as taxiway restrictions dependent on runway configurations often occur. For example, Figure 5.6 presents the current layout of Schiphol with different geometric situations.

---

[2] Baldur's Gate: https://www.baldursgate.com/

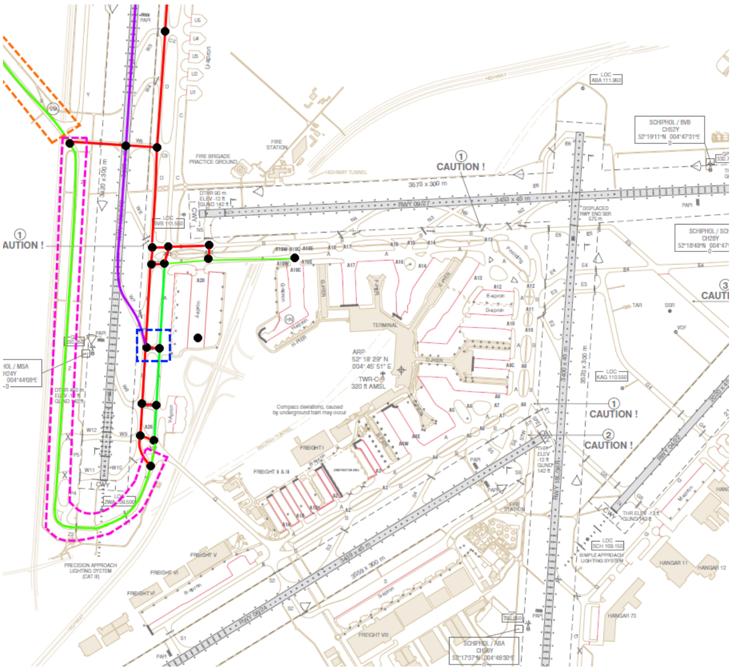

Figure 5.6: Different slidable instances at Schiphol [3]

If one considers a hypothetical scenario where the light green line is a shortest path $\pi(u)$ for an agent taxiing from runway 18R to the Hotel pier, then clearly, the slidability of regions along its route differ.

Firstly, the orange section which is the taxiway from runway 18L is not slidable as there are no other taxiway segments which link the runway to the rest of the airport. Secondly, the pink region is also not slidable as runway crossing are not possible due to runway 18C being active. However, if runway 18C was not active, then this region is slidable as the shortest path would be be crossing runway 18C. Thirdly, the region in blue also poses a dynamic constraint issue on this path due to an inbound (arrival) aircraft vacating runway 18C which will be further elaborated upon now. The alternative routes were hashed in the planning phase. However, it could be the case that arrival aircraft temporarily restrict the slidability of the agent following the green path, as they block certain segments of the alternative paths when the runway is vacated. These dynamic operational constraints indicate that further coordination must be made between operational environments such as runway modes of operation determine the slidability of agents, as certain slidable routes may become suddenly blocked by new traffic appearing (which takes operational priority as they are unable to stop on the runway). This raises issues that must be addressed with this method, especially with respect to the runway exit coordination. It may be the case that, for example, the MAPP algorithm may only be useful for the confined inner rings of Schiphol, which are located away from active runways.

5.4.2. MAPF-DP: MAPF with Delay Probabilities

Multi-Agent Path Finding with Delay Probabilities (MAPF-DP) is a recent formalization of the plan execution approach, utilizing 2 classes of decentralized plan-execution policies in order to execute plans whilst avoiding conflicts. This method is presented in [69]. This is a form of centralized planning with a decentralized plan execution.

Each agent a_i has a local state x at each time step. This local state is a time index such that each agent knows its current local state and is able to receive messages from some of the other agents about their local states. Furthermore, $l_i(x)$ refers to the vertex assigned to local state x of an agent a_i. A valid MAPF-DP plan is defined as: two agents are never scheduled to be in the same vertex at the same time index, therefore the vertices of two agents in the same local state are different. Additionally, an agent is never scheduled to be in the same vertex as any other agent at the next time point, therefore the vertex of the local state $x+1$ of two agents is not the same. These plans are coordinated using a new MAPF-DP solver called Approximate Minimization in Expectation (AME), which is based on CBS (discussed in section 5.5), incorporates focal searches and results in small average plan execution times. The plans are executed according to a plan execution schedule, which, for each agent, has a probability p_i chance of no action (move or wait) being determined, resulting in the agent remaining delayed in its current vertex.

Based on this, two plan execution policies are presented. Fully Synchronized Policies (FSPs) execute the policies in "lockstep". They provide a "GO" command to an agent if the agent has not entered its last local state, and all other agents have either reached their respective last local states, or have left their local states which precede the local state of the agent. Understandably, this requires agents to be constantly communicating as to when agents have left local states. This results in large average times to execute individual paths as agents are forced to wait for other agents to complete their local state transitions, and agents need to determine (through messages) the local states of all other agents.

Improved execution policies are Minimum Communication Policies (MCPs) that take the shortcomings of FSPs into account, by reducing the need to wait and the amount of communication required between agents. A transitive reduction of the valid MAPF-DP plans is constructed, which highlights the interdependencies between local states between agents. In this way, agents carry out their plans until a specific local state, at which they must wait for a specific message from another agent. When they have received such a message, they are able to continue with their plan. This significantly reduces the amount of communication and time taken to carry out the plan, as only certain identified local states are required to wait until a message is received.

Experiments were performed using this method, in environments very similar to the warehouse structure presented in Figure 5.15. Utilizing the presented AME MAPF solver in the MAPF-DP had a success rate of 80% in such scenarios, with an average plan execution time of approximately 118s for 35 agents in 10 instances, and a runtime of 0.12s to generate the solutions. In further experiments, a 30x30 grid with 10% obstacles is used. The probability of not executing a plan is increased, and the runtime, approximate time to execute the plan, and number of messages all increase almost linearly. Finally, when increasing the number of agents from 50 to 150 (in steps of 50), the success rate decreases from 94% to 68% to 10% respectively, with the runtime increasing from 0.17s to 4.7s to 134s. Interestingly, the average time to execute the plan does not increase so drastically, increasing from 75s to 87s to 96s. Finally, utilizing MCP instead of FSP results in almost a factor 2 less average time to execute plans, and almost 90 times less communication (messages) being required.

It is questionable whether plan execution which is based on a time step by step schedule is applicable in operational environments at Schiphol as each aircraft does not have the ability to communicate or know the specific plans of each other aircraft at each time point. However, it is possible for decentralized elements on the airport's surface, for example, to be able to form this type of communication and therefore MCPs are more realistic to be applied. Furthermore, it is not entirely understood how feasible of an assumption it is that the probabilistic element of aircraft's chance of no action (move or wait) being determined is, especially in an airport environment.

5.4.3. Push and Rotate

Push and Rotate is a planning and coordinating algorithm that is presented in [72]. It is a complete algorithm that builds upon the Push and Swap algorithm [73], which guarantees a solution (albeit not optimum) for instances having at least two unoccupied vertices.

First, a pre-processing phase is undertaken in which the graph is divided into sub graphs, and then agents are assigned to the subgraphs. The manner in which the graph is split is not arbitrary. Graphs are sub-divided into biconnected components, where agents within biconnected components can exchange positions. Finally, an order with which agents are planned for is determined.

After the pre-processing is completed, the agents are "moved" (virtually) to their destinations, in a similar approach to the MAPP method presented in subsection 5.4.1 in order to determine the sequence of actions which agents must undertake to reach their destinations. For each agent in the ordered list, as previously mentioned, the shortest path is computed from its origin to destination, after which it is attempted to be moved forward along this path. If other agents are encountered, i.e. which block the forward motion, then an action needs to be taken depending on whether or not the blocking agent has a higher priority than the moving agent. If the blocking agent is of lower priority, then that agent is attempted to be moved to an empty vertex. If this is not possible, or the blocking agent is of higher priority, then a "swap" operation is attempted as shown in Figure 5.7.

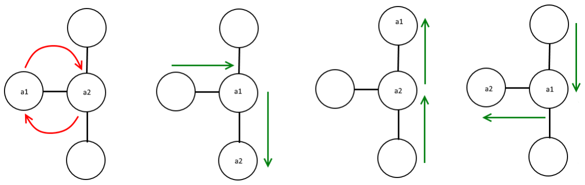

Figure 5.7: Example of a swap operation.

It is important to note that only the agents $a1$ and $a2$, as depicted in the figure, should be moved and other agents should not. Furthermore, a detection is carried out which determines if a cycle of agents requires to move forward. If this is the case, then these agents are advanced by a "rotate" operation, as shown in Figure 5.8.

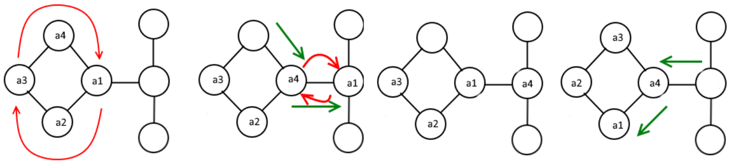

Figure 5.8: Example of a rotate operation.

Additionally, this rotate operation is performed on agents that have reached their destination and may need to be moved out of the way for another agent's path. In this way, the agent is simply rotated around to its destination (again).

The method is experimented on a video game map, presented in Figure 5.9, has 13765 vertices and varies the agents from 100 to 2000 in steps of 100 agents.

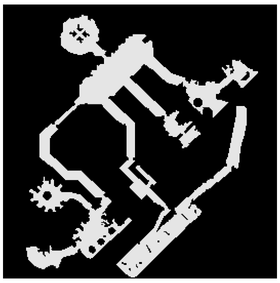

Figure 5.9: Baldur's Gate II map used for experimentation [12]

The results of this show that the number of moves required to solve all agent paths varies linearly with the number of agents, requiring almost 250000 moves for 2000 agents. Interestingly, the authors also applied the MAPP method (see subsection 5.4.1) on the same map, and the results for this show that the number of moves varies exponentially with the number of agents, reaching almost 2500000 moves for 200 agents and 750000 moves for 1500 agents. In terms of runtime performance, the Push and Rotate method is the best performing, having an average runtime of a couple of seconds for all number of agents (up to and including 2000). The MAPP method varies linearly, having a runtime of approximately 150s with 2000 agents.

As with the MAPP method, the feasibility of the geometrical nodes at Schiphol is questionable. Furthermore, aircraft cannot U-turn in any type of operation on their own power due to the taxiway width limitations, and may require towing to do so. Additionally, this results in them having to travel extra distances which is not efficient.

This method also discusses the fact that if an agent at a destination needs to be moved, then it can be rotated. However, aircraft which have arrived at their destination cannot be moved, either because they are waiting for departure at a runway, or because they have shut down the engines at the gate. Therefore, this is not possible to be carried out in order to avoid these types of conflicts.

5.5. Coordination After Planning

Coordination after planning refers to validating a set of individual paths of agents to discover what sequence of actions may lead to conflicts. Based on this, appropriate plan modifications must be made in order to find a satisfactory set of individual plans that are conflict free, and help achieve the global goal of the system.

5.5.1. CBS: Conflict Based Search

Conflict Based Search (CBS) is a MAPF algorithm that is presented in [13, 74]. The basic principle of this approach is based upon constructing structured search trees to determine if and where conflicts occur after the individual plans of agents have been created. Based on this, alternative paths are determined in order to resolve the conflicts by assigning different paths constraints to agents.

CBS is a high and low level algorithm which uses a Constraint Tree (CT). The nodes in the CT tree contain:

- A set of constraints: contains information on the agents, the vertext of collision and the timepoint of collision.
- A solution: A set of all individual paths of all agents
- The total cost: Summation of all individual agent path costs of the current solution

The high level searches the nodes in the CT and validates the solution to determine any collisions. If, during the validation, a conflict is determined, then the node being considered is declared a non-goal node and then branched into two child nodes with constraints. These constraints are time and location based which are used in the low level search to avoid the conflict point either by moving to adjacent nodes, or by waiting at the current node. After the high level search has completed, and a non-goal node has been declared, then the low level processes the CT node. The low level search is associated with determining the individual paths of agents based on their origin, destination and constraints. After the node has been processed by the low level, it the high level validation is then performed again. If no conflicts are found, then the node is considered a goal node and the optimum [13, 74] solution has been found. Figure 5.10 presents a flow overview of this algorithm.

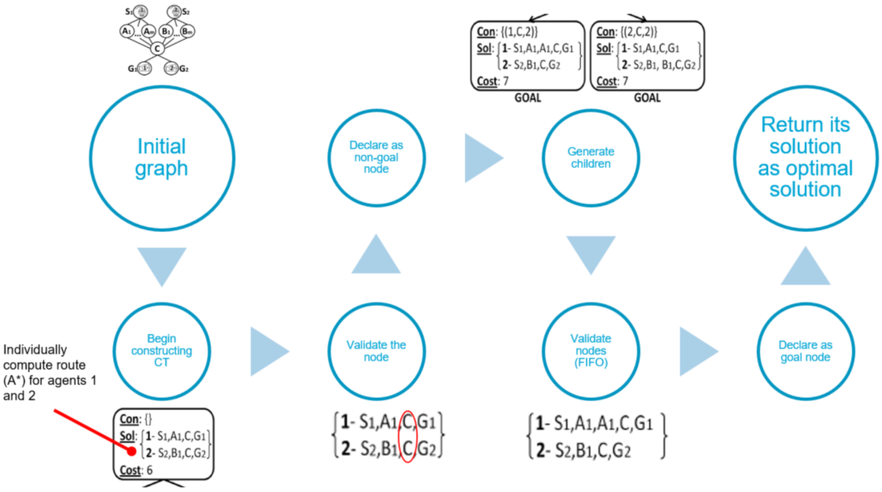

Figure 5.10: Flow of the CBS algorithm

An experiment was performed in order to benchmark the CBS algorithm to other established MAPF algorithms. Figure 5.11 presents the experimental results as published in the literature [13].

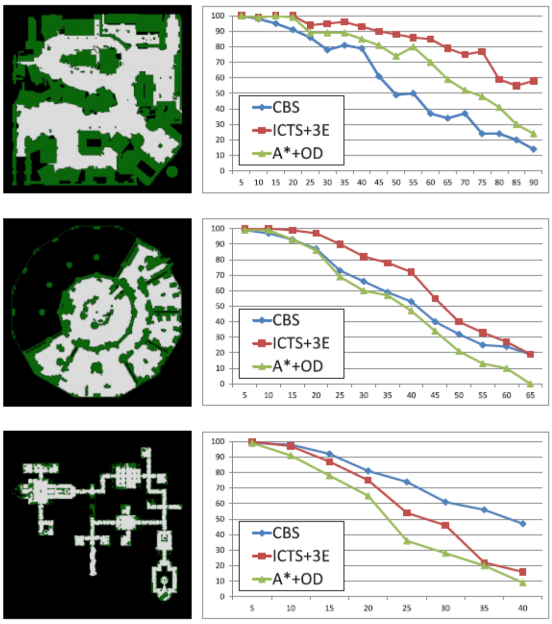

Figure 5.11: Experimental results as presented in the CBS paper [13]

The three maps on which the CBS algorithm was experimented on are depicted on the left of Figure 5.11, the x-axis shows the number of agents, and the y axis shows the success rate (% of solved agent plans). Qualitatively, by observation, one can understand the difference in the maps. The topmost map has wide spaces as well as corridors, the middle map has primarily wide spaces, and the bottommost map has narrow corridors. The results presented in the literature indicate that CBS is worst performing for wide spaces with corridors (top map), and best performing on maps with narrow corridors and potential bottlenecks. All maps also show deteriorating performance with increasing number of agents. The best performance is for the top map, as it has a success rate for the biggest increase in agents (shallowest linear gradient), indicating that the CBS in maps similar to this geometry is able to cope with the most number of agents.

In terms of time performance, the literature only presents the runtime for computing the solution for 13 agents on an 8x8 grid. This was stated to be 36.2s, lower than that for the ICTS3 algorithm (46s), where the A*+OD algorithm was not able to find a solution.

However, the most interesting map, qualitatively, is the bottom map as this is similar to the layout of Schiphol airport. In terms of the success rate, CBS performs 5 times better than A*+OD in the hardest, 40 agent case, and 3 times better than ICTS+3E, at approximately 48%.

As previously mentioned, conflicts are both time and location based. In other words, when two individual plans have the same location at the same time point, they are assumed to collide when executing their plans. From a theoretical viewpoint, this is fairly straight forward to understand as perfect plan execution is assumed. However, in reality, it is rare that plans are executed perfectly, without delays and without errors. Aircraft on the airport taxi at different speeds, and flight crews have different execution policies due to their individual personalities, airline policies, aircraft types etc. For this reason, accurately estimating the exact time point at which an aircraft with corresponding flight crew reaches a designated vertex in their route is difficult. For this reason, perhaps including time windows which can account for these imperfect plan executions may be interesting to consider. This concept is further discussed in the MAPF-POST and MAPF-DP methods, presented in subsection 5.5.3 and subsection 5.4.2.

Furthermore, as previously mentioned, plans are coordinated by either moving to adjacent nodes, or waiting at the current node. There may be scenarios which create long waiting times at a node, in order to resolve the conflict. One could expect that if this node is positioned close to the graph entry/exit points position by runways or the gates, safety bottlenecks may occur such as aircraft not being able to vacate the runway in time. This could be reduced by relaxing the problem to not find optimal solutions, but to find suboptimal solutions which prevent such scenarios of extensive waiting periods.

5.5.2. CBS Improvements

A lot of supporting literature was found into various additions and improvements to the original CBS method, presented in subsection 5.5.1. As the improvements significantly improve the runtime performance, and success rate for this algorithm (in addition to this seeming to currently be one of the most active and state-of-the-art MAPF approaches) these improvements will now be presented.

Optimal Variants

Meta Agent Conflict Based Search (MA-CBS) [13] is a slight variation on the initial CBS method. Instead of always branching into children nodes (with respective constraints), another option is used when a solution is returned as invalid (see subsection 5.5.1). This option is called "Merge" which consists of merging two conflicting agents into a single meta-agent. Once this merging has occurred, the low level search is then executed for this new meta-agent, using an optimal MAPF solver. The decision to branch or merge is established by a "merging policy". This merging policy can be designed in a number of ways, but the way presented in the literature is based on number of conflicts. For example, if agents a_1 and a_2 have a number of conflicts greater than a conflict limit parameter, B, then they should be merged into a meta-agent. Otherwise, they should be branched as per the CBS method. Figure 5.12 presents the flow of this algorithm.

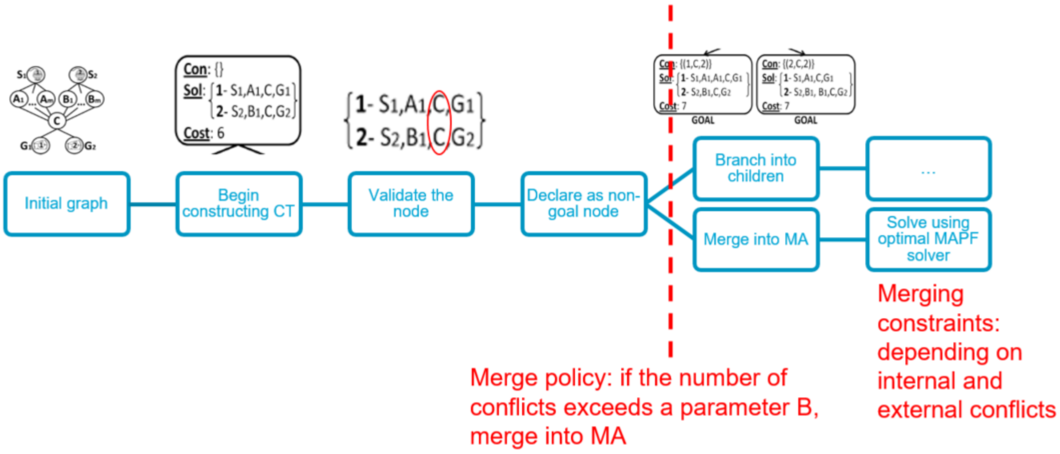

Figure 5.12: MA-CBS flow [13]

The same experiment as per the CBS method in subsection 5.5.1 is performed but now using the MA-CBS. Figure 5.13 presents the results for the same three maps, as published directly in the paper.

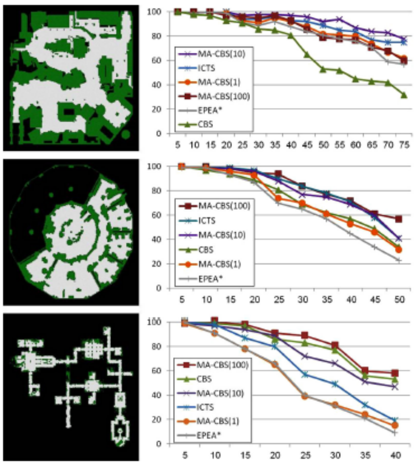

Figure 5.13: Experimental results as presented in the MA-CBS paper [13]

In all three maps, the MA-CBS(B) method is the best performing in terms of success rate. In the topmost map, the MA-CBS is almost 2.5 times better than the normal CBS method in the worst case of 75 agents, is almost 2 times better than CBS in the middle map, and almost 1.2 times better in the bottom map.

The results suggest that merging agents into meta-agents has the best effect in maps with open spaces and some lanes (top and middle maps). There is not much improvement in the bottom map, which is the map that is deemed to qualitatively resemble Schiphol airport. This suggests that, perhaps, there is not a major improvement in using this variant instead of the basic CBS version, in terms of this research at Schiphol airport, as the results only slightly improved in this map. Therefore, this further emphasizes the suspicion that this low level of success rate in such maps may be linked to the optimality of solutions.

The Improved Conflict-Based Search (ICBS) [75] presents three overall improvements to the (MA-)CBS method. These improvements are as follows:

- Merge and Restart (MR): In the MA-CBS, when the merge decision is reached, then disregard the current CT and construct a new CT using the new merged meta-agent as a single agent for the entire tree.
- Prioritizing Conflicts (PR): Prioritize conflicts based on three categories (cardinal, semi-cardinal and non-cardinal), and hierarchically solve conflicts based on the categorisation of them.
- Bypass (BP): Bypass a conflict if possible.

Experimental results in an 8x8 grid with 15% obstacles showed that the best performing ICBS variant was the MA-CBS+BP+MR. The experiment was performed on an 8x8 grid with 15% obstacles. The worst case of 45 agents presented the highest success rate of any other combination of improvements, with a success rate of approximately 90%, whilst taking approximately 0.5s to compute the solution. Comparing this to the CBS performance of 13 agents in 36.2s on a similar 8x8 grid, discussed in subsection 5.5.1, shows that this ICBS has a better performance.

Bounded Suboptimal Variants

The Enhanced Conflict Based Search (ECBS) [14] is a CBS variant that reduces the optimal solution aspect of the CBS, making it a bounded suboptimal algorithm that guarantees a solution which is no larger than a given constant greater than the optimal solution cost. The optimal CBS method, as previously shown in subsection 5.5.1, produces relatively low success rates for high number of agents, thus indicating scalability and runtime issues. A focal search in combination with a bounding factor is used in this variant.

Focal searches consist of two lists of nodes: OPEN (regular OPEN-list of A*) and FOCAL (which consists of a subset of OPEN). Two functions f_1 and f_2 are used for this. f_1 determines which nodes are in FOCAL. This method uses a suboptimality factor w in determining which nodes are in FOCAL. All nodes n in OPEN which are $f_1(n) \leq w \cdot f_{1,min}$ are in FOCAL. f_2 is used to choose which node from FOCAL to expand. Based on this, the returned solution has a cost of $w \cdot C^*$, where C^* is the optimal solution. This search is used in the low level of the CBS, which in turn enables the high level CBS search to determine which nodes to expand in this suboptimal way, instead of expanding all of them as was the case in CBS.

Indeed, an experiment is presented in the literature, in order to to compare this performance to that of the CBS. In a 32x32 grid, with 20% obstacles, the CBS is only able to ensure a 100% success rate for 15 agents at most, where as the ECBS is able to ensure 100% success rate for up to 60 agents. This indicates that reducing the optimality of the method with an arbitrary constant, and thus bounding the optimality, results in a significant increase in success rate. Furthermore, this ECBS method was applied to the bottom map which is similar to Schiphol airport, described in subsection 5.5.1 and in Figure 5.13. These results, taken directly from the paper, are presented in Figure 5.14.

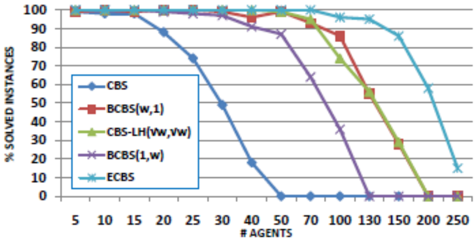

Figure 5.14: Experimental results as presented in the ECBS paper [14]

As can be seen, the ECBS algorithm applied to the Schiphol-like map has significantly better performance than the CBS method, but also from the MA-CBS presented in Figure 5.13. The ECBS method is able to resolve

100% of the individual plans of up to 70 agents, and approximately 87% of the plans of 150 agents. This is significantly better than the MA-CBS method which is able to solve approximately 100% of the plans for 15 agents only (after which it significantly decreases, as seen in Figure 5.13) and of CBS which solves 100% of the plans for 15 agents only (as seen in this figure, Figure 5.14). This suggests that the using sub-optimal solvers such as ECBS is most effective for having the highest success rates in Schiphol-like maps.

Following from this ECBS method, another method called Enhanced Conflict Based Search with Highways (ECBS + HWY) [15] was found. This uses the ECBS structure with additional logic in the heuristic function in the low level search. The effect of this is to alter the global heuristic values for certain paths of nodes that are defined as "highways". The benefit of this is that agent paths are encouraged to be formed along these highways. This interesting in areas with limited paths, directed paths, or bottlenecks. An experiment is performed in the paper on a grid-lane structure, where 150 agents need to travel from one end of the grid to the other. The structure is taken from the published paper is presented in Figure 5.15.

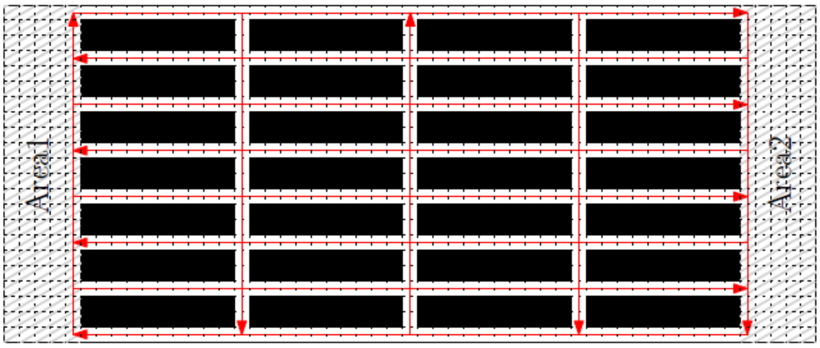

Figure 5.15: Highway structure [15]

Results are presented in the paper which show that incorporating highways in such narrow lane areas reduces the solution cost by 5% (thus approximating improving the optimality of the solution) and can double the success rate when not using highways. The average runtime required to find valid solution (in the best performing bound constant w case) was approximately 151s.

Additionally, the ECBS+HWY may provide a better method of implementing the directional flows in the inner ring of Schiphol airport, instead of creating directed edges within the graphs, as discussed in the coordination before planning in section 5.3 and chapter 2.

This might be a more flexible way of imposing the taxiway directions as it encourages the agents' paths to follow these flows. The benefit of this way of imposing directional flows, and the benefit of not imposing these flows as directional edges in a graph, is that if there are blockages or sections of these cannot be followed as specified in the figure, then it is still possible for paths to be planned against these flows, although a higher cost will be undertaken. This suggests that benefits of relaxing the law (as discussed in section 5.3) can be enjoyed potentially resulting in higher success rates of routes. When there is no blockage, however, the flows will provide a lower cost for the paths of agents, and thus will be followed. Furthermore, if necessary, additional, dynamically varying highways can be constructed depending on the mode of operations and availability of the taxiway network, or the "gravity" (heuristic contribution) of the highways can be altered. For example, a "high gravity" for the flows results in "more encouragement" for paths to follow the flows, where as a "low gravity" has the opposite effect. This gives further flexibility with respect to determining how strictly the highways could be followed. Continuing with this idea, literature [76] presents methods for automatically generating highways (learning) in this MAPF, in a similar manner to the resilient literature found and presented in section 4.2. In such a way, the highways can be modified based on the local geometrical environment (based on the taxiing aircraft, for example), therefore suggesting that this is desirable from a resilient behaviour viewpoint.

5.5.3. MAPF-POST: Post MAPF processing

MAPF-POST is a method that coordinates individual plans by post processing them, and is presented in [68]. This coordination method actually aims at minimizing the amount of time needed to complete the paths whilst ensuring collision-free paths, by determining a plan-execution schedule for each of the agents. Although MAPF solutions result in a coordinated plan execution schedule, it is almost impossible to precisely execute the plans due to imperfect plan-execution capabilities (such a unforeseen delays or flight crew/aircraft variations in the scope of this research). Therefore, the post processing of these results taking into account velocity limits and additional slack (to ensure safety and incorporate delays) resulting in a plan execution schedule is proposed in this method.

First, the collision free MAPF solution is determined, using any MAPF solver. From this, an augmented Temporal Plan Graph (TPG) is constructed (augmented to include additional safety buffers for spacing) which relates the individual plans of the MAPF solution, and their related interdependencies. Based on this, a Simple Temporal Network (STN) is then constructed which uses $t = \frac{d}{v_{max}}$ as lower time bounds (constraints) at which agents must pass through certain nodes. Following from this, the STN, with associated time constraints, is formulated as a Linear Programming (LP) problem and solved. The outcome of this is a plan execution schedule that takes into accounts the different speeds of the agents, as well as additional safety spacing buffers. Figure 5.16 presents a flow of this method.

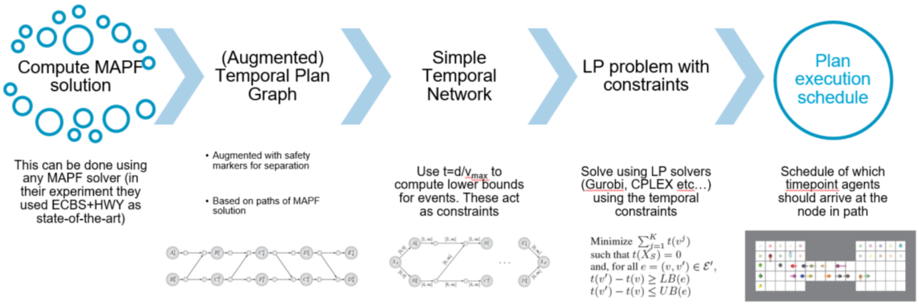

Figure 5.16: Flow of the MAPF-POST algorithm

This approach was evaluated in simulations but also implemented in reality, using robots[3]. In a comparable experiment, 100 agents were used in a layout very similar to that of Figure 5.15 in the ECBS method described in the suboptimal variants of the CBS in subsection 5.5.2. For this case, this method took approximately 6mins to compute the plan execution schedule, with an average time to complete the plans of 88s. This shows that this method took a significantly longer time to compute the coordinated solution, when compared to the ECBS+HWY which took approximately 151s, which is also interesting as this MAPF-POST only considered 100 agents, whereas ECBS+HWY considered 150. However, no times with respect to how long each agent took to complete their adjusted plans are stated for the ECBS+HWY, and thus it is not possible to compare the (relatively) short plan time of 88s of this method.

In general, this approach is flexible in that it can be applied on top of any optimal MAPF solution, such as that of the CBS in subsection 5.5.1, in order to address the issue of imperfect plan execution. Including the elements of lower bounds as well as additional spacing buffers does improve the feasibility of the method, which therefore takes into account the issues associate with delays in the execution. In addition to this, for suboptimal MAPF solvers such as the ECBS+HWY presented in the experimental validation, not only does it improve the feasibility of the imperfect plan execution, but it also plays a role in improving the overall optimality of the solution by minimizing a cost function with respect to time, for example. These benefits indicate that this is a flexible method that is able to approximate the real world operations at airports by the use of these imperfections.

5.6. Trade-Off

This section presents the trade-off of all the previously discussed mechanisms for cooperative coordination in multi-agent systems.

5.6.1. Criteria

In order to further quantify and compare the different coordination mechanisms found, a trade-off is performed, based on a scoring system of 1 - 3. Each mechanism is scored based on three areas: implementation, performance, and connection ability with resilience mechanisms. These are further sub-divided into more specific criteria for each topic and will now be described. The higher the score, the more suitable the algo-

---

[3] Videos of both the simulations and the robot experiments can be found at: youtu.be/mV3BqnelqDU

rithm is deemed to be.

Complexity
Complexity of the mechanism relates to the ease of implementation. Some mechanisms require significant amounts of programming, system design and architectures, as there are many steps linked with carrying out the mechanism. Other mechanisms require less. Furthermore, there may be multiple inputs and outputs and communication mechanisms required in order to execute the coordination. Furthermore, complexity is related to project risk as the higher the complexity associated with a mechanism, the more likely it is that more time or certain elements cannot be included. Therefore, the lower the complexity with an associated mechanism, the higher the score given to the mechanism.

Availability
Availability relates to the amount of resources associated with the mechanism. This can be in the form of other literatures, but also relates to if there are already sample codes or GitHub repositories that contain programming aspects of the mechanism. The higher the score, the more available the resources are.

Resources
Resources relate to the computational resources required in order to both implement and execute the mechanism. Specifically, this relates to how open-source required software is, and how much RAM and hard drive memory and processing power is required in order to execute it. The higher the score, the less resources that are required.

Scalability
The mechanisms vary in terms of computational time, success rate, solution costs based on the number of agents for which plans must be coordinated. Scalability relates to how well the mechanisms are able to scale with increasing number of agents. This is interesting to consider as the airport environment has varying and sporadic traffic flows: there may be periods with low amounts of aircraft agents, followed by periods with high amounts of aircraft agents. The discussion of the mechanisms in the aforementioned sections already suggests that all mechanisms are able to cope well with low levels of agents. The mechanisms, however, significantly differ with a high number of agents. For this reason scalability is considered as how well mechanisms are able to cope with large amounts of agents. The better, in terms of computational time and success rate, a mechanism is able to cope with large amounts of agents, the higher the score.

Realism
Realism relates to if realistic operational aspects are already included in a mechanism. For example, imperfect plan execution or kinematic constraints are realistic operational aspects as they occur in almost all operational environments. Although it may be possible to integrate these operational dynamics, this criterion assesses the literature-found ability of the mechanism, as is. The more the operational aspects that a mechanism contains, the higher the score.

Feasibility
Feasibility relates to the the applicability of the mechanism to a graph similar to Schiphol Airport. In some cases, only simple graphs or graphs with specific geometric limitations were required by certain mechanisms. These raise questions with regards to how feasible they are to be applied to Schiphol Airport, or whether the airport surface must be over-simplified in order to accommodate them. The higher the feasibility score, the more suitable the mechanism is to be applied on a graph similar to that of Schiphol Airport.

Assumptions
This criterion relates to the amount of (known) operational assumptions associated with a mechanism. This primarily relates to assumptions such as all agents moving at a fixed constant speed, instantaneously changing direction, executing plans perfectly, or allowing backwards motions. The higher the score, the less the operational assumptions that are associated with the mechanism.

Success Rate
Success rate, as discussed in the previous subsections, relates to how many of the total individual agent plans were successfully coordinated to become conflict free. This is of interest as mechanisms with higher

success rates indicate that they are able to provide conflict free paths for more agents than others. This will be considered for the "hardest" case which is the case where the most amount of agents need to be coordinated. The higher the score, the higher the success rate of the mechanism.

Conflict Resolution

Conflict resolution relates to the location at which the mechanism resolves conflicts. The maximum score is given to mechanisms that offer conflict free routes at vertices and edges, with the least score given to mechanisms that do not offer conflict resolved solutions.

Flexibility

Flexibility relates to how much elements of a mechanism are able to adapt and result in changes at an operational level. This is interesting to evaluate how able a mechanism is to modify system functions, which is related to resilient behaviour. For example, if there are multiple elements that can change operational aspects (such as resolving conflicts but also directing traffic along predefined routes), then a mechanism is considered to be flexible. The more flexible the mechanism, the higher the score.

Compatibility

Compatibility relates to how possible it is to alter parts of the mechanism in terms of inputs, outputs and inter-system blocks, and therefore how able the mechanism is able to be connected to other resilience mechanisms beyond this mechanism. A mechanism that is more able to be altered with respect to inputs, outputs and inter-system blocks is, potentially, more able to be connected to resilience, and other decentralized control mechanism elements, and vice versa. The higher the score, the more compatible it is with other mechanisms.

5.6.2. Trade-off

The previously mentioned criteria are applied and each mechanism is evaluated. Table 5.1 presents the trade-off table for the coordination mechanisms.

Table 5.1: Trade-off of cooperative coordination mechanisms

	DEC-A*	CBS	MA-CBS	ECBS	ECBS+HWY	MAPF-POST	MAPP	MAPF-DP	Push & Rotate
Implementation									
Complexity	3	2	1	2	1	1	2	1	2
Availability	1	3	2	2	2	1	2	2	3
Resources	2	2	2	3	3	1	2	1	2
Performance									
Scalability	3	2	2	2	2	2	1	2	2
Realism	1	2	2	2	3	2	2	3	1
Feasibility	2	2	2	2	3	3	1	2	1
Assumptions	1	2	2	2	2	2	1	3	2
Success Rate	1	1	2	2	2	2	2	2	2
Solution	3	3	3	2	2	2	1	2	1
Conflict Resolution	1	3	3	3	3	3	2	2	2
Connection Ability									
Flexibility	2	1	2	2	3	1	1	2	1
Compatibility	2	2	2	2	2 3	2	1	2	1
Total	22	25	25	26	29	22	18	24	20

As can be seen, the ECBS+HWY mechanism results in the highest evaluation score of 29, with the ECBS and (MA-)CBS receiving second and third place respectively. For this reason, it could be recommended that the ECBS+HWY suboptimal variant of the CBS with highways be used for the routing service of the Follow-the-Greens (as presented in chapter 3). However, this mechanism mainly had the highest score due to the good connection ability criteria (to resilience mechanisms) due to the HWY concept. Furthermore, there is much complexity with this mechanism, and it is built on CBS. For this reason, it is recommended to first use CBS+HWY in order to to assess if this results in sufficient performance (such as high success rates and short computation times) as the core CBS is both less complex to implement, yields an optimal solution, and is also able to incorporate the HWY feature. If it does not yield acceptable results, then the CBS should further be developed into the ECBS variant. Therefore, the CBS+HWY is proposed to be used in this MSc study. Furthermore, for highway determination, the learning highway mechanisms presented in subsection 5.5.2 could also be incorporated in order to determine appropriate highways based on traffic levels as well as runway reconfigurations, in order to adapt in a similar manner mentioned to the literature found in the resilient behaviour section, section 4.2. This is included in the research methodology in chapter 7.

6

Research Proposal

Now that the main literature which is relevant to this MSc study has been reviewed, a research proposal which this MSc will focus upon can be presented. section 6.1 presents the research objective and section 6.2 presents the main research question and associated sub-questions.

6.1. Research Objective

Airports are complex environments with fixed resources (such as taxiways and runways), which are constantly exposed to different types of adverse events. In order to cope with these, resilient behaviour (see chapter 4) is required, in order for the operations to adapt and ensure the system goal of maintaining the safe and efficient flow of ground traffic during adverse events.

One regularly occurring (planned or unplanned) adverse event that airport surface operations must be able to cope with is that of runway reconfigurations, which occur from many contributing factors as discussed in section 2.3. This results in an altered airport situation with different taxi routings which raises issues with respect to conflicting traffic flows which have to be re-routed in order to accommodate the new runway configuration, for example. During these events, airport surface movement operations should be altered in order to result in resilient behaviour and maintain the global goal of the safe and efficient flow of traffic from origin to destination on the airport's surface.

Currently, ATCOs at airports closely resemble a centralized system where one ATCO controls and commands aircraft in their sector in order to ensure the safe and efficient flow of traffic. Naturally, workloads and performance of operations vary and possibly deteriorate with the amount of traffic and number of movements under a controller's control. Adverse events such as runway changes further introduce complexity and workload for the ATCO's human performance as new routing and traffic solutions must be found. One change in system architecture in order to improve these factors is that of decentralized control, where decision making is shifted from the tower control to a lower, local level by placing multiple interacting virtual agent controllers on the taxiway system which operate and issue commands on local information in order to accomplish global goals. This may result in higher autonomy and allows, potentially, a better organization of the system, especially in the case of adverse events. Furthermore, this presents a shift in the ATCO's role from a very active control role to more of a supervisory role.

One decentralized control outcome of the SESAR programme is the concept of an A-SMGCS called Follow-the-Greens, presented in chapter 3. This is a decentralized control system for airport surface movement operations based on the taxiway lighting system. During the SESAR validation of this concept (as presented in section 3.2), improvements in terms of capacity, environment, human performance, safety and predictability of airport surface movement operations through the use of this system were confirmed.

However, as mentioned in section 3.4, although this decentralized control system is very promising and state-of-the-art, no results with respect to how well it is able to cope with adverse events, nor what mechanisms are required in order to do so were explicitly found. A system is considered resilient if it has the intrinsic ability to adapt its functioning in response to such adverse events so that it can sustain the required operations. Therefore, a decentralized control system such as the Follow-the-Greens concept should result in resilient airport surface movement operations during runway reconfigurations in order sustain the core goals of the safe and efficient operations of airport movements.

For this reason, the research objective is formulated as follows:

> *"To investigate the contribution of an airport surface decentralized control system to the resilience of airport surface movement operations by assessing how well cooperative coordination mechanisms deal with both planned and unplanned runway reconfigurations."*

This research objective, therefore, aims on further elaborating, operationalizing and building upon on the Follow-the-Greens decentralized control concept with respect to resilience, primarily focussing on cooperative coordination and resilience related mechanisms as presented in chapter 4 and chapter 5 respectively. The adverse events that will be considered are planned and unplanned runway reconfigurations, as presented in section 2.3, and Schiphol airport will be used as a case study for airport surface movement operations.

6.2. Research Questions

Following from the research objective mentioned in section 6.1, the main research question that is associated is as follows:

> *"How effective, with respect to the performance of airport surface movement operations, can an airport surface decentralized control system and its cooperative coordination mechanisms be in achieving resilient airport surface movement operations during planned and unplanned runway reconfigurations?"*

In order to answer the main research question, the following key questions have been derived, which directly stem from this literature study.

1. What are important elements for airport surface movement operations?
    (a) What are the standard airport surface movement operating procedures?
    (b) How is the airport ground infrastructure utilized?
    (c) What elements, interactions and processes are involved?
2. How do runway reconfigurations take place?
    (a) What factors influence runway reconfigurations?
    (b) What are the issues associated with runway reconfigurations?
    (c) What are the possible runway reconfiguration combinations?
    (d) What are the differences between planned and unplanned runway reconfigurations?
    (e) What procedures are associated with runway reconfigurations?
3. How can an airport surface decentralized such as the Follow-the-Greens control system be characterized?
4. Which mechanisms and techniques that are associated with achieving resilient behaviour are most suitable for a resilient response of airport surface movement operations to a runway reconfiguration?
5. What mechanisms and techniques for cooperative coordination in multi-agent systems are most suitable for an airport decentralized control system?
6. How can cooperative coordination mechanisms be expanded to achieve resilient behaviour in airport surface movement operations during planned and unplanned runway reconfigurations?
7. To what extent can real-world airport surface movement operations, planned and unplanned runway reconfigurations and an augmented airport decentralized control system be integrated in a model?
    (a) To what extent can (socio-)technical system elements can be simplified?
    (b) Which (socio-)technical system elements cannot be modelled?
    (c) How can runway reconfiguration effects propagate in a multi agent system?

8. When simulating planned and unplanned runway reconfigurations, how do the outcomes of the simulation model compare to the performance of airport surface movement operations before the runway reconfigurations?
    (a) What metrics will be used to assess the comparison?
    (b) What metrics will be used to assess the resilient behaviour of such outcomes?
    (c) Which runway reconfiguration scenario will be simulated?
9. What recommendations can be made to the Follow-the-Greens A-SMGCS system with respect to resilient responses following runway reconfigurations?

7

Research Methodology

This chapter presents the proposed methodology to be used in order to fulfil the research objective and answer the research question presented in section 6.1 and section 6.2 respectively. Firstly, section 7.1 presents the proposed modelling approach, section 7.2 presents the scope of the research including assumptions, section 7.3 presents the model development methodology, and finally section 7.4 presents the simulation, analysis and conclusions methodology.

7.1. Modelling Approach

The aim of this literature study report is to review the state-of-the-art approaches which are relevant to the research objective. Now that relevant areas have been established, and literature has been reviewed, it is possible to determine a modelling approach.

7.1.1. Modelling Technique

From the presented literature, it is clear that airport surface movement operations, as well as the decentralized control of Follow-the-Greens form a complex socio-technical multi-agent system. Multiple interacting elements appear both in conventional airport surface operations, as described in chapter 2, as well as in the internals of decentralized control systems (as discussed in chapter 3).

Furthermore, an elaborate Agent Based Model (ABM) of the ground operations at Schiphol Airport has already been designed [4], which serves as a solid platform to expand upon. This will be discussed further in subsection 7.3.1. Additionally, in the achieving resilient behaviour in systems chapter in section 4.2, multiple literatures were found that confirm and support the feasibility of ABM for resilience related studies.

This, in combination with the fact that elements, interactions and interdependencies exist between airport surface movement operations, runway reconfigurations and decentralized control systems, suggests that it is indeed possible to develop an overall agent based socio-technical simulation which is able to simulate the effects required by the research objective. Furthermore, it also enables the design of inter-connected mechanisms, as a computer programming language is able to be used for the development of such mechanisms, and is also able to simulate the effects of them. For this reason, the primary modelling technique which will be used to fulfil the research questions will be ABM.

7.2. Research Scope

Airport surface movement operations as well as runway reconfigurations at Schiphol were elaborated upon in chapter 2. Furthermore the Follow-the-Greens decentralized control concept was discussed in chapter 3. Culminating from the literature reviews presented in those chapters, the following assumptions are presented:

- Planned and unplanned runway reconfiguration will be based on operationally allowable runway configurations as presented in section 2.3.
- It is assumed that the amount of vehicle operations (on the airport's surface/airside such as towing operations) are significantly less than aircraft movements, and thus may be excluded. Therefore, only aircraft will be considered.

- Aircraft will carry out plans perfectly, without deviations or errors in plan execution. Therefore, aircraft will carry out all commands of the AGL with no mistakes.
- Apron operations, including pushback, will not be considered as the taxiway distance between an aircraft's stand and the apron exit is small, thus there is little room for control through the AGL. Therefore, aircraft will only taxi from the apron entry/exit (for departure aircraft) and to the apron entry/exit (for arrival aircraft).
- A simplified surface layout of the published Schiphol airport infrastructure will be used, where the aprons are not considered.
- Arrival aircraft are assumed to require a route from the taxiway point at which they vacated the landing runway, to the apron exit/entry. Departure aircraft are assumed to require a route from the apron entry/exit to the holding point of an active departure runway.
- Aircraft will begin to taxi once the command "follow the greens" has been issued to them. If the command has not been given, departure aircraft will hold position at the apron exit. Arrival aircraft will vacate the runway towards the side closest to the main terminal aprons, and hold after the runway hold short markers on the taxiway's surface.
- Aircraft will adhere to the rules of the air as presented in chapter 2.
- Figure 3.6 presented in section 3.5 will form the basis for constructing the ABM of the socio-technical system, as it is created using the core aspects of airport surface movement operations including the Follow-the-Greens decentralized control(as presented in chapter 3). However, the following simplifications can be made and the overall simplification is presented in Figure 7.1:
  - The surveillance service will not be modelled as it will be assumed that the location of each aircraft at each timepoint is known and the interrogation of the aircraft's transponder is therefore not required.
  - The airport safety support service will not be modelled as this is an advisory service that presents alerts to the ground controller, which must accordingly react based on the alerts. Instead, the types of situations which create these events will be included during the coordination of the plans in the routing service and guidance service. In particular the CATC functions will be part of the routing service which will also resolve these conflict predictions, and the RMCA and CMAC functions will be part of the guidance service.
  - The ground controller will be modelled as a managerial agent, which coordinates and interacts with the services, outbound planner and runway controller, as well as with aircraft.
  - Only the interaction of the outbound planner will be modelled as an augmented interaction of an input of a flight identification from a flight schedule of flights that have received startup clearance, have push backed and will request taxi clearance imminently. This will be communicated to the ground controller.
  - Only the interaction of the runway controller will be modelled as an interaction where a flight is transferred from the ground controller to the runway controller when it reaches the departure runway, or when it requires to cross an active runway. When an arrival aircraft vacates the runway (or has finished crossing), the interaction from the runway controller to the ground controller will be a transferring of the flight identification. These flights will be generated in accordance with the current and active runway configuration.
  - The mobile information database will contain all information about flights, including those that are communicated from the outbound planner and runway controller, as well as all surveillance related information of aircraft.
  - Planned runway reconfigurations will be in the form of an information transfer containing a runway configuration and an indication of a time window within which they will be activated, issued 0.5 - 1 hour before the activation window (as is done at Schiphol, discussed in section 2.3). Only the interaction from the airport operators/ANSP element to the airport operation status element will be modelled which will contain this planned runway reconfiguration information.
  - Unplanned runway reconfigurations will be in the form of an information transfer containing the runway configuration and time at which they changed from the interaction of the airport operators/ANSP element to the operation status element.
  - Inputs to the model are highlighted in the green circles, and outputs are highlighted in the red circles, as shown in Figure 7.1.

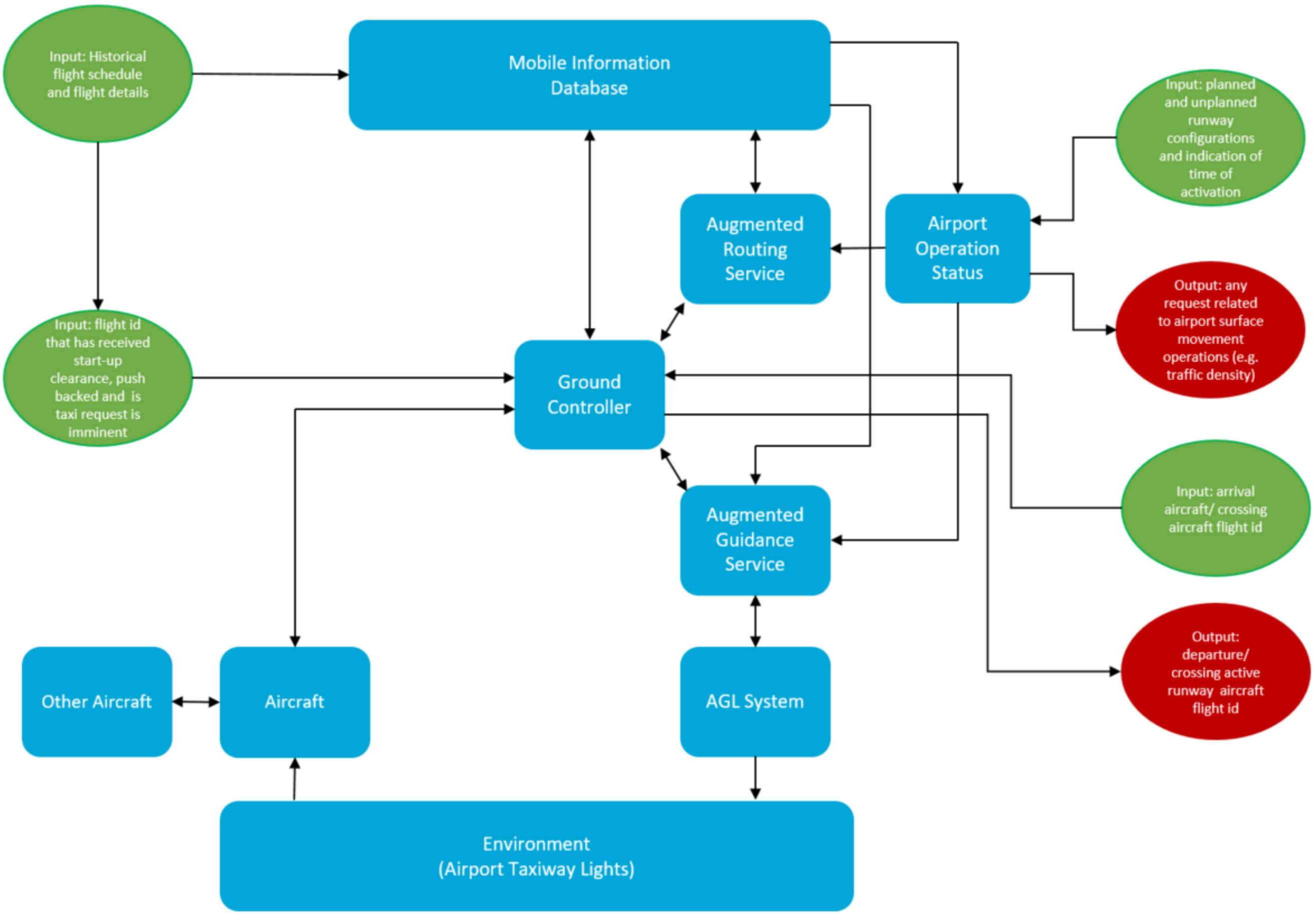

Figure 7.1: Simplified socio-technical representation of the Follow the Greens A-SMGCS which will be used as a basis for the model design, including model input and outputs

7.3. Model Development

This section presents the main steps and methodologies that will be undergone during this MSc study.

7.3.1. Augmenting and Operationalizing Cooperative Coordination Mechanisms Towards Resilient Behaviour

Using the basis of the model design presented in Figure 7.1, as well as the knowledge gained from this literature study report, it is clear that the routing service must be augmented to utilize and operationalize the selected cooperative coordination mechanism of the CBS+HWY mechanism chosen in chapter 5, as well as the mechanisms which result in resilient behaviour, presented in chapter 4. This is will be done by:

- Identifying indicators, primarily based on ICAO KPIs [55], which can be used to monitor airport surface movement operations and resilient behaviour before and after runway reconfigurations.
- Defining nominal/undisturbed (i.e. when no runway reconfiguration has occurred) performance levels for airport surface movement operations.
- Determining possible modifications such that the airport surface movement operations can adapt during planned and unplanned runway reconfigurations.
- Determining how highways can be generated/learned to achieve these modifications.
- Determining mechanisms which are able to execute the previously mentioned modifications utilizing the CBS+HWY coordination mechanism.
- Quantifying and comparing the effectiveness of such modifications with respect to the performance of airport surface movement operations. Therefore, the resilient behaviour associated with each modification can be assessed.
- Including anticipation elements for planned runway reconfigurations.

- Developing and implementing the previous bullet points in an overall augmented CBS+HWY mechanism in the Routing Service.
- Determining a plan execution method which will be shown by the AGL system.
- Determining when coordination and planning will occur.

The aim of this methodology is to expand the CBS+HWY cooperative coordination mechanism such that it is able to create adaptations in the airport surface movement operations as a response to planned and unplanned runway reconfigurations in a resilient manner. This will form the core mechanism of the augmented routing service presented in Figure 7.1.

7.3.2. Development of the Airport Surface Movement Decentralized Control System Model

Once the augmented routing service mechanism has been specifically developed, the rest of the elements and interactions presented in Figure 7.1 must be developed and the ABM constructed. This will be done in a common ABM approach which includes the following steps:

- Description of the environment.
- Description of the agents, properties and internal processes.
- Description of interactions between agents.
- Development of a runway reconfiguration generator which will be used as an input to the model for planned and unplanned runway reconfigurations. This will include the time window concerns, as discussed in subsection 2.3.3.
- Development of an imminent taxi request generator based on a flight schedule.
- Development of an arrival aircraft/crossing aircraft generator based on the active runway configurations.
- Representation of the Schiphol airport airport surface infrastructure as a graph with edges (taxiways) and nodes (taxiway junctions, runway holding points, apron entry/exit points).
- Development of the augmented guidance service utilizing decentralized intersection agents (AGL System) to illuminate planned paths for aircraft and create the Follow-the-Green effect as well as the augmented functions as discussed in section 7.2.
- Development of the remaining elements: Mobile Information Database, Ground Controller, Airport Operation Status, AGL System, Environment, and Aircraft elements as well as associated interactions.
- Development of the outputs of the model.
- Determination of verifcation and validation strategies on both a local and global level of the model.
- Determination of appropriate levels of abstraction between the elements.

The TU Delft has access to an already existing ABM Schiphol airport ground simulator, which has been used by former researchers for their studies [4, 35]. This has been validated and verified throughout their projects and therefore is of particular interest to be used as a baseline model for this MSc study.

Familiarisation with this simulator will be undergone using supporting material [4] (which was the last researcher to implement changes into the simulator and therefore has the most up-to-date documentation of this). Furthermore, the source code has already been acquired and a preliminary review of it has been performed, which includes installing it and its library dependencies and running basic experimental simulations. Although it was determined that the source code is complex and interconnected, it was also observed that it was thoroughly commented. For this reason studying the source code will also be performed to further support the documented literature.

The aim of this familiarisation will be to gain a working understanding of the simulator, as well as its implementations, current limitations, assumptions, architecture and simulation setup. Following from this, key elements which must be altered and changed in the source code will be determined, as well as expansions of new features such as the re-working of runway reconfigurations. The aim of this will be to modify the existing simulator such that it takes the form of the desired ABM.

7.4. Simulation, Analysis and Conclusions

Once the model has been developed, runway reconfiguration simulations will be run for the operationally allowable runway combinations, presented in chapter 2, utilizing the runway reconfiguration input model presented in subsection 7.3.2, and will be used to generate simulation results for both planned and unplanned runway reconfigurations. The results will describe how effective the behaviour of the model is with respect to the deviations in performance of airport surface movement operations from the nominal/undisturbed (no runway reconfiguration) conditions. Based on this, comparisons with respect to how the model deals with the respective planned and unplanned runway reconfigurations will be analysed and elaborated upon.

Furthermore, if it is required (and data is accessible), performance indicators of the simulations as well as from the real-world data analysis will be used in order to further validate and compare the simulation performance results with respect to reality. Data from sources presented in section 2.4 can be used for this. The outcome of this is to draw further conclusions and recommendations, especially highlighting and finding explanations for the differences between the real world and model results.

Finally conclusions and recommendations based on these analyses will be determined and the feasibility of the model with respect to the Follow-the-Greens implementation will be discussed.

III

Supporting Work

8

MAS Simulation Results

8.1. Simulation Specifications

Table 8.1 presents the variable values that were used in order to generate the results in the Master of Science Thesis Paper in Part I, as well as all subsequent results within this chapter.

Table 8.1: Variables used in the MAS simulations.

Category	Variable	Description	Value
Kinematics	V_{max}	Maximum taxi speed of the Aircraft Agents	15.4m/s
	V_{turn}	Turn speed for turns which required to be slowed down for	5.14m/s
	a_{accel}	Acceleration of Aircraft Agents	$0.26 m/s^2$
	a_{decel}	Deceleration of Aircraft Agents	$0.78 m/s^2$
	$\theta_{maxturn}$	Angle of turn beyond which v_{turn} should be utilized	30°
Mechanisms	T_{RO}	Runway occupancy time	60s
	T_{window}	CBS anticipated conflict detection window	15s
	n_{PM}	PM HWY generation threshold	2 aircraft
	$T_{PMpersist}$	PM HWY time to persist	300s
	n_{CB}	CB HWY generation threshold	3 commands
	$T_{CBpersist}$	CB HWY time to persist	180s

The variables of the kinematics category are chosen based on the Flight Crew Training Manuals (FCTM) of two typical aircraft which regularly Schiphol Airport's surface, namely the FCTMs of the Airbus A320 family aircraft [77] and the Boeing 737 [78] aircraft.

The variables of the mechanisms category are chosen based on calibrations during the development of the MAS simulation. The values are also chosen such that they are large enough for the different effects of the mechanisms to be able to propagate throughout the simulation. Furthermore, the value of T_{window} is chosen in order to enforce a 15s safety margin between the anticipated conflict resolutions.

8.2. Taxi Time Results

This section presents additional taxi time results of the eight simulated days.

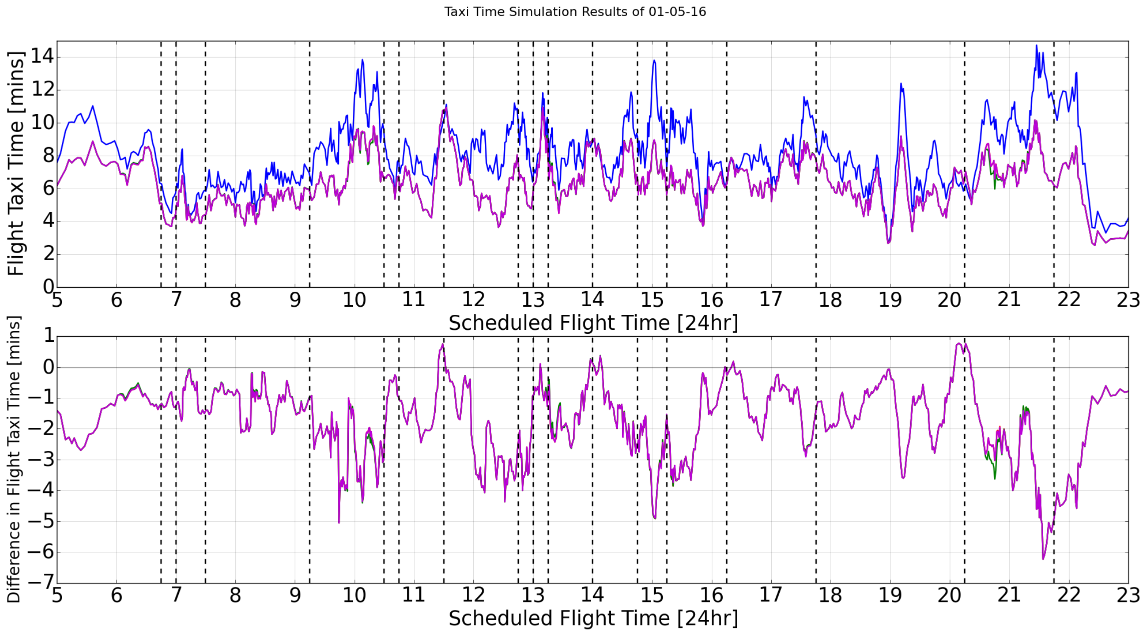

Figure 8.1: Taxi Time MAS Simulation Results of 01-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

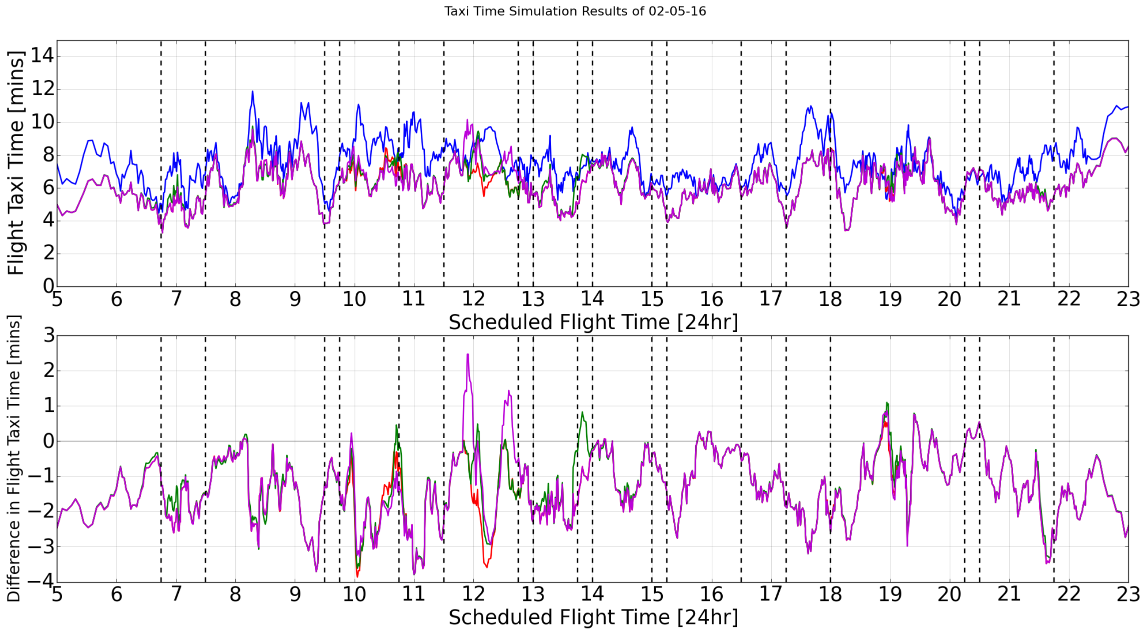

Figure 8.2: Taxi Time MAS Simulation Results of 02-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

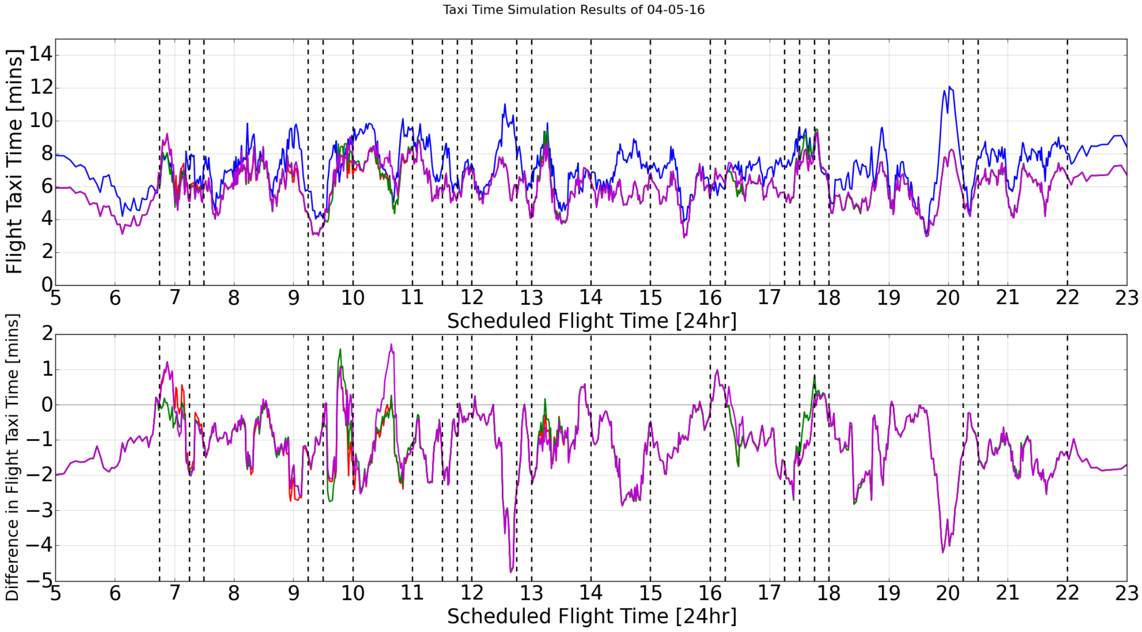

Figure 8.3: Taxi Time MAS Simulation Results of 04-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

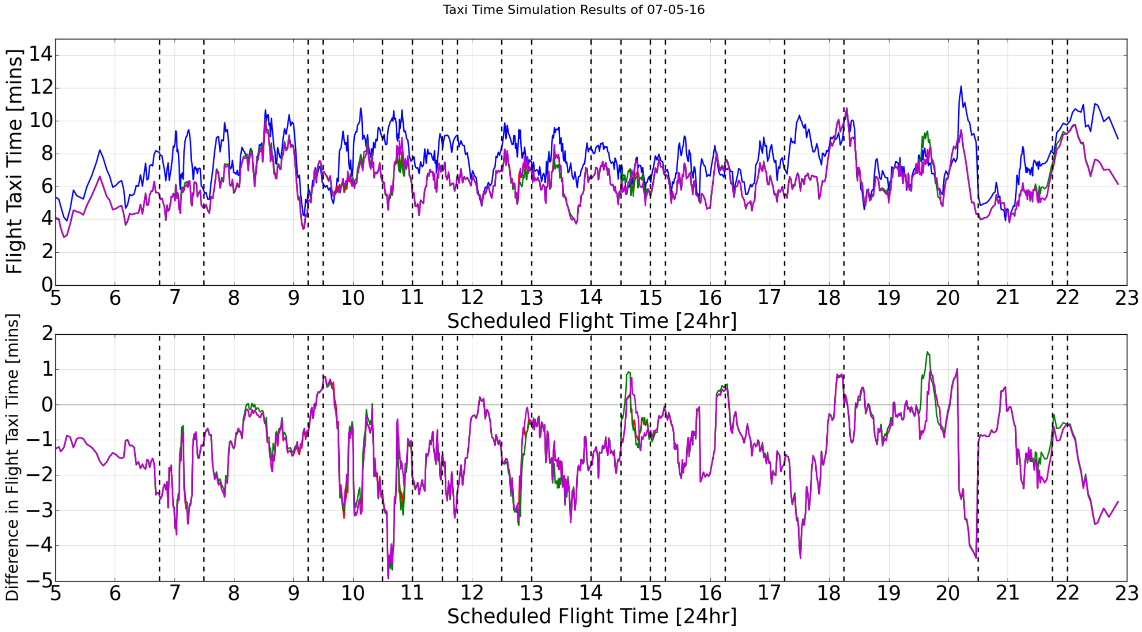

Figure 8.4: Taxi Time MAS Simulation Results of 07-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

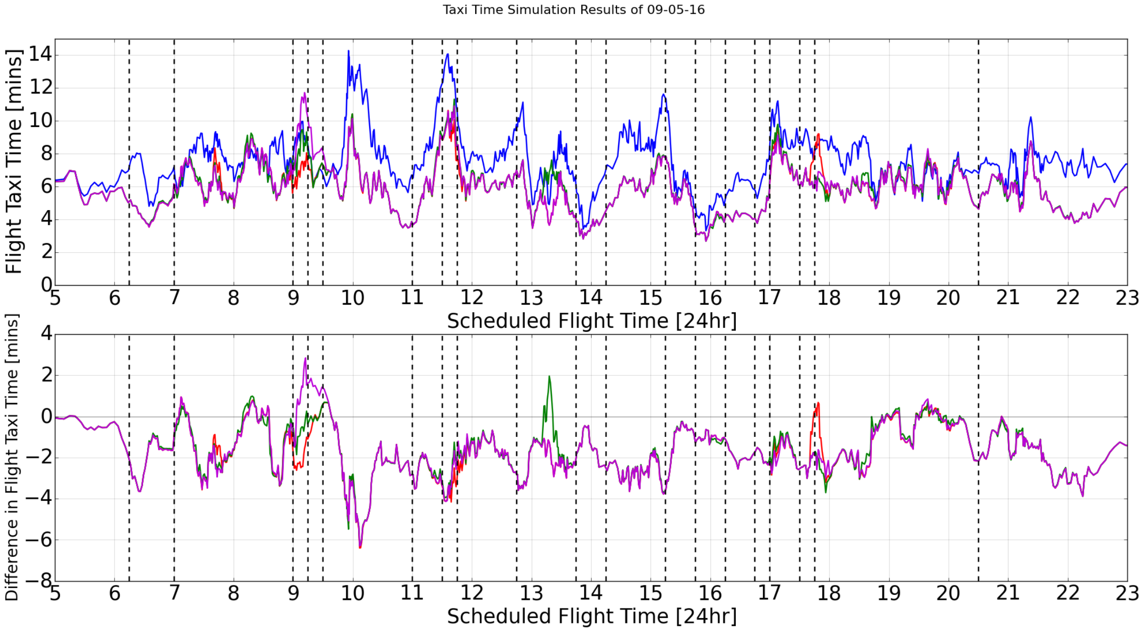

Figure 8.5: Taxi Time MAS Simulation Results of 09-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

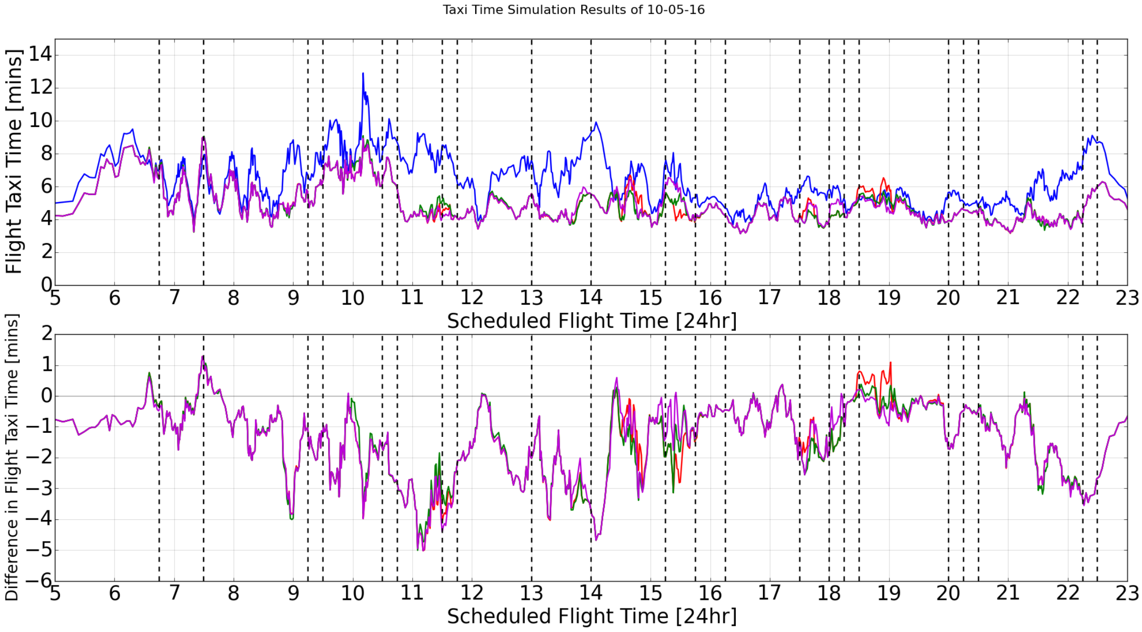

Figure 8.6: Taxi Time MAS Simulation Results of 10-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

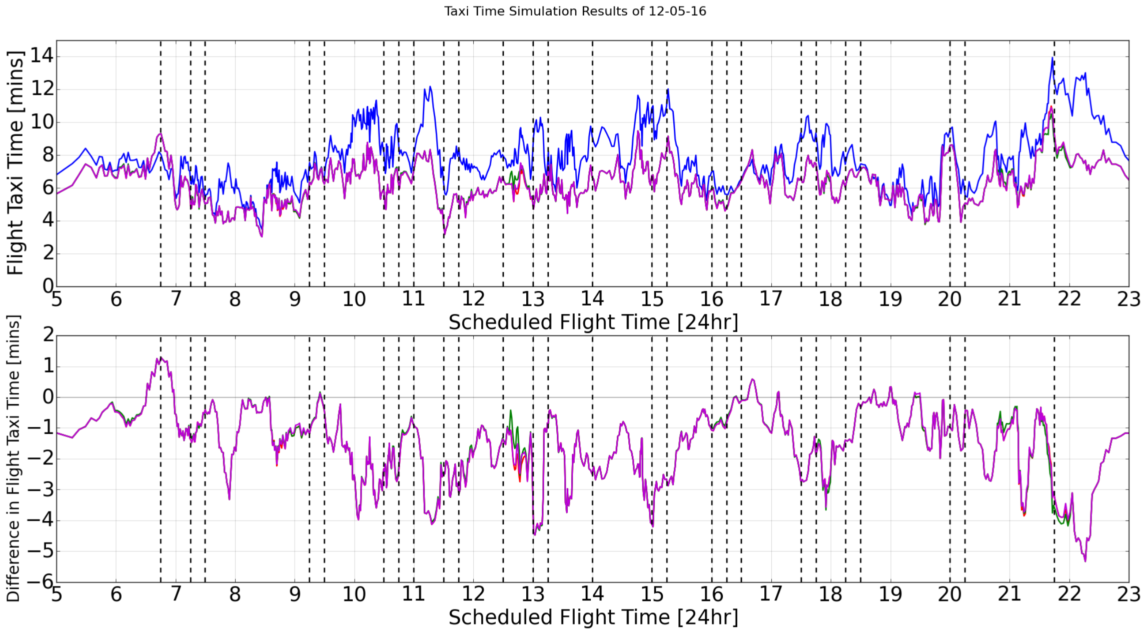
Taxi Time Simulation Results of 12-05-16

Figure 8.7: Taxi Time MAS Simulation Results of 12-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

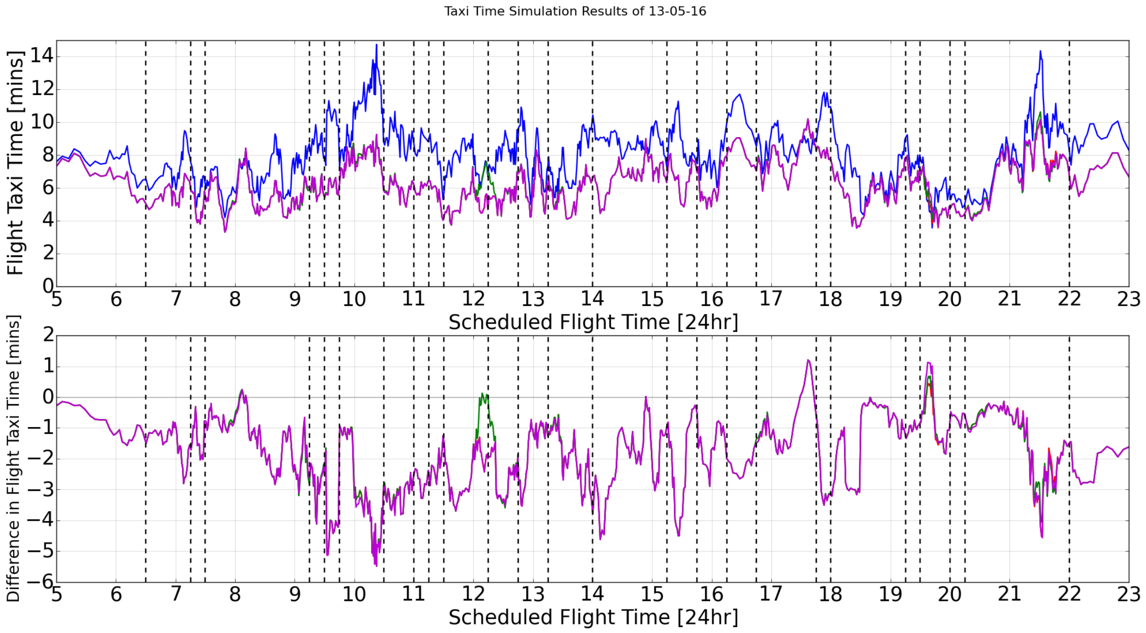
Taxi Time Simulation Results of 13-05-16

Figure 8.8: Taxi Time MAS Simulation Results of 13-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

8.3. Taxi Distance Results

This section presents the taxi distance results for the eight simulated days.

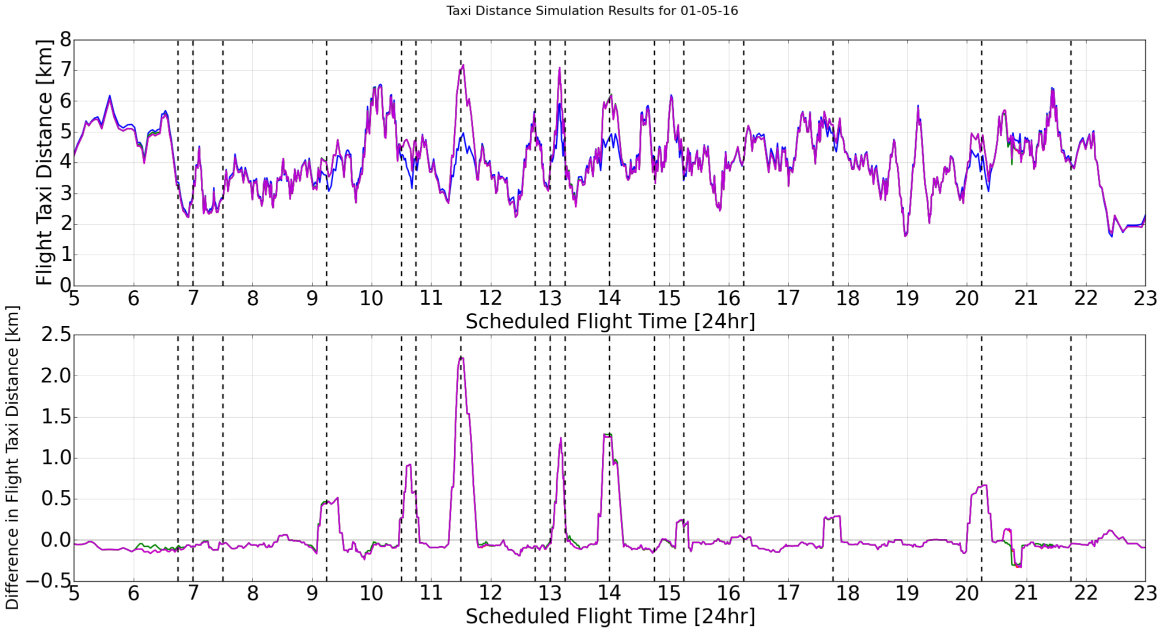

Figure 8.9: Taxi Distance MAS Simulation Results of 01-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

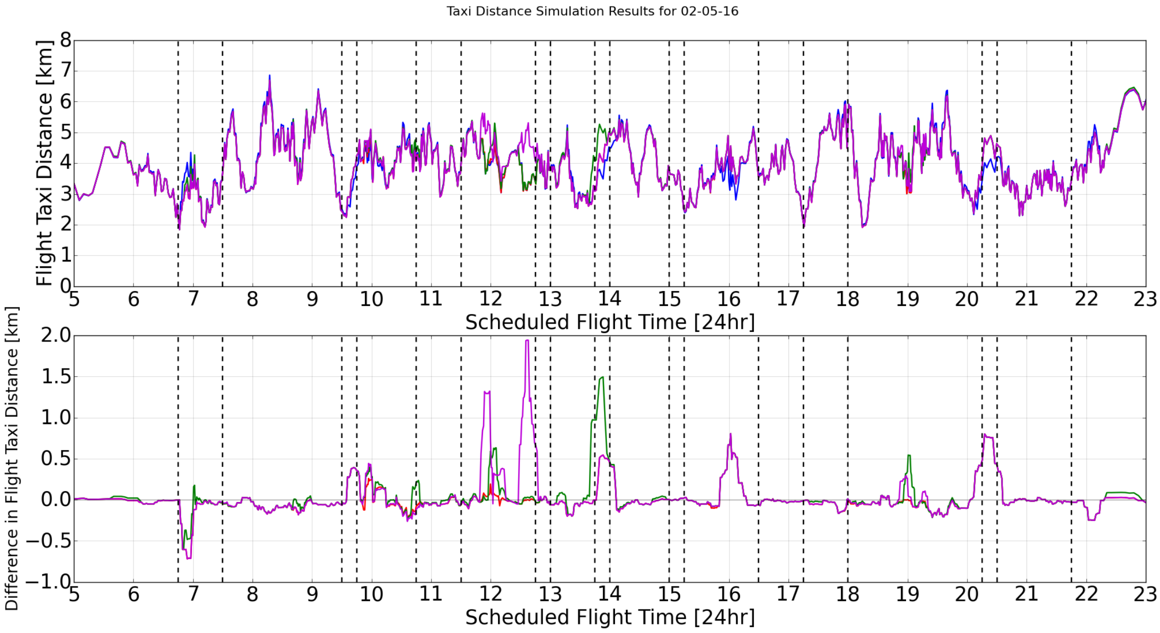

Figure 8.10: Taxi Distance MAS Simulation Results of 02-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

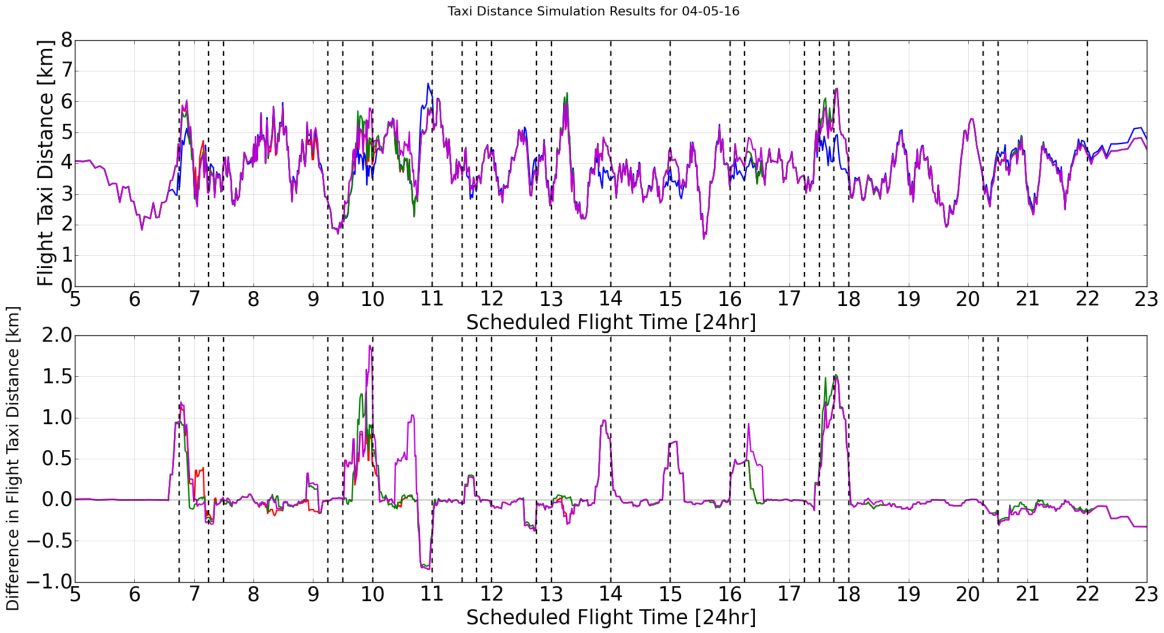

Figure 8.11: Taxi Distance MAS Simulation Results of 04-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

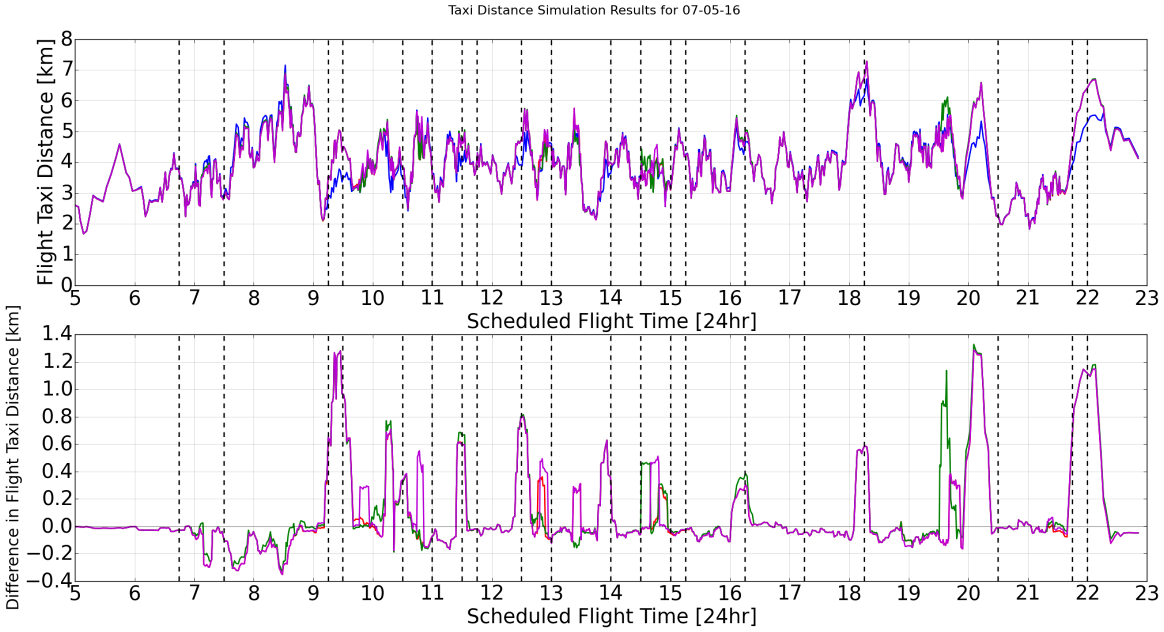

Figure 8.12: Taxi Distance MAS Simulation Results of 07-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

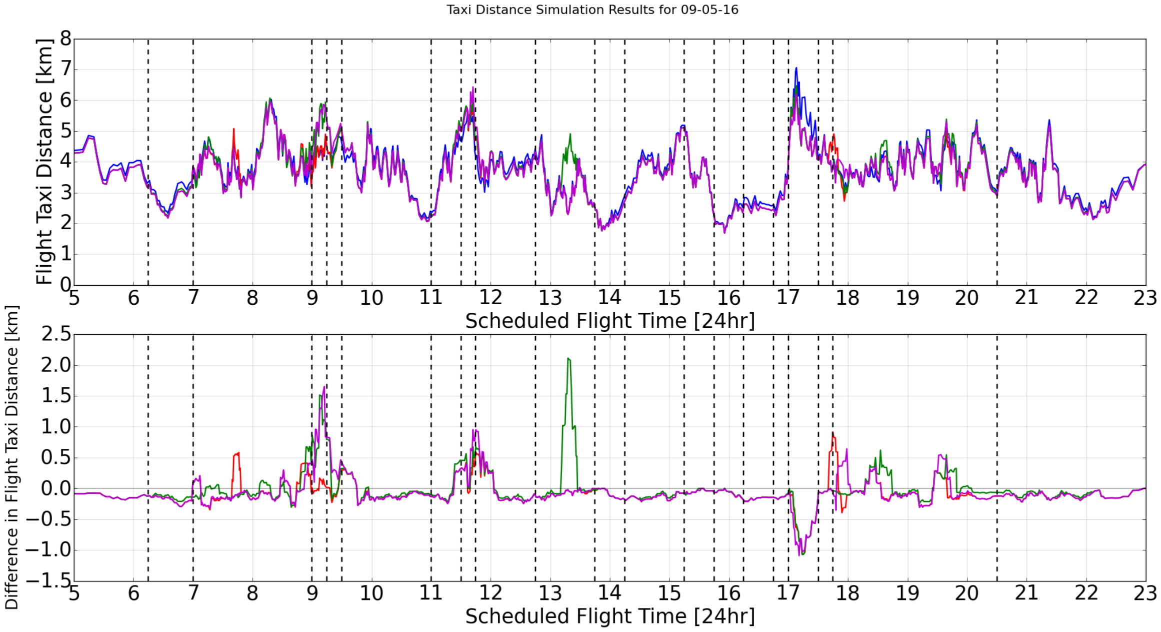

Figure 8.13: Taxi Distance MAS Simulation Results of 09-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

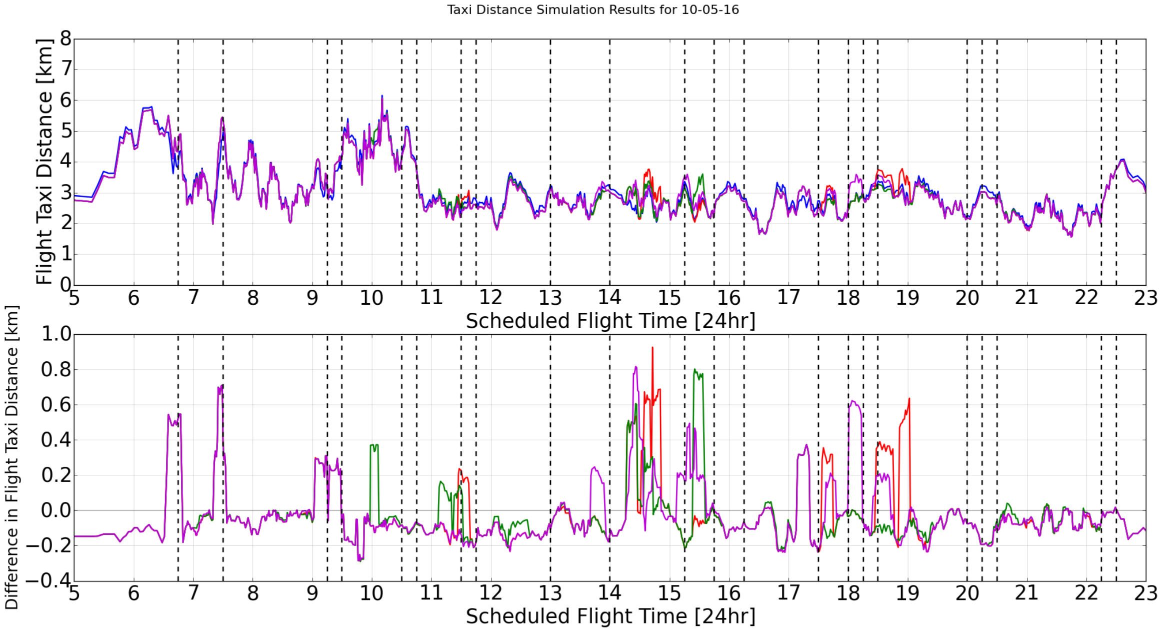

Figure 8.14: Taxi Distance MAS Simulation Results of 10-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

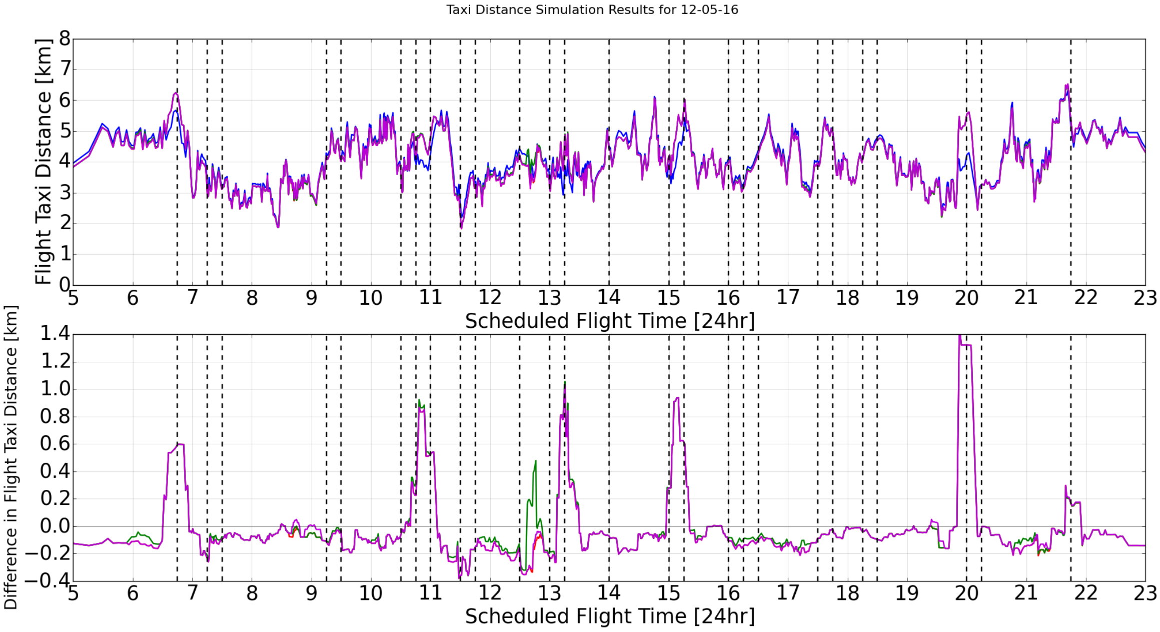

Figure 8.15: Taxi Distance MAS Simulation Results of 12-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

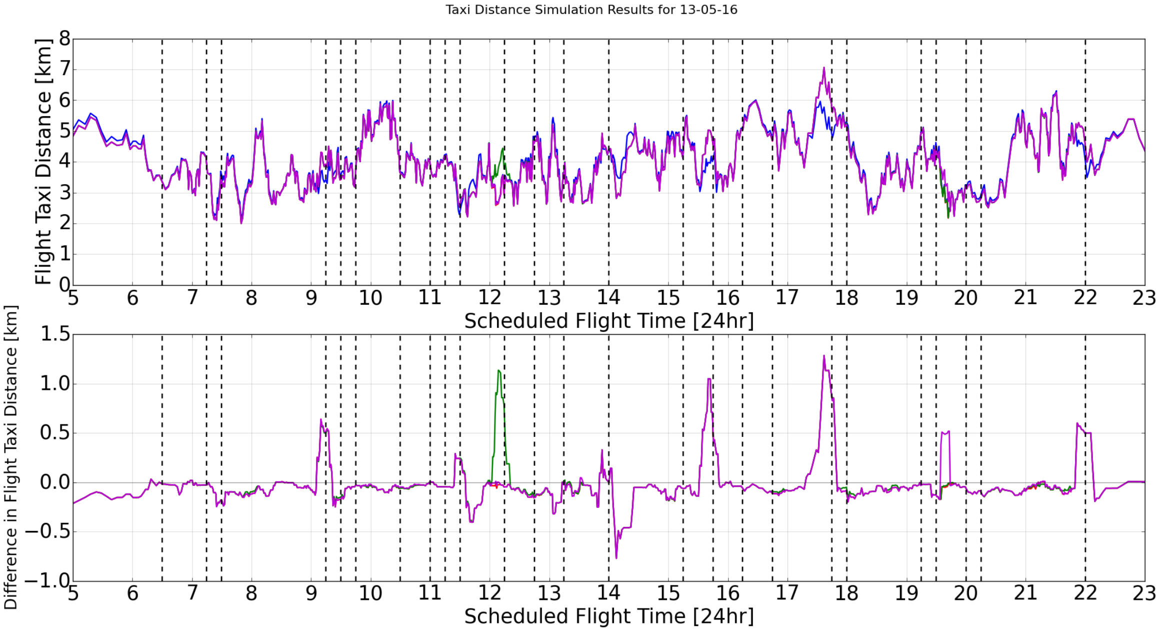

Figure 8.16: Taxi Distance MAS Simulation Results of 13-05-16. Blue line: real-world performance, red line: CBS mechanism performance, green line: CBS + PM HWYs mechanism performance, purple line: CBS + CB HWYs performance. Dotted vertical lines represent runway reconfiguration events.

8.4. Performance Deviations for All Runway Reconfiguration Events

This section presents the runway reconfigurations events for each of the eight simulated days. In this case, $t_w = 15$ min is used, to conform with the results presented in the paper in Part I.

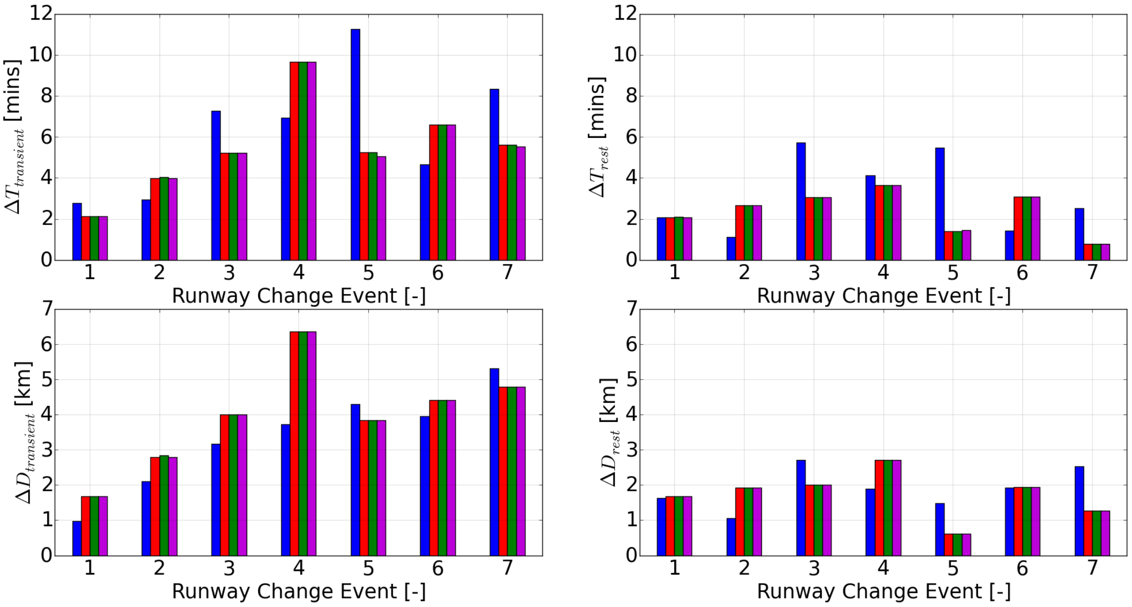

Figure 8.17: Runway reconfigurations considered on 01-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

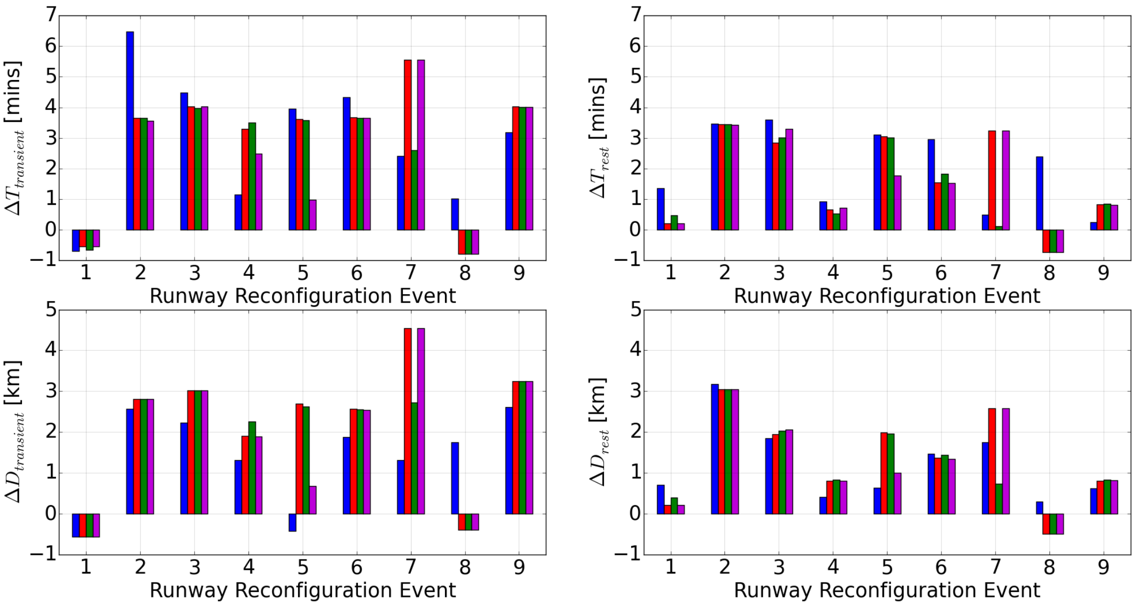

Figure 8.18: Runway reconfigurations considered on 02-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

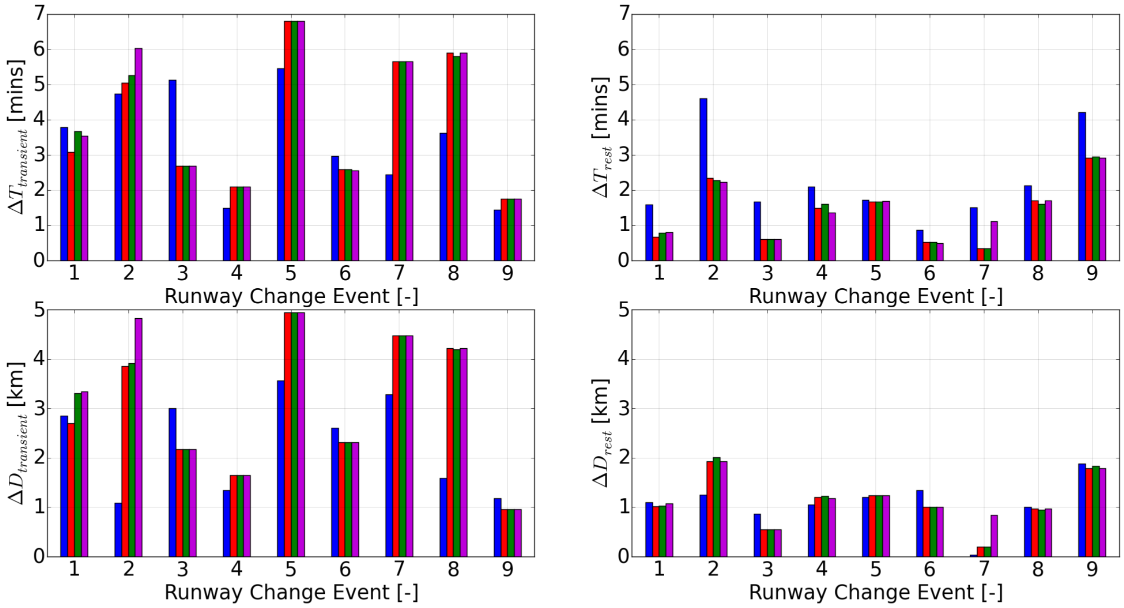

Figure 8.19: Runway reconfigurations considered on 04-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

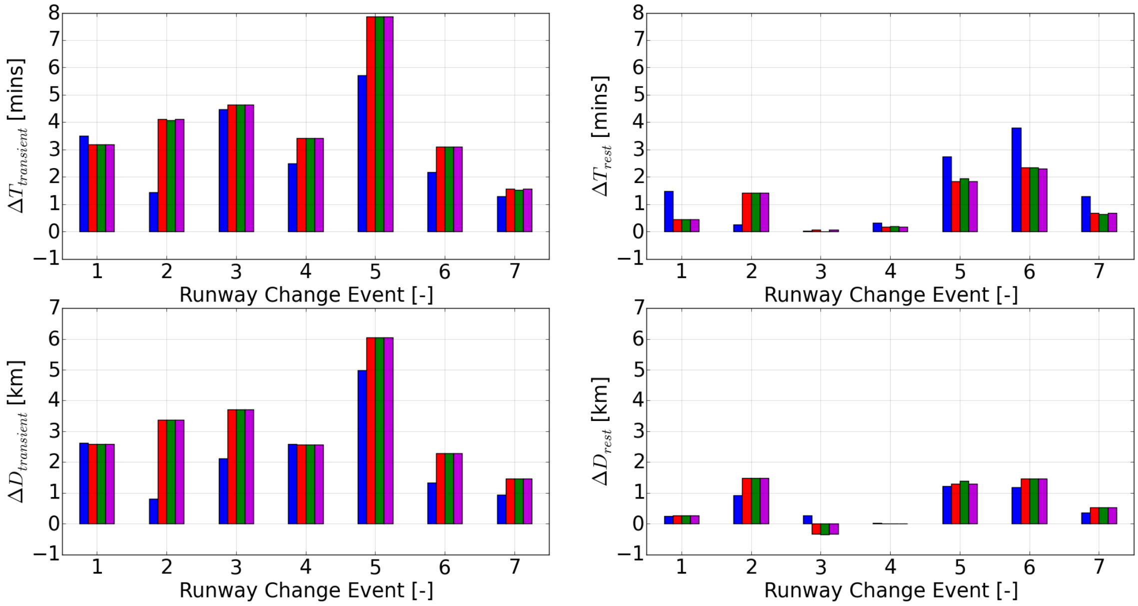

Figure 8.20: Runway reconfigurations considered on 07-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

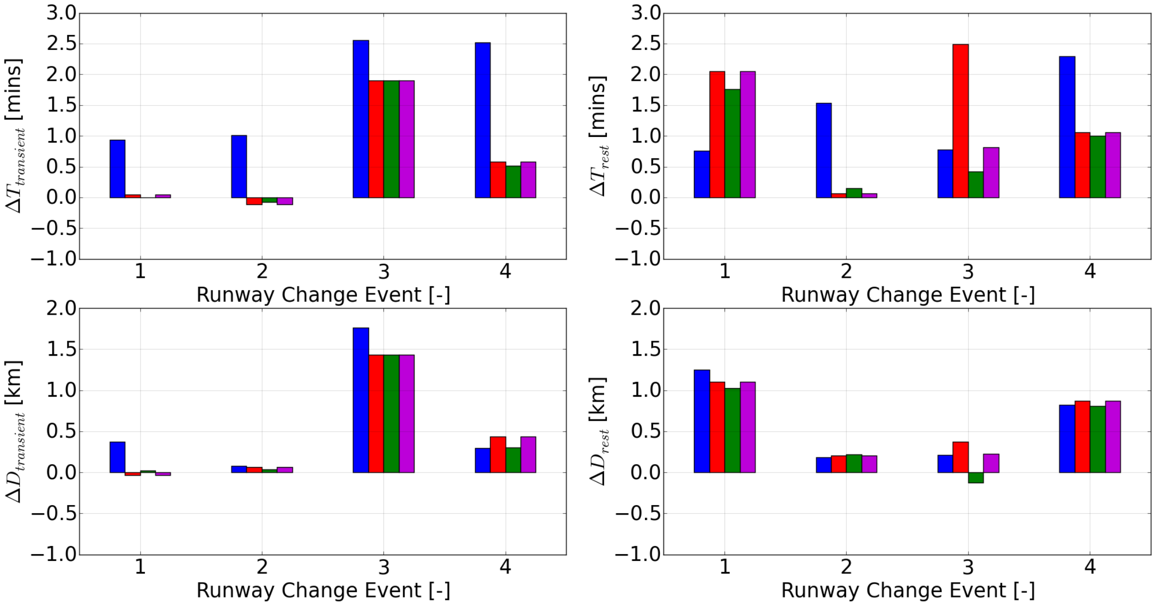

Figure 8.21: Runway reconfigurations considered on 09-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

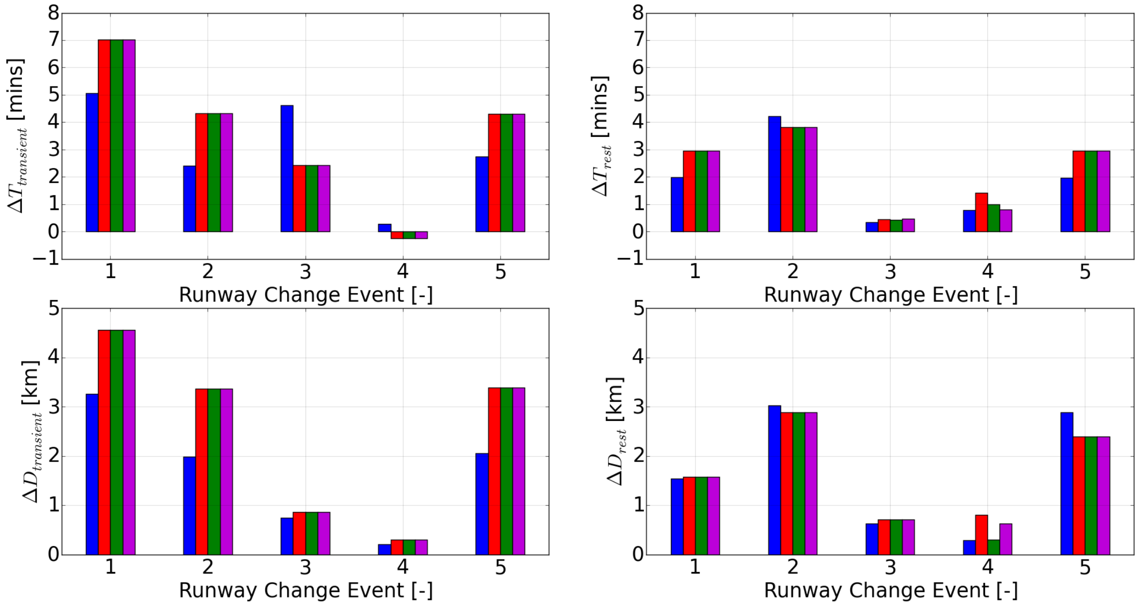

Figure 8.22: Runway reconfigurations considered on 10-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

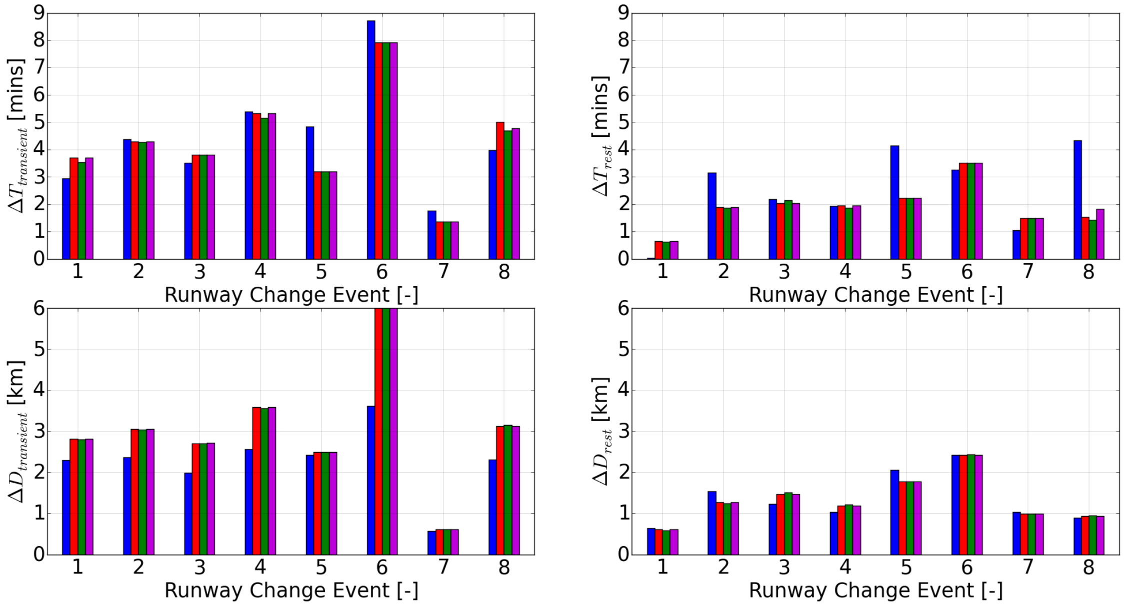

Figure 8.23: Runway reconfigurations considered on 12-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

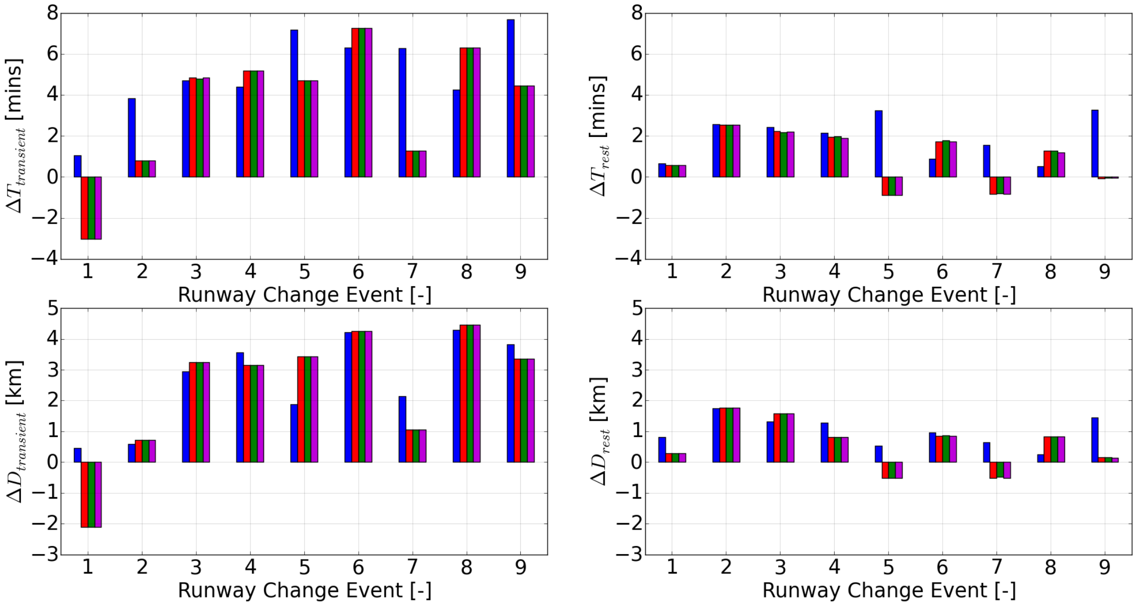

Figure 8.24: Runway reconfigurations considered on 13-05-16. Blue lines: real-world performance, red line: CBS mechanism performance, green line: CBS+PM HWYs mechanism performance, purple line: CBS+CB HWYs mechanism performance.

8.5. Deviation Analysis for 02-05-16

Table 8.2: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on 02-05-16, $t_w = 10$ min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	3.38	1.92	3.58	2.27
CBS	3.30	2.53	2.65	2.09
CBS+PM HWYs	3.27	2.55	2.59	2.11
CBS+CB HWYs	3.22	2.53	2.67	2.09

Table 8.3: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on 02-05-16, $t_w = 20$ min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	2.85	1.35	2.05	1.09
CBS	2.80	2.07	1.51	1.10
CBS+PM HWYs	2.64	2.00	1.36	1.02
CBS+CB HWYs	2.52	1.92	1.49	1.04

Table 8.4: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on 02-05-16, $t_w = 25$ min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	3.63	1.45	1.97	1.52
CBS	3.77	2.89	2.35	1.66
CBS+PM HWYs	3.58	2.79	2.15	1.57
CBS+CB HWYs	3.43	2.64	2.31	1.59

Table 8.5: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on 02-05-16, $t_w = 30$ min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	4.24	1.18	1.23	0.94
CBS	2.92	2.27	1.46	0.99
CBS+PM HWYs	2.88	2.26	1.40	0.98
CBS+CB HWYs	2.45	1.93	1.62	1.04

8.6. Deviation Analysis for All Days

Table 8.6: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on All Simulated Days, $t_w = 10$ min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	$\mu = 4.11, \sigma = 2.23$	$\mu = 2.31, \sigma = 1.32$	$\mu = 3.44, \sigma = 1.74$	$\mu = 1.93, \sigma = 1.03$
CBS	$\mu = 3.91, \sigma = 2.29$	$\mu = 2.87, \sigma = 1.69$	$\mu = 2.65, \sigma = 1.81$	$\mu = 1.90, \sigma = 1.26$
CBS+PM HWYs	$\mu = 3.93, \sigma = 2.29$	$\mu = 2.89, \sigma = 1.69$	$\mu = 2.62, \sigma = 1.75$	$\mu = 1.89, \sigma = 1.25$
CBS+CB HWYs	$\mu = 3.90, \sigma = 2.31$	$\mu = 2.89, \sigma = 1.69$	$\mu = 2.64, \sigma = 1.80$	$\mu = 1.90, \sigma = 1.26$

Table 8.7: MAS Simulation A-test Values of the Average Resilient Responses for the Considered Runway Reconfiguration Events of All Days, $t_w = 10$ min

Planning and Coordination Mechanism	A-test value of $\Delta T_{transient}$	A-test value of $\Delta D_{transient}$	A-test value of ΔT_{rest}	A-test value of ΔD_{rest}
CBS	0.48	0.62	0.37	0.47
CBS+PM HWYs	0.48	0.62	0.36	0.46
CBS+CB HWYs	0.48	0.62	0.37	0.47

Table 8.8: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on All Simulated Days, $t_w = 20$ min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	$\mu = 3.96, \sigma = 2.45$	$\mu = 1.99, \sigma = 1.22$	$\mu = 1.93, \sigma = 1.16$	$\mu = 1.00, \sigma = 0.74$
CBS	$\mu = 3.41, \sigma = 2.33$	$\mu = 2.49, \sigma = 1.70$	$\mu = 1.31, \sigma = 1.15$	$\mu = 0.91, \sigma = 0.79$
CBS+PM HWYs	$\mu = 3.57, \sigma = 2.33$	$\mu = 2.6, \sigma = 1.66$	$\mu = 1.27, \sigma = 1.08$	$\mu = 0.88, \sigma = 0.77$
CBS+CB HWYs	$\mu = 3.53, \sigma = 2.31$	$\mu = 2.57, \sigma = 1.66$	$\mu = 1.26, \sigma = 1.12$	$\mu = 0.90, \sigma = 0.77$

Table 8.9: MAS Simulation A-test Values of the Average Resilient Responses for the Considered Runway Reconfiguration Events of All Days, $t_w = 20$ min

Planning and Coordination Mechanism	A-test value of $\Delta T_{transient}$	A-test value of $\Delta D_{transient}$	A-test value of ΔT_{rest}	A-test value of ΔD_{rest}
CBS	0.46	0.61	0.36	0.46
CBS+PM HWYs	0.47	0.66	0.36	0.44
CBS+CB HWYs	0.47	0.63	0.36	0.46

Table 8.10: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on All Simulated Days, t_w = 25 min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	$\mu = 3.98, \sigma = 2.25$	$\mu = 1.92, \sigma = 1.10$	$\mu = 1.81, \sigma = 1.29$	$\mu = 1.12, \sigma = 0.61$
CBS	$\mu = 3.67, \sigma = 1.92$	$\mu = 2.69, \sigma = 1.45$	$\mu = 1.39, \sigma = 1.10$	$\mu = 1.03, \sigma = 0.71$
CBS+PM HWYs	$\mu = 3.64, \sigma = 1.89$	$\mu = 2.70, \sigma = 1.41$	$\mu = 1.36, \sigma = 1.01$	$\mu = 1.03, \sigma = 0.69$
CBS+CB HWYs	$\mu = 3.64, \sigma = 1.94$	$\mu = 2.69, \sigma = 1.46$	$\mu = 1.26, \sigma = 1.07$	$\mu = 1.02, \sigma = 0.69$

Table 8.11: MAS Simulation A-test Values of the Average Resilient Responses for the Considered Runway Reconfiguration Events of All Days, t_w = 25 min

Planning and Coordination Mechanism	A-test value of $\Delta T_{transient}$	A-test value of $\Delta D_{transient}$	A-test value of ΔT_{rest}	A-test value of ΔD_{rest}
CBS	0.46	0.67	0.42	0.45
CBS+PM HWYs	0.46	0.67	0.41	0.45
CBS+CB HWYs	0.45	0.66	0.42	0.45

Table 8.12: MAS Simulation Average Resilient Responses for the Runway Reconfiguration Events on All Simulated Days, t_w = 30 min

Planning and Coordination Mechanism	Average $\Delta T_{transient}$ [min/event]	Average $\Delta D_{transient}$ [km/event]	Average ΔT_{rest} [min/event]	Average ΔD_{rest} [km/event]
Real-World	$\mu = 3.46, \sigma = 2.14$	$\mu = 1.82, \sigma = 1.26$	$\mu = 1.35, \sigma = 1.10$	$\mu = 0.93, \sigma = 0.57$
CBS	$\mu = 3.46, \sigma = 2.29$	$\mu = 2.53, \sigma = 1.58$	$\mu = 1.24, \sigma = 0.97$	$\mu = 0.89, \sigma = 0.61$
CBS+PM HWYs	$\mu = 3.43, \sigma = 2.26$	$\mu = 2.56, \sigma = 1.57$	$\mu = 1.17, \sigma = 0.88$	$\mu = 0.87, \sigma = 0.62$
CBS+CB HWYs	$\mu = 3.39, \sigma = 2.35$	$\mu = 2.49, \sigma = 1.63$	$\mu = 1.24, \sigma = 0.97$	$\mu = 0.88, \sigma = 0.63$

Table 8.13: MAS Simulation A-test Values of the Average Resilient Responses for the Considered Runway Reconfiguration Events of All Days, t_w = 30 min

Planning and Coordination Mechanism	A-test value of $\Delta T_{transient}$	A-test value of $\Delta D_{transient}$	A-test value of ΔT_{rest}	A-test value of ΔD_{rest}
CBS	0.51	0.66	0.52	0.50
CBS+PM HWYs	0.51	0.66	0.51	0.48
CBS+CB HWYs	0.51	0.65	0.51	0.49

8.7. Considered Runway Reconfiguration Events for 02-05-16

Table 8.14: Considered Runway Reconfiguration Events for 02-05-16.

Runway reconfiguration event number	Time at which it occurred
1	06:45
2	07:30
3	09:45
4	10:45
5	11:30
6	13:00
7	14:00
8	17:15
9	18:00

9

Additional Elements

9.1. CBS Implementation Issues

Whilst running the MAS simulations, an implementation issue was identified and will now be further elaborated upon. This issue resulted in deadlock scenarios upon the airport's surface. It was determined that the cause of this was not because of any shortcomings of the mechanisms but of the implementation of the the CBS algorithm. Furthermore, these issues did not occur due to a runway reconfiguration event. This was deduced as as the deadlock scenario did not occur at a time point close to any runway reconfiguration ($t \leq t_0 \pm t_w$). Instead, the failure occurred due to the geometrical graph layout shown in Fig. 9.1.

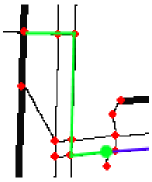

Figure 9.1: Geometrical implementation issue. Green line represents route of arrival aircraft, blue line represents departure route of aircraft. The green node represents the conflict node.

The indicated green node was the conflict node. Although a head-on collision should have been detected, it was not as the T_{window} was too small and the node was not declared as a conflict node until the time point where both Aircraft Agents were travelling directly towards it (and due to the geometry of this particular intersection, no alternative paths for either Aircraft Agent was possible). This situation also occurred for other nodes upon the airport's surface where no alternative paths were possible from them.

This shows that further interdependencies exist between the varying graph geometry and the implementation of the main CBS mechanism. Possible solutions to this could be to either group nodes which do not have alternative paths from them as a meta-agent with some of their neighbours.

9.2. Release Mechanism Property

This property involves interactions between: 1. Entry/Exit Agents and ATC Agents, 2. Entry/Exit Agents and Aircraft Agents.

Furthermore, this property is executed if the agent has communicated a route to an Aircraft Agent, at the same time point as the *Route Generation Property* in Part I.

The agent observes its location in the graph. If the agent is located at a runway holding point, then the agent releases the Aircraft Agent.

Otherwise, if the agent is located at an entry/exit point, then the agent determines if there are any Aircraft Agents on the the red edges and travelling in the direction as shown in Fig. 9.2. If there are no Aircraft Agents, then the agent releases the Aircraft Agent. Otherwise, the agent waits until the edges are free, and then releases the Aircraft Agent.

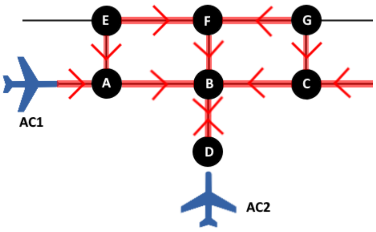

Figure 9.2: Release mechanism protected area. The highlighted red edges must be free.

9.3. Release Mechanism Example and Implementation Comments

Fig. 9.2 is used as an example. The Entry/Agent at node D communicates with the ATC Agent at node B and its neighbouring ATC Agents at nodes A, F and C. If any of these ATC Agents has Aircraft Agents in their control and the Aircraft Agent is heading in the direction indicated by the arrows in Fig. 9.2 (i.e. towards node B), then the Entry/Exit Agent does not release $AC2$ into the taxiway network. Otherwise, the agent releases $AC2$. In this way, any aircraft entering or in the the red edge zone and heading in the direction indicated by the arrows in Fig. 9.2 prevent the release of $AC2$.

It is important to note that, after observing the emergent effects of the MAS, and to account for the unevenly distributed geometries of the graph, the ATC Agent neighbours which the agent communicates with are expanded to look at more nodes in specific directions. Table 9.1 presents these neighbours, where the IDs correspond to those from [4].

Table 9.1: Additional entry mechanism neighbouring nodes

ATC Agent ID	Additional ATC Neighbour ID
6	34, 35, 28
8	45, 83, 48
7	48, 49, 50, 83, 47, 96, 51, 53
5	38, 74, 77, 76
20	74, 27, 75, 77, 85, 73, 29, 25, 79
1	105, 104, 103
15	104, 103
22	40, 86, 87

9.4. Airport Operation Status Agent Runway Edge Combinations

Table 9.2: Airport Operation Status Runway Edge Lookup Table

Runway in use	Arrival/Departure	Removed taxiway graph edge
18C	Arrival	W5 and Y
36C	Arrival	W5 and Z
18C	Departure	W5 and Z
36C	Departure	W5 and Y

9.5. Possible Flight Types for Point-Merge Highways

Table 9.3: Possible Flight Types for Point-Merge Highways

Flight Type ID	Arrival/Departure	Destination
1	Arrival	North Piers
2	Arrival	East Piers
3	Arrival	South Piers
4	Departure	Runway 36L
5	Departure	Runway 36C
6	Departure	Runway 18C
7	Departure	Runway 24
8	Departure	Runway 06
9	Departure	Runway 09
10	Departure	Runway 27
11	Departure	Runway 18L
12	Departure	Runway 36R
13	Departure	Runway 22
14	Departure	Runway 04

10

Recommendations for Future Work

This chapter presents recommendations for future work. These are as follows:

- **Improving the CBS implementation** As was previously mentioned, the implemented structural graph geometry had some shortcomings which resulted in particular failures of the current implementation of the CBS algorithm. An analysis of some of these situations has been presented in the previous chapters. These should be further investigated and improved in order to be able to enable a wider variety of operational situations to be explored.

- **Improving predictive elements** A core element of the CBS implementation is the prediction of times at which aircraft will traverse taxiway junctions (nodes within the graph). Although an initial forward simulation has been implemented, effects from varying speeds, slowing due to speed control commands from other ATC Agents, as well as from maintaining visual separation from other aircraft have not been fully captured. For this reason, it is recommended that investigations into alternative ways in order to generate these predictions be undertaken, such as using machine learning or statistical prediction techniques.

- **Anticipated runway reconfigurations** The results highlighted that the lack of anticipation in the actual runway reconfiguration event was not beneficial. Instead, it is recommended that the runway reconfigurations can be anticipated by the MAS model, such that the system can re-organize prior to the actual runway reconfiguration event occurring.

- **Mechanism adaptability** This study utilized fixed values for the variables and parameters which are used within the cooperative coordination, such as the conflict time window, highway generation thresholds. For this reason, it is recommended to investigate adaptive ways for these values to be chosen, instead of having fixed value. The MAS could learn or dynamically change values based on certain criteria.

- **Variable aircraft kinematics** In a similar ideology to the previous point, fixed Aircraft Agent kinematics were used, such as fixed maximum speeds and accelerations. In order to make the simulations more realistic, variable values for these could be used based on specific aircraft types (e.g., a Boeing 747 having a lower maximum taxi speed and lower acceleration than a Boeing 737). This could be used to assess the operationalization of the proposed mechanisms within this study.

- **Predefined highways** This study did not include any pre-defined or fixed highways. Instead, highways were automatically generated from the emergent behaviour of the MAS model. However, as highways were demonstrated to be promising with respect to resilience, investigations should be carried out in order to investigate whether fixed highways based on specific runway reconfigurations are more advantageous and effective than generating highways automatically.

- **Update real-world data** More up-to-date data is recommended to be used, with specific data sources for runway reconfiguration events, such that they do not need to be inferred from trajectory data. In this way, better comparisons can be made and more realistic MAS simulations can be run.

- **Other complex airports** Other, more complex airport surface movement operations and infrastructures are recommended to be utilized in order to assess how the mechanisms perform in different operational environments.

Bibliography

[1] EUROCONTROL. Airport cdm implementation, 2012.

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

[3] Air traffic control the netherlands. `http://www.ais-netherlands.nl/aim/2018-10-25-AIRAC/html/index-en-GB.html`. Accessed: 25-10-2018.

[4] T.E.H. Noortman. Agent-based modelling of an airport's ground surface movement operation. Master's thesis, Delft University of Technology, the Netherlands, 2018.

[5] P. Roling. Lecture slides: Ae4446 airport operations, runway design, February 2018.

[6] by Adam Aladdin Kes47, Based on File:Decentralization.jpg. Graphical comparison of a centralised (a) and a decentralised (b) system. `https://commons.wikimedia.org/w/index.php?curid=35018016`. Accessed on 20-12-2018.

[7] EUROCONTROL. Eurocontrol specification for advanced-surface movement guidance and control system (A-SMGCS) services, 2018.

[8] Anne-Nynke Blok, Alexei Sharpanskykh, and Matthieu Vert. Formal and computational modeling of anticipation mechanisms of resilience in the complex sociotechnical air transport system. *Complex Adaptive Systems Modeling*, page 6:7, 2018.

[9] Mohammad Al-Zinati and Rym Zalila-Wenkstern. A resilient agent-based re-organizing traffic network for urban evacuations. *Advances in Practical Applications of Agents, Multi-Agent Systems, and Complexity*, pages 42–58, 2018.

[10] Mohamad EL Falou, Maroua Bouzid, and Abdel-Illah Mouaddib. DEC-A*: a decentralized multiagent pathfinding algorithm. *2012 IEEE 24th International Conference on Tools with Artificial Intelligence*, pages 516–523, 2012.

[11] Ko-Hsin Cindy Wang and Adi Botea. MAPP: a scalable multi-agent path planning algorithm with tractability and completeness guarantees. *Journal of Artificial Intelligence Research*, 42:55–90, 2011.

[12] Bolder's gate II map: Ar0603sr. `https://movingai.com/benchmarks/bgmaps/index.htmll`. Accessed: 04-11-2018.

[13] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R. Sturtevant. Conflict-based search for optimal multi-agent pathfinding. *Artificial Intelligence*, 219:40–66, 2015.

[14] Max Barer, Guni Sharon, and Roni Stern adn Ariel Felner. Suboptimal variants of the conflict-based search algorithm for the multi-agent pathfinding problem. *Proceedings of the Seventh Annual Symposium on Combinatorial Search*, 2014.

[15] Liron Cohen, Tansel Uras, and Sven Koenig. Feasibility study: Using highways for bounded-suboptimal multi-agent path finding. *Proceedings of the Eighth Annual Symposium on Combinatorial Search*, 2015.

[16] EUROCONTROL. Eurocontrol seven-year forecast february 2018: Flight movements and service units 2018-2024, 2018.

[17] EUROCONTROL. Coda digest q1 2018: All-causes delay and cancellations to air transport in europe q1 2018, 2018.

[18] T. E. H. Noortman. Agent-based modelling of airport's ground surface movement: Literature study, 2017.

[19] Schiphol. 100 years of schiphol. `https://www.schiphol.nl/en/you-and-schiphol/page/over-100-years-of-schiphol/`. Accessed on 02-12-2018.

[20] Schiphol. Airport facts about dutch international airport. `https://www.schiphol.nl/en/route-development/page/amsterdam-airport-schiphol-airport-facts/`. Accessed on 02-12-2018.

[21] EUROCONTROL. Amsterdam schiphol airport adopts a-cdm. `https://www.eurocontrol.int/news/amsterdam-schiphol-airport-adopts-cdm`. Accessed on 13-11-2018.

[22] ICAO. Annex 14: Aerodrome design and operations. Technical report, International Civil Aviation Organization, 2016.

[23] Eurocontrol. Icao standard phraseology. Technical report, EUROCONTROL.

[24] ICAO. Annex 2: Rules of the air. Technical report, International Civil Aviation Organization, 2005.

[25] LVNL. Runway use. `https://en.lvnl.nl/environment/runway-use`. Accessed on 07-12-2018.

[26] ICAO. Aerodrome meteorological observation and forecast study group (amofsg). `https://www.icao.int/safety/meteorology/amofsg/AMOFSG%20Meeting%20Material/AMOFSG.10.SN.014.5.en.pdf`. Accessed on 25-11-2018.

[27] LVNL. Runway use. `https://en.lvnl.nl/safety/achieving-safety/weather-conditions`. Accessed on 23-11-2018.

[28] KNMI. Uurgegevens van het weer in nederland. `https://projects.knmi.nl/klimatologie/uurgegevens/selectie.cgi`. Accessed on 01-12-2018.

[29] KNMI. Daggegevens van het weer in nederland. `http://projects.knmi.nl/klimatologie/daggegevens/selectie.cgi`. Accessed on 01-12-2018.

[30] G. Ballester Valor. Metar reports selection query. `https://www.ogimet.com/metars.phtml.en`. Accessed on 01-12-2018.

[31] FlightRadar24. Flight tracker. `https://www.flightradar24.com/`. Accessed on 01-12-2018.

[32] Chun xia Dou, Da wei Hao, Bao Jin, Wei qian Wang, and Na An. Multi-agent-system-based decentralized coordinated control for large power systems. *International Journal of Electrical Power & Energy Systems*, 58:130–139, 2014.

[33] Jamie King, R.K. Pretty, and R.G Gosine. Coordinated execution of tasks in a multiagent environment. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 33:615–619, 2003.

[34] Alvaro Camara, Daniel Castro Silva, Eugenio Oliveira, and Pedro Henriques Abreu. Comparing centralized and decentralized multi-agent approaches to air traffic control. *Proceedings of the 28th European Simulation and Modelling Conference*, pages 189–193, 2014.

[35] H. Udluft. *Decentralization in Air Transportation.* PhD thesis, 2017.

[36] EUROCONTROL. Single european sky timeline. `https://www.eurocontrol.int/articles/single-european-sky-timeline`. Accessed on 07-10-2018.

[37] SESARJU. Sesar joint undertaking. `https://www.sesarju.eu/`. Accessed on 07-10-2018.

[38] ATRiCS Advanced Traffic Solutions GmbH. Follow the greens. `http://followthegreens.com/`. Accessed on 07-10-2018.

[39] 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. *Advances in Intelligent Systems and Computing*, 484(X):837–849, 2016.

[40] Adriano Covizzi and Francesco Modafferi. SESAR IVT expert report performed for exercise VP-759. Technical report, Eurocontrol, 2015.

[41] ATRiCS. ATRiCS advanced traffic solutions. `https://www.atrics.com/`. Accessed on 08-10-2018.

[42] Follow-the-greens: The future of guidance is green. Powerpoint presentation, 2017. SEAC European Airport Consortium.

[43] ICAO. Advanced surface movement guidance and control systems (a-smgcs) manual. 1, 2004.

[44] J. Bergstrom, R. van Winsen, and E. Henriqson. On the rationale of resilience in the domain of safety: A literature review. *Reliability Engineering and System Safety*, 141:131–141, 2015.

[45] A.R. Blok. Towards formalization and operationalization of resilience. Master's thesis, Delft University of Technology, Delft, the Netherlands, 2018.

[46] David D. Woods. Four concepts for resilience and the implications for the future of resilience engineering. *Reliability Engineering and System Safety*, 141:5–9, 2015.

[47] Sybert H. Stroeve and Mariken H.C. Everdij. Agent-based modelling and mental simulation for resilience engineering in air transport. *Safety Sceince*, 93:29–49, 2017.

[48] Jonas Lundberg and Bjorn JE Johansson. Systematic resilience model. *Reliability Engineering and System Safety*, 141:22–32, 2015.

[49] E. Hollnagel. *Resilience engineering in practice: a guidebook*. Ashgate, Farnham, 2011.

[50] Sarah Dunn and Sean M. Wilkinson. Increasing the resilience of air traffic networks using a network graph theory approach. *[Transportation Research Part E*, 90:39–50, 2016.

[51] Gloria Pumpuni-Lenss, Timothy Blackburn, and Andreas Garstenauer. Resilience in complex systems: Anagent-based approach. *Systems Engineering*, 20:158–172, 2017.

[52] G. Pezzulo, M.V. Butz, C. Castelfranchi, and R. Falcone. *The Challenge of Anticipation*. Springer, 2008.

[53] R. Westrum. A topology of resilience situations. *Resilience engineering: concepts and precepts*, pages 55–65, 2006.

[54] Edoardo Filippone, Francesco Gargiulo, Angela Errico, Vittorio Di Vito, and Domenico Pascarella. Resilience management problem in ATM systems as a shortest path problem. *Journal of Air Transport Management*, 56:57–65, 2016.

[55] International Civil Aviation Authority. *Manual on Global Performance of the Air Navigation System*. 2009.

[56] Kenneth A. Pettersen and Paul R. Schulman. Drift, adaptation, resilience and reliability: Toward an empirical clarification. *Safety Science*, 117:460–468, 2019.

[57] A. Sharpanskykh. Lecture slides: Coordination in multiagent systems, multiagent planning and scheduling, March 2017.

[58] Nick R. Jennings. Commitments and conventions: The foundation of coordination in multiagent systems. *Department of Electronic Engineering, Queen Mary and Westfield College, University of London*, 1993.

[59] DR. Wei Chen, Dr. Renato Levy, and Dr. Keith S. Decker. An integrated multi-agent coordination including planning, scheduling and task execution. 2007.

[60] Thomas W. Malone and Keving Crowston. The interdisciplinary study of coordination. *ACM Computing Surveys (CSUR)*, 1994.

[61] N. R. Jennings. Coordination techniques for distributed artificial intelligence. *Department of Electronic Engineering, Queen Mary and Westfield College, University of London*, 1996.

[62] Mathijs de Weerdt, Adriaan ter Mors, and Cees Witteveen. Multi-agent planning an introduction to planning and coordination. *Dept. of Software Technology, Delft University of Technology*, 2005.

[63] Akshay Kumar Guruji, Himansh Agarwal, and D. K. Parsediya. Time-efficient A* algorithm for robot path planning. *Procedia Technology*, 23:144–149, 2016.

[64] Xiao Cui and Hao Shi. A*-based pathfinding in modern computer games. *IJCSNS International Journal of COmputer Science and Network Security*, 11:125–130, 2011.

[65] Marcelo Becker, Carolina Meirelles Dantas, and Weber Perdigao Macedo. Obstacle avoidance procedure for mobile robots. *Association for the Advancement of Artifical Intelligence Techincal Report*, pages 19–25, 2012.

[66] Meir Goldenberg, Ariel Felner, Roni Stern, Guni Sharon, and Jonathan Schaeffer. A* variants for optimal multi-agent pathfinding. *Association for the Advancement of Artifical Intelligence Techincal Report*, pages 19–25, 2012.

[67] Esra Erdem, Doga G. Kisa, Umut Oztok, and Peter Schuller. A general formal framework for pathfinding problems with multiple agents. *Proceedings of the Twenty-Seventh Conference on Artificial Intelligence*, pages 290–296, 2013.

[68] Wolfgang Honig, T. K. Satish Kumar, Liron Cohen, Hang Ma, Hong Xu, Nora Ayanian, and Sven Koenig. Multi-agent path finding with kinematic constraints. *Proceedings of the Twenty-Sixth International Conference on Automated Planning and Scheduling (ICAPS 2016)*, pages 477–485, 2016.

[69] Hang Ma, T. K. Satish Kumar, and Sven Koenig. Multi-agent path finding with delay probabilities. *Association for the Advancement of Artifical Intelligence*, 2017.

[70] Will S. Briggs and Diane J. Cook. Modularity and communication in multiagent planning. *University of Texas at Arlington*, 1999.

[71] Yoav Shoham and Moshe Tennenholtz. On social laws for artificial agent societies: off-line design. *Artificial Intelligence*, 73, 1995.

[72] Boris de Wilde, Adriaan W. ter Mors, and Cees Witteveen. Push and rotate: a complete multi-agent pathfinding algorithm. *Journal of Artificial Intelligence Research*, 51:443–492, 2014.

[73] Ryan Luna and Kostas E. Bekris. Push and swap: Fast cooperative path-finding with completeness guarantees. *IJCAI'11 Proceedings of the Twenty-Second international joint conference on Artificial Intelligence*, 1:294–300, 2011.

[74] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R. Sturtevant. Conflict-based search for optimal multi-agent pathfinding. *AAAI'12 Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 563–569, 2012.

[75] Eli Boyarski, Ariel Felner, Roni Stern, Guni Sharon, David Tolpin, Oded Betzalel, and Eyal Shimony. ICBS: Improved conflict-based search algorithm for multi-agent pathfinding. *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, 2015.

[76] Liron Cohen, Tansel Uras, T.K. Satish Kumar, Hong Xu, Nora Ayanian, and Sven Koenig. Improved solvers for bounded-suboptimal multi-agent path finding. *Proceeding IJCAI'16 Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 3067–3074, 2016.

[77] Airbus. A318/a319/a320/a321 flight crew training manual, 2008.

[78] The Boeing Company. 737 flight crew training manual, 2007.